

MODELO ACADÉMICO DE LA FORMACIÓN LABORAL BÁSICA EN

DESARROLLO DE SOFTWARE



4^{to} **Semestre** **2025 – 2026B**

MÓDULO II. **BASE DE DATOS Y DESARROLLO WEB**

DIDÁCTICA PARA EL ESTUDIANTE

Nombre _____

Plantel _____ *Grupo* _____ *Turno* _____



COLEGIO DE BACHILLERES DE TABASCO

MTRO. FERNANDO YRYS HERNÁNDEZ
Director General

MTRA. ANA FANNY HERNÁNDEZ MONTERO
Directora Académica

LIC. CELIS DEL CARMEN RANERO LARA
Subdirectora de Planeación Académica

MTRA. NATIVIDAD VERTIZ HERNÁNDEZ
Jefa de Departamento de Capacitación para el Trabajo

FORMACIÓN LABORAL BÁSICA EN DESARROLLO DE SOFTWARE **MÓDULO II. BASE DE DATOS Y DESARROLLO WEB**

SUBMÓDULO I. DESARROLLO DE APLICACIONES DE ESCRITORIO CON BASE DE DATOS **SUBMÓDULO II. PROGRAMACIÓN WEB**

En la realización del presente material participaron:

COORDINADOR: MTRO. JORGE ARMANDO FLORES PALACIOS. PLANTEL 6
ASESORA: MTRA. MARÍA DEL ROSARIO HERNÁNDEZ ALEJANDRO. PLANTEL 2

EDICIÓN. 2025-2026B.

Revisado por.
LIC. MARÍA MARGARITA GARCÍA VIDAL

Este material fue elaborado bajo la coordinación y supervisión del Departamento de Capacitación para el Trabajo de la Dirección Académica del Colegio de Bachilleres de Tabasco y el contenido del mismo es gratuito, sin fines de lucro y con la responsabilidad de su buen uso para fines educativos

www.cobatab.edu.mx





CONTENIDO

MENSAJE DEL DIRECTOR GENERAL	I
PRESENTACIÓN	II
PROGRAMA DE ESTUDIOS DE LA FORMACIÓN LABORAL BÁSICA EN DESARROLLO DE SOFTWARE	III
FUNDAMENTACIÓN	IV
JUSTIFICACIÓN DE LA FORMACIÓN LABORAL BÁSICA EN DESARROLLO DE SOFTWARE	IX
ENFOQUE DE LA FORMACIÓN LABORAL BÁSICA.	XII
UBICACIÓN DE LA FORMACIÓN LABORAL	XVII
MAPA DE LA CAPACITACIÓN DESARROLLO DE SOFTWARE	XVIII
COMPETENCIAS LABORALES BÁSICAS	XIX
PROCESO DE EVALUACIÓN BAJO EL ENFOQUE EN COMPETENCIAS.....	XXII
EVALUACIÓN POR COMPETENCIAS	XXIV
INSTRUMENTOS DE EVALUACIÓN	XXIV
ROL DEL ESTUDIANTE	XXVI
SIMBOLOGÍA.....	XXVII
TEMARIO	XXIX
 SUBMÓDULO I. DESARROLLO DE APLICACIONES DE ESCRITORIO CON BASE DE DATOS	1
DOSIFICACIÓN PROGRAMÁTICA	2
CRITERIOS DE EVALUACIÓN	3
SITUACIÓN DIDÁCTICA.....	4
EVALUACIÓN DIAGNÓSTICA	6
 LECTURA NO 1. BIENVENIDA A LAS BASES DE DATOS	9
ACTIVIDAD 1. BASE DE DATOS	12
LECTURA NO 2. SOFTWARE MANEJADORES DE BASE DE DATOS	15
ACTIVIDAD 2. CUADRO COMPARATIVO MANEJADOR BD	18
INSTALACIÓN Y USO BÁSICO DE UN DBMS	20
LECTURA NO 3. TIPOS DE BASE DE DATOS	24
ACTIVIDAD 3. TABLA COMPARATIVA TIPOS Y MODELOS BD	29
LECTURA NO 4. ESTRUCTURA DE BASE DE DATOS.....	32
MODELADO BÁSICO DE DATOS.....	35



Educación que genera cambio

PRÁCTICA 1. MODELADO BÁSICO DE BASES DE DATOS.....	38
LECTURA NO 5. ESTRUCTURA DE BASE DE DATOS RELACIONALES	41
LECTURA NO 6. COMANDOS SQL Y LENGUAJE DE MANIPULACIÓN DE DATOS	45
COMANDOS DE MENÚ	48
ACTIVIDAD 4. PRESENTACIÓN ELECTRÓNICA.	57
COMANDOS SQL Y LENGUAJE DE MANIPULACIÓN DE DATOS	57
EJEMPLO GUIADO CON CÓDIGO SQL “HOLA MUNDO DE BASE DE DATOS”	59
PRÁCTICA 2. CREACIÓN DE BASE DE DATOS	60
EJEMPLO GUIADO CON CÓDIGO SQL “MANEJO DE REGISTROS”	64
PRÁCTICA 3. BASE DE DATOS CON INSTRUCCIONES SQL	66
ACTIVIDAD 5. ELEMENTOS DE UN PROYECTO BD SQL	69
LECTURA NO 6. INFORMES.....	72
ACTIVIDAD 6. GENEREMOS INFORMACIÓN	76
DATALAB: CREANDO CONEXIONES	81
 SUBMÓDULO II. PROGRAMACIÓN WEB.....	85
DOSIFICACIÓN PROGRAMÁTICA	86
CRITERIOS DE EVALUACIÓN	88
SITUACIÓN DIDÁCTICA.....	89
EVALUACIÓN DIAGNÓSTICA	90
 LECTURA NO 1. CONCEPTOS BÁSICOS PÁGINAS WEB Y SITIOS WEB.....	92
ACTIVIDAD 1. TABLA COMPARATIVA: ELEMENTOS DE LA WEB	99
LECTURA NO 2. ELEMENTOS DE UN PÁGINA WEB.....	101
ACTIVIDAD 2. IDENTIFICA LOS ELEMENTOS DE LAS PÁGINAS WEB.....	105
LECTURA NO 3. ESTÁNDARES DE DISEÑO WEB (W3C)	107
ACTIVIDAD 3. INFOGRAFÍA “ESTÁNDARES DE DISEÑO WEB”	112
LECTURA NO 4. LA ESTRUCTURA DE LA WEB	114
ACTIVIDAD 4. ESTRUCTURANDO MI PÁGINA WEB	120
PRÁCTICA 1. MI PRIMER PÁGINA WEB.....	122
LECTURA NO 5. ELEMENTOS BÁSICOS DE LAS PÁGINAS WEB	125





Educación que genera cambio

ACTIVIDAD 5. APRENDO A IDENTIFICAR LOS ELEMENTOS BÁSICOS: TÍTULOS, PÁRRAFOS, IMÁGENES, HIPERVÍNCULOS, VIDEOS, SONIDOS.....	131
PRÁCTICA 2. MI PÁGINA WEB CON MULTIMEDIA	133
PRÁCTICA 3. MEJORANDO MI PÁGINA WEB	136
LECTURA NO 6. ESTILOS: FUENTE, TAMAÑO Y COLORES	139
PRÁCTICA 4. ENCHULANDO MI WEB	144
LECTURA NO 7. USO DE TABLAS EN HTML.....	148
HTML ENCABEZADOS DE LA TABLA	154
PRÁCTICA 5. TABLAS EN HTML	155
LECTURA NO 8. USO DE CLASES, MENÚS Y FORMULARIOS EN HTML	160
ACTIVIDAD 6. CLASES, MENÚS Y FORMULARIOS	172
PRÁCTICA 6. CLASES, MENÚS Y FORMULARIOS.....	175
LECTURA NO 9. ¿QUÉ ES JAVASCRIPT?	179
ACTIVIDAD 7. ¿QUÉ ES JAVASCRIPT?	186
LECTURA NO 10. INSTRUCCIONES EN JAVASCRIPT	189
ACTIVIDAD 8. INSTRUCCIONES EN JAVASCRIPT	202
EJEMPLO GUIADO	203
LECTURA NO 11. ¿CÓMO PUBLICO MI PÁGINA WEB?	206
GREENBYTES – “GYTES VERDES”, TECNOLOGÍA AL SERVICIO DEL AMBIENTE	221
PRÁCTICA 7. PUBLICANDO MI PÁGINA WEB	223
ANEXOS	229



GUÍA DE ESTUDIANTES

Estimada comunidad estudiantil del Colegio de Bachilleres de Tabasco: quiero hacer énfasis en nuestra Institución, la cual tiene 5 décadas forjando el aprendizaje de cada uno de los jóvenes, hoy muchos de los que han pasado por nuestras aulas de clases son profesionistas exitosos.

Dentro del marco del 50 aniversario COBATAB, me permito como Director General enviarles un cordial saludo y dirigirme a ustedes a través de esta guía de estudio diseñada para acompañarlos como una herramienta en su formación académica y personal.

En esta etapa de su vida, el aprendizaje autónomo es clave para descubrir sus talentos, fortalecer sus habilidades y construir un futuro con propósito. Cada actividad, cada lectura y cada reto que encuentren en esta guía de estudio está pensado para impulsar su Desarrollo integral.

Les invito a asumir con responsabilidad su proceso de aprendizaje, a colaborar con sus compañeros, docentes y a mantener siempre una actitud de respeto y apertura. En el Colegio de Bachilleres de Tabasco promovemos la equidad y creemos firmemente en el potencial de cada uno de ustedes.

Esta guía de estudio busca ser clara y accesible, con objetivos definidos y contenidos relevantes para su contexto. Les animo a aprovecharla al máximo, a preguntar, reflexionar y aplicar lo aprendido en su vida diaria.

Confío en su capacidad para superar desafíos, alcanzar metas y contribuir positivamente a su comunidad. Cuentan con el apoyo de sus docentes y de todos los que integramos esta institución.

Con entusiasmo y compromiso, sigamos construyendo juntos una educación de calidad, que genera cambio.

Con aprecio y confianza,

Fernando Yrys Hernández
Director General
Colegio de Bachilleres de Tabasco



Presentación

Desde su creación, el Colegio de Bachilleres de Tabasco tiene como objetivo impartir educación de nivel medio superior en la modalidad de bachillerato general, operando desde el año 2009 un plan de estudios bajo un enfoque por competencias a través del Marco Curricular Común definiendo así los rasgos del perfil del egresado de la EMS en donde las competencias genéricas en su conjunto delinean el perfil deseable del joven bachiller.

En este sentido, los planteamientos del Marco Curricular Común de la Educación Media Superior (MCCEMS)¹ buscan una formación integral en el estudiantado mediante el desarrollo de la capacidad creadora, productiva, libre y digna del ser humano, con amor al país, a su cultura e historia. Por ello, el Bachillerato General plantea las diversas Unidades de Aprendizaje Curricular (UAC) y Formaciones para el Trabajo para que con sus estudiantes egresados y egresadas contribuya al logro de su objetivo específico, el cual radica en la “conformación de una ciudadanía reflexiva, con capacidad de formular y asumir responsabilidades de manera comunitaria, interactuar en contextos plurales y propositivos, trazarse metas y aprender de manera continua y colaborativa”.

El Colegio de Bachilleres de Tabasco a través de su Competente de Formación Laboral Básica ofrece en ocho de sus planteles la capacitación de **Desarrollo de Software** con la cual se busca desarrollar en el estudiantado competencias laborales básicas, que les permitan aplicar en forma integrada los conocimientos, destrezas, habilidades, actitudes y valores con responsabilidad y autonomía para desenvolverse en contextos específicos del desarrollo personal, académico, social y profesional en situaciones de la vida común, de estudio o trabajo a lo largo de la vida.

En ese contexto, conforme a las necesidades sociales, económicas y demográficas en relación y congruencia con los objetivos y líneas de acción del programa institucional para el Colegio de Bachilleres de Tabasco, se presenta la formación laboral básica en **Desarrollo de Software**, específica del bachillerato general, con objetivos delimitados acorde a las características del subsistema y la necesidad del conglomerado social en el cual será aplicado, situación que deviene

¹ El cual puede ser consultado a través del siguiente enlace:

<https://educacionmediasuperior.sep.gob.mx/work/models/sems/Resource/13516/1/image s/Documento%20base%20MCCEMS.pdf>



Educación que genera cambio

de tendencias académicas que obligan a este subsistema a tener en consideración las áreas de conocimiento y perfil profesional y su estrecha relación con la demanda ocupacional y la oferta laboral, esto con plena convicción de que los estudiantes del Colegio de Bachilleres de Tabasco concluyan satisfactoriamente con sus estudios de educación media superior e ingresen a sus estudios de nivel superior con herramientas y conocimientos que son de interés general conforme al área geográfica y entorno social en el que se desarrollen; por lo que el presente documento se encuentra conformado en apartados mediante los cuales describe la justificación y los elementos claves para su implementación en el aula.

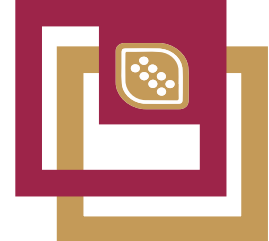
Programa de Estudios de la Formación Laboral Básica en Desarrollo de Software

Semestre	Tercero, Cuarto, Quinto y Sexto	
Créditos	56 totales / 14 por semestre	
Componente	De formación laboral	
Nivel de formación laboral	Básica	
Tiempo asignado	Mediación docente	Estudio independiente
	448 h. totales 112 h. por semestre	112 h. totales 28 h. por semestre
Sector productivo	Servicios profesionales, Científicos y Técnicos. ²	



² Acorde al RENECE – Registro Nacional de Estándares de Competencia por Sector Productivo. Disponible en <https://conocer.gob.mx/renece-registro-nacional-de-estandares-de-competencia-porsector-productivo/>





Fundamentación

La Dirección General del Colegio de Bachilleres de Tabasco en coordinación con la Dirección General del Bachillerato (DGB), acorde a la Nueva Escuela Mexicana (NEM) y al Marco Curricular Común de la Educación Media Superior (MCCEMS), y en su responsabilidad de enfrentar tanto los nuevos retos como los objetivos que de estos se desprenden, actualiza el presente Programa de estudios, el cual responde a una visión de educación integral, pertinente, de calidad y excelencia.

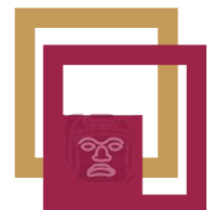
Dicho programa forma parte del Componente de Formación Laboral Básica del Bachillerato General, el cual busca ser un espacio vinculado con el sector productivo, permitiendo al estudiantado no solo cumplir con su trayecto educativo, sino construir su proyecto de vida, con mayores posibilidades de inserción en el mercado de trabajo.

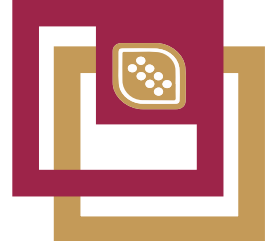
Esto atendiendo al mandato constitucional que en materia educativa, con base en la Reforma Constitucional a los artículos 3°, 31° y 73° de la Constitución Política de los Estados Unidos Mexicanos y de la emisión de la Legislación Secundaria, publicadas el 30 de septiembre de 2019 en el Diario Oficial de la Federación, determinan la reorientación del Sistema Educativo Nacional para “garantizar el derecho a la educación con un enfoque de derechos humanos y de igualdad sustantiva, para incidir en la cultura educativa mediante la corresponsabilidad y el impulso de transformaciones sociales dentro de la escuela y en la comunidad”.

Bajo este contexto, es que el Componente de Formación Laboral Básica, adquiere mayor relevancia, pues tendrá como ejes rectores:

- **Enfoque en competencias**, que busca desarrollar las capacidades y habilidades necesarias para el desempeño laboral.
- **Enfoque humanista**, que valora y respeta la diversidad y la dignidad del estudiantado, su potencial creativo, su participación, su bienestar integral y su compromiso social.

Estos enfoques se orientan a promover una educación inclusiva, equitativa y de calidad, que favorezca el aprendizaje permanente y el máximo logro de los aprendizajes.





Así pues, el Componente de Formación Laboral Básica, busca desarrollar en el estudiantado competencias laborales básicas, que les permitan aplicar en forma integrada los conocimientos, destrezas, habilidades, actitudes y valores con responsabilidad y autonomía para desenvolverse en contextos específicos del desarrollo personal, académico, social y profesional en situaciones de la vida común, de estudio o trabajo a lo largo de la vida.³

Para lograr dicho propósito, lo que a continuación se presenta es una serie de competencias laborales básicas las cuales permitirán al personal docente el abordaje de aprendizajes con la amplitud y profundidad acorde a los diversos contextos.

Es decir, a partir de estas competencias, se diseñarán estrategias de enseñanza-aprendizaje que permita a las y los estudiantes ser capaces de conducir su vida hacia su futuro con bienestar y satisfacción, con sentido de pertenencia social, conscientes de los problemas sociales, económicos y políticos que aquejan al país, pero también de su entorno inmediato, dispuestos a participar de manera responsable y decidida en los procesos de democracia participativa y a comprometerse en las soluciones de las problemáticas que los aquejan y que tengan la capacidad de aprender a aprender en el trayecto de su vida.⁴

Para lo cual, es de primordial importancia visualizar que debe existir una articulación contextualizada del Componente Laboral Básico, con el Currículum Fundamental y el Ampliado, para garantizar así la transferencia de los conocimientos, experiencias, habilidades, capacidades, actitudes y valores.

Es decir, esta articulación permitirá:

- La transversalidad del conocimiento adquirido en el Currículum Fundamental con las competencias laborales básicas requeridas en el mercado laboral.
- Fomentar el aprendizaje en contextos diversos, utilizando métodos y estrategias de aprendizaje.
- Centrar las necesidades del mercado laboral.

³ Subsecretaría de Educación Media Superior. (2023b). El currículum laboral en la educación media superior. SEP.

⁴ Subsecretaría de Educación Media Superior. (2023b). El currículum laboral en la educación media superior. SEP.



Educación que genera cambio

- Fomentar la interacción entre la vida educativa y la comunidad, a fin de propiciar la adquisición de conocimientos y competencias, de acuerdo con el desarrollo biopsicosociocultural del estudiantado.
- Aprovechar, mediante el Programa Aula, Escuela, Comunidad (PAEC), todas las oportunidades para poner en práctica lo que se ha aprendido; su aplicación no se limitará solo a los recursos sociocognitivos y a las áreas de conocimiento, sino que abarcará los aspectos funcionales (competencias laborales) y recursos socioemocionales.
- Mejorar la relación entre la escuela y los sectores productivos.⁵

El componente de Formación Laboral Básico aporta al estudiantado elementos que le permiten iniciarse en diversos aspectos del sector productivo, fomentando una actitud positiva hacia el trabajo y en su caso, su integración al mismo. Los módulos que conforman este programa son el resultado del trabajo colegiado con personal docente que imparte esta capacitación en los diferentes subsistemas coordinados por esta Dirección General, quienes brindan su experiencia y conocimientos buscando responder a los diferentes contextos existentes en el país, así como a la formación de una ciudadanía socialmente útil, para que el estudiantado cuente con la opción de iniciar una ruta laboral que le promueva una proyección hacia las diferentes modalidades laborales.

Aunado a ello, en virtud de que la Educación Media Superior debe favorecer la convivencia, el respeto a los derechos humanos y la responsabilidad social, el cuidado de las personas, el entendimiento del entorno, la protección del medio ambiente, la puesta en práctica de habilidades productivas para el desarrollo integral de los seres humanos, la actualización del presente programa de estudios, incluye temas transversales que según Figueroa de Katra (2005), enriquecen la labor formativa de manera tal que conectan y articulan los saberes de los distintos sectores de aprendizaje que dotan de sentido a los conocimientos disciplinares, con los temas y contextos sociales, culturales y éticos presentes en su entorno; buscan mirar toda la experiencia escolar como una oportunidad para que los aprendizajes integren sus dimensiones cognitivas y formativas, favoreciendo de esta forma una educación incluyente y con equidad.

⁵ Subsecretaría de Educación Media Superior. (2023b). El currículum laboral en la educación media superior. SEP.



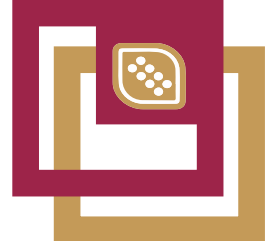
Educación que genera cambio

De igual forma, con base en el fortalecimiento de la educación para la vida, se abordan dentro de este programa de estudios los temas transversales, mismos que se clasifican a través de ejes temáticos de los campos Social, Ambiental, Salud y Habilidad Lectora como el componente básico, con la particularidad de que se complementan con características propias de la formación para el trabajo. Dichos temas no son únicos ni pretenden imitar el quehacer educativo en el aula, ya que es necesario tomar en consideración temas propios de cada comunidad, por lo que el personal docente podrá considerar ya sea uno o varios, en función del contexto escolar y de su pertinencia en cada submódulo:

- Eje transversal Emprendimiento: se sugiere retomar temas referentes a la detección de oportunidades y puesta en práctica de acciones que contribuyen a la demostración de actitudes tales como iniciativa, liderazgo, trabajo colaborativo, visión, innovación y creatividad promoviendo la responsabilidad social.
- Eje transversal Vinculación Laboral: se recomienda abordar temas referentes a la realización de acciones que permiten al estudiantado identificar los sitios de inserción laboral o autoempleo.
- Eje transversal Iniciar, Continuar y Concluir sus estudios de nivel superior: se recomienda abordar temas referentes a los mecanismos que permiten al estudiantado reflexionar sobre la importancia de darle continuidad a sus estudios superiores.

Asimismo, otro aspecto importante que promueve el programa de estudios es la interdisciplinariedad entre asignaturas del mismo semestre, en donde diferentes disciplinas se conjuntan para trabajar de forma colaborativa para la obtención de resultados en los aprendizajes esperados de manera integral permitiendo al estudiantado confrontarse a situaciones cotidianas aplicando dichos saberes de forma vinculada.

Por otro lado, en cada submódulo se observa la relación de las competencias genéricas y profesionales básicas, los conocimientos, las habilidades y actitudes que darán como resultado los aprendizajes esperados, permitiendo llevar de la mano al personal docente con el objetivo de generar un desarrollo progresivo no sólo de los conocimientos, sino también de aspectos actitudinales.

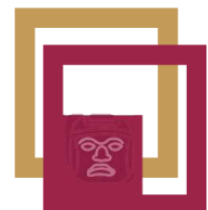


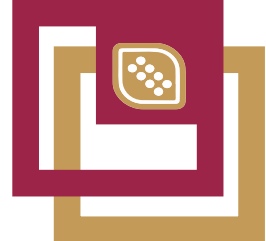
Educación que genera cambio

En este sentido, el rol docente dentro del proceso de enseñanza – aprendizaje, tiene un papel fundamental, como lo establece el Acuerdo Secretarial 4474, ya que el profesorado que imparte el componente de formación profesional, es quien facilita el proceso educativo al diseñar actividades significativas que promuevan el desarrollo de las competencias (conocimientos, habilidades y actitudes); propicia un ambiente de aprendizaje que favorece el conocimiento social, la colaboración, la toma responsable de decisiones y al perseverancia a través del desarrollo de habilidades socioemocionales del estudiantado, tales como la confianza, seguridad, autoestima, entre otras, propone estrategias disciplinares y transversales en donde el objetivo no es la formación de técnicos en diferentes actividades productivas, sino la promoción de las diferentes competencias profesionales básicas que permitan a la población estudiantil del Bachillerato General tener alternativas para iniciar una ruta a su integración laboral, favoreciendo el uso de herramientas tecnológicas de la información y la comunicación; así como el diseño de instrumentos de evaluación que atiendan al enfoque por competencias.

Es por ello que la Dirección General del Bachillerato a través del Trabajo colegiado busca promover una mejor formación docente a partir de la creación de redes de gestión escolar, analizar los indicadores del logro académico del estudiantado, generar técnicas exitosas de trabajo en el aula, compartir experiencias de manera asertiva, exponer problemáticas comunes que presenta el estudiantado respetando la diversidad de opiniones y mejorar la práctica pedagógica, donde es responsabilidad del profesorado realizar secuencias didácticas innovadoras a partir del análisis de los programas de estudio, promoviendo el desarrollo de habilidades socioemocionales y el abordaje de temas transversales de manera interdisciplinar; rediseñar las estrategias de evaluación y generar materiales didácticos.

Finalmente, este programa de estudios brinda herramientas disciplinares y pedagógicas al personal docente, quienes deberán, a través de los elementos antes mencionados, potenciar el papel de los educandos como gestores autónomos de su propio aprendizaje, promoviendo la participación creativa de las nuevas generaciones en la economía, en el ámbito laboral, la sociedad y la cultura, reforzar el proceso de formación de la personalidad, construir un espacio valioso para la adopción de valores y el desarrollo de actitudes positivas para la vida.





Educación que genera cambio

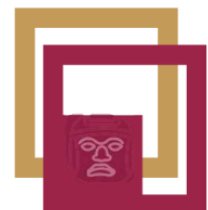
Justificación de la Formación Laboral Básica en Desarrollo de Software

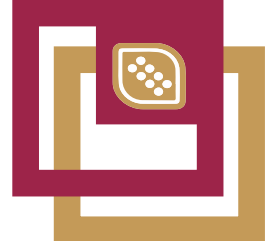
La Formación Laboral Básica de Desarrollo de Software se ubica dentro de los Recursos Sociocognitivos, dentro de las disciplinas de Comunicación y Matemáticas, las cuales en el MCCEMS tienen en el proceso de aprendizaje, la función de ampliar, potenciar y consolidar el conocimiento de la experiencia; permitiendo aprovechar y ampliar los conocimientos de las Áreas de Conocimiento, asimismo, los Recursos Sociocognitivos contribuyen a desarrollar capacidades, destrezas, habilidades, actitudes y valores en las y los estudiantes, brindando la posibilidad de construir la propia experiencia, para que sepan qué hacer con el conocimiento que tienen, sepan actuar, entendiendo lo que hacen, comprendiendo cómo participar y colaborar, asumiendo la responsabilidad de las acciones realizadas, sus implicaciones y consecuencias, y transformando los contextos locales y comunitarios en pro del bien común.

Con base en ello, a lo largo del Componente de Formación Laboral Básico, se brindará una formación de carácter genérico y transversal que le permita al estudiantado incorporarse al sector productivo con actividades contextualizadas, de autonomía y responsabilidad mínima, fomentando una actitud positiva hacia el trabajo y en su caso, su integración al mismo.

De manera específica, a través de la Formación Laboral Básica en Desarrollo de Software, tiene como eje acercar al estudiantado al desarrollo científico - tecnológico en el sector de los servicios profesionales, científicos y técnicos, como en las Tecnologías de la Información. Evaluando estrategias de solución a problemáticas en diversas áreas, utilizando software de programación de alto nivel, clasificando y diseñando el software mediante diversos lenguajes de programación, para mejorar su entorno, demostrando conciencia social y responsabilidad ante dichas problemáticas.

Con base en lo anterior, esta Formación Laboral Básica tiene el propósito general de: Evaluar estrategias para el desarrollo de software, de manera creativa e innovadora; identificando, clasificando y diseñando software mediante diversos lenguajes de programación, para mejorar su entorno, demostrando conciencia social y responsabilidad ante las problemáticas de su contexto, fomentando el trabajo colaborativo y propositivo.





Educación que genera cambio

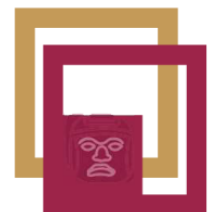
El uso de los lenguajes de programación en esta capacitación demuestra el manejo en forma avanzada de dichos lenguajes para la solución de diversas problemáticas de la vida cotidiana y del contexto, con un desarrollo metodológico, creativo y comunicativo en pro del beneficio personal y social.

De esta manera, las personas egresadas de esta Formación Laboral Básica en Desarrollo de Software se pueden integrar a la vida laboral tanto en instituciones públicas como privadas en los siguientes perfiles ocupacionales: en educación, ciberseguridad, visualización y gestión de datos, comercio electrónico, aplicaciones móviles, robótica y en el desarrollo web.

Los módulos que conforman este programa son el resultado del trabajo colegiado con personal docente que imparte esta capacitación en los diferentes subsistemas coordinados por esta Dirección General, quienes brindan su experiencia y conocimientos buscando responder a los diferentes contextos existentes en el país, así como a la formación de una ciudadanía socialmente útil, para que el estudiantado cuente con la opción de iniciar una ruta laboral que le promueva una proyección hacia las diferentes modalidades laborales.

La capacitación de Desarrollo de software busca que el estudiantado desarrolle las competencias profesionales en el desarrollo de sistemas y aplicaciones para Internet y celular, en donde también se desarrollan las competencias genéricas, la interdisciplinaridad y los ejes transversales de vinculación laboral, emprendimiento y la continuación de sus estudios a nivel superior.

La Capacitación de Desarrollo de Software en la formación para el trabajo del estudiantado está basada en las Normas Técnicas de Competencia Laboral (NTCL) del Consejo Nacional de Normalización y Certificación de Competencias Laborales (CONOCER), son una necesidad para cumplir con las exigencias del mundo actual y de los sectores productivos, porque hoy en día se exige tener trabajadores calificados, capaces de desarrollar en todo momento las áreas de la organización en la cual están inmersos, promoviendo los productos o servicios en el entorno nacional o internacional, proporcionando las herramientas y técnicas que son básicas para los egresados del nivel medio superior, que les va a permitir vencer todas las fronteras e incorporarse al mundo globalizado por medio de la programación, así como de las Tecnologías de la Información y de la Comunicación (TIC'S) y de la utilización de las Tecnologías del Aprendizaje y del Conocimiento (TAC'S).





Educación que genera cambio

Además, para consolidar la formación profesional y el compromiso con el medio ambiente, enfrentando los desafíos del mundo actual en materia de producción y consumo responsable; se consideran las metas de los 17 objetivos de la Agenda 2030 de acuerdo con la situación contextualizada de cada comunidad, para hacer más y mejores cosas con menos recursos, incrementando las ganancias netas de bienestar de las actividades económicas mediante la reducción de la utilización de los recursos, la degradación y la contaminación durante todo el ciclo de vida, logrando al mismo tiempo una mejor calidad de vida.⁶

De igual manera, con el desarrollo de las Tecnologías diversas y el desarrollo de habilidades como el pensamiento crítico, la toma de decisiones, la resolución de problemas, la comunicación efectiva y el trabajo colaborativo. La inteligencia artificial (IA) es la base a partir de la cual se imitan los procesos de inteligencia humana mediante la creación y la aplicación de algoritmos creados en un entorno dinámico de computación. Con esto se pretende que los ordenadores piensen y actúen como los humanos. Tomando tres componentes fundamentales:

- Sistemas computacionales.
- Datos y gestión de estos.
- Algoritmos para tomar decisiones (IA avanzados, código).⁷

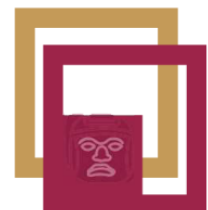
Con todo esto, la Formación Laboral Básica de Desarrollo de Software pretende atender las demandas de desarrollo en materia de cómputo y servicios a través de tecnologías aplicadas, así como el desarrollo científico y tecnológico para el progreso humano social y ambiental productivo.

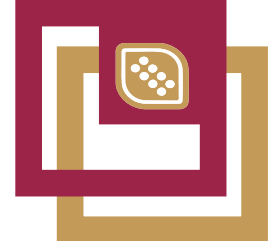
De esta manera, los estudiantes de Desarrollo de Software pueden proporcionar soluciones de software: aplicaciones, programas y sistemas que permitan atender las necesidades sociales y ambientales para bien de todos. Consolidando las competencias laborales en torno a la situación del contexto de los estudiantes, así como del país.

El Desarrollo de Software considera la metodología STEAM, donde los estudiantes de la capacitación pueden desarrollar proyectos que integran las disciplinas de ciencia, tecnología, ingeniería, arte y matemáticas (STEAM). Con la particularidad de la interdisciplinariedad vinculando los saberes y conocimientos en la vinculación con las instituciones de formación profesional universitarias.

⁶ La Agenda 2030 y los Objetivos de Desarrollo Sostenible. Una oportunidad para América Latina y el Caribe.

⁷ Inteligencia Artificial aplicada a la solución de problemas nacionales. Consuelo Doddoli, Ciencia UNAM-DGDC.





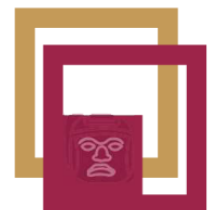
Enfoque de la Formación Laboral Básica.

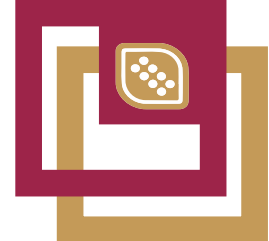
La Formación Laboral Básica de Desarrollo de Software se ubica dentro de los Recursos Sociocognitivos, dentro de las disciplinas de Comunicación y Matemáticas, las cuales en el MCCEMS tienen en el proceso de aprendizaje, la función de ampliar, potenciar y consolidar el conocimiento de la experiencia; permitiendo aprovechar y ampliar los conocimientos de las Áreas de Conocimiento, asimismo, los Recursos Sociocognitivos contribuyen a desarrollar capacidades, destrezas, habilidades, actitudes y valores en las y los estudiantes, brindando la posibilidad de construir la propia experiencia, para que sepan qué hacer con el conocimiento que tienen, sepan actuar, entendiendo lo que hacen, comprendiendo cómo participar y colaborar, asumiendo la responsabilidad de las acciones realizadas, sus implicaciones y consecuencias, y transformando los contextos locales y comunitarios en pro del bien común.

Con base en ello, a lo largo del Componente de Formación Laboral Básico, se brindará una formación de carácter genérico y transversal que le permita al estudiantado incorporarse al sector productivo con actividades contextualizadas, de autonomía y responsabilidad mínima, fomentando una actitud positiva hacia el trabajo y en su caso, su integración al mismo.

Con base en ello, a lo largo del Componente de Formación Laboral Básico, se brindará una formación de carácter genérico y transversal que le permita al estudiantado incorporarse al sector productivo con actividades contextualizadas, de autonomía y responsabilidad mínima, fomentando una actitud positiva hacia el trabajo y en su caso, su integración al mismo.

De manera específica, a través de la Formación Laboral Básica en Desarrollo de Software, tiene como eje acercar al estudiantado al desarrollo científico - tecnológico en el sector de los servicios profesionales, científicos y técnicos, como en las Tecnologías de la Información. Evaluando estrategias de solución a problemáticas en diversas áreas, utilizando software de programación de alto nivel, clasificando y diseñando el software mediante diversos lenguajes de programación, para mejorar su entorno, demostrando conciencia social y responsabilidad ante dichas problemáticas.





Con base en lo anterior, esta Formación Laboral Básica tiene el **propósito general** de: Evaluar estrategias para el desarrollo de software, de manera creativa e innovadora; identificando, clasificando y diseñando software mediante diversos lenguajes de programación, para mejorar su entorno, demostrando consciencia social y responsabilidad ante las problemáticas de su contexto, fomentando el trabajo colaborativo y propositivo.

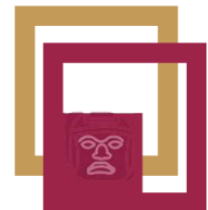
El uso de los lenguajes de programación en esta capacitación demuestra el manejo en forma avanzada de dichos lenguajes para la solución de diversas problemáticas de la vida cotidiana y del contexto, con un desarrollo metodológico, creativo y comunicativo en pro del beneficio personal y social.

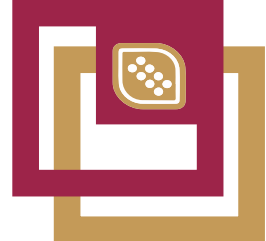
De esta manera, las personas egresadas de esta Formación Laboral Básica en Desarrollo de Software se pueden integrar a la vida laboral tanto en instituciones públicas como privadas en los siguientes perfiles ocupacionales: en educación, ciberseguridad, visualización y gestión de datos, comercio electrónico, aplicaciones móviles, robótica y en el desarrollo web.

Los módulos que conforman este programa son el resultado del trabajo colegiado con personal docente que imparte esta capacitación en los diferentes subsistemas coordinados por esta Dirección General, quienes brindan su experiencia y conocimientos buscando responder a los diferentes contextos existentes en el país, así como a la formación de una ciudadanía socialmente útil, para que el estudiantado cuente con la opción de iniciar una ruta laboral que le promueva una proyección hacia las diferentes modalidades laborales.

La capacitación de Desarrollo de software busca que el estudiantado desarrolle las competencias profesionales en el desarrollo de sistemas y aplicaciones para Internet y celular, en donde también se desarrollan las competencias genéricas, la interdisciplinaridad y los ejes transversales de vinculación laboral, emprendimiento y la continuación de sus estudios a nivel superior.

La Capacitación de Desarrollo de Software en la formación para el trabajo del estudiantado está basada en las Normas Técnicas de Competencia Laboral (NTCL) del Consejo Nacional de Normalización y Certificación de Competencias Laborales (CONOCER), son una necesidad para cumplir con las exigencias del mundo actual y de los sectores productivos, porque hoy en día se exige tener trabajadores calificados, capaces de desarrollar en todo momento las áreas de la organización en la cual están inmersos, promoviendo los productos o servicios en el entorno





Educación que genera cambio

nacional o internacional, proporcionando las herramientas y técnicas que son básicas para los egresados del nivel medio superior, que les va a permitir vencer todas las fronteras e incorporarse al mundo globalizado por medio de la programación, así como de las Tecnologías de la Información y de la Comunicación (TIC'S) y de la utilización de las Tecnologías del Aprendizaje y del Conocimiento (TAC'S).

Además, para consolidar la formación profesional y el compromiso con el medio ambiente, enfrentando los desafíos del mundo actual en materia de producción y consumo responsable; se consideran las metas de los 17 objetivos de la Agenda 2030 de acuerdo con la situación contextualizada de cada comunidad, para hacer más y mejores cosas con menos recursos, incrementando las ganancias netas de bienestar de las actividades económicas mediante la reducción de la utilización de los recursos, la degradación y la contaminación durante todo el ciclo de vida, logrando al mismo tiempo una mejor calidad de vida.⁸

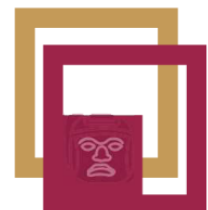
De igual manera, con el desarrollo de las Tecnologías diversas y el desarrollo de habilidades como el pensamiento crítico, la toma de decisiones, la resolución de problemas, la comunicación efectiva y el trabajo colaborativo. La inteligencia artificial (IA) es la base a partir de la cual se imitan los procesos de inteligencia humana mediante la creación y la aplicación de algoritmos creados en un entorno dinámico de computación. Con esto se pretende que los ordenadores piensen y actúen como los humanos. Tomando tres componentes fundamentales:

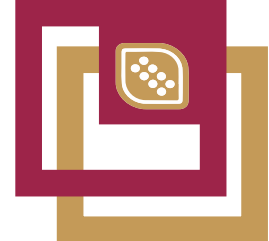
- Sistemas computacionales.
- Datos y gestión de estos.
- Algoritmos para tomar decisiones (IA avanzados, código).⁹

Con todo esto, la Formación Laboral Básica de Desarrollo de Software pretende atender las demandas de desarrollo en materia de cómputo y servicios a través de tecnologías aplicadas, así como el desarrollo científico y tecnológico para el progreso humano social y ambiental productivo.

⁸ La Agenda 2030 y los Objetivos de Desarrollo Sostenible. Una oportunidad para América Latina y el Caribe.

⁹ Inteligencia Artificial aplicada a la solución de problemas nacionales. Consuelo Doddoli, Ciencia UNAM-DGDC.





De esta manera, los estudiantes de Desarrollo de Software pueden proporcionar soluciones de software: aplicaciones, programas y sistemas que permitan atender las necesidades sociales y ambientales para bien de todos. Consolidando las competencias laborales en torno a la situación del contexto de los estudiantes, así como del país.

El Desarrollo de Software considera la metodología STEAM, donde los estudiantes de la capacitación pueden desarrollar proyectos que integran las disciplinas de ciencia, tecnología, ingeniería, arte y matemáticas (STEAM). Con la particularidad de la interdisciplinariedad vinculando los saberes y conocimientos en la vinculación con las instituciones de formación profesional universitarias.

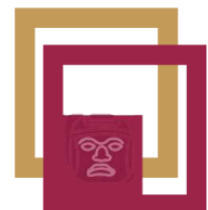
El contenido de la capacitación de Desarrollo de software se divide en cuatro módulos, impartidos a partir del tercer semestre con una carga de 7 horas semanales, cada módulo se integra por dos submódulos en los que se busca desarrollar en el estudiantado la creación de programas con características avanzadas, utilizando C++, Java, Visual .NET, así como la creación de aplicaciones móviles y juegos, además de diseño y programación en robótica, esto con el fin de desarrollar software con bases de datos y creación de páginas web comunicando ideas e información en el entorno laboral y escolar.

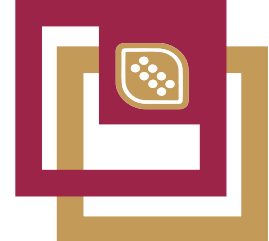
El Módulo I. Programación básica, el estudiantado revisará sistemas de información mediante software de programación de alto nivel.

El Módulo II. Base de datos y desarrollo web, en este apartado se analizará la creación de bases de datos y sitios web mediante el software de programación adecuado.

El Módulo III. Aplicaciones para web, el estudiantado utilizará software de aplicación y elementos multimedia para crear sitios web.

El Módulo IV. Aplicaciones móviles, en este apartado se construirá soluciones para aplicaciones móviles -y juegos, así como propuestas para robótica básica respetando su arquitectura y anatomía. Todas las competencias mencionadas hacen posible en el egresado tener los conocimientos, técnicas, métodos y lenguajes necesarios en el desarrollo de software para ingresar a estudios superiores y desempeñarse de forma eficiente, además de desarrollar las habilidades y actitudes necesarias para la realización de una actividad productiva socialmente útil como auxiliar en áreas de desarrollo de software en diferentes instituciones públicas o privadas.





La Capacitación de Desarrollo de Software en la formación laboral básica del estudiantado está basada en las Normas Técnicas de Competencia Laboral (NTCL) del Consejo Nacional de Normalización y Certificación de Competencias Laborales (CONOCER), son una necesidad para cumplir con las exigencias del mundo actual y de los sectores productivos, porque hoy en día se exige tener trabajadores calificados, capaces de desarrollar en todo momento las áreas de la organización en la cual están inmersos, promoviendo los productos o servicios en el entorno nacional o internacional, proporcionando las herramientas y técnicas que son básicas para los egresados del nivel medio superior, que les va a permitir vencer todas las fronteras e incorporarse al mundo globalizado por medio de la programación, así como de las Tecnologías de la información y de la comunicación (TIC'S) y de la utilización de las Tecnologías del aprendizaje y del conocimiento (TAC'S).

Así mismo se señalan algunas normas de competencia laboral vigentes, avaladas por el Consejo Nacional de Normalización y Certificación de Competencias Laborales, que podrán servir de guía para el desarrollo de los módulos:

NMX-J-515-ANCE-2021, Equipos De Control Y Distribución-Requisitos Generales De Seguridad.

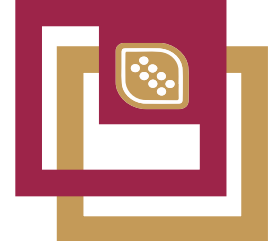
NOM-003-SCFI-2014 "Productos Eléctricos – especificaciones de seguridad".

UINF0947.01 Operar las herramientas de cómputo en ambiente de red. **EC0835** Ejecución de software con codificación de comandos y datos orientada a objetos. **EC0160** Desarrollo de código de software.



Ubicación de la Formación Laboral

PRIMER SEMESTRE					SEGUNDO SEMESTRE					TERCER SEMESTRE					CUARTO SEMESTRE					QUINTO SEMESTRE					SEXTO SEMESTRE								
UNIDAD DE APRENDIZAJE CURRICULAR	HD	HI	HT	C	UNIDAD DE APRENDIZAJE CURRICULAR	HD	HI	HT	C	UNIDAD DE APRENDIZAJE CURRICULAR	HD	HI	HT	C	UNIDAD DE APRENDIZAJE CURRICULAR	HD	HI	HT	C	UNIDAD DE APRENDIZAJE CURRICULAR	HD	HI	HT	C	UNIDAD DE APRENDIZAJE CURRICULAR	HD	HI	HT	C				
LA MATERIA Y SUS INTERACCIONES	4	1	5	8	CONSERVACIÓN DE LA ENERGÍA Y SUS INTERACCIONES CON LA MATERIA	4	1	5	8	ECOSISTEMAS: INTERACCIONES, ENERGÍA Y DINÁMICA	4	1	5	8	REACCIONES QUÍMICAS: CONSERVACIÓN DE LA MATERIA EN LA FORMACIÓN DE NUEVAS SUSTANCIAS	4	1	5	8	LA ENERGÍA EN LOS PROCESOS DE LA VIDA DIARIA	4	1	5	8	ORGANISMOS: ESTRUCTURAS Y PROCESOS. HERENCIA Y EVOLUCIÓN BIOLÓGICA	4	1	5	8				
CIENCIAS SOCIALES I	2	0.5	2.5	4	CIENCIAS SOCIALES II	2	0.5	2.5	4						CONCIENCIA HISTÓRICA I. PERSPECTIVAS DEL MÉXICO ANTIGUO. LOS CONTEXTOS GLOBALES	3	0.75	3.75	6	CONCIENCIA HISTÓRICA II. MÉXICO DURANTE EL EXPANSIONISMO CAPITALISTA	3	0.75	3.75	6	CONCIENCIA HISTÓRICA III. LA REALIDAD ACTUAL EN PERSPECTIVA HISTÓRICA	3	0.75	3.75	6				
CULTURA DIGITAL I	3	0.75	3.75	6	CULTURA DIGITAL II	2	0.5	2.5	4						* TALLER DE CULTURA DIGITAL	1	0.25	1.25	2														
PENSAMIENTO MATEMÁTICO I	4	1	5	8	PENSAMIENTO MATEMÁTICO II	4	1	5	8						PENSAMIENTO MATEMÁTICO III	4	1	5	8														
LENGUA Y COMUNICACIÓN I	3	0.75	3.75	6	LENGUA Y COMUNICACIÓN II	3	0.75	3.75	6	LENGUA Y COMUNICACIÓN III	3	0.75	3.75	6	* PENSAMIENTO LITERARIO	3	0.75	3.75	6		3	0.75	3.75	6	**	3	0.75	3.75	6				
INGLÉS I	3	0.75	3.75	6	INGLÉS II	3	0.75	3.75	6	INGLÉS III	3	0.75	3.75	6	INGLÉS IV	3	0.75	3.75	6	**	3	0.75	3.75	6	**	3	0.75	3.75	6				
HUMANIDADES I	4	1	5	8	HUMANIDADES II	4	1	5	8	HUMANIDADES III	5	1.25	6.25	10	* ESPACIO Y SOCIEDAD	3	0.75	3.75	6	**	3	0.75	3.75	6	**	3	0.75	3.75	6				
* LABORATORIO DE INVESTIGACIÓN	3	0.75	3.75	6	* TALLER DE CIENCIAS I	4	1	5	8	* TALLER DE CIENCIAS II	3	0.75	3.75	6	CIENCIAS SOCIALES III	2	0.5	2.5	4	**	3	0.75	3.75	6	**	3	0.75	3.75	6				
										✓ PROGRAMACIÓN EN C++	3	0.75	3.75	6	✓ DESARROLLO DE APLICACIONES DE ESCRITORIO CON BASE DE DATOS	3	0.75	3.75	6	**	3	0.75	3.75	6	**	3	0.75	3.75	6				
										✓ PROGRAMACIÓN EN JAVA	4	1	5	8	✓ PROGRAMACIÓN WEB	4	1	5	8	✓ APLICACIONES PARA WEB	3	0.75	3.75	6	✓ PROGRAMACIÓN MÓVIL Y JUEGOS	3	0.75	3.75	6				
- CURRÍCULUM AMPLIADO					- CURRÍCULUM AMPLIADO					- CURRÍCULUM AMPLIADO					- CURRÍCULUM AMPLIADO					✓ PROGRAMACIÓN PARA WEB					✓ DISEÑO Y PROGRAMACIÓN EN ROBÓTICA								



Mapa de la Capacitación Desarrollo de Software

Módulo II: Base de Datos y Desarrollo Web

Módulo I: Programación Básica

Submódulo 1	Programación en C++
	48 horas de mediación docente 12 horas de estudio independiente 6 créditos
Submódulo 2	Programación en Java
	64 horas de mediación docente 16 horas de estudio independiente 8 créditos

Nota: Elaboración propia.

Submódulo 1	Desarrollo de Aplicaciones de Escritorio con Base de Datos
	48 horas de mediación docente 12 horas de estudio independiente 6 créditos
Submódulo 2	Programación Web
	64 horas de mediación docente 16 horas de estudio independiente 8 créditos

Nota: Elaboración propia.

Módulo III: Aplicaciones Web

Submódulo 1	Aplicaciones para Web
	48 horas de mediación docente 12 horas de estudio independiente 6 créditos
Submódulo 2	Herramientas de Diseño para Web
	64 horas de mediación docente 16 horas de estudio independiente 8 créditos

Nota: Elaboración propia.

Módulo IV: Aplicaciones Móviles

Submódulo 1	Programación Móvil y Juegos
	48 horas de mediación docente 12 horas de estudio independiente 6 créditos
Submódulo 2	Diseño y Programación en Robótica
	64 horas de mediación docente 16 horas de estudio independiente 8 créditos





Educación que genera cambio

Competencias Laborales Básicas

Hacen referencia a la capacidad para aplicar conocimientos, destrezas, habilidades, actitudes y valores en el desarrollo personal, académico, social y profesional en situaciones de la vida común, de estudio o trabajo para que el estudiantado desarrolle la formación fundamental o laboral básica, que les permite desempeñar **funciones laborales de nivel dos** de competencia, aplicando soluciones a problemas simples en contextos conocidos y específicos.

Tienen validez oficial dentro del Sistema Educativo Nacional (SEN), lo cual se expresa con la emisión del documento que acredita su formación.

Estas competencias se caracterizan por:

- **Pertinencia:** Atiende a las necesidades del sector productivo y son valoradas.
- **Relevancia:** Favorece la empleabilidad y emprendimientos productivos, sin disparidades de género, étnicas o exclusión de grupos vulnerables.
- **Coherencia:** Acorde al tipo educativo de media superior.
- **Suficiencia:** Abarca un proceso completo e independiente a las demás funciones que desempeña quien egresa de la formación para el trabajo, técnica o tecnológica en el sector productivo.
- **Autonomía:** Faculta para el análisis y toma de decisiones.
- **Responsabilidad:** Capacidad para asumir compromisos orientados al logro de objetivos y metas laborales.
- **Variedad:** Abarca la ejecución de actividades rutinarias – no rutinarias, predecibles – impredecibles – contextos diversos.
- **Complejidad:** Moviliza recursos cognitivos, procedimentales y actitudinales en diferentes niveles para la ejecución de actividades y funciones.



Educación que genera cambio

De igual forma es importante señalar, que, para el caso de la Formación Laboral Básica, como se señaló anteriormente, se logrará en el estudiantado el nivel de competencia dos, el cual se caracteriza por:

- ✓ Realización de actividades programadas
- ✓ Aplicación de habilidades cognitivas y de comunicación para recibir, transmitir y recordar información
- ✓ Utiliza técnicas, materiales, herramientas y equipamiento que no requieren un nivel de especialización para realizar actividades en contextos conocidos, además del uso de tecnologías de la información y comunicación básicas, actuando con ética, con un enfoque de sostenibilidad y responsabilidad sobre su entorno.¹⁰

1. Descubre sistemas de información mediante la codificación y compilación de instrucciones pertinentes utilizando software de programación, de forma creativa y responsable, en diversos ámbitos.

CLBDS1.

2. Predice sistemas de información a través de sus fundamentos, utilizando software de programación que cumpla con los requerimientos de funcionalidad y rendimiento en diferentes contextos, siendo creativo y reflexionando sobre las consecuencias que se deriven de su toma de decisiones.

CLBDS2.

3. Estructura bases de datos relacionadas con situaciones sociales de su comunidad utilizando la programación y mostrando un comportamiento metódico, además de organizado, con la finalidad de promover un pensamiento crítico y analítico.

CLBDS3.

4. Construye sitios web creativos y funcionales mediante software de diseño web, para transmitir información a gran escala de forma responsable y empática en diversos contextos.

CLBDS4.

5. Prueba la funcionalidad de los sitios web, acorde a las necesidades y requerimientos de los usuarios, para el manejo de datos a gran escala, actuando con eficiencia y reflexionando sobre diferentes posturas de conducirse en el contexto.

CLBDS5.

¹⁰ Subsecretaría de Educación Media Superior. (2023b). El currículum laboral en la educación media superior. SEP.



6. Explica la creación de sitios web creativos a través de la utilización de software de aplicación y elementos multimedia para la optimización de páginas web, favoreciendo la solución a diversas situaciones de su entorno, actuando de manera consciente y previniendo riesgos.

CLBDS6.

7. Recomienda soluciones mediante software de aplicación, para aplicaciones móviles y juegos, relacionándose con sus semejantes de forma colaborativa, mostrando disposición al trabajo metódico y organizado en diferentes contextos.

CLBDS7.

8. Selecciona propuestas de robótica básica, a través de sus fundamentos, arquitectura y anatomía, en cualquier ámbito, afrontando retos y asumiendo la frustración como parte del proceso.

CLBDS8.

Así mismo se señalan algunas normas de competencia laboral vigentes, avaladas por el Consejo Nacional de Normalización y Certificación de Competencias Laborales, que podrán servir de guía para el desarrollo de los módulos:

NMX-J-515-ANCE-2021, Equipos De Control Y Distribución-Requisitos Generales De Seguridad.

NOM-003-SCFI-2014 "Productos Eléctricos – especificaciones de seguridad".





Educación que genera cambio

Proceso de Evaluación bajo el Enfoque en Competencias

La evaluación por competencias es un proceso que permitirá mediante la obtención de evidencias conocer el dominio de conocimientos, habilidades y actitudes socioafectivas desarrolladas por el estudiantado.

Dicho proceso deberá ser formativo e integral, es decir, permitirá visualizar no solo el saber, saber hacer y saber ser, sino también el bagaje histórico y cultural lo cual permitirá al estudiantado la comprensión de la realidad social y laboral de los sectores y de la comunidad para a partir de ello lograr su intervención y aporte.

Para ello lo que a continuación se presentan son los principios que orientarán el proceso de evaluación:

1. **Validez:** debe existir correlación entre los resultados de la evaluación y los resultados esperados en situaciones laborales reales.
2. **Confiabilidad:** producir resultados consistentes al evaluar en momentos diferentes y en diversos contextos.
3. **Accesibilidad:** facilitar el acceso a cualquier persona que pueda ser capaz de demostrar el desarrollo de la competencia.
4. **Comunicación:** dar a conocer previamente las condiciones en que se va a evaluar, y posteriormente, los resultados mediante la retroalimentación.
5. **Equidad:** evitar cualquier práctica discriminatoria, es decir, el estudiantado será evaluado bajo los mismos criterios e indicadores.
6. **Flexibilidad:** adaptarse a diferentes modalidades y opciones de formación, así como a las características y necesidades del estudiantado.¹¹

Así pues para poder llevar a cabo lo aquí planteado, se propone la utilización de una amplia gama de instrumentos que permitan visualizar tanto el proceso como el resultado final del aprendizaje

¹¹ Subsecretaría de Educación Media Superior. (2023). El currículum laboral en la educación media superior. SEP.



Educación que genera cambio

del estudiantado, entre los que se encuentran: rúbricas, pruebas de ejecución, portafolios de evidencias, diario de campo o bitácora, organizadores gráficos, ensayos, resolución de ejercicios y problemas, exámenes o pruebas tipo saber, exposición, método de casos, proyectos y debates o discusiones dirigidas, entre otros.

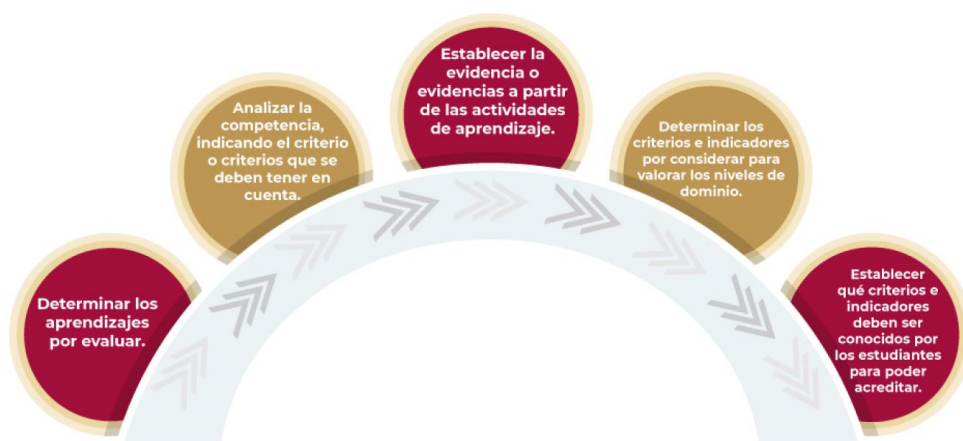
De igual forma, es necesario que la evaluación contemple:

- Autoevaluación: cuando el estudiantado valora su desarrollo y la forma en que aprendió.
- Coevaluación: a través de la retroalimentación entre pares, fomentando la cooperación, la colaboración, la empatía y la crítica constructiva.
- Heteroevaluación: emitida por el personal docente en función de los conocimientos, habilidades, actitudes y valores ponderados en los instrumentos de evaluación.



Finalmente, respecto a los pasos para evaluar las competencias laborales básicas, se presenta a continuación una propuesta elaborada por la Subsecretaría de Educación Media Superior (2023b), la cual ilustra la ruta a seguir y constituye una referencia para el personal docente.

Pasos para evaluar competencias laborales



Fuente: Subsecretaría de Educación Media Superior, 2023b.





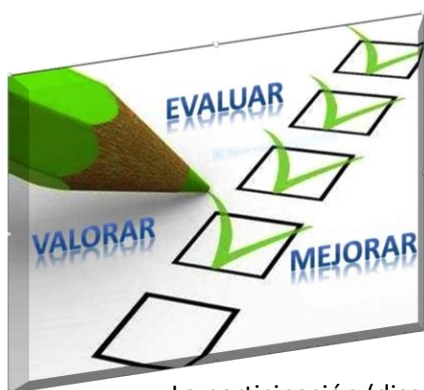
Educación que genera cambio

Evaluación por Competencias

Para evaluar por competencias, se debe favorecer el proceso de formación a través de:

- **La Evaluación Diagnóstica:** se realiza antes de algún proceso educativo (curso, secuencia o segmento de enseñanza) para estimar los conocimientos previos del estudiantado, identificar sus capacidades cognitivas con relación al programa de estudios y apoya al personal docente en la toma de decisiones para el trabajo en el aula.
- **La Evaluación Formativa:** se lleva a cabo durante el proceso educativo y permite precisar los avances logrados en el desarrollo de competencias por cada estudiante y advierte las dificultades que encuentra durante el aprendizaje. Tiene por objeto mejorar, corregir o reajustar su avance y se fundamenta, en parte, en la autoevaluación. Implica una reflexión y un diálogo con el estudiantado acerca de los resultados obtenidos y los procesos de aprendizaje y enseñanza que le llevaron a ello; permite estimar la eficacia de las experiencias de aprendizaje para mejorarlas y favorece su autonomía.
- **La Evaluación Sumativa:** se realiza al final de un proceso o ciclo educativo considerando el conjunto de diversas evidencias que surgen de los aprendizajes logrados.

Instrumentos de Evaluación



Con el fin de que el estudiantado muestre el saber hacer que subyace en una competencia, las Sugerencias de Evidencia de Evaluación permiten establecer una estrategia de evaluación, por lo tanto, contienen elementos observables que deben ser considerados en la evaluación tales como:

La participación (discurso y comunicación, compromiso, empeño e iniciativa, cooperación).

- Las actividades generativas (trabajo de campo, proyectos, solución de casos y problemas, composición de textos, arte y dramatizaciones).





Educación que genera cambio

- Las actividades de análisis (comprensión e integración de conceptos como interpretación, síntesis y clasificación, toma de decisiones, juicio y evaluación, creación e invención y pensamiento crítico e indagación).

Para ello se consideran instrumentos que pueden agruparse principalmente en (Díaz-Barriga, 2014):

- Rúbricas:** Son guías que describen las características específicas de lo que se pretende evaluar (productos, tareas, proyectos, exposiciones, entre otras) precisando los niveles de rendimiento que permiten evidenciar los aprendizajes logrados de cada estudiante, valorar su ejecución y facilitar la retroalimentación.
- Portafolios:** permiten mostrar el crecimiento gradual y los aprendizajes logrados con relación al programa de estudios, centrándose en la calidad o nivel de competencia alcanzado y no en una mera colección al azar de trabajos sin relación. Estos establecen criterios y estándares para elaborar diversos instrumentos para la evaluación del aprendizaje ponderando aspectos cualitativos de lo cuantitativo.

Los trabajos que se pueden integrar en un portafolio y que pueden ser evaluados a través de rúbricas son: ensayos, videos, series de problemas resueltos, trabajos artísticos, trabajos colectivos, comentarios a lecturas realizadas, autorreflexiones, reportes de laboratorio, hojas de trabajo, guiones, entre otros, los cuales deben responder a una lógica de planeación o proyecto.

Con base en lo anterior, los programas de estudio de la Dirección General del Bachillerato al incluir elementos que enriquecen la labor formativa tales como la transversalidad, las habilidades socioemocionales y la interdisciplinariedad trabajadas de manera colegiada y permanentemente en el aula, consideran a la evaluación formativa como eje central al promover una reflexión sobre el progreso del desarrollo de competencias del alumnado. Para ello, es necesario que el personal docente brinde un acompañamiento continuo con el propósito de mejorar, corregir o reajustar el logro del desempeño del bachiller sin esperar la conclusión del semestre para presentar una evaluación final.





Rol del Estudiantado

El rol del estudiantado en el proceso educativo no se limita simplemente a recibir información y repetirla, sino que debe ser un agente activo en la construcción de su propio conocimiento y de su identidad. En este sentido, no sólo se trata de aprender a leer y escribir; implica aprender a narrar y comprender su propia vida, tanto como autor o autora de su historia personal, como testigo de su contexto social y cultural. Este proceso es fundamental para que el estudiantado se convierta en un sujeto consciente y crítico de su realidad.

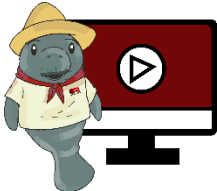
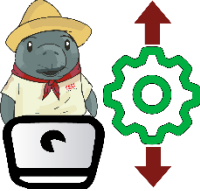
La educación es un motor de transformación social, pero también puede perpetuar las desigualdades existentes al tratar a todos y todas por igual sin considerar la diversidad inherente al estudiantado. La educación debe empoderarles, dándoles las condiciones necesarias para reconocer y cuestionar las desigualdades que les rodean.

Si las y los estudiantes son insertados en una educación que no considera su clase, sexo, género, etnia, lengua, cultura, capacidad, condición migratoria, religión o cualquier otro aspecto de su identidad, es muy probable que se apropien de la idea de que “la escuela no es para ellos y ellas”, ya que se enfrentarían constantemente a comentarios o actitudes que les califican de incapaces, ignorantes, indolentes o inútiles terminando por creerlo y asumirlo como verdad. Esta autodesvalorización es una barrera significativa para su desarrollo ya que puede llevar a creer que el conocimiento y la sabiduría pertenecen únicamente a las y los “profesionales” y no reconocen el valor de su propio conocimiento y experiencia.

El rol de las y los estudiantes, entonces, debe ser el de un sujeto activo que desafía y transforma estas narrativas opresivas que fomentan las desigualdades. Debe aprender a valorar su propia voz y experiencia, y a reconocer su capacidad para conocer y transformar su realidad. La educación debe ser un proceso liberador que les permita verse a sí mismos o mismas como agentes de transformación social, capaces de escribir su propia historia y de participar activamente en la construcción de una sociedad más justa y humana.



SIMBOLOGÍA

Icono	Descripción
	Lectura: Indica que se realizará la lectura que contiene información de los conocimientos básicos del Programa de Estudio.
	Material audio visual: Representa recursos adicionales al contenido de los conocimientos básicos del Programa de Estudio a través de un video mostrado por el docente, indicado por QR.
	Actividad: Es momento de realizar una actividad teórica – escrita que contribuye a evidenciar el propósito del aprendizaje esperado de los conocimientos básicos del Programa de Estudio.
	Práctica: Hagamos la práctica guiada, que contribuye a la aplicación de los saberes obtenidos con relación al conocimiento básico del Programa de Estudio, se acompaña con el instrumento de evaluación correspondiente.
	Docente explica: Se sugiere que esta sección el docente profundice los conocimientos para adecuarlos a su contexto.
	Actividad SIGA: Indica el producto que se contempla para la plataforma SIGA.



Icono	Descripción
	Evaluación Diagnóstica: Presenta el momento para llevar a cabo la evaluación diagnóstica del submódulo.
	Instrumento de evaluación: Documento en el que tanto el estudiante como el docente observan los criterios con los que se evaluará el producto a realizar.
	Encuadre: Presenta cada aspecto y el porcentaje de calificación con el que se evaluará el submódulo.
	Dosificación: Muestra las fechas en el que los conocimientos básicos del Programa de Estudio se van a desarrollar.
	PAEC: Proyecto Escolar Comunitario
	Cuadernillo: Estudio Independiente

Temario

Submódulo I. Desarrollo de Aplicaciones de Escritorio con Base de Datos

Introducción a las bases de datos.

- Fundamentos de las bases de datos.
- Diferencia entre hojas de cálculo y base de datos.

Instalación del software manejador de base de datos.

Tipos de base de datos.

Estructura de base de datos relacionales.

- Campos.
- Registros.
- Tablas.
- Redundancia de datos.

Comandos de menú y SQL en el manejo de registros para:

- Insertar.
- Borrar.
- Modificar.
- Consultar.

Creación y actualización de base de datos.



Submódulo II. Programación Web

Diseño web.

- Conceptos básicos: Página web, sitio, dominios, URL, hosting, applets.
- Elementos de una página web.
- Estándares de diseño web (W3C).

HTML y CSS

- Estructura básica de una página web: html, head, title, body.
- Elementos básicos de una página web: títulos, párrafos, imágenes, hipervínculos, vídeo, sonido, fuentes, tamaño, colores.
- Estructuras en una página web: tablas, divs, clases, menús, artículos, secciones, formularios, márgenes y posición de los elementos en la página web.

Desarrollo de páginas web con software de aplicación.

Publicación de páginas web.





Submódulo I. Desarrollo de Aplicaciones de Escritorio con Base de Datos



Propósito del Módulo

Elaborar diferentes bases de datos de forma responsable, mediante software de programación para desarrollar soluciones a diferentes problemáticas del contexto y desarrollar sitios web creativos, funcionales mediante software de diseño, que coadyuven en la transmisión de información en la vida cotidiana, mostrando un comportamiento responsable, aceptando diferentes puntos de vista.



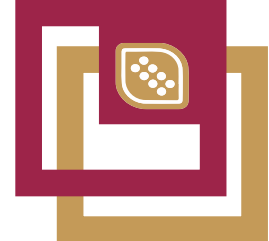
Sugerencias de Evidencia de Evaluación

- ❖ Prepara el software con base de datos relacional en forma metódica y organizada para su desarrollo, reflexionando sobre la forma de conducirse en el contexto.
- ❖ Plantea comandos de menú e instrucciones SQL, en forma crítica, para el manejo de registros en la base de datos; responsabilizándose de sus decisiones en diferentes contextos.

Competencia Laboral Básica

CLBDS3. Estructura con acompañamiento de su docente, bases de datos relacionadas con situaciones sociales de su comunidad utilizando la programación y mostrando un comportamiento metódico, además de organizado, con la finalidad de promover un pensamiento crítico y analítico.





DOSIFICACIÓN PROGRAMÁTICA

DESARROLLO DE SOFTWARE

Formación Laboral Básica: Desarrollo de Software

Módulo II: Base de Datos y Desarrollo Web

Submódulo I: Desarrollo de Aplicaciones de Escritorio con Base de Datos Clave: C4AE

Semestre: 4to. **Turno:** Matutino/Vespertino **Periodo:** 2025 – 2026B



Submódulo	Momento	Tiempo (minutos)	Conocimientos	Semana	Fecha inicio	Observaciones
SUBMÓDULO I. DESARROLLO DE APLICACIONES DE ESCRITORIO CON BASE DE DATOS	Apertura	50 min	Bienvenida. Encuadre Evaluación diagnóstica.	1	03 – 06 de febrero de 2026	Inicio del semestre 03 de febrero.
		100 min	Introducción a las bases de datos.			
		100 min	Fundamentos de las bases de datos.			
		100 min	Diferencia entre hojas de cálculo y base de datos			
	Desarrollo	350 min	Instalación del software manejador de base de datos. Tipos de base de datos.	2	09 – 13 de febrero de 2026	
		350 min	Estructura de base de datos relacionales. - Campos. - Registros	3	16 – 20 de febrero de 2026	
		350 min	Estructura de base de datos relacionales. - Tablas - Redundancia de datos	4	23 – 26 de febrero de 2026	



Submódulo	Momento	Tiempo (minutos)	Conocimientos	Semana	Fecha inicio	Observaciones
SUBMÓDULO I. DESARROLLO DE APLICACIONES DE ESCRITORIO CON BASE DE DATOS	Desarrollo	350 min	Comandos de menú y SQL en el manejo de registros para: - Insertar. - Borrar - Modificar	5	02 – 06 de marzo de 2026	
		350 min	Comandos de menú y SQL en el manejo de registros para: Consultar Creación y actualización de base de datos.	6	09 – 13 de marzo de 2026	
	Cierre	350 min	Creación y actualización de base de datos. Situación Didáctica	7	17 – 20 de marzo de 2026	 Suspensión oficial 16 de marzo.

Criterios de Evaluación

Situación didáctica:

DataLab: Creando Conexiones

30%

Actividades:

25%

Actividad 1. Base de Datos.

5%

Actividad 2. Manejadores de Bases de Datos.

5%

Actividad 3. Tipos y Modelos de Bases de Datos.

5%

Actividad 4. Presentación Electrónica. Comandos SQL y Lenguaje de Manipulación de Datos.

5%

Actividad 5. Elementos de un proyecto BD SQL.

3%

Actividad 6. Generemos Información.

2%

Prácticas:

25%

Práctica 1. Modelado Básico de Bases de Datos.

5%

Práctica 2. Creación de Base de Datos.

10%

Práctica 3. Base de Datos con instrucciones SQL.

10%

Examen:

20%

Vamo a trabaja



Situación Didáctica

DATALAB: CREANDO CONEXIONES

Propósito:

Elaborar en equipos de 6 integrantes, un Modelo de Base de Datos relacional donde apliquen los fundamentos y el lenguaje SQL para crear un proyecto que analice datos ambientales, con el fin de generar propuestas tecnológicas que contribuyan al cuidado del medio ambiente de su comunidad e interpretar la información utilizando el software manejador de base de datos y socializarlo en el aula.

Contexto:

Tabasco, estado conocido como el Edén del Sureste Mexicano; está enfrentándose al crecimiento en la población y con ello diferentes problemáticas ambientales que exigen la necesidad de hacer más eficiente la comunicación de la información para mantener las áreas verdes, los espacios acuíferos sanos y contribuir en la generación de soluciones a las problemáticas latentes.

En la actualidad se han identificado diversas problemáticas ambientales como el aumento en la generación de residuos, la disminución de áreas verdes, cambios en la calidad del aire y variaciones en el consumo de agua. Aunque existen datos dispersos en reportes, mediciones aisladas y registros institucionales, **no hay un sistema organizado que permita analizarlos de manera integral** para comprender qué está ocurriendo realmente y tomar decisiones informadas.

Las autoridades locales, docentes de los diversos centros educativos y grupos ciudadanos coinciden en que la falta de información estructurada dificulta proponer acciones efectivas. Sin datos claros es imposible responder preguntas como:



	• ¿Qué zonas generan más residuos y por qué? • ¿Cómo varía la calidad del aire a lo largo del día o del mes? • ¿Qué factores están relacionados con el aumento del consumo de agua? • ¿Cómo impactan estas situaciones en la salud y el entorno? Frente a esta necesidad, el Colegio de Bachilleres de Tabasco, a través de la Formación Laboral Básica de Desarrollo de Software, propone que los estudiantes desarrollen un sistema de base de datos relacional que permita registrar, organizar y analizar datos ambientales reales o simulados, con el fin de generar información útil que oriente soluciones para el cuidado del entorno. De este modo, las y los estudiantes utilizarán SQL para crear conexiones entre datos y revelar patrones invisibles a simple vista, contribuyendo con un proyecto tecnológico de impacto comunitario.
Conflicto Cognitivo	¿Cómo las redes sociales almacenan y gestionan millones de perfiles de usuarios, publicaciones, comentarios, imágenes y conexiones entre personas? ¿Cómo se pueden relacionar los datos? ¿Qué beneficios tiene el uso de las bases de datos y el lenguaje SQL para el trabajo de proyectos medio ambientales?





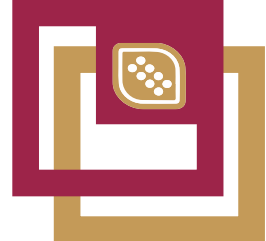
Evaluación Diagnóstica



Instrucciones: Lee cuidadosamente los siguientes cuestionamientos y selecciona en base a tus conocimientos la respuesta que consideres correcta.

- ¿Es una colección de información estructurada y organizada de forma que permita de manera fácil su gestión y actualización?
 - Registro
 - Tabla
 - Base de datos
 - Archivo
- ¿Base de datos que permite establecer interconexiones lógicas entre datos que están almacenados en tablas, y por medio de las cuales se pueden vincular estás?
 - Jerárquica
 - Relacional
 - De red
 - Transaccionales
- ¿Son conjuntos de campos que contiene valores relacionados con algún tema o especie de dato en particular??
 - Tabla
 - Relación
 - Registros
 - Datos
- ¿Lenguaje responsable de hacer consultas y editar la información almacenada en un determinado sistema de gestión de base de datos?
 - SQL
 - HTML
 - JavaScript
 - PHP
- Es la base de datos que enlaza registros en forma de estructura de árbol. En donde un nodo padre puede tener varios nodos hijos.
 - Jerárquica
 - Relación
 - De red
 - Transaccionales
- Es la unidad mínima de información a la cual se puede acceder en una base de datos.
 - Registro
 - Tabla
 - Campo
 - Relación
- Son objetos de la base de datos que contiene todos sus datos.
 - Registro
 - Tabla
 - Campo
 - Relación





8. Sirve para extraer información de una base de datos permitiendo manipularla.

A) Modificaciones	C) Consultas
B) Inserción	D) Eliminación
9. Esta instrucción del lenguaje SQL permite agregar nuevas filas de datos a las tablas existentes.

A) Insert	C) Delete
B) Update	D) From
10. Esta instrucción del lenguaje SQL sirve para actualizar los registros existentes en una tabla.

A) Insert	C) Delete
B) Update	D) From
11. Es un objeto de la base de datos que resulta útil para presentar la información de esta.

A) Informe	C) Tabla
B) Registro	D) Campo
12. Este software sirve para crear y manipular la base de datos.

A) Administrador de archivos	B) Sistema manejador de base de datos
B) Programa de aplicación	C) Sistema de archivos.
13. Es la persona que tiene el control central de la base de datos.

A) Administrador de la base de datos	C) Programador de aplicaciones
B) Usuario final	D) Usuarios
14. Es un campo o un conjunto de campos que identifica de forma única cada una de las filas de una tabla de una base de datos.

A) Clave secundaria	C) Clave primaria o llave primaria
B) Clave foránea	D) Identificador
15. Es un nivel de abstracción de datos que describe que datos se almacenan en la base de datos y la relación que existe entre esos datos.

A) Nivel físico	C) Nivel lógico
B) Nivel de vistas	D) Nivel máquina
16. Indica. Es la instrucción del lenguaje SQL que permite recuperar datos de una tabla.

A) Insert	C) Delete
B) Update	D) Select





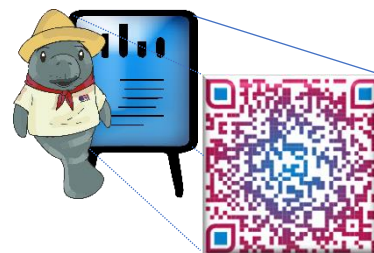


Lectura No 1. Bienvenida a las Bases de Datos

Instrucciones: Realiza la Lectura 1. “Bienvenida a las Bases de Datos” subrayando las ideas principales, y al finalizar realiza la Actividad 1 – “Base de Datos”



LECTURA 1. “BIENVENIDA A LAS BASES DE DATOS”



¿Qué es un Sistema de Base de Datos?



Es un sistema computarizado cuya finalidad general es **almacenar información y permitir a los usuarios recuperar y actualizar esa información** con base en peticiones. La información en cuestión puede ser cualquier cosa que sea de importancia para el individuo u organización; en otras palabras, todo lo que sea necesario para auxiliarle en el proceso general de su administración. (Date, 2001)

Los términos datos e información pueden ser tratados como sinónimos. Algunos autores prefieren distinguir entre ambos, utilizando datos para referirse a lo que está en realidad almacenado en la base de datos e información para referirse al significado de esos datos, cómo lo entiende algún usuario. La diferencia es importante tan importante que parece preferible hacer la explicación donde sea necesario en vez de depender de una diferenciación un tanto arbitraria entre dos términos que son en esencia sinónimos. (Date, 2001)



Los datos dentro de los tipos más comunes de bases de datos en funcionamiento hoy en día se modelan típicamente en filas y columnas en una serie de tablas para que el procesamiento y la consulta de datos sean eficientes, la cantidad de estos tres elementos está limitada únicamente por la capacidad de Hardware del Sistema de Base de Datos. Luego se puede acceder, administrar, modificar, actualizar, controlar y organizar fácilmente los datos. La mayoría de las bases de datos utilizan lenguaje de consulta estructurado (SQL) para escribir y consultar datos. (Oracle México, s.f.)



¿Qué interviene en una Base de Datos?

Las bases de datos tienen tres elementos fundamentales, componente físico, componente lógico y usuario. El **componente físico** consta del hardware de un equipo de cómputo, principalmente los discos duros, procesadores y memoria principal, estos elementos conforman el almacenamiento físico de la base de datos. El **componente lógico** consta de los elementos de software, generalmente se conoce a este programa como el administrador de base de datos o el servidor de base de datos o más comúnmente conocido como Sistema de Administración de Base de Datos (DBMS).



El último elemento que interviene en una base de datos es el **usuario**. Estos son los responsables de escribir los programas de aplicación de base de datos en algún lenguaje de programación como C++, JAVA, C# o algún otro lenguaje de alto nivel. Estos programas acceden a la base de datos emitiendo la solicitud apropiada al DBMS (por lo regular instrucciones SQL) para realizar algún proceso con los datos almacenados. (Date, 2001)

¿Qué es el lenguaje de consulta estructurado (SQL)?

SQL es un lenguaje de programación usado por casi todas las bases de datos relacionales para consultar, manipular y definir datos, y para proporcionar control de acceso. SQL se desarrolló por primera vez en IBM en la década de 1970 con Oracle como uno de los principales contribuyentes, lo que llevó a la implementación del estándar ANSI de SQL. SQL ha generado muchas extensiones por parte de compañías como IBM, Oracle y Microsoft. Aunque SQL todavía se usa ampliamente en la actualidad, comienzan a aparecer nuevos lenguajes de programación. (Oracle México, s.f.)



La gestión de base de datos es un proceso complicado, que se ha facilitado considerablemente por el lenguaje SQL. Como su nombre completo (Structured Query Language) implica, el lenguaje SQL es responsable de hacer consultas y editar la información almacenada en un determinado sistema de gestión de base de datos. (UIV, s.f.)





Este lenguaje de programación está conformado por lo que comúnmente se le conoce como instrucciones, que permiten la interacción entre el usuario y la base de datos.

¿Cuál es la diferencia entre una base de datos y una hoja de cálculo?

Las bases de datos y las hojas de cálculo (como Microsoft Excel) son dos formas convenientes de almacenar información. Las principales diferencias entre las dos son:

- Cómo se almacenan y manipulan los datos
- Quién puede acceder a los datos
- Cuántos datos se pueden almacenar

Las hojas de cálculo se diseñaron originalmente para un usuario, y sus características lo reflejan. Son muy buenas para un solo usuario o un pequeño número de usuarios que no necesitan manipular una gran cantidad de datos complicados. Las bases de datos, por otro lado, están diseñadas para contener colecciones mucho más grandes de información organizada, cantidades masivas en ocasiones. Las bases de datos permiten a múltiples usuarios al mismo tiempo acceder y consultar los datos de forma rápida y segura utilizando una lógica y un lenguaje altamente complejos. (Oracle México, s.f.)



ACTIVIDAD 1. BASE DE DATOS



Instrucciones: Después de realizar la Lectura No 1. Elige una de las dos actividades que se te presentan a continuación.

Actividad A

Elabora un juego didáctico interactivo siguiendo los siguientes pasos:

1.- Selecciona 10 conceptos mencionados en la lectura, considera que los conceptos deben ser cortos y entendibles.

2.- Ingresa a la herramienta EducaPlay (o similar).



3.- Si aún no tienes cuenta en la plataforma es necesario que realices un registro de un usuario nuevo. Puedes registrarte utilizando tu cuenta institucional en la opción de Microsoft.

Crea y comparte tu mundo de juegos educativos

¿Aún no tienes cuenta? [Regístrate gratis](#), es fácil y rápido.

Inicia sesión

Correo electrónico

Contraseña

☐ Mantener sesión iniciada en este navegador

Entrar

¿Has olvidado tu contraseña?

Acceder con Google

¿No estás registrado? Obtén tu cuenta gratis

4.- Después de ingresar a la plataforma con tu cuenta, debes dar clic en la opción "Crear Actividad".



5.- De las opciones que se te muestran en pantalla selecciona entre las opciones de Crucigrama y Sopa de Letras.



Sopa de letras

Esconde palabras entre un cuadrado lleno de letras para que las encuentren.



Crucigrama

Forma filas y columnas de palabras entrecruzadas para que rellenen sus letras.

6.- Siguiendo las instrucciones crea un Crucigrama o Sopa de Letras agregando los conceptos y definiciones identificados previamente.



7.- Comparte con tu maestro y compañeros tu nuevo juego.



Educación que genera cambio

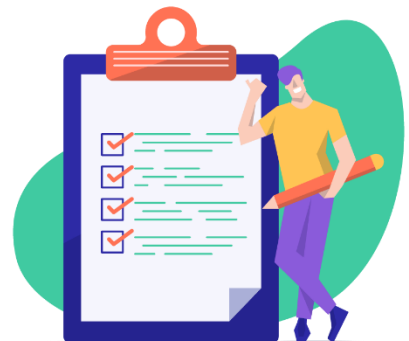


Actividad B

Responde las siguientes preguntas o genera tu propio cuestionario, aplícalo a tus compañeros y al finalizar comenta tus respuestas con tus compañeros de clase.



1. ¿Cuál consideras que sea la utilidad de los Sistemas de Base de Datos desde el aspecto laboral?
2. ¿Cuál consideras que sea el componente más importante en una base de datos?
3. ¿Como estudiante de la capacitación de Desarrollo de Software, ¿Por qué considerarías relevante e importante aprender sobre la Creación y Administración de Base de Datos?
4. ¿Qué es SQL?
5. ¿Cómo se modelan típicamente los datos en una base de datos?



 INSTRUMENTO DE EVALUACIÓN LISTA DE COTEJO 1 Actividad 1: "Base de Datos" 						
<i>Actividad A</i>						
DATOS GENERALES						
Nombre(s) del alumno(s)				Matrícula(s)		
Producto: Juego didáctico interactivo				Fecha		
Submódulo 1: Desarrollo de Aplicaciones de Escritorio con Base de Datos				Periodo: 2025 - 2026B		
Nombre del docente				Firma del docente		
CRITERIO	INDICADORES	VALOR OBTENIDO		PONDERACIÓN	CALIF	OBSERVACIONES Y/O SUGERENCIAS DE MEJORA
		SÍ	NO			
1	Elige de 6 a 10 conceptos para la elaboración de las preguntas.			1		
2	Presenta un excelente uso del software para la creación de su juego didáctico.			2		
3	Presenta al menos 5 iconos o imágenes para mejorar la calidad de su juego didáctico.			2		
4	Presenta una actitud positiva en la realización y aplicación de su juego didáctico.			1		
5	No presenta faltas ortográficas.			1		
6	Considera las opiniones de sus compañeros para mejorar el juego didáctico.			2		
7	Entrega en tiempo y forma.			1		
CALIFICACIÓN						





Lectura No 2. Software Manejadores de Base de Datos

Instrucciones: Realiza la Lectura 2. “Software Manejadores de Base de Datos” subrayando las ideas principales, y al finalizar realiza la Actividad 2 – “Cuadro Comparativo Manejador de Base de Datos”



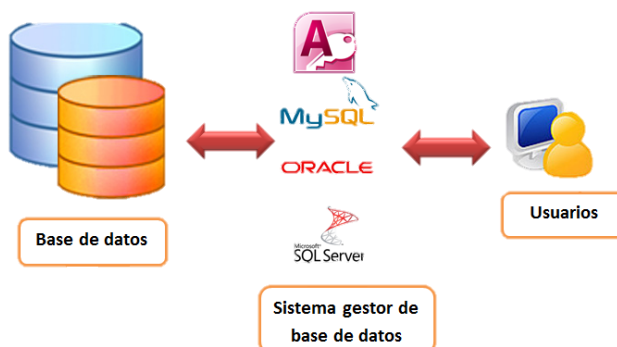
LECTURA 2. “SOFTWARE MANEJADORES DE BASE DE DATOS”



¿Qué es un Data Base Management System (Sistema Manejador de Base de Datos)?

La propia base de datos y el sistema gestor de base de datos o SGBD (también llamado DBMS, del inglés database management system) conforman lo que se denomina el sistema de base de datos (a veces, se utiliza simplemente el término base de datos para denominarlo). En términos generales, un SGBD es un software que sigue un modelo de sistema de base de datos y, por lo tanto, resulta decisivo a la hora de configurarla, administrarla y utilizarla. Solo cuando el sistema gestor de base de datos está instalado y configurado, los usuarios pueden introducir y consultar los datos. Los permisos de lectura y escritura, así como las funciones de administración generales, se establecen mediante las interfaces específicas de la aplicación y el lenguaje de definición de datos correspondiente. El más conocido de estos lenguajes es SQL (Structured Query Language, lenguaje de consulta estructurada).

Tareas, funciones y propiedades del sistema gestor de base de datos



El sistema de gestión de base de datos es el componente más importante de un sistema de base de datos. Sin él, no sería posible administrar, controlar o supervisar la base de datos. Este software también es responsable de gestionar todos sus permisos de lectura y escritura. Un término que suele utilizarse mucho para resumir las funciones y propiedades de las transacciones de los sistemas gestores de base de datos es **ACID**, siglas de los términos en inglés *atomicity*, *consistency*, *isolation* y *durability* (es decir, atomicidad, consistencia, aislamiento y permanencia). Estos cuatro conceptos engloban los requisitos más importantes de un SGBD:



- La atomicidad o **integridad** describe la propiedad de “todo o nada” de los SGBD, por la que todas las fases de una transacción deben finalizarse por completo y en el orden correcto para que esta sea válida.
- La **consistencia** implica que las transacciones completadas no afecten la estabilidad de la base de datos, lo que requiere supervisarlas constantemente.
- El **aislamiento** es la propiedad que asegura que las transacciones no obstaculicen a las demás, de lo que, por lo general, se encargan algunas funciones de bloqueo.
- La **permanencia** implica que todos los datos queden almacenados permanentemente en el SGBD, no solo después de una transacción correcta, sino también o especialmente en caso de error o caída del sistema. Los registros de las transacciones, donde quedan anotados todos los procesos del SGBD, son fundamentales para garantizar la permanencia.

Ejemplos de SGBD: sistemas más populares

De entre los numerosos sistemas gestores de bases de datos que existen, estos son los 15 más populares y utilizados (ionos.mx):

DBMS - Most Popular Database Management Systems

- | | | |
|-------------------------------------|----------------------|---|
| • Microsoft Access (relacional) | ORACLE
DATABASE | • MongoDB (orientado a documentos) |
| • Microsoft SQL Server (relacional) | | |
| • MySQL (relacional) | mongoDB | • PostgreSQL (combina relacional y orientado a objetos) |
| • Oracle Database (relacional) | MySQL | • Firebird (relacional) |
| • OrientDB (orientado a documentos) | Microsoft SQL Server | • InterSystems Caché (combina relacional y orientado a objetos) |
| • CouchDB (orientado a documentos) | | • InterSystems IRIS (combina relacional y orientado a objetos) |
| • Db2 de IBM (relacional) | PostgreSQL | |
| • IMS de IBM (jerárquico) | | |
| • IBM Informix (relacional) | Microsoft Access | |
| • MariaDB (relacional) | | |
| • Sybase ASE (relacional) | IBM DB2 | |



BIBLIOGRAFÍA

Date, C. J. (2001). *Introducción a los Sistemas de Base de Datos* (25 ed.). Pearson.

Date, C. J. (2004). *An introduction to database systems* (8th ed.). Addison-Wesley.

Elmasri, R., & Navathe, S. B. (2016). *Fundamentals of database systems* (7th ed.). Pearson.

Garcia-Molina, H., Ullman, J. D., & Widom, J. (2009). *Database systems: The complete book* (2nd ed.). Pearson.

Harrington, J. L. (2016). *Relational database design and implementation* (4th ed.). Morgan Kaufmann.

IONOS. (2020). sistema gestor de base de datos (SGBD). 12.05.20, de IONOS Sitio web: <https://www.ionos.mx/digitalguide/hosting/cuestiones-tecnicas/sistema-gestor-de-base-de-datos-sgbd/>

Oracle México. (s.f.). Obtenido de ¿Qué es una base de datos?: <https://www.oracle.com/mx/database/what-is-database.html>

UIV, E. d. (s.f.). Universidad Internacional de Valencia. Obtenido de Lenguaje SQL, historia y conceptos básicos. <https://www.universidadviu.com/lenguaje-sql-historia-conceptos-basicos/>

Rob, P., & Coronel, C. (2012). *Database systems: Design, implementation, and management* (9th ed.). Cengage Learning.



ACTIVIDAD 2. CUADRO COMPARATIVO MANEJADOR BD



Instrucciones: Después de instalar el software sugerido en la Lectura No 2. Investiga al menos 5 manejadores o gestores de bases de datos para realizar una tabla comparativa considerando los siguientes aspectos:

Nombre del software, icono (imagen del software), sistemas operativos en los que funciona, observaciones (descripción) y enlace de descarga (URL)



MANEJADORES DE BASES DE DATOS

Nº	Icono (imagen del software)	Nombre del software	Descripción (Observaciones)	Sistema(s) Operativo(s)	URL (Enlace de descarga)
1					
2					
3					
4					
5					

CONCLUSIÓN:

Menciona el software que va a instalar y utilizar, las características y herramientas que posee.



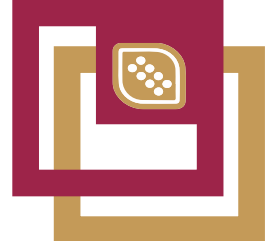


INSTRUMENTO DE EVALUACIÓN
LISTA DE COTEJO 2
Actividad 2: "Cuadro Comparativo Manejador BD"



DATOS GENERALES						
Nombre(s) del alumno(s)				Matrícula(s)		
Producto: Cuadro comparativo Manejador BD				Fecha		
Submódulo 1: Desarrollo de Aplicaciones de Escritorio con Base de Datos				Periodo: 2025 - 2026B		
Nombre del docente				Firma del docente		
CRITERIO	INDICADORES	VALOR OBTENIDO		PONDERACIÓN	CALIF	OBSERVACIONES Y/O SUGERENCIAS DE MEJORA
		SÍ	NO			
1	Respetar la estructura de la tabla indicada por el docente			1		
2	Presenta los 5 administradores o gestores de bases de datos.			2		
3	Presenta los 5 iconos o imágenes de los administradores o gestores de bases de datos.			2		
4	Las descripciones corresponden a cada tipo de Base de Datos.			1		
5	No presenta faltas ortográficas.			1		
6	Elabora una conclusión personal y entendible.			2		
7	Entrega en tiempo y forma.			1		
CALIFICACIÓN						





Instalación y uso Básico de un DBMS



Ahora que ya tienes idea de lo que es un programa que permite la gestión de base de datos, te mostraremos como instalar el programa necesario para crear y administrar Bases de Datos.



« Instalación en PC »

La aplicación utilizada en Escritorio será XAMPP.

- XAMPP es una distribución de Apache completamente gratuita y fácil de instalar que contiene MariaDB, PHP y Perl.

Al instalar esta aplicación obtendremos el DBMS el cual en este caso es MariaDB que es un motor de base de datos derivado de MySQL.

El Software MariaDB es de uso Libre. También se obtiene un IDE que permite la creación y administración de base de datos por medio de comandos SQL. El IDE en cuestión se denomina phpMyAdmin.



Procedimiento.

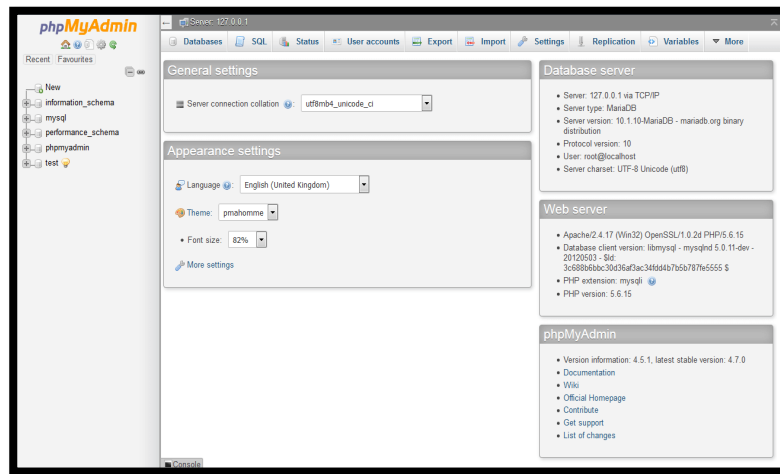
- 1.- Descarga XAMPP de la página oficial dependiendo la versión de tu sistema operativo y procesador.
- 2.- Una vez descargado el archivo .exe debes ejecutarlo haciendo doble clic en él, es recomendable desactivar el antivirus y verificar que se ejecuta el archivo exe como administrador en Windows.
- 3.- Después de lo anterior deberá iniciar el asistente de instalación en el cual debes hacer clic en Next.
- 4.- La siguiente opción te mostrará una serie de opciones que se pueden evitar instalar, te sugerimos dejar estas opciones por default y dar clic en Next.
- 5.- En este paso deberás elegir el directorio en el cual instalar XAMPP, por default se crea una carpeta con el mismo nombre en la unidad C:. Es recomendable dejar esta misma opción y dar clic en Next.
- 6.- Después de esto iniciara la instalación de la aplicación en nuestra computadora.



7.- Es posible que surjan avisos de bloqueo de FireWall. En este caso es necesario marcar la casilla de Redes Privadas. Después dar clic en permitir el acceso.

8.- Después de esto la instalación terminará y el programa deberá poderse ejecutar para mostrar su panel de control.

9.- Para poder utilizar el programa es necesario dar clic en star en los servicios de Apache y MySQL. Al iniciar estos servicios podrás ir a tu navegador web y escribir lo siguiente en la barra de Navegación. *localhost/phpmyadmin/ esto abrirá el IDE que permitirá ejecutar comandos SQL en la base de datos.*



Descarga el software



<https://www.apachefriends.org/es/index.html>

« Móvil Android »

En los dispositivos Android podemos instalar una aplicación similar a la instalada en el PC.

En este caso utilizaremos la aplicación denominada AwebServer.



ANDROID

- 1.- Descargar e instalar de la tienda oficial la aplicación AwebServer.
- 2.- Después de instalar deberás ingresar a la aplicación y dar clic en el botón Iniciar esto iniciará el servidor Web.
- 3.- A continuación, deberás ir al ícono MySQL y dar clic en iniciar. Esto iniciará el servidor Base de Datos.
- 4.- Una vez iniciada la Base de Datos deberás dar clic en el enlace con números y terminación :8080/mysqladmin esto abrirá el navegador de tu dispositivo y cargará una página de logeo.
- 5.- En las opciones de ingreso debes escribir como usuario la palabra: root y como contraseña la palabra: root.
- 6.- La aplicación phpMyAdmin abrirá con todas las opciones que permiten la ejecución de comandos SQL.



Descarga el software



AWebServer (Http Web Server
Apache PHP Sql)

Sylkat Herramientas

★★★★★ 305

Todos

Contiene anuncios

Esta app es compatible con todos tus dispositivos.

Agregar a la lista de deseos

Instalar

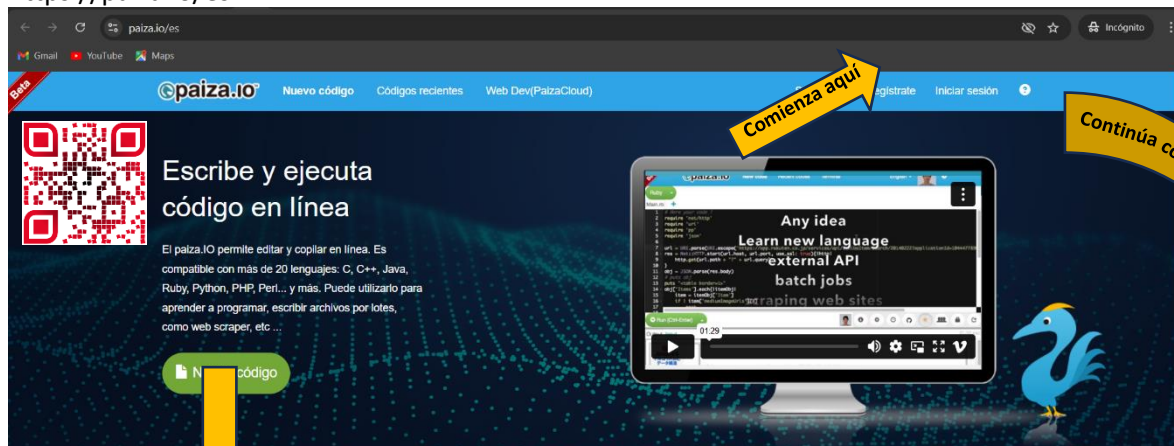


https://play.google.com/store/apps/details?id=com.sylkat.apache&hl=es_GT



Educación que genera cambio

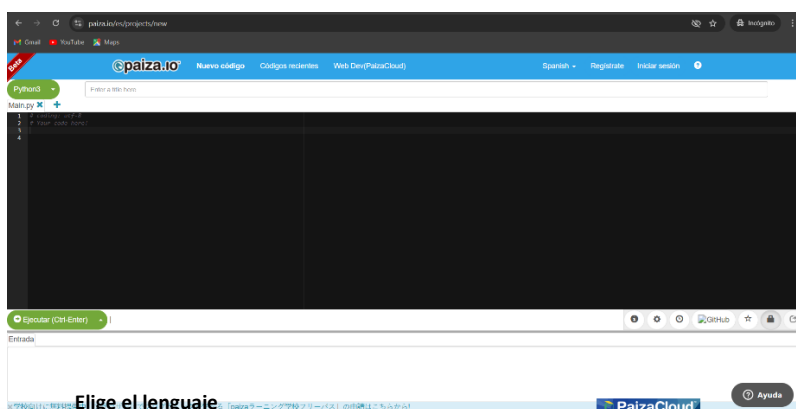
https://paiza.io/es



For web development, package installation,... visit PaizaCloud Cloud IDE!

Usa el IDE de desarrollo online

PaizaCloud



Elige el lenguaje

PaizaCloud

Register

GitHub Iniciar sesión

Register

Username

Email

Password

Password confirmation

Register

Forgot your password?

Didn't receive confirmation instructions?





Lectura No 3. Tipos de Base de Datos

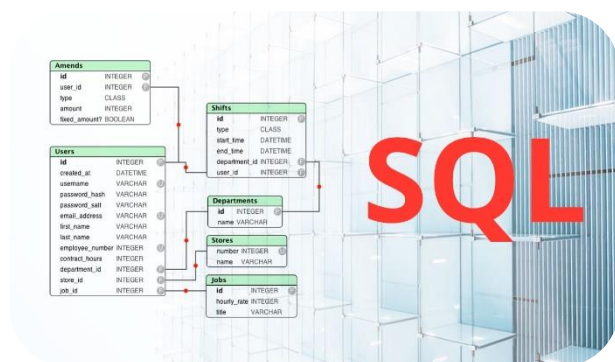
Instrucciones: Realiza el análisis de la siguiente información, posteriormente, completa la Actividad 3 Tabla Comparativa Tipos y Modelos de Datos.



LECTURA 3. "TIPOS DE BASE DE DATOS"



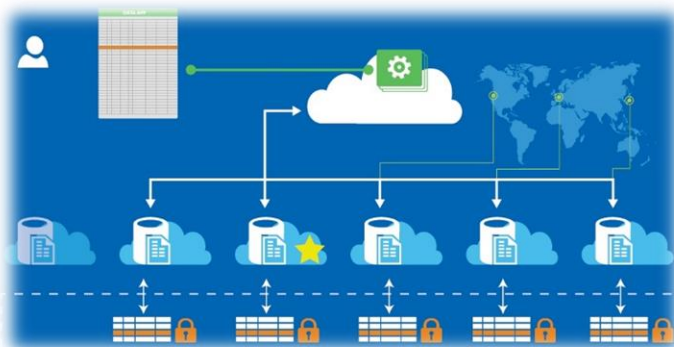
Hay muchos tipos o modelos diferentes de bases de datos. La mejor base de datos para una organización específica depende de cómo la organización pretende utilizar los datos.

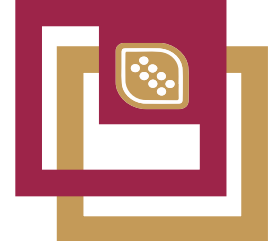


Bases de datos relacionales. Las bases de datos relacionales se popularizaron en los años ochenta. Los elementos de una base de datos relacional se organizan como un conjunto de tablas con columnas y filas. La tecnología de base de datos relacional proporciona la manera más eficiente y flexible de acceder a información estructurada.

Bases de datos orientadas a objetos. La información en una base de datos orientada a objetos se representa en forma de objetos, como en la programación orientada a objetos.

Bases de datos distribuidas. Una base de datos distribuida consta de dos o más archivos ubicados en diferentes sitios. La base de datos puede almacenarse en múltiples computadoras, ubicadas en la misma ubicación física o dispersas en diferentes redes.





Al estar las bases de datos en distintos lugares físicos, pueden agregarse sitios nuevos a la red sin afectar las operaciones de otros sitios. Tal flexibilidad permite que la compañía se expanda con relativa facilidad y rapidez. Aunque esto también puede terminar comprometiendo la seguridad de los datos almacenados, ya que se puede tener menos control para acceder a ellos físicamente. (Clavijero Edu, s.f.)

Almacenes de datos. Un almacén de datos es un tipo de base de datos diseñada específicamente para consultas y análisis rápidos, y funciona como un depósito central de datos. Su procesamiento requiere de tiempos prolongados.



Bases de datos NoSQL. Una NoSQL, o una base de datos no relacional, permite que los datos no estructurados y semiestructurados se almacenen y manipulen, a diferencia de una base de datos relacional, que define cómo deben componerse todos los datos insertados en la base de datos. Las bases de datos NoSQL se hicieron populares a medida que las aplicaciones web se hacían más comunes y más complejas.



Bases de datos orientadas a grafos. Una base de datos orientada a grafos almacena datos en términos de entidades y las relaciones entre entidades.

Bases de datos OLTP. Una base de datos OLTP es una base de datos analítica y rápida diseñada para un gran número de transacciones realizadas por múltiples usuarios.

Estas son solo algunas de las varias docenas de tipos de bases de datos en uso hoy en día. Otras bases de datos menos comunes se adaptan a funciones científicas, financieras u otras funciones muy específicas. Además de los diferentes tipos de bases de datos, los cambios en los enfoques de desarrollo de tecnología y los importantes avances, como la nube y la automatización, llevan a las bases de datos en direcciones totalmente nuevas. **Algunas de las bases de datos más recientes incluyen:**



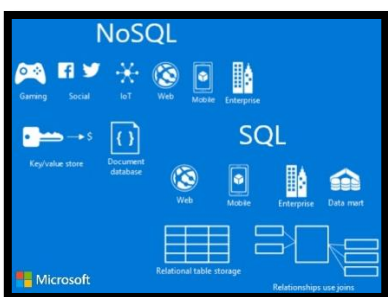
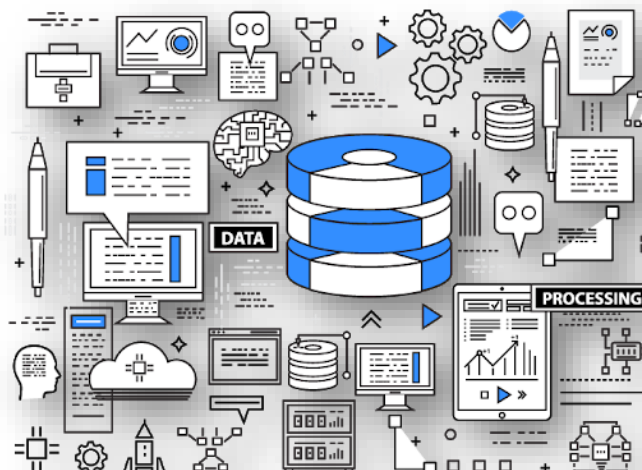
Bases de datos de código abierto.

Un sistema de base de datos de código abierto es aquel cuyo código fuente es de código abierto; dichas bases de datos podrían ser bases de datos SQL o NoSQL.

Bases de datos en la nube.

Una base de datos en la nube es una colección de datos, ya sean estructurados o no estructurados,

que reside en una plataforma de computación en la nube privada, pública o híbrida. Hay dos tipos de modelos de base de datos en la nube: tradicional y database as a service (DBaaS). Con DBaaS, las tareas administrativas y el mantenimiento son realizados por un proveedor de servicios.

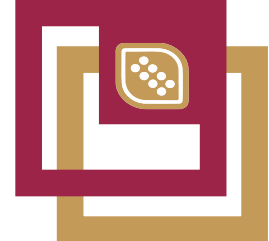


Base de datos multimodelo. Las bases de datos multimodelo combinan diferentes tipos de modelos de base de datos en un único back-end integrado. Esto significa que pueden acomodar varios tipos de datos.

Base de datos documental/JSON. Diseñadas para almacenar, recuperar y administrar información orientada a documentos, las bases de datos documentales son una forma moderna de almacenar datos en formato JSON en lugar de filas y columnas.

Bases de datos independientes. Las bases de datos independientes, el tipo de base de datos más nuevo e innovador (también conocidas como bases de datos autónomas), se basan en la nube y utilizan el aprendizaje autónomo para automatizar el ajuste, la seguridad, las copias de seguridad, las actualizaciones y otras tareas de administración de rutina de las bases de datos que tradicionalmente realizan los administradores de bases de datos. (Oracle México, s.f.)





Para estudio de esta materia el modelo de base de datos a utilizar será el Relacional.

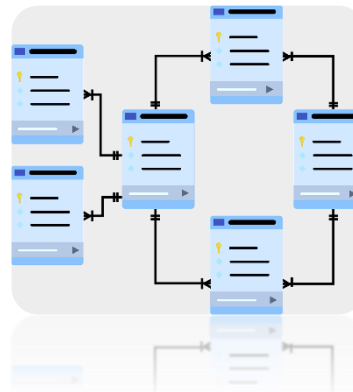
Modelo Relacional

En los primeros años de las bases de datos, cada aplicación almacenaba datos en su propia estructura única. Cuando los desarrolladores querían crear aplicaciones para usar esos datos, tenían que saber mucho sobre la estructura de datos particular para encontrar los datos que necesitaban.

Estas estructuras de datos eran ineficientes, difíciles de mantener y difíciles de optimizar para ofrecer un buen rendimiento de la aplicación. El modelo de base de datos relacional se diseñó para resolver el problema de varias estructuras de datos arbitrarias.

El **modelo relacional** proporcionó una forma estándar de representar y consultar datos que cualquier aplicación podría utilizar. Desde el principio, los desarrolladores reconocieron que la principal fortaleza del modelo de base de datos relacional estaba en el uso de tablas,

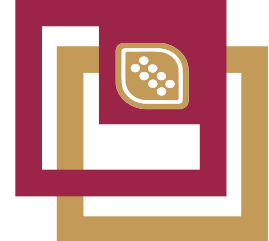
que eran una forma intuitiva, eficiente y flexible de almacenar y acceder a información estructurada. Con el tiempo, cuando los desarrolladores comenzaron a utilizar el lenguaje de consulta estructurado (SQL) para escribir y consultar datos en una base de datos, surgió otra fortaleza del modelo relacional. Durante muchos años, se utilizó ampliamente el SQL como lenguaje para consultas de bases de datos. El SQL, que se basa en el álgebra relacional, proporciona un lenguaje matemático internamente consistente que facilita la mejora del rendimiento de todas las consultas de la base de datos. En comparación, otros enfoques deben definir consultas individuales.



Beneficios de las bases de datos relacionales

Las organizaciones de todo tipo y tamaño utilizan el modelo relacional simple pero poderoso para una amplia variedad de necesidades de información. Las bases de datos relacionales se utilizan para hacer seguimiento de los inventarios, procesar transacciones de comercio electrónico, administrar grandes cantidades de información de clientes de misión crítica y mucho más. Se puede considerar una base de datos relacional para cualquier necesidad de información en la que los puntos de datos se relacionen entre sí y se deban administrar de una manera segura, consistente y basada en reglas. Las bases de datos relacionales existen desde la década de 1970. Actualmente, las ventajas del modelo relacional continúan convirtiéndolo en el modelo más aceptado para bases de datos.





BIBLIOGRAFÍA

Coronel, C., & Morris, S. (2020). *Sistemas de bases de datos: Diseño, implementación y administración* (13ª ed.). Cengage Learning.

Date, C. J. (2004). *Introducción a los sistemas de bases de datos* (8ª ed.). Pearson Educación.

Oracle México. (s.f.). Obtenido de ¿Qué es una base de datos?:

<https://www.oracle.com/mx/database/what-is-database.html>

Elmasri, R., & Navathe, S. B. (2016). *Fundamentos de sistemas de bases de datos* (7ª ed.). Pearson.

Tema 1.4 Ventajas de las bases de datos distribuidas contra las bases de datos centralizadas - Bases de datos distribuidas - **Instituto Consorcio Clavijero. (s. f.).**

https://cursos.clavijero.edu.mx/cursos/080_bdd/modulo1/contenidos/tema1.4.html?opc=0

UIV, E. d. (s.f.). Universidad Internacional de Valencia. Obtenido de Lenguaje SQL, historia y conceptos básicos.

<https://www.universidadviu.com/lenguaje-sql-historia-conceptos-basicos/>



Actividad 3. Tabla Comparativa Tipos y Modelos BD

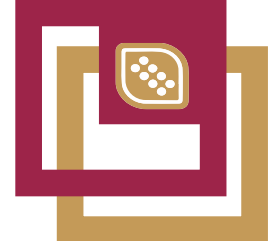


Instrucciones: Completa la siguiente tabla comparativa utilizando la información de la Lectura 3. Elige 5 tipos de Base de Datos para completar la tabla. Al finalizar elabora una breve conclusión



Tipo	Descripción	Ventaja	Desventaja
1.- Relacional	<i>Se organizan como un conjunto de tablas con columnas y filas</i>	<i>Eficiente y Flexible el acceder a la información.</i>	<i>Modelo no muy actual.</i>





Educación que genera cambio



Instrucciones: Al finalizar elabora una breve conclusión

CONCLUSIÓN

This image shows a single sheet of white paper with horizontal blue ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

“EDUCACION QUE GENERA CAMBIO”





INSTRUMENTO DE EVALUACIÓN
LISTA DE COTEJO 3
Actividad 3: Tabla comparativa
"Tipos y modelos de Base de Datos"



DATOS GENERALES						
Nombre(s) del alumno(s)				Matrícula(s)		
Producto: Tabla comparativa "Tipos y Modelos de BD"				Fecha		
Submódulo 1: Desarrollo de Aplicaciones de Escritorio con Base de Datos				Periodo: 2025 - 2026B		
Nombre del docente				Firma del docente		
CRITERIO	INDICADORES	VALOR OBTENIDO		PONDERACIÓN	CALIF	OBSERVACIONES Y/O SUGERENCIAS DE MEJORA
		SÍ	NO			
1	Respetar la estructura de la tabla indicada por el docente.			1		
2	Se utilizan la totalidad de los elementos solicitados.			2		
3	Se contrasta con claridad las ventajas con las desventajas.			2		
4	Las descripciones corresponden a cada tipo de Base de Datos.			2		
5	No presenta faltas ortográficas.			1		
6	Elabora una conclusión personal y entendible.			2		
CALIFICACIÓN						





Lectura No 4. Estructura de Base de Datos

Instrucciones: Realiza el análisis de la siguiente información, posteriormente, contesta las preguntas que se solicitan.



LECTURA 4. "ESTRUCTURA DE BASE DE DATOS"



Una base de datos relacional es una recopilación de elementos de datos con relaciones predefinidas entre ellos. Estos elementos se organizan como un conjunto de tablas con columnas y filas. Las tablas se utilizan para guardar información sobre los objetos que se van a representar en la base de datos. Cada columna de una tabla guarda un

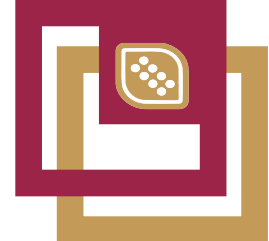


determinado tipo de datos y un campo almacena el valor real de un atributo. Las filas de la tabla representan una recopilación de valores relacionados de un objeto o una entidad. Cada fila de una tabla podría marcarse con un identificador único denominado clave principal, mientras que filas de varias tablas pueden relacionarse con claves extranjeras. Se puede obtener acceso a estos datos de muchas formas distintas sin reorganizar las propias tablas de la base de datos. (AMAZON, s.f.)

Compromiso y atomicidad

Las bases de datos relacionales manejan las reglas y políticas comerciales en un nivel muy detallado, con políticas estrictas sobre el compromiso (es decir, hacer un cambio permanente en la base de datos). Por ejemplo, considere una base de datos de inventario que hace seguimiento de tres partes que siempre se usan juntas. Cuando se extrae una parte del inventario, las otras dos también se deben extraer. Si una de las tres partes no está disponible, ninguna de las partes debe quitarse; las tres partes deben estar disponibles antes de que la base de datos se comprometa.





Una base de datos relacional no se comprometerá por una parte hasta que sepa que puede comprometerse por las tres. Esta capacidad de compromiso multifacética se llama atomicidad. La atomicidad es la clave para mantener la precisión de los datos en la base de datos y garantizar que cumpla con las reglas, regulaciones y políticas de la empresa.

Procedimientos almacenados y bases de datos relacionales



El acceso a los datos implica muchas acciones repetitivas. Por ejemplo, una consulta simple para obtener información de una tabla de datos puede necesitar repetirse cientos o miles de veces para producir el resultado deseado. Estas funciones de acceso a los datos requieren algún tipo de código para acceder a la base de datos. Los desarrolladores de aplicaciones no desean escribir un código nuevo para estas funciones en cada aplicación nueva. Afortunadamente, las bases de datos

relacionales permiten procedimientos almacenados, que son bloques de código a los que se puede acceder con una simple llamada de aplicación. Por ejemplo, un solo procedimiento almacenado puede proporcionar un etiquetado de registro consistente para usuarios de varias aplicaciones. Los procedimientos almacenados también pueden ayudar a los desarrolladores a garantizar que ciertas funciones de datos en la aplicación se implementen de una manera específica.



Bloqueo de bases de datos y concurrencia

Pueden surgir conflictos en una base de datos cuando varios usuarios o aplicaciones intentan cambiar los mismos datos al mismo tiempo. Las técnicas de bloqueo y concurrencia reducen la posibilidad de conflictos mientras mantienen la integridad de los datos.



El bloqueo evita que otros usuarios y aplicaciones accedan a los datos mientras se actualizan. En algunas bases de datos, el bloqueo se aplica a toda la tabla, lo que crea un impacto negativo en el rendimiento de la aplicación.



Educación que genera cambio



Otras bases de datos, como las bases de datos relacionales de Oracle, aplican bloqueos a nivel de registro, lo que deja disponibles los otros registros dentro de la tabla, lo que ayuda a garantizar un mejor rendimiento de la aplicación.

La concurrencia gestiona la actividad cuando varios usuarios o aplicaciones realizan consultas al mismo tiempo en la misma base de datos. Esta capacidad proporciona el acceso correcto a los usuarios y las aplicaciones de acuerdo con las políticas definidas para el control de datos. (Oracle, s.f.)

Estructura básica de una base de datos

Columna o atributo

Fila o registro

Nombre	CIF	Teléfono
Ejemplo 1	123456789	000000000
Ejemplo 2	987654321	999999999

Garantías de ACID

- **Atomicidad:**
 - La ejecución de cada transacción es atómica:
 - Se realizan todas las acciones o no se realiza ninguna
- **Consistencia:**
 - Cada transacción debe preservar la integridad
 - La base de datos satisfacen todas las restricciones después de una transacción
- **Aislamiento (Isolation):**
 - Una transacción no puede afectar otra
- **Durabilidad:**
 - Una vez que haya un COMMIT, la base de datos debe persistir los cambios



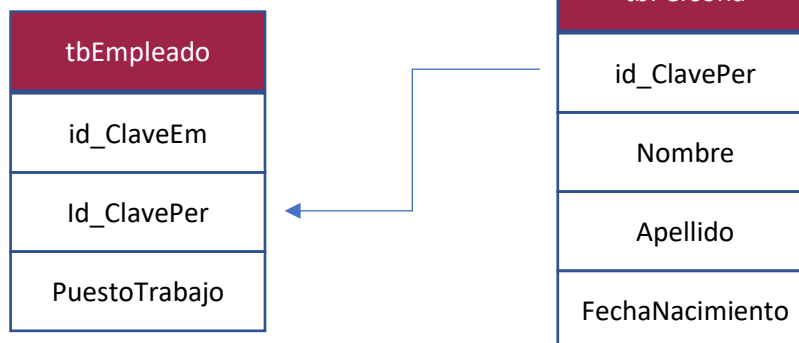
Modelado Básico de Datos

Modelar una base de datos es un proceso que puede tornarse un poco extenso y complejo si este no se hace de forma correcta. Por tal razón es necesario que antes de crear y gestionar base de datos se tenga noción de la forma en que se debe estructurar una base de datos de forma básica. El tema es extenso y con muchas vertientes, para propósito de los contenidos a tratar no se profundizará mucho en el mismo. A continuación, puedes ver el ejemplo de una base de datos llamada **bdTrabajadores** en el cual se almacena información referente de personas que laboran en una empresa. En el ejemplo se puede apreciar que la base de datos está compuesta por dos tablas que se relacionan entre sí.



- La tabla **Persona (tbPersona)** contiene las columnas **id_ClavePer**, **Nombre**, **Apellido**, **FechaNacimiento**.
- La tabla **Empleado (tbEmpleado)** contiene las columnas **id_ClaveEm**, **idClavePer**, **PuestoTrabajo**.

Las columnas que en su nombre llevan la preposición **id**, representan la clave que identifica a cada entidad en la tabla. Como podrás notar en la **tbEmpleado** se encuentran dos id, esto debido a que el primero **id_ClaveEm** hace referencia a la clave única del empleado de la empresa. Esta misma tabla cuenta con un **id_ClavePer**, la cual puedes notar es la misma id que se encuentra en la **tbPersona**. Lo anterior es porque los empleados en la empresa tienen distintos puestos de trabajo, pero todos tienen **Nombre**, **Apellido** y **Fecha de Nacimiento**. Las características descritas anteriormente son la parte fundamental del modelo Relacional.



Base de datos Trabajadores (bdTrabajadores)



Educación que genera cambio

Como se mencionó en lecturas anteriores una entidad es el conjunto de datos que se encuentra en una fila de la tabla. Cada entidad está identificada por una clave única.

Cada fila es una entidad.

Id_ClavePer	Nombre	Apellido	FechaNacimiento
1	Juan de Jesús	Solís Gutierrez	25/12/1999
2	Fernando	Zapata Xicoténcatl	06/12/2004

Id_ClaveEm	Id_ClavePer	PuestoTrabajo
10	2	Gerente de Ventas
20	1	Vendedor

En las tablas que se mostraron anteriormente se puede apreciar que gracias al uso de la relación entre tablas de la Base de Datos se evita la duplicidad de estos, lo que permite que la base de datos sea más eficiente al momento de almacenar y procesar información. De las tablas anteriormente puedes determinar, **¿cuál es el nombre del gerente de Ventas? ¿cuál es el nombre del vendedor?**



Si lograste dar respuesta a las preguntas anteriores, has comprendido una parte fundamental para el modelado de Base de Datos. En el ejemplo mostrado se utilizan pocas tablas y pocos datos en ellas. Pero una base de datos puede almacenar tantas tablas y tantos datos como capacidad de Hardware tenga el Servidor. Cuando se trabaja con tal cantidad de datos en un mismo lugar se hace fundamental tener el orden adecuado para cada elemento.

tabla

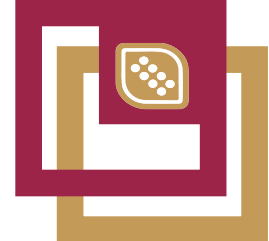
Cve. cliente	Nombre	Direccion	Ciudad	Estado
1	Alfredo Godínez	Fresnillo #47	Veracruz	Veracruz
2	Gabriela Mora	El crespo #81	Guadalajara	Jalisco
3	Alejandra Avalos	Casa Mata #1	Morelia	Michoacan
4	Jaime Quintero	Miraflores #23	Uruapan	Michoacan
5	Carlos Miranda	Rio Bravo #95	Matamoros	Tamaulipas

registro

campo clave

campo





BIBLIOGRAFÍA

AMAZON. (s.f.). Obtenido de ¿Qué es una base de datos Relacional?:

<https://aws.amazon.com/es/relational-database/>

https://cursos.clavijero.edu.mx/cursos/080_bdd/modulo1/contenidos/tema1.4.html?opc=0

IntelDig. (2018). Cómo modelar datos en una BBDD. 2018, de Tecnologías información Sitio web:

<https://www.tecnologias-informacion.com/como-modelar-datos.html#>

Microsoft 365. (2021). Definir una relación en un diagrama de modelo de base de datos. 2021, de

Microsoft Sitio web: <https://acortar.link/4IPTre>

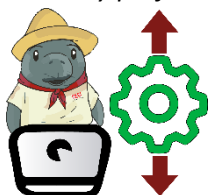
Oracle México. (s.f.). Obtenido de ¿Qué es una base de datos?:

<https://www.oracle.com/mx/database/what-is-database.html>



Práctica 1. Modelado Básico de Bases de Datos

Instrucciones: Modela base de datos relacionales, resolviendo con ellas situaciones del contexto escolar y profesional. Sigue los pasos que se mencionan.



PRÁCTICA 1. "MODELADO BÁSICO DE BASES DE DATOS"

- Agrúpate en equipos de 5 estudiantes y **acuerden una situación del contexto escolar o profesional** en el que se puedan utilizar bases de datos para almacenar y procesar información. Si no logran decidir qué situación pueden representar en una base de datos pide asesoría al docente.
- Una vez determinada la base de datos, deberán realizar el modelado de esta.
- Primero determinen la cantidad de tablas que tendrá tu base de datos, procura que no sean menos de 3 o más de 10 para motivos prácticos.
- Después de tener las tablas, debes determinar cuáles son los datos que deberá contener cada tabla, incluyendo los identificadores de cada una y los que permitan relacionarlas entre sí.
- Usando todo lo anterior, deberás representar la base de datos en el Modelo Relacional.
- Con el modelo relacional elaborado deberán preparar una exposición en la que muestren el modelo obtenido, así como el proceso realizado para obtenerlo. En dicha exposición deberán justificar las decisiones tomadas al momento de realizar el modelaje de la base de datos.



Considera los fundamentos de la normalización de bases de datos



Nota: La exposición deberá realizarle utilizando medios digitales, en caso de no ser posible utilizar un editor de presentaciones electrónicas o alguna otra aplicación, el material para la exposición puede ser elaborado en papel bond o cartulina.





Escribe el contexto de la Base de Datos a diseñar y anota el nombre de las Tablas de Tu Base de Datos, así como los datos que contendrán cada una.

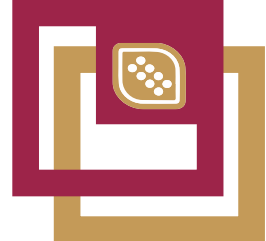
Situación del Contexto de la Base de Datos:



No.	NOMBRE DE LA TABLA	DATOS QUE CONTENDRA LA TABLA
1		
2		
3		
4		
5		
6		
7		
8		
9		
10		

REPRESENTACION DEL MODELO RELACIONAL DE LA BASE DE DATOS





 COBATAB		INSTRUMENTO DE EVALUACIÓN LISTA DE COTEJO 4 Práctica 1: “Modelado Básico de Bases de Datos”					
DATOS GENERALES							
Nombre(s) del alumno(s)					Matrícula(s)		
Producto: Presentación electrónica y/o impresa					Fecha		
Submódulo 1: Desarrollo de Aplicaciones de Escritorio con Base de Datos					Periodo: 2025 - 2026B		
Nombre del docente					Firma del docente		
CRITERIO	INDICADORES	VALOR OBTENIDO		PONDERACIÓN	CALIF	OBSERVACIONES Y/O SUGERENCIAS DE MEJORA	
		SÍ	NO				
1	Las tablas de la BD cuentan con datos identificadores.			1			
2	El estudiante justifica con claridad y precisión su modelo.			1			
3	El modelo cuenta con más de 3 tablas y por lo menos 2 están relacionadas entre sí.			1			
4	Se elabora material para exponer claro y entendible.			1			
5	El modelo presentado representa una base de datos que realmente puede ser creada por los mismos estudiantes.			1			
6	Las diapositivas o presentación electrónica cumplen con un diseño.			2			
7	La información se relaciona con el modelo de datos presentado.			1			
8	Realiza una presentación de forma seria y su información es entendible para el público que lo observa.			2			
	CALIFICACIÓN						





Lectura No 5. Estructura de Base de Datos Relacionales

Instrucciones: Realiza la lectura señalando lo más relevante para realizar la presentación electrónica solicitada.



LECTURA 5. "ESTRUCTURA DE BASE DE DATOS RELACIONALES"



Vamos a ponernos de acuerdo con el vocabulario: no es lo mismo una base que una tabla, ni es lo mismo un registro que un campo.

Un gran repositorio de datos, llamado **base de datos**; es un conjunto de datos estructurados apropiadamente y relacionados entre sí. En la base de datos pueden aparecer diferentes tipos de registros. Éstos, almacenan un **recurso** que es cualquier activo que es de valor para una organización y que incurre en costos.

La estructura de la base de datos, llamado **modelo lógico** de la base de datos es el tercer nivel de discusión. En este nivel se habla acerca de **metadatos**, o datos acerca de datos. Para cada conjunto de entidad en el modelo conceptual, se tiene un **tipo de registro** en el modelo lógico de la base de datos. Por ejemplo, para el conjunto de **entidad Estudiante** en la universidad, se tendría un tipo de registro Estudiante. Un tipo de registro contiene muchos **tipos ítem de datos**, cada uno de los cuales representa un atributo de una entidad. Para el tipo de registro Estudiante, los tipos ítem de datos podrían ser estId, estNombre, direccion, telefono, edad, credits, gpa (promedio) y adviser (consejero). Un ítem de datos es la unidad más pequeña nominada de los datos almacenados. Otras palabras que a veces se usan para los ítems de datos son *elemento de datos*, *campo* o *atributo*.



Equipos - BaseEjemplo - OpenOffice.org Table Data View

IdEquipo	Nombre
1	LocosDelBalón
2	Imparables
3	Simuladores
4	Los mejores
5	Luchadores

Annotations: A red box highlights the entire table content, labeled 'Registro'. A red arrow points to the 'Imparables' cell, labeled 'Dato'. A red arrow points to the 'IdEquipo' column header, labeled 'Campo'.

Δ Campo

Por lo general, **campo** significa un conjunto de bytes adyacentes identificados como la ubicación física para un ítem de datos, así que tiene un significado más físico que el término ítem de datos. **Atributo** por lo general se refiere a una característica de una entidad en el modelo conceptual, pero, dado que hay una correspondencia entre atributos e ítems de datos, con frecuencia las dos palabras son intercambiables. Los ítems de datos en ocasiones se agrupan para formar

agregados de datos, que son grupos nominados de ítems de datos dentro de un registro.



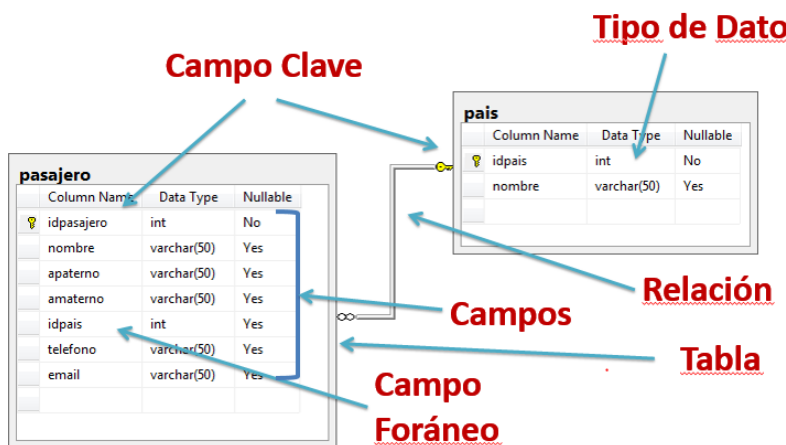
Educación que genera cambio

Para un registro *Empleado*, puede haber un agregado de datos llamado empAdd, que consiste en los ítems de datos *calle*, *ciudad*, *estado* y *codigo postal*. Los agregados de datos permiten hacer referencia al grupo de ítems de datos como un todo o a los ítems individuales en el grupo.

Entenderemos por campo una parte de información que está integrada en una tabla. Por ejemplo, la tabla «Proveedores» puede estar formada por los siguientes campos: Nombre, Apellidos, Dirección, Teléfono, Ciudad, Provincia, Comunidad y Código Postal y el título de éstos se encontraría situado en la parte superior de la ventana.

Es conveniente que sepas que los datos organizados en campos individuales te facilitarán la adquisición de información. A continuación, pasaremos a ver los tipos de campos que nos podemos encontrar:

- ❖ **Campos autonuméricos:** son aquellos que como su propio nombre indica, incluye números. Es un campo que nos puede ser de mucha utilidad pues podemos tener, por ejemplo, a nuestros proveedores de forma ordenada numéricamente, para lo cual crearíamos un campo numérico para cada uno de ellos, que se incrementaría automáticamente en el siguiente proveedor que incluyamos en la tabla.



- ❖ **Campos calculados:** son aquellos que nos van a pedir realizar operaciones numéricas entre los mismos campos. Para poder realizar esto, sólo tendremos que hacer clic sobre una celda de campo vacía y escribir ahí la operación que vamos a realizar.
- ❖ **Campos de búsqueda:** son aquellos que nos facilitan el trabajo a la hora de introducir datos en la hoja de datos de una tabla o formulario. Nos sirve para buscar una información en otro lugar.
- ❖ **Campos de clave primaria:** son aquellos que nos identifican de forma única cada registro de una tabla. Es decir, podremos tener varios proveedores y cada uno de ellos con su identificación (ID). Al añadir el campo ID a la tabla, el Gestor de Base de Datos evitará, por ejemplo, que dos proveedores tengan la misma identificación.

I Registro

Un registro por otro lado es una colección nominada de ítems de datos relacionados y/o agregados de datos. Como se mencionó antes, usualmente hay un tipo de registro para cada conjunto de entidad, y un tipo ítem de dato para cada atributo. Las relaciones también se pueden representar mediante tipos de registro, pero hay otras formas para representar relaciones.

Los registros son los conjuntos de todos los campos de una fila de la tabla. Cada fila o renglón se llama registro. Las bases de datos se organizan en tablas.

Cada registro debe tener un campo que lo identifique de forma única.

O Tablas

Estructura matricial conformada por campos y Registros

Las tablas son los elementos fundamentales y básicos de las bases de datos, en ellas se encuentra almacenada toda la información.

ALUMNO : Tabla				
COD_ALUMNO	CODIGO_GRUPO	AP_ALUMNO	AM_ALUMNO	NOM_ALUMNO
A0001	00004	ABANTO	CARDENAS	YAMIT
A0002	00001	BENITES	FUENTES	MIRIAN
A0003	00001	CASTRO	HUERTA	CARLOS
A0004	00002	BARRIOS	BARRIOS	JUAN
A0005	00002	MARTINEZ	MALAGUER	GINO
A0006	00002	MURILLO	PEREZ	MARTHA

Registros
O
Tuplas

Campos(columnas)

Una tabla está constituida por **registros** que contienen toda la información referida a un elemento u objeto y cada registro está constituido por **campos**, que son las unidades básicas de las bases de datos y almacenan cada uno de los datos de los registros.

§ Redundancia de datos

Cuando se almacena en una base de datos, la información se integra de modo que múltiples copias de los mismos datos no se almacenan a menos que sea necesario. Se permite alguna redundancia limitada para mantener las conexiones lógicas entre los ítems de datos o para mejorar el rendimiento. Para poner un caso, en el ejemplo de universidad que se discutió, la ID del estudiante aparecía tanto en la tabla Student como en la tabla Enroll.



El sistema de gestión de la base de datos “sabe acerca” de dicha repetición. Una base de datos de ordinario no tiene múltiples copias de registros enteros, a diferencia de un sistema de archivos, donde distintos departamentos podían tener duplicados de archivos enteros.

FIGURA 1.1(a)

La tabla Student

Student				
stuld	lastName	firstName	major	credits
S1001	Smith	Tom	Historia	90
S1002	Chin	Ann	Mat	36
S1005	Lee	Perry	Historia	3
S1010	Burns	Edward	Arte	63
S1013	McCarthy	Owen	Mat	0
S1015	Jones	Mary	Mat	42
S1020	Rivera	Jane	CSC	15

Enroll		
stuld	classNumber	grade
S1001	ART103A	A
S1001	HST205A	C
S1002	ART103A	D
S1002	CSC201A	F
S1002	MTH103C	B
S1010	ART103A	
S1010	MTH103C	
S1020	CSC201A	B
S1020	MTH101B	A

FIGURA 1.1(d)

La tabla Enroll

Consistencia de datos

Un efecto de eliminar o controlar la redundancia es que los datos son consistentes. Si un ítem de datos aparece sólo una vez, cualquier actualización a su valor necesita realizarse sólo una vez, y todos los usuarios tendrán acceso al mismo nuevo valor. Si el sistema tiene cierta redundancia controlada, cuando recibe una actualización a un ítem que aparezca más de una vez con frecuencia puede realizar actualizaciones en cascada. Esto significa que automáticamente actualizará cada ocurrencia de dicho ítem, lo que mantiene consistente a la base de datos. Por ejemplo, si se cambia la ID de un estudiante en la tabla Student, los registros Enroll para dicho estudiante se actualizarán para mostrar la nueva ID en forma automática.

REDUNDANCIA E INCONSISTENCIA DE LOS DATOS

Redundancia.
La misma información es almacenada varias veces en la misma base de datos

PERSONAL		
Cedula	Nombre	Dirección
7492837	Ana	Calle 1
9836384	Pedro	Avenida 3
8038464	José	Urb. Los ...
PROFESORES		
Cedula	Nombre	Dirección
7492837	Ana Maria	Calle 55
9836384	Pedro	Avenida 3

Inconsistencia.
Consiste en que no todas las copias redundantes contienen la misma información





Lectura No 6. Comandos SQL y Lenguaje de Manipulación de Datos

Instrucciones: Realiza la lectura señalando lo más relevante para realizar la presentación electrónica solicitada.



LECTURA 6. "COMANDOS SQL Y LENGUAJE DE MANIPULACIÓN DE DATOS"



Lenguaje de manipulación de datos.

Un lenguaje de manipulación de datos (Data Manipulation Language) o DML en inglés) es un lenguaje proporcionado por el sistema de gestión de base de datos que permite a los usuarios llevar a cabo las tareas de consulta o manipulación de los datos, organizados por el modelo de datos que se adecuado. El lenguaje de manipulación de los datos más popular hoy en día es SQL usado para recuperar y manipular datos en una base de datos relacional.

Clasificación de los DML.

El lenguaje de manipulación de datos se clasifica en dos grandes grupos:

- ❖ Lenguajes de consulta procedimentales. En este tipo de lenguaje el usuario da instrucciones al sistema para que realice una serie de procedimientos u operaciones en la base de datos para calcular un resultado final.
- ❖ Lenguajes de consulta no procedimentales. En este tipo de lenguaje el usuario describe la información deseada sin un procedimiento específico para obtener esa información.

DML

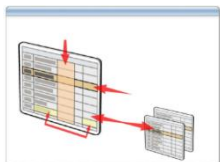
INSERT
UPDATE
DELETE
SELECT



Lenguaje SQL.

La programación SQL permite interactuar con una base de datos. El lenguaje de consulta estructurado SQL es el lenguaje de base de datos más implementado y valioso para cualquier persona involucrada en la programación informática o que usa bases de datos para recopilar y organizar información.





La programación SQL se puede usar para compartir y administrar datos, en particular la información que está organizada en tablas que se encuentra en los sistemas de administración de bases de datos relacionales. Mediante el uso de SQL, se puede:

- ❖ Consultar, actualizar y reorganizar datos.
- ❖ Crear y modificar la estructura de un sistema de base de datos.
- ❖ Controlar el acceso a sus datos.

Puede parecer algo similar al funcionamiento de una hoja de cálculo, el objetivo de SQL es diferente, ya que permite compilar y administrar datos en volúmenes mayores. Mientras que las hojas de cálculo pueden volverse complicadas con demasiada información que llena muchas celdas, las bases de datos SQL permiten gestionar hasta miles de millones de datos.

Alguien que es experto en programación SQL puede almacenar datos acerca de cada cliente con el que haya trabajado su empresa, desde características clave hasta detalles sobre ventas; recuperando, de forma instantánea cualquier información que sea requerida sobre él.

El lenguaje de consulta estructurado y todo lo relacionado con la programación SQL son el día de hoy, aspectos claves para las empresas.

SQL por sus siglas en inglés significa Lenguaje de Consulta Estructurada (Structured Query Language), es un lenguaje de programación diseñado para actualizar, obtener y calcular información en bases de datos relacionales.



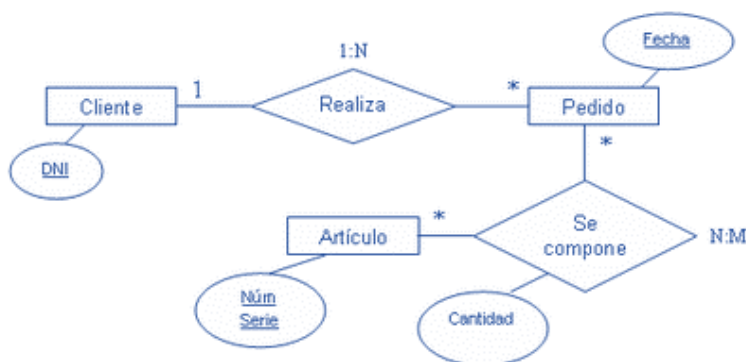
De acuerdo con ANSI (American National Standards Institute), es el lenguaje estándar para sistemas de gestión de bases de datos relacionales. Las sentencias SQL se utilizan para realizar tareas como actualizar datos en una base de datos o recuperar datos de una base de datos. Algunos sistemas de gestión de bases de datos relacionales comunes que utilizan SQL son: Oracle, MySQL, Microsoft SQL Server, Access, Ingres, etc.

La mayoría de los sistemas de base de datos utilizan SQL, la mayoría de ellos también tienen sus propias extensiones propietarias que usualmente sólo se utilizan en su sistema. Sin embargo, los comandos SQL estándar como Seleccionar, Insertar, Actualizar, Eliminar, Crear y Eliminar se pueden utilizar para lograr casi todo lo que se necesita hacer con una base de datos.

Un sistema de base de datos relacional contiene uno o más objetos denominados tablas. Los datos o información para la base de datos se almacenan en estas tablas. Las tablas se identifican de forma única por sus nombres y están formadas por columnas y filas. Las columnas contienen el nombre de la columna, el tipo de datos y cualquier otro atributo para la columna. Las filas contienen los registros o datos de las columnas.



Al crear una base de datos, se usan tablas separadas para diferentes tipos de entidades. Pero también necesitamos tener relaciones entre estas tablas. Estas relaciones deben estar representadas en la base de datos. Cuando se recupera datos con SQL, se necesita utilizar ciertos tipos de consultas JOIN para obtener lo que se necesite.



Hay varios tipos de relaciones de bases de datos:

- ❖ Relaciones uno a uno.
- ❖ Una a muchas y muchas a una.
- ❖ Muchas a muchas.
- ❖ Relaciones de autoreferencia.

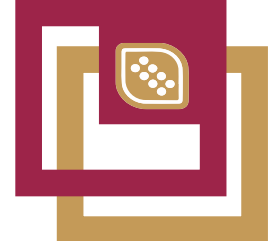
Al seleccionar datos de varias tablas con relaciones, se utiliza la consulta **JOIN**. Hay varios tipos de JOIN:

- ❖ Cross Joins.
- ❖ Natural Joins.
- ❖ Inner Joins.
- ❖ Left (Outer) Joins.
- ❖ Right (Outer) Joins.

Las bases de datos más comunes basadas en SQL son:

- ❖ **MySQL**: es una base de datos SQL de código abierto, desarrollada por una empresa sueca **MySQL AB**.
- ❖ **Oracle**: es un sistema de gestión de bases de datos relacional desarrollado por Oracle Corporation.
- ❖ **Access**: Es un software de gestión de base de datos de nivel de entrada.





Comandos de menú

Aunque SQL se considera un sublenguaje debido a su naturaleza de no procesamiento, aun así, es un lenguaje completo que le permite crear y mantener objetos en una base de datos, asegurar esos objetos y manipular la información dentro de los objetos. Un método común usado para categorizar las instrucciones SQL es dividir las de acuerdo con las funciones que realizan. Basado en este método, SQL se separa en tres tipos de instrucciones:

◆ Lenguaje de definición de datos (DDL, Data Definition Language)

Las instrucciones DDL se usan para crear, modificar o borrar objetos en una base de datos como tablas, vistas, esquemas, dominios, activadores, y almacenar procedimientos. Las palabras clave en SQL más frecuentemente asociadas con las instrucciones DDL son **CREATE**, **ALTER** y **DROP**. Por ejemplo, se usa la instrucción *CREATE TABLE* para crear una tabla, la instrucción *ALTER TABLE* para modificar las características de una tabla, y la instrucción *DROP TABLE* para borrar la definición de la tabla de la base de datos.

◆ Lenguaje de control de datos (DCL, Data Control Language)

Las instrucciones DCL permiten controlar quién o qué (un usuario en una base de datos puede ser una persona o un programa de aplicación) tiene acceso a objetos específicos en la base de datos. Con DCL, puede otorgar o restringir el acceso usando las instrucciones **GRANT** o **REVOKE**, los dos comandos principales en DCL. Las instrucciones DCL también permiten controlar el tipo de acceso que cada usuario tiene a los objetos de una base de datos. Por ejemplo, puede determinar cuáles usuarios pueden ver un conjunto de datos específico y cuáles usuarios pueden manipular esos datos.

◆ Lenguaje de manipulación de datos (DML, Data Manipulation Language)

Las instrucciones DML se usan para recuperar, agregar, modificar o borrar datos almacenados en los objetos de una base de datos. Las palabras clave asociadas con las instrucciones DML son **SELECT**, **INSERT**, **UPDATE** y **DELETE**, las cuales representan los tipos de instrucciones que probablemente son más usadas. Por ejemplo, puede usar la instrucción *SELECT* para recuperar datos de una tabla y la instrucción *INSERT* para agregar datos a una tabla.



Crear una tabla

- Vamos a crear una tabla llamada empleado en alguna base de datos, por ejemplo test.

```
test=> CREATE TABLE empleado (
test(>     nombre VARCHAR(20),
test(>     apellido VARCHAR(20),
test(>     sueldo FLOAT,
test(>     sexo CHAR(1),
test(>     edad INTEGER,
test(>     transaccion TIMESTAMP DEFAULT
test(>     CURRENT_TIMESTAMP,
test(>     ingreso DATE);
```

COMANDOS TIPO DDL	
COMANDO	DEFINICION
CREATE	Crea nuevas bases de datos, tablas, campos e índices.
DROP	Elimina bases de datos, tablas, campos e índices.
ALTER	Modifica estructura, diseño de una base de datos

COMANDOS TIPO DML	
COMANDO	DEFINICION
SELECT	Consulta registros dentro de la base de datos
INSERT	Inserta o agrega datos en una tabla o entidad
DELETE	Elimina registros de una entidad bajo ciertos criterios.
UPDATE	Modifica o actualiza los datos existentes dentro de las entidades.





Las principales funcionalidades de SQL como lenguaje de definición son la creación, modificación y borrado de las tablas que componen la base de datos, así como de los índices, vistas, sinónimos, permisos, etc., que se puedan definir sobre las mismas. Algunos comandos para el trabajo básico con tablas son:

- ❖ **CREATE TABLE:** Se utiliza para crear una tabla.

```
CREATE DATABASE nbBasedeDatos
[ ON
    [ PRIMARY ] [ <esp_fichero> [ ,...n ]
    [ , <grupo> [ ,...n ] ]
[ LOG ON { < esp_fichero > [ ,...n ] } ]
]
[ COLLATE nbintercalacion]
[ WITH <external_access_option> ]
];
```



CREATE [TEMPORARY] TABLE [nombre de la tabla] ([definiciones de columna]) [parámetros de la tabla].

```
CREATE TABLE "nombre_tabla"
("columna 1" "tipo_de_datos_para_columna_1",
"columna 2" "tipo_de_datos_para_columna_2",
... );
```



- ❖ **SHOW TABLES:** Se utiliza para mostrar tablas.
- ❖ **DROP TABLE** <nombre de tabla>: Sirve para mostrar la estructura de una tabla.

```
DROP {DATABASE | SCHEMA} [IF EXISTS] db_name
```

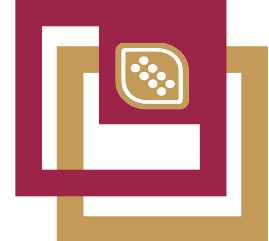
```
DROP TABLE NOMBRE_DE_TABLA;
```

DROP TABLE no puede utilizarse para eliminar una tabla a la que se hace referencia en una restricción **FOREIGN KEY**. Primero hay que eliminar la restricción **FOREIGN KEY** de la tabla a la que se hace referencia. No se puede utilizar la instrucción **DROP TABLE** en tablas del sistema.

- ❖ **ALTER TABLE:** Se utiliza para modificar la definición (estructura) de una tabla mediante la modificación de la definición de sus columnas. El comando ALTER se utiliza para realizar las siguientes funciones: a) añadir, modificar columnas de la tabla, b) agregar y quitar restricciones, c) habilitar y deshabilitar las restricciones. La sintaxis para agregar una columna es

```
ALTER TABLE table_name ADD column_name datatype;
```





La sintaxis para eliminar una columna es:

```
ALTER TABLE table_name DROP column_name;
```

La sintaxis para modificar una columna es:

```
ALTER TABLE table_name MODIFY column_name datatype;
```

- ❖ **ALTERN COLUMN:** Especifica que la columna con nombre va a cambiarse o modificarse. La columna que se vaya a modificar no puede ser ninguna de las siguientes:
 - A) timestamp data type. Una columna con un tipo de datos timestamp.
 - B) La columna ROWGUIDCOL de la tabla.
 - C) Una columna calculada o usada en una columna calculada.
 - D) varchar, nvarchar, or, varbinary, data type, the data type is not changed, and the new size is equal to or greater than the old size, or if the column is changed from not null to null. Se usa en las estadísticas generadas por la instrucción CREATE STATISTICS a menos que la columna sea del tipo de datos varchar, nvarchar o varbinary, que el tipo de datos no se cambie y que el nuevo tamaño sea igual o mayor que el tamaño anterior, o que la columna se cambie de NOT NULL a NULL. Se tiene que quitar primero las estadísticas con la instrucción DROP STATISTICS. Las estadísticas generadas automáticamente por el optimizador de consultas se quitan automáticamente con ALTER COLUMN.
 - E) Una columna utilizada en una restricción PRIMARY KEY o [FOREIGN KEY] REFERENCES.
 - F) Una columna utilizada en una restricción CHECK o UNIQUE. Se permite el cambio de longitud de una columna de longitud variable en una restricción CHECK o UNIQUE.
 - G) Una columna asociada a la definición predeterminada. No obstante, la longitud, precisión o escala de una columna puede cambiarse si el tipo de datos no se cambia.

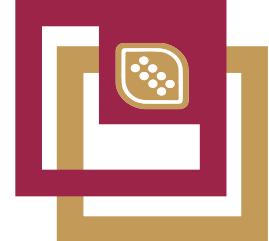


Tipos de comandos de SQL.

Para comprender lo que puede hacer una base de datos SQL hay que conocer los diferentes tipos de comandos que tiene. Los comandos del lenguaje SQL se dividen según su función en estos 5 tipos:

- ❖ DDL (Data Language): definen el esquema o estructura de la base de datos. Ejemplos: CREATE (crear), ALTER (alterar), DROP (eliminar objetos), RENAME (renombrar), TRUNCATE (quita todos los registros de una tabla, incluidos los espacios de los registros eliminados), COMMENT (comentar) entre otros.
- ❖ DQL (Data Query Language): sirven para hacer consultas sobre los datos en el esquema de objetos. Un objeto puede ser desde un resultado de búsqueda a una tabla. El propósito del comando es el de establecer una relación, basada en la consulta, dentro de la estructura de la base de datos, como la función de búsqueda. Ejemplo: SELECT (recuperar registros de la base de datos).





- ❖ DML (Data Manipulation Language): tratan la manipulación de datos presentes en la base de datos. La mayoría de los comandos pertenecen a este tipo. Ejemplo: INSERT (insertar un objeto), DELETE (eliminar registros), UPDATE (actualizar), CALL (llamar), MERGE (3 en 1, inserta, elimina y actualiza), LOCK TABLE (bloquear tabla), EXPLAIN PLAN (determina el plan de acceso).
- ❖ DCL (Data Control Language): se encargan de los derechos, los permisos y otros controles del sistema de la base de datos. Ejemplo: GRANT (proporcionar privilegios, acceso a un usuario), REVOKE (revocar el derecho de acceso dado a un usuario), etc.
- ❖ TCL (Transaction Control Language): sirve para las transacciones con la base de datos. Con estos comandos se puede llevar un control sobre otros comandos y cómo afectan a la base de datos. Ejemplo: COMMIT (llevar a cabo una transacción), ROLLBACK (revertir una transacción en caso de que ocurra algún error), SAVEPOINT (establecer un punto de rescate dentro de una transacción)

Comandos básicos de SQL.

Todos los tipos de SQL tienen algunos comandos básicos. Los más comunes son:

- ❖ SELECT: Permite seleccionar los datos para descargar.
- ❖ WHERE: Permite seleccionar que filtro aplicar a los datos que se van a descargar.
- ❖ INSERT: Permite insertar datos.
- ❖ DELETE: Permite borrar datos.
- ❖ UPDATE: Permite actualizar los datos.



SELECT | SELECCIONAR.

La sentencia **SELECT** como se mencionó nos permite consultar los datos almacenados en una tabla de la base de datos.

Forma básica:

```
SELECT [ALL | DISTINCT ]
      <nombre_campo> [{,<nombre_campo>}]
FROM <nombre_tabla>|<nombre_vista>
     [{,<nombre_tabla>|<nombre_vista>}]
[WHERE <condición> [{ AND|OR <condición>}]]
[GROUP BY <nombre_campo> [{,<nombre_campo> }]]
[HAVING <condición> [{ AND|OR <condición>}]]
[ORDER BY <nombre_campo>|<indice_campo> [ASC | DESC]
        [{,<nombre_campo>|<indice_campo> [ASC | DESC ]}]]
```



- ❖ **SELECT.** Palabra clave que indica que la sentencia SQL que se desea ejecutar es de selección.
- ❖ **ALL.** Indica que se desea seleccionar todos los valores. es el valor por defecto y no suele especificarse casi nunca.
- ❖ **DISTINCT.** Indica que queremos seleccionar sólo los valores distintos.
- ❖ **FROM.** Indica la tabla o tablas desde la que queremos recuperar los datos. En el caso, de que existan más de una tabla se denomina a la consulta “consulta combinada” o “join”. En estas consultas combinadas es necesario aplicar una condición de combinación a través de una cláusula WHERE.
- ❖ **WHERE.** Especifica una condición que debe cumplirse para que los datos sean devueltos por la consulta. Admite los operadores lógicos AND y OR.
- ❖ **GROUP BY.** Especifica la agrupación que se da a los datos. Se usa siempre en combinación con funciones agregadas.
- ❖ **HAVING.** Especifica una condición que debe cumplirse para que los datos sean devueltos por la consulta. Su funcionamiento es similar al de WHERE pero aplicado al conjunto de resultados devueltos por la consulta. Debe aplicarse siempre junto a **GROUP BY** y la condición debe estar referida a los campos contenidos en ella.
- ❖ **ORDER BY.** Presenta el resultado ordenado por las columnas indicadas. El orden puede expresarse con **ASC** (orden ascendente) y **DESC** (orden descendente). El valor predeterminado es **ASC**.

Para formular una consulta a la tabla Coches y recuperar los campos matrícula, marca, modelo, color, numero, kilómetros, num_plazas, se tiene que ejecutar la siguiente consulta. Los datos se devuelven ordenados por marca y por modelo en orden ascendente, de menor a mayor. La palabra clave **FROM** indica que los datos son recuperados de la tabla Coches.

SELECT Matrícula, marca, modelo, color, numero_kilometros, num_plazas **FROM** Coches **ORDER BY** marca,modelo;

También se puede realizar la Consulta simplificada a través de un comodín de Campos (*). El uso del asterisco indica que se quiere que la consulta devuelva todos los campos que existen en la tabla y los datos serán devueltos ordenados por marca y modelo.

SELECT * FROM Coches **ORDER BY** marca,modelo;

Cláusula WHERE.

Es la instrucción que nos permite filtrar el resultado de una sentencia **SELECT**. Habitualmente no se desea obtener toda la información existente en la tabla, lo que se desea es obtener sólo la información que nos resulte útil en ese momento. La cláusula **WHERE** filtra los datos antes de ser devueltos por la consulta.

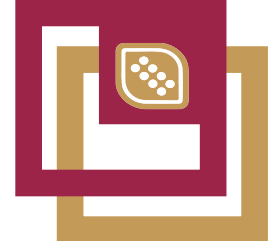
Cuando en la cláusula **WHERE** queremos incluir un tipo texto, se tiene que incluir dicho valor entre comillas simples. Cuando se desea consultar un coche en concreto, se agrega una cláusula **WHERE**.

Ejemplo

```
SELECT nombres, apellidos, edad
FROM estudiantes
WHERE edad > 21
```

	NOMBRES	APELLIDOS	EDAD
1	Esteban	Riquelme	22
2	Ernesto	Cardenas	25





En donde se especifica una o varias condiciones que deben cumplirse para que la sentencia **SELECT** devuelva los datos. En este caso, la consulta devolverá sólo los datos del coche con matrícula para que la consulta devuelva sólo los datos del coche con matrícula MF-234-ZD o bien la matrícula FK-965-LZ. Se puede utilizar la cláusula **WHERE** sola o en combinación con tantas condiciones como se quiere.

SELECT Matrícula, marca, modelo, color, numero_kilometro, num_plazas **FROM** Coches **WHERE** Matrícula = 'MF-234-ZD' **OR** Matrícula = 'FK-965-LZ';

La condición **WHERE** puede ser negada a través del operador lógico **NOT**. La siguiente consulta devolverá todos los datos de la tabla Coches, menos el que tenga la matrícula MF-234-ZD.

SELECT Matrícula, marca, modelo, color, numero_kilometros, num_plazas **FROM** Coches **WHERE** **NOT** Matrícula = 'MF-234-ZD';

SELECT Matrícula, marca, modelo, color, numero_kilometros, num_plazas
FROM Coches
WHERE Matrícula = 'MF-234-ZD'
OR Matrícula = 'FK-938-ZL';

La siguiente consulta utiliza la condicional **DISTINCT**, la cual devolverá todos los valores distintos formados por los campos marca y modelo de la tabla Coches.

SELECT DISTINCT marca, modelo **FROM** Coches;

Clausula ORDER BY.

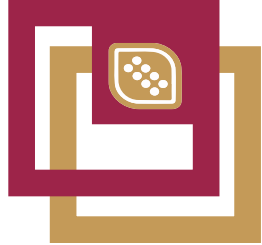
La cláusula **ORDER BY** nos permite especificar el orden en el que son devueltos los datos. Se puede especificar la ordenación ascendente o descendente a través de las palabras clave **ASC** y **DESC**. La ordenación depende del tipo de datos que esté definido en la columna, de esta forma un campo numérico será ordenado como tal, u un alfanumérico se ordena de la "A" a la "Z", a pesar de que su contenido sea numérico. El valor predeterminado es **ASC** pero se puede cambiar especificándolo al realizar la consulta.

SELECT Matrícula, marca, modelo, color, numero_kilometros, num_plazas **FROM** Coches
ORDER BY marca '**ASC**', modelo '**DESC**';

SELECT	Selección de datos
INSERT UPDATE DELETE MERGE	DML: Lenguaje de manipulación de datos
CREATE ALTER DROP RENAME TRUNCATE	DDL: Lenguaje de definición de datos
COMMIT ROLLBACK SAVEPOINT	Control de transacciones
GRANT REVOKE	DCL: Lenguaje de control de datos

En esta consulta, se seleccionan todos los campos de la tabla Coches, ordenándolos por los campos macar y modelo, en donde marca está en forma ascendente y modelo en forma descendente.





SELECT Matrícula, marca, modelo, color, numero_kilometros, num_plazas **FROM** Coches **ORDER BY** 2;

En esta consulta selecciona todos los campos de la tabla Coches, los ordena por el campo marca, ya que aparece en segundo lugar dentro de la lista de campos que componen la **SELECT**.

INSERT | INSERTAR.

Una sentencia **INSERT** de **SQL** agrega uno o más registros a una y sólo una tabla en una base de datos relacional.

```
INSERT INTO 'tablatura' ('columna1', ['columna2,... '])  
VALUES ('valor1', ['valor2,...'])
```

O tambien se puede utilizar como:

```
INSERT INTO tablatura VALUES ('valor1','valor2')
```

La cantidad de columnas y valores tienen que ser iguales. Si una columna no se especifica, se le asigna el valor por omisión. Los valores especificados, se le asigna el valor por omisión. Los valores especificados o implícitos por la sentencia **INSERT** deben satisfacer todas las restricciones que pueden ser aplicadas. Si hay error de sintaxis o si alguna de las restricciones es incorrecta, no se agrega la fila y devuelve error.



INSERT INTO agenda_telefonica (nombre, numero) **VALUES** ('Roberto Aguirre', 4886850);

Cuando se especifican todos los valores de una tabla, se puede utilizar la sentencia corta:

INSERT INTO nombreTabla **VALUES** ('valor1, ['valor2...']);

Para ello supongamos que en la tabla 'agenda_telefonica', 'nombre' y 'numero' son las únicas columnas.

INSERT INTO agenda_telefonica **VALUES** ('Jhonny Aguilar', 080473968);

Una característica de **SQL** (desde **SQL-92**) es el uso de constructores fijos para insertar múltiples filas a la vez, con una sola sentencia **SQL**.

INSERT INTO "tabla" ("columna", ["columna2..."]) **VALUES** ("valor1a", ["valor1b..."]), ("value2a", ["value2b,..."]),...:

Esta característica está soportada por **DB2**, **PostgreSQL** (desde la versión 8.2), **MySQL** y **H2**.





Para ello supongamos que en la tabla 'agenda_telefonica', 'nombre' y 'numero' son las únicas columnas.

INSERT INTO agenda_telefonica **VALUES** ('Roberto Aguirre', 4886850'); ('Alejandro Solís', 4556550');

Lo cual se puede hacer por las sentencias:

INSERT INTO agenda_telefonica **VALUES** ('Roberto Aguirre', '4886850'); **INSERT INTO** agenda_telefonica **VALUES** ('Alejandro Solís', '4556550');

UPDATE | ACTUALIZAR.

Esta sentencia es utilizada para modificar los valores de un conjunto de registros existentes en una tabla.

```
UPDATE My_table SET field1 = 'updated value asd' WHERE field2 = 'N';
```

DELETE | BORRAR.

Esta sentencia es utilizada para borrar uno o más registros existentes en una tabla.

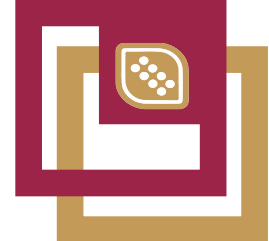
```
DELETE FROM tabla WHERE columna1 = 'valor1';
```

Ejemplo:

DELETE FROM My_table **WHERE** field2 = 'N';

```
SELECT [{ALL|DISTINCT}]
<nombre_campo>[, <nombre_campo>...]
FROM {<nombre_tabla>|<nombre_vista>}[,
{<nombre_tabla>|<nombre_vista>}...]
[WHERE <condición> [{AND|OR} <condición>...]]
[GROUP BY <nombre_campo>[, <nombre_campo>...]]
[HAVING <condición> [{AND|OR} <condición>...]]
[ORDER BY {<nombre_campo>|<indice_campo>} [{ASC|DESC}][,
{<nombre_campo>|<indice_campo>} [{ASC|DESC}]]];
```





BIBLIOGRAFÍA.

Comandos SQL. (2018). Obtenido de 4D doc center: <https://doc.4d.com/4Dv15/4D/15/Comandos-SQL.200-2288063.es.html>

Comandos SQL Básicos. (2020). Obtenido de MariaDB: <https://mariadb.com/kb/es/basic-sql-statements/>

Conceptos básicos de SQL. (s.f.). Obtenido de Geotalleres: https://geotalleres.readthedocs.io/es/latest/conceptos-sql/conceptos_sql.html

Date, C. J. (2001). Introducción a los sistemas de bases de datos. Séptima edición. Editorial Pearson México. ISBN: 978-9684444195

Jacobs, Paige. (2018). SQL. Guía completa para principiantes de la programación SQL con ejercicios y estudio de casos. Edición Kindle. Kindle Amazon. México.

Oppel, A., & Sheldon, R. (2010). Fundamentos de SQL. Ciudad de México: McGraw-Hill.

Ricardo, C. M. (2009). Base de Datos. Ciudad de México: Mc Graw Hill.

Rob, Peter, Coronel, Carlos. (2003). Sistemas de bases de datos: diseño, implementación y administración. Primera edición. Editorial Cengage Learning Editores S.A. de C.V. México. ISBN: 9789706862860

SQL Commands. (12 de noviembre de 2020). Obtenido de postgresql: <https://www.postgresql.org/docs/9.1/sql-commands.html>

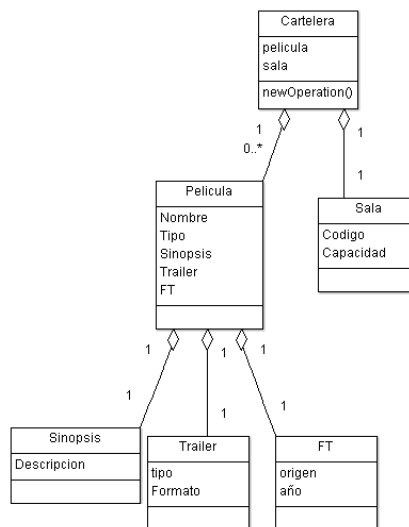


ACTIVIDAD 4. PRESENTACIÓN ELECTRÓNICA. COMANDOS SQL Y LENGUAJE DE MANIPULACIÓN DE DATOS

Instrucciones: Al finalizar la lectura y señalar lo más relevante, realiza la presentación electrónica de la misma.

Trabajo en equipo:

- ❖ Considera el mismo equipo que trabajaste en la práctica 1, elije uno de los tipos de instrucciones SQL o de acuerdo con la organización que tiene el docente.
- ❖ Una vez determinado los acuerdos, empezar con el diseño de la presentación electrónica.
- ❖ Primero elijan el diseño de dicha presentación.
- ❖ Incluir imágenes y la información más relevante de la lectura.
- ❖ Incluye instrucciones de SQL con ejemplos.
- ❖ Integra el modelo diseñado y corregido de la práctica 1.
- ❖ Con la presentación elaborada preparar una exposición en la que muestren: Presentación, Índice, Justificación, Introducción, el desarrollo de la presentación, Conclusión y Bibliografía.
- ❖ El tiempo de exposición es de 10 minutos máximo.



Nota: La exposición deberá realizarle utilizando medios digitales, en caso de no ser posible utilizar un editor de presentaciones electrónicas o alguna otra aplicación, el material para la exposición puede ser elaborado en papel bond o cartulina.





INSTRUMENTO DE EVALUACIÓN
LISTA DE COTEJO 5
Actividad 4. Presentación electrónica.
Comando SQL y lenguaje de manipulación de datos



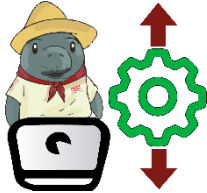
DATOS GENERALES

Nombre(s) del alumno(s)				Matrícula(s)		
Producto: Presentación electrónica y/o impresa				Fecha		
Submódulo 1: Desarrollo de Aplicaciones de Escritorio con Base de Datos				Periodo: 2025 - 2026B		
Nombre del docente				Firma del docente		
CRITERIO	INDICADORES	VALOR OBTENIDO		PONDERACIÓN	CALIF	OBSERVACIONES Y/O SUGERENCIAS DE MEJORA
		SÍ	NO			
1	La presentación cuenta con diseño y datos de identificación.			2		
2	El estudiante justifica con claridad y precisión su presentación.			2		
3	La presentación tiene índice, introducción, desarrollo.			2		
4	La presentación es clara, entendible y con imágenes acordes al tema.			2		
5	La exposición es clara, entendible y con el tiempo solicitado.			2		
CALIFICACIÓN						



Ejemplo Guiado con código SQL “Hola Mundo de Base de Datos”

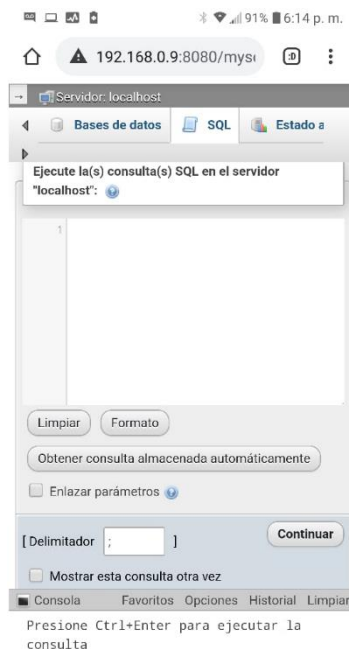
Instrucciones: Realiza el siguiente ejemplo respetando cada una de las indicaciones.



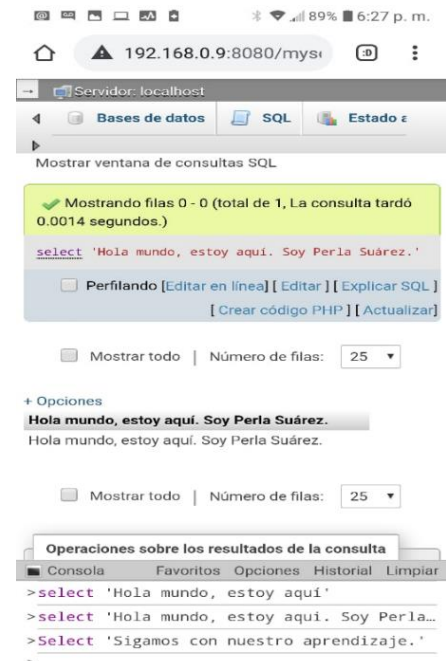
EJEMPLO GUIADO 1. “HOLA MUNDO DE BASE DE DATOS”



- ❖ Ingresar XAMPP, AWebServer o <https://paiza.io/es>.
- ❖ Guarda la base de datos con el nombre **SQLHolaMundoJAFP4H** en donde JAFP son las iniciales de tu nombre y H el grupo.
- ❖ Al estar en la pantalla principal dar clic en SQL.
- ❖ Aparece la siguiente pantalla:
- ❖ En esta pantalla escribir lo siguiente:
Select ‘Hola mundo, estoy aquí. Soy Perla Suárez.’;
Select ‘Es un gusto saludarte.’;
Select ‘Sigamos con nuestro aprendizaje.’;
- ❖ En donde en lugar de *Perla Suárez* escribirás tu nombre.
- ❖ Se ejecuta la consulta y se muestra lo siguiente:

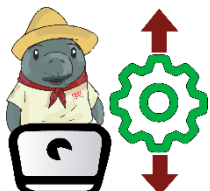


- ❖ Ahora agrega dos mensajes más.
- ❖ Ejecuta la consulta.
- ❖ Agrega otro mensaje.
- ❖ Ejecuta la consulta



Práctica 2. Creación de Base de Datos

Instrucciones: Utiliza el software XAMPP, AWebServer o Paiza para crear una base de datos con lo estudiado anteriormente.



PRÁCTICA 2. "CREACIÓN DE BASE DE DATOS"



Conocimientos:

- ❖ XAMPP, AWebServer o Paiza.
- ❖ Tablas
- ❖ Atributos
- ❖ Consultas.

Software a utilizar.

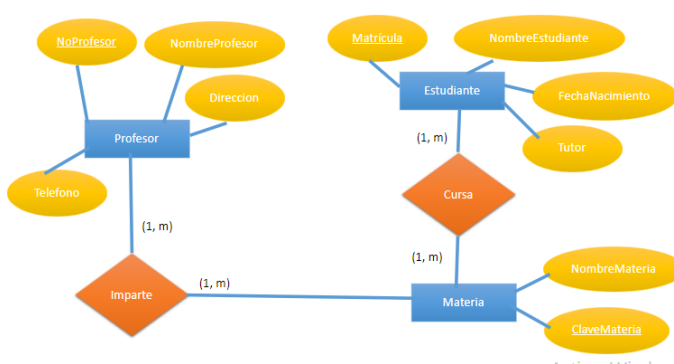
XAMPP, AWebServer o Paiza.

Situación planteada.

Realizar la base de datos para llevar el control de los Profesores que imparten las materias y los estudiantes que cursan dichas materias. En donde se tiene los siguientes datos:

- ❖ Profesor (NoProfesor, Nombre, Direccion, Telefono. CURP)
- ❖ Materia (ClaveMateria, Nombre, NoProfesor)
- ❖ Estudiante (Matrícula, Nombre, FechaNacimiento, Tutor, CURP)

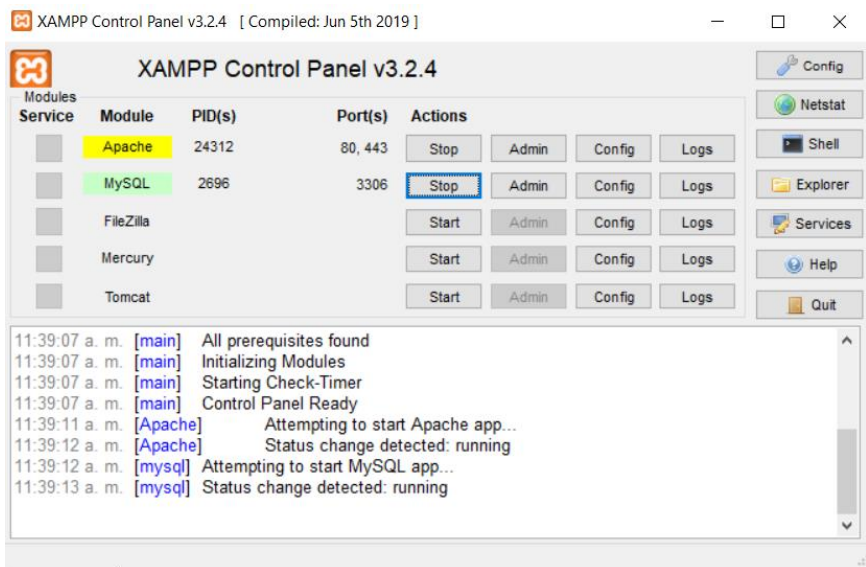
En donde un Profesor imparte varias Materias y las Materias pueden ser impartidas por varios Profesores. El modelo Entidad-Relación es el siguiente.



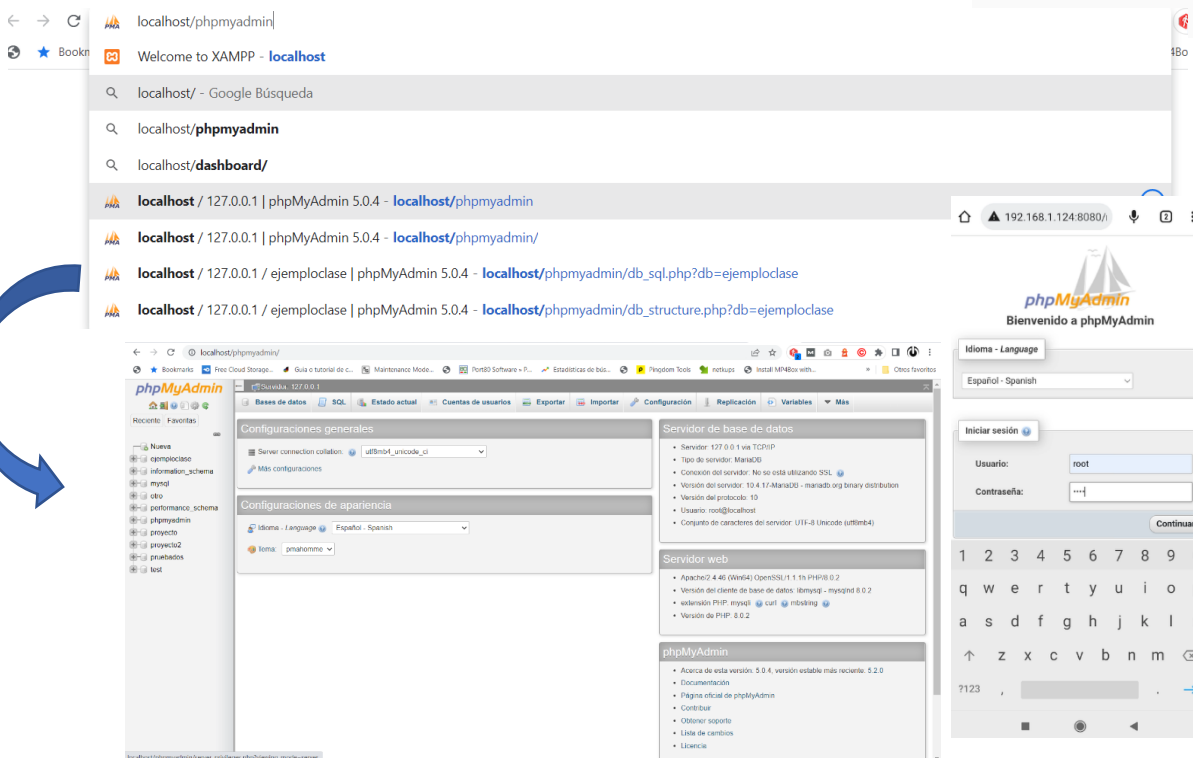
Educación que genera cambio

Procedimiento.

- ❖ Ingresar a XAMPP, AWebServer o Paiza.



- ❖ Crear una base de datos nueva con el nombre **Practica2CreacionBDJAFP4H** en donde JAFP son las iniciales de tu nombre y H el grupo.



- ❖ Crea la tabla Profesor con sus atributos. Debe quedar de la siguiente forma.

Consola Favoritos Opciones Historial Limpiar

Presione Ctrl+Enter para ejecutar la consulta

```
>CREATE TABLE `Practica2`.`Profesor` ( `No...
```

- ❖ Crea la tabla Materia con sus atributos.
- ❖ Crea la tabla Estudiante con sus atributos.
- ❖ Crea la tabla Cursa con los siguientes datos:
Cursa (ClaveMateria, Matrícula)
- ❖ Introduce 7 datos a cada tabla. No olvidando que las tablas están relacionadas por NoProfesor, ClaveMateria y Matrícula.
- ❖ Crea la consulta MateriasProfesor en donde se muestre lo siguiente:
Consulta MateriasProfesor(ClaveMateria, NombreMateria, NoProfesor, NombreProfesor)
La cual debe llevar la siguiente estructura:
SELECT ClaveMateria, NombreMateria, NoProfesor, NombreProfesor FROM Materia, Profesor);
- ❖ Crear la consulta MateriasEstudiante en donde se muestra lo siguiente:
Consulta MateriasEstudiante(ClaveMateria, NombreMateria, Matrícula, NombreEstudiante)
- ❖ Crear la consulta ProfesorMateriaEstudiante en donde se muestre lo siguiente:
Consulta ProfesorMateriaEstudiantes(NoProfesor, NombreProfesor, ClaveMateria, NombreMateria, Matrícula, NombreEstudiante).





INSTRUMENTO DE EVALUACIÓN
LISTA DE COTEJO 6
Práctica 2. Creación de Base de Datos

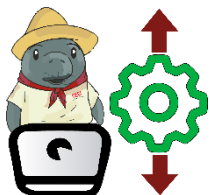


DATOS GENERALES						
Nombre(s) del alumno(s)				Matrícula(s)		
Producto: Documento electrónico y/o impreso				Fecha		
Submódulo 1: Desarrollo de Aplicaciones de Escritorio con Base de Datos				Periodo: 2025 - 2026B		
Nombre del docente				Firma del docente		
CRITERIO	INDICADORES	VALOR OBTENIDO		PONDERACIÓN	CALIF	OBSERVACIONES Y/O SUGERENCIAS DE MEJORA
		SÍ	NO			
1	Crea la base de datos con XAMPP, AWebServer o https://paiza.io/es .			2		
2	Crea las tablas indicadas.			2		
3	Crea la tabla de relación Cursa.			2		
4	Introduce en cada una de las tablas los registros solicitados.			2		
5	Realiza todas las consultas solicitadas con la instrucción SELECT de SQL.			2		
CALIFICACIÓN						



Ejemplo Guiado con código SQL “Manejo de Registros”

Instrucciones: Realiza el siguiente ejemplo respetando cada una de las indicaciones.



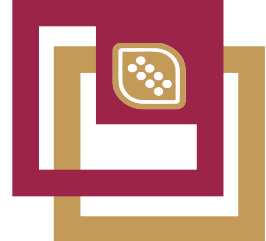
EJEMPLO GUIADO 2. “MANEJO DE REGISTROS”

❖ Ingresar XAMPP, AWebServer o Paiza.

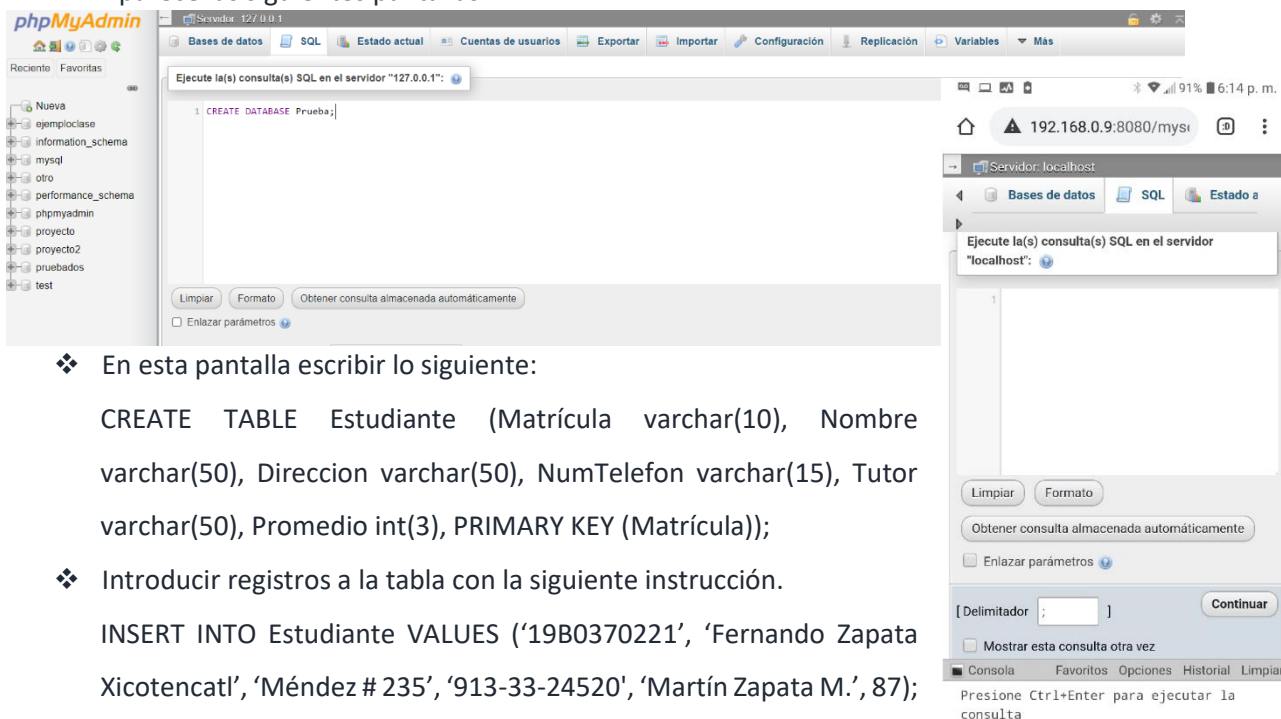
- ❖ Guarda la base de datos con el nombre **SQLInBoMoCoJAFP4H** en donde JAFP son las iniciales de tu nombre y H el grupo.

- ❖ Al estar en la pantalla principal dar clic en **SQL**.





❖ Aparece las siguientes pantallas:



❖ En esta pantalla escribir lo siguiente:

CREATE TABLE Estudiante (Matrícula varchar(10), Nombre varchar(50), Direccion varchar(50), NumTelefon varchar(15), Tutor varchar(50), Promedio int(3), PRIMARY KEY (Matrícula));

❖ Introducir registros a la tabla con la siguiente instrucción.

INSERT INTO Estudiante VALUES ('19B0370221', 'Fernando Zapata Xicotencatl', 'Méendez # 235', '913-33-24520', 'Martín Zapata M.', 87);

Introducir 5 registros más.

❖ Realizar las siguientes consultas.

SELECT Matrícula, Nombre, Direccion, Tutor FROM Estudiante;

SELECT Matrícula, Nombre, Promedio FROM Estudiante;

❖ Modificar el registro de la siguiente forma:

UPDATE Estudiante SET Direccion='Juarez # 1', Promedio=92 WHERE Matrícula='19B0370221';

❖ Borra el registro con la siguiente instrucción.

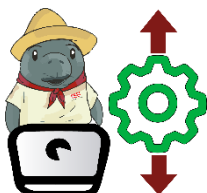
DELETE FROM Estudiante WHERE Matrícula='19B0370221';

❖ De esta forma con instrucciones SQL dentro de XAMPP o AWebServer, se crean tablas, se insertan, consultan, modifican y eliminan registros de la tabla. Recuerden que en la aplicación hay que dar clic en SQL para hacer todas las operaciones indicadas.



Práctica 3. Base de Datos con Instrucciones SQL

Instrucciones: Utiliza el software XAMPP, AWebServer o Paiza para crear una base de datos con instrucciones SQL.



PRÁCTICA 3. "BASE DE DATOS CON INSTRUCCIONES SQL"

Conocimientos:

- ❖ XAMPP, AWebServer o Paiza.
- ❖ SQL
- ❖ Tablas
- ❖ Atributos
- ❖ Crear tablas.
- ❖ Insertar, consultar, modificar y eliminar registros con SQL



Software a utilizar.

XAMPP, AWebServer o Paiza.



Situación planteada.

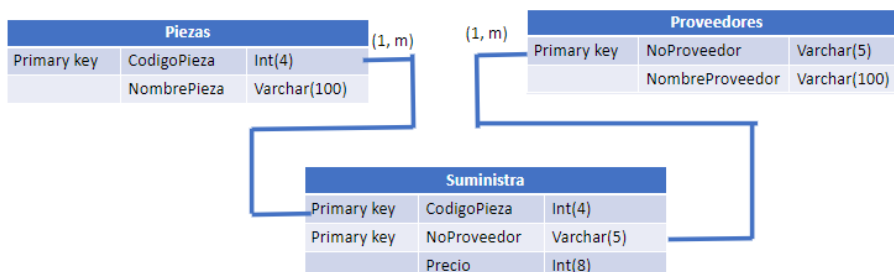
Realizar la base de datos para llevar el control de las Piezas que Suministran los Proveedores. En donde se tiene los siguientes datos:

- ❖ Piezas (Codigo, Nombre)
- ❖ Suministra (CodigoPieza, NoProveedor, Precio)
- ❖ Proveedores (NoProveedor, Nombre)

En donde un Proveedor Suministra varias Piezas.



La relación en las tablas es la siguiente:



Procedimiento.

- ❖ Ingresar a XAMPP, AWebServer o Paiza.
- ❖ Crear una base de datos nueva con el nombre Practica3BDInstruccionesSQLJAFP4H en donde JAFP son las iniciales de tu nombre y H el grupo.
- ❖ La creación de tablas, así como la inserción, consulta, modificación y eliminación de registros es con instrucciones SQL.
- ❖ Crea la tabla Piezas con sus atributos.
- ❖ Crea la tabla Proveedores con sus atributos.
- ❖ Crea la tabla Suministra con sus atributos.
- ❖ Introduce 10 datos a cada tabla.
- ❖ Hacer una consulta para mostrar los nombres de todas las piezas.
- ❖ Hacer una consulta para mostrar todos los proveedores.
- ❖ Hacer una consulta para obtener el nombre de los proveedores que suministran una pieza X y el precio de la misma.
- ❖ Hacer una consulta para obtener el nombre de las piezas suministradas por el proveedor cuyo NoProveedor sea X. La X es por un NoProveedor que exista en la Tabla.
- ❖ Insertar en la tabla Suministra los siguientes datos: NoProveedor X, CodigoPieza X, Precio=7.



- ❖ Modificar el registro anterior cambiando el Precio a 12.
- ❖ Borrar un registro en la tabla Piezas.
- ❖ Hacer una consulta para mostrar todo el contenido de la tabla Suministra.





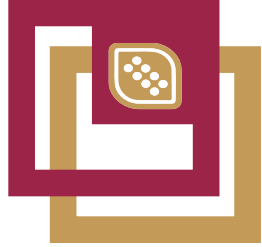
INSTRUMENTO DE EVALUACIÓN
LISTA DE COTEJO 7
Práctica 3. Base de Datos con Instrucciones SQL



DATOS GENERALES

Nombre(s) del alumno(s)				Matrícula(s)		
Producto: Documento electrónico y/o impreso				Fecha		
Submódulo 1: Desarrollo de Aplicaciones de Escritorio con Base de Datos				Periodo: 2025 - 2026B		
Nombre del docente				Firma del docente		
CRITERIO	INDICADORES	VALOR OBTENIDO		PONDERACIÓN	CALIF	OBSERVACIONES Y/O SUGERENCIAS DE MEJORA
		SÍ	NO			
1	Crea la base de datos con XAMPP, AwebServer o Paiza.			1		
2	Crea las tablas indicadas con la instrucción CREATE TABLE.			2		
3	Introduce en cada una de las tablas los registros solicitados con la instrucción INSERT INTO.			2		
4	Realiza todas las consultas solicitadas con la instrucción SELECT de SQL.			2		
5	Modifica el registro solicitado con la instrucción UPDATE.			2		
6	Elimina el registro solicitado con la instrucción DELETE.			1		
CALIFICACIÓN						





Actividad 5. ELEMENTOS DE UN PROYECTO BD SQL



Instrucciones: En equipo de 6 integrantes aplicarán los elementos básicos de un proyecto de base de datos para resolver un caso sencillo, integrando conceptos de diseño conceptual, lógico y físico.

Caso propuesto:

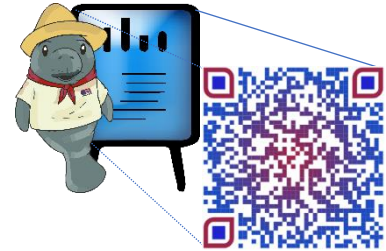
Diseña una base de datos para un sistema de gestión de una tienda de videojuegos. La base de datos debe permitir:

- Registrar los videojuegos (nombre, género, consola, precio).
- Registrar los clientes (nombre, teléfono, correo electrónico).
- Registrar las ventas (cliente, videojuego comprado, fecha de venta).

Procedimiento.

❖ Portada del Proyecto.

- Crea una portada que incluya:
 - Nombre del proyecto.
 - Nombres completos.
 - Fecha de entrega.



❖ Introducción.

- Escribe un párrafo breve con el objetivo del proyecto.

❖ Diseño conceptual.

- Identifica las entidades (Videojuegos, Clientes, Ventas).
- Define sus atributos principales.
- Representa las relaciones entre ellas en un pequeño diagrama Entidad-Relación (puede hacerlo a mano o con herramientas como Lucidchart o papel cuadriculado).

❖ Diseño lógico.

- Define un esquema básico de tablas.



❖ **Consultas básicas.**

- *Escribe dos ejemplos de consultas SQL que puedan realizarse en la base de datos.*
Por ejemplo: Mostrar todos los videojuegos disponibles.

❖ **Entrega Final.**

- *Los estudiantes entregarán un documento (en Word o PDF) que incluya los elementos mencionados: portada, introducción diagrama conceptual y ejemplos de consultas SQL.*

❖ **Producto esperado:**

- *Un documento que incluya:*
 - *Una portada con los datos básicos.*
 - *Una breve introducción con el objetivo del proyecto.*
 - *Un diagrama ER.*
 - *Un esquema lógico de las tablas.*
 - *Dos consultas SQL simples.*





INSTRUMENTO DE EVALUACIÓN
LISTA DE COTEJO 8
Actividad 5. Elementos de un proyecto BD SQL.



DATOS GENERALES						
Nombre(s) del alumno(s)				Matrícula(s)		
Producto: Documento electrónico y/o impreso				Fecha		
Submódulo 1: Desarrollo de Aplicaciones de Escritorio con Base de Datos				Periodo: 2025 - 2026B		
Nombre del docente				Firma del docente		
CRITERIO	INDICADORES	VALOR OBTENIDO		PONDERACIÓN	CALIF	OBSERVACIONES Y/O SUGERENCIAS DE MEJORA
		SÍ	NO			
1	El documento cuenta con diseño y datos de identificación.			2		
2	El documento presenta una introducción clara y acorde al contenido.			2		
3	Presenta el diseño conceptual a través del diagrama Entidad-Relación.			2		
4	Integra el diseño lógico mediante las tablas y los campos.			2		
5	Muestra al menos dos consultas a través de las sentencias SQL.			2		
CALIFICACIÓN						



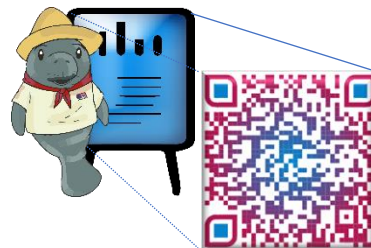


Lectura No 6. Informes

Instrucciones: Lee en forma cuidadosa la lectura subraya las ideas principales y al finalizar realiza la Actividad 6. “Generemos información”.



LECTURA 6. “INFORMES”



Los informes se usan para dar formato a los datos resumirlos y presentarlos. Un informe responde a una pregunta específica como: ¿cuánto dinero se recibe por cada cliente en este año? o ¿en qué ciudad residen nuestros vendedores? A cada informe se le puede dar formato para presentar la información de la manera más clara posible.

Se puede ejecutar un informe y siempre se reflejan datos actuales de la base de datos. Se les da formato a los informes para imprimirlos, pero también puede verse en pantalla, exportarse a otro programa o enviarse como datos adjuntos en un correo electrónico.

Los informes constituyen un modo de ver, formatear y resumir la información contenida en la base de datos. Se puede crear un sencillo informe de números de teléfono de todos los contactos o un informe del total de las ventas en distintos estados y períodos de tiempo.

Un informe es un objeto de base de datos que resulta útil para presentar la información de la base de datos con alguno de los siguientes propósitos:

- ❖ Mostrar o distribuir un resumen de los datos.
- ❖ Archivar instantáneas de los datos.
- ❖ Aportar detalles sobre un registro concreto.
- ❖ Crear etiquetas.



Es posible crear un informe “no enlazado” que no muestre datos, pero para se da por sentado que un informe está enlazado a un origen de datos como una tabla o consulta. El diseño de un informe está enlazado a un origen de datos como una tabla o consulta. El diseño de un informe se divide en secciones. Para crear mejores informes, se tiene que comprender cómo funciona cada sección.



A continuación, se muestra un resumen de las secciones y sus usos:

Sección	Cómo se muestra la sección al imprimirse	Dónde se puede usar la sección
Encabezado del informe	Al inicio del informe	El encabezado del informe se usa para ofrecer información que normalmente aparecería en una página de portada, como un logotipo o un título y una fecha. Cuando se coloca un control calculado que usa la función de agregado de suma en el encabezado del informe, el resultado de la suma se calcula para todo el informe. El encabezado del informe se imprime antes del encabezado de página.
Encabezado de página	Al principio de cada página	El encabezado de página se usa para repetir el título del informe en todas las páginas
Encabezado de grupo	Al inicio de cada grupo de registros nuevos	El encabezado de grupo se usa para imprimir el nombre del grupo. En un informe que se agrupa por producto se usa el encabezado de grupo para imprimir el nombre del producto. Cuando se coloca un control calculado que usa la función de agregado de suma en el encabezado de grupo, la suma se calcula para el grupo actual. Se puede tener varias secciones de encabezado de grupo en un informe en función del número de niveles de agrupación que se haya agregado.
Detalle	Aparece una vez por cada fila del origen de registros	Aquí se colocan los controles que constituyen el cuerpo principal del informe
Pie del grupo	Al final de cada grupo de registros	El pie de grupo se usa para imprimir información de resumen de un grupo. Puede tener varias secciones de pie de grupo en un informe en función del número de niveles de agrupación que se haya agregado.
Pie de página	Al final de cada página	El pie de página se usa para imprimir números de página o información sobre cada página.
Pie del informe	Al final del informe	El pie de informe se usa para imprimir totales de los informes u otra información de resumen de todo el informe.

Es más sencillo crear informes significativos cuando la base de datos está bien estructurada.

Los informes proporcionan de forma útil la información que se encuentra en la base de datos, de forma similar a las consultas. Los informes se generan desde las tablas o consultas de la base de datos. Pueden contener todos los campos de la tabla o consulta, o sólo un grupo seleccionado de campos. Los informes pueden ser estáticos o



dinámicos. Los informes estáticos contienen los datos de los campos seleccionados en el momento de generar el informe mientras que los dinámicos se pueden actualizar para que muestren los últimos datos.

Por ejemplo, un informe de gastos del mes anterior es un informe estático porque está basado en datos específicos que no cambian. Sin embargo, un informe de datos de combustible es un informe dinámico porque depende de datos que cambian.

Todos los informes se basan en una tabla o consulta única. Por ello, se debe decidir que campos quiere usar en el informe. Si se quiere usar campos de tablas diferentes, antes hay que combinar los campos en una consulta, después se puede crear un informe basado en esa consulta.

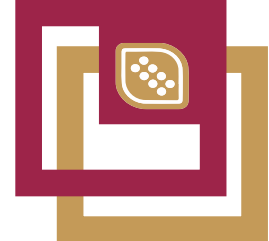


Contesta las siguientes preguntas.

1. ¿Qué es un informe y cuál es su función en una base de datos?

¿Por qué se menciona que el informe está enlazado?





¿Para qué sirve el encabezado de grupo en un informe?

BIBLIOGRAFÍA.

Date, C. J. (2001). *Introducción a los Sistemas de Base de Datos*. PEARSON, Prentice Hall.

Manual de referencia de MySQL 5.7

Silberschatz, A. (2002). *Fundamentos de Base de Datos*. España: Mc Graw Hill.

Tutorial. Optimización de consultas en MySQL.



Actividad 6. Generemos Información



Instrucciones: Realiza la Lectura “Informes” de forma detallada subraya los más importante anotando las ideas principales y con base en ello analiza y resuelve lo siguiente.

Software para utilizar.

XAMP, AWebServer o Paiza.



Procedimiento.

- ❖ Ingresa a XAMP, AWebServer o Paiza.
- ❖ Crea la base de datos con el nombre Curso actual. **Da clic en Crear.**

- ❖ Crea la tabla con el nombre Mensajes_curso.

- ❖ Introduce los siguientes campos con el tipo de dato que se especifica.

Nombre	Tipo	Longitud/Valores	Predeterminado	Cotejamiento	Atributos
Id	INT		Ninguno		
Nombre	VARCHAR		Ninguno		
Email	VARCHAR		Ninguno		
Mensaje	TEXT		Ninguno		

Nombre de la tabla: Mensajes_curso Agregar 1 columna(s) Continuar

Longitud/Valores	Predeterminado	Cotejamiento	Atributos	Nulo	Índice	Comentarios
4	Ninguno			☐	PRIMARY ☒	
60	Ninguno			☐		
60	Ninguno			☐		
	Ninguno			☐		

❖ Ahora pulsamos el botón Grabar o Guardar.

Nombre de la tabla: Mensajes_curso Agregar 1 columna(s) Continuar

60	Ninguno			☐		
60	Ninguno			☐		
	Ninguno			☐		

Procesando petición

Cotejamiento: Motor de almacenamiento: InnoDB

Previsualizar SQL Guardar

❖ La pantalla que se muestra es la siguiente.

Examinar Estructura SQL Buscar Insertar Exportar Importar Privilegios Operaciones Más

Estructura de tabla Vista de relaciones

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra	Acción
1	Id	int(4)			No	Ninguna		AUTO_INCREMENT	Cambiar Eliminar Más
2	Nombre	varchar(60)	utf8mb4_general_ci		No	Ninguna			Cambiar Eliminar Más
3	Email	varchar(60)	utf8mb4_general_ci		No	Ninguna			Cambiar Eliminar Más
4	Mensaje	text	utf8mb4_general_ci		No	Ninguna			Cambiar Eliminar Más

Seleccionar todo Para los elementos que están marcados: Examinar Cambiar Eliminar Primaria Único Índice

Texto completo Agregar a columnas centrales Eliminar de las columnas centrales

Imprimir Planteamiento de la estructura de tabla Hacer seguimiento a la tabla Mover columnas Normalizar

Agregar 1 columna(s) después de Mensaje Continuar

Educación que genera cambio

- ❖ Para insertar un primer registro con datos dentro de los campos de nuestra tabla, damos clic en Insertar.

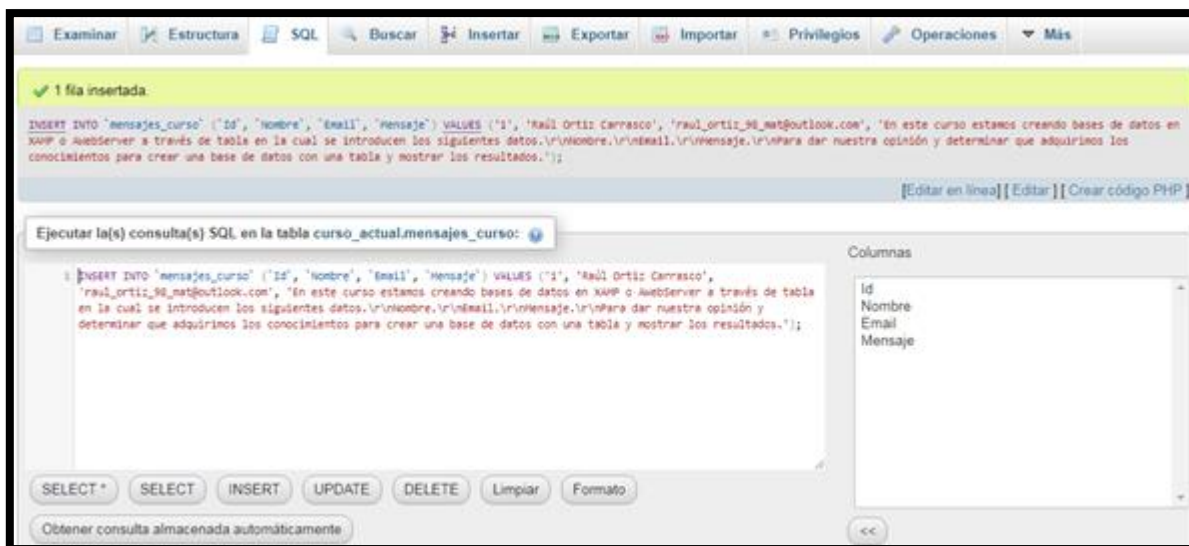
The screenshot shows a database management interface with a menu bar at the top containing 'Examinar', 'Estructura', 'SQL', 'Buscar', 'Insertar' (highlighted with a red circle), 'Exportar', 'Importar', 'Privilegios', 'Operaciones', and 'Más'. Below the menu bar, there are tabs for 'Estructura de tabla' and 'Vista de relaciones'. A table structure is displayed with columns: '#', 'Nombre', 'Tipo', 'Cotejamiento', 'Atributos', 'Nulo', 'Predeterminado', 'Comentarios', 'Extra', and 'Acción'. The table has four rows: 1. 'Id' (int(4)), 2. 'Nombre' (varchar(60)), 3. 'Email' (varchar(60)), and 4. 'Mensaje' (text). The 'Email' row is highlighted in yellow and shows 'Cargando...' in the 'Atributos' column. Below the table, there are various action buttons like 'Seleccionar todo', 'Examinar', 'Cambiar', 'Eliminar', 'Primaria', 'Único', 'Índice', and a 'Continuar' button at the bottom right.

- ❖ Se introduce el primer registro con los siguientes datos.

The screenshot shows the 'Insertar' (Insert) form in the same database management interface. The form has columns for 'Columna', 'Tipo', 'Función', 'Nulo', and 'Valor'. The 'Id' column has a value of '1'. The 'Nombre' column has a value of 'Raúl Ortiz Carrasco'. The 'Email' column has a value of 'raul_ortiz_98_mat@outlook.com'. The 'Mensaje' column is empty. A text box on the right side of the form contains the following text: 'En este curso estamos creando bases de datos en XAMP o localhost a través de tabla en la cual se introducen los siguientes datos. nombre. Email. Mensaje. Para dar nuestra opinión y determinar que adquirimos los conocimientos para crear una base de datos con una tabla y mostrar los resultados.' A 'Continuar' button is located at the bottom right of the form.

- ❖ Damos clic en el botón continuar y aparece lo siguiente.

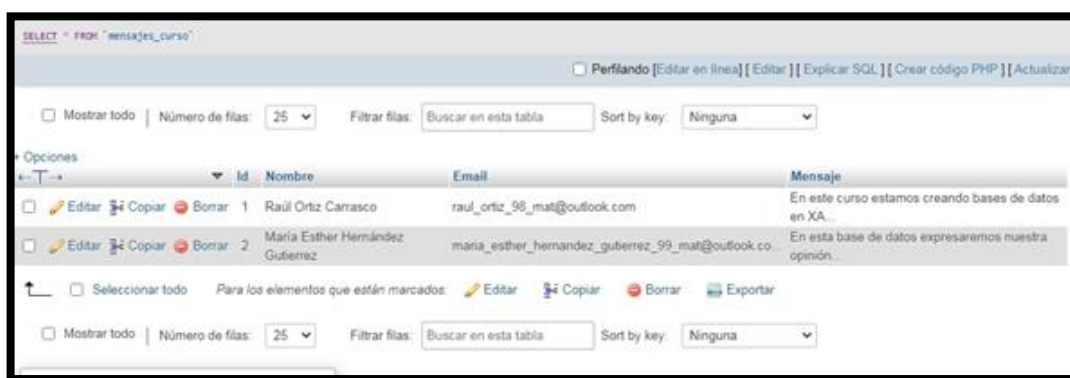




- ❖ Introduce 7 registros más en la tabla. Por medio de la instrucción Insert de SQL.



- ❖ Al finalizar dar clic en Examinar y después en imprimir para que muestre como quedarían los datos.



SELECT * FROM `mensajes_curso`			
Id	Nombre	Email	Mensaje
1	Raúl Ortiz Carrasco	raul_ortiz_98_mat@outlook.com	En este curso estamos creando bases de datos en XA...
2	María Esther Hernández Gutierrez	maria_esther_hernandez_gutierrez_99_mat@outlook.co...	En esta base de datos expresaremos nuestra opinión...

Volver Imprimir

- ❖ Este sería un informe.





INSTRUMENTO DE EVALUACIÓN
LISTA DE COTEJO 9
Actividad 6. Informe.



DATOS GENERALES

Nombre(s) del alumno(s)				Matrícula(s)		
Producto: Documento electrónico y/o impreso				Fecha		
Submódulo 1: Desarrollo de Aplicaciones de Escritorio con Base de Datos				Periodo: 2025 - 2026B		
Nombre del docente				Firma del docente		
CRITERIO	INDICADORES	VALOR OBTENIDO		PONDERACIÓN	CALIF	OBSERVACIONES Y/O SUGERENCIAS DE MEJORA
		SÍ	NO			
1	El documento cuenta con diseño y datos de identificación.			2		
2	El documento presenta las capturas de la creación del informe.			2		
3	Los comandos para realizar el informe.			2		
4	Presenta los datos organizados y con un formato entendible.			2		
5	Muestra los registros considerados de la base de datos.			2		
CALIFICACIÓN						





SIGA

DataLab: Creando Conexiones



Propósito:

Elaborar en equipos de 6 integrantes, un Modelo de Base de Datos relacional donde apliquen los fundamentos y el lenguaje SQL para crear un proyecto que analice datos ambientales, con el fin de generar propuestas tecnológicas que contribuyan al cuidado del medio ambiente de su comunidad e interpretar la información utilizando el software manejador de base de datos y socializarlo en el aula.

Instrucciones: Con los conocimientos que has adquirido a lo largo del submódulo, es momento de diseñar en equipo, un modelo de base de datos que les permita utilizar SQL para crear conexiones entre datos y revelar patrones invisibles a simple vista, contribuyendo con un proyecto tecnológico de impacto comunitario y se realice la gestión de recursos; para recolectar la información necesaria desarrollar su Sistema de base de datos.

¿Qué debemos hacer?



Considera los datos, las tablas y las relaciones que tienen para analizar la información que se requiere.

Considera el diseño y modelo para la base de datos que deben realizar.



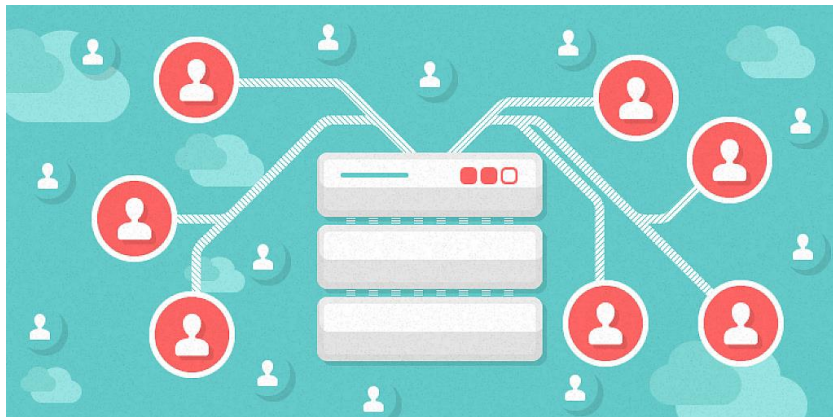
Seguidamente utiliza el software proporcionado o el que consideres pertinente para crear las tablas junto a las relaciones. Puedes utilizar el código SQL para lograrlo.



Educación que genera cambio

Después elaboren el (los) formulario (s) que consideren pertinentes para poder almacenar, eliminar, consultar o modificar los registros dentro de las tablas. Recuerda que los formularios son los que permiten interactuar con la Base de Datos.

Para finalmente, es necesario diseñar los informes; con ellos es posible presentar los datos de manera digital o impresa,



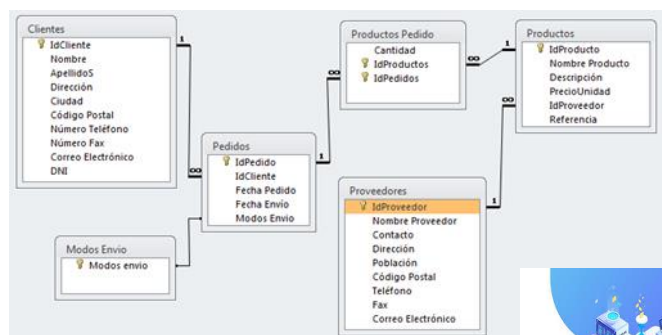
dando respuesta a las solicitudes que el usuario requiere analizar.

Es momento de crear la base de datos.



Así como las entidades con sus respectivos atributos.

Sus relaciones.





COLEGIO DE BACHILLERES DE TABASCO PLANTEL No _____

RÚBRICA DE EVALUACIÓN REPORTÁNDOME



Capacitación Desarrollo de software	Módulo II. Programación Web	Submódulo 1. Desarrollo de Aplicaciones de Escritorio con Base de Datos
--	--------------------------------	--

Situación didáctica. DataLab: Creando Conexiones.		Nombre de los estudiantes:	
Docente:			
Semestre. 4to	Grupo.	Turno.	Fecha.
Competencia Laboral Básica			
CLBDS3. Estructura con acompañamiento de su docente, bases de datos relacionadas con situaciones sociales de su comunidad utilizando la programación y mostrando un comportamiento metódico, además de organizado, con la finalidad de promover un pensamiento crítico y analítico.			
Evidencia de aprendizaje. Documento electrónico / Base de datos			

Instrucciones: Marque con una (X) para resaltar si cumplió con los criterios de la actividad. Sume los puntos para obtener la calificación.

INDICADORES	%	CRITERIOS			
		EXCELENTE	BUENO	SUFICIENTE	INSUFICIENTE
Creación de base de datos	20%	Inserta 4 tablas con los campos bien definidos y el tipo de dato de cada uno de ellos. Así como su longitud. 20	Inserta al menos 3 tablas solicitadas con la mayoría de los campos bien definidos y el tipo de dato de cada uno de ellos. Así como su longitud. 15	Inserta al menos 2 tablas con algunos campos y el tipo de dato de cada uno de ellos. Así como su longitud. 10	Inserta al menos 1 tabla con pocos campos y sus tipos de datos. 5
Inserción y eliminación de datos.	30%	Realiza la inserción de 17 registros y la eliminación de 2 registros con lenguaje SQL. 30	Realiza la inserción de 13 registros y la eliminación de 2 registros con lenguaje SQL. 25	Realiza la inserción de 9 registros y la eliminación de 2 registros con lenguaje SQL. 15	Realiza la inserción de 5 registros y la eliminación de 1 registro con lenguaje SQL. 5



INDICADORES	%	CRITERIOS			
		EXCELENTE	BUENO	SUFICIENTE	INSUFICIENTE
Consultas en la base de datos	20%	Crea las 2 consultas con los datos solicitados en lenguaje SQL y respetando la estructura de una consulta. 20	Crea las 2 consultas con el 80% de los datos solicitados en lenguaje SQL y respetando la estructura de una consulta. 15	Crea las 2 consultas con el 50% de los datos solicitados en lenguaje SQL y respetando la estructura de una consulta. 10	Crea las 2 consultas con menos del 50% de los datos solicitados. 5
Informes en la base de datos	20%	Genera los 2 informes con los datos solicitados respetando su estructura. 20	Genera los 2 informes con el 80% de los datos solicitados respetando su estructura. 15	Genera los 2 informes con el 50% de los datos solicitados respetando su estructura. 10	Genera los 2 informes con menos del 50% de los datos solicitados. 5
Todo el equipo ha participado en la creación de la base de datos	10%	Todos y cada uno de los miembros del equipo han participado y colaborado en la creación de la base de datos. 10	La mayoría de los integrantes del equipo han participado y colaborado en la creación de la base de datos. 8	Pocos miembros del equipo han participado y colaborado en la creación de la base de datos. 5	La construcción de la base de datos ha sido realizada por un solo miembro del equipo y los demás observan. 3
Total	100%	100	83	50	23

Firma del evaluador. _____

Conflicto cognitivo

¿Cómo las redes sociales almacenan y gestionan millones de perfiles de usuarios, publicaciones, comentarios, imágenes y conexiones entre personas?

¿Cómo se pueden relacionar los datos?

¿Qué beneficios tiene el uso de las bases de datos y el lenguaje SQL para el trabajo de proyectos medio ambientales?



Submódulo II. Programación Web



Propósito del Módulo

Elaborar diferentes bases de datos de forma responsable, mediante software de programación para desarrollar soluciones a diferentes problemáticas del contexto y desarrollar sitios web creativos, funcionales mediante software de diseño, que coadyuven en la transmisión de información en la vida cotidiana, mostrando un comportamiento responsable, aceptando diferentes puntos de vista.



Sugerencias de Evidencia de Evaluación

- ❖ Establece HTML y CSS en forma creativa, para la construcción de una página web; demostrando respeto por las opiniones de los demás y aportando ideas en la solución de problemas.
- ❖ Propone el software de aplicación para el diseño web, demostrando su creatividad e innovación, para la construcción de páginas web, siendo responsable de sus decisiones y de la comunicación de información en distintos contextos.

Competencia Laboral Básica

CLBDS4. Construye sitios web creativos y funcionales mediante software de diseño web, para transmitir información a gran escala de forma responsable y empática en diversos contextos.



DOSIFICACIÓN PROGRAMÁTICA

DESARROLLO DE SOFTWARE

Formación Laboral Básica: Desarrollo De Software

Módulo II: Programación Web

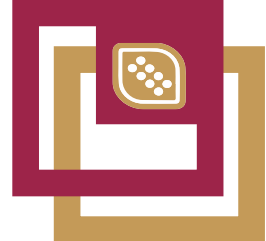
Submódulo II. Programación Web: C4PW

Semestre: 4to. Turno: Matutino/Vespertino Periodo: 2025 - 2026B



Submódulo	Momento	Tiempo (minutos)	Conocimientos	Semana	Fecha inicio	Observaciones
SUBMÓDULO II. PROGRAMACIÓN WEB	Apertura	50 min	Bienvenida Encuadre Reglamento Instrumentos de evaluación.	8	23 – 27 de marzo de 2026	Evaluación sumativa.
		50 min	Evaluación diagnóstica.			
		250 min	Diseño web • Conceptos básicos: ▪ Páginas web. ▪ Sitio web ▪ Dominio URL ▪ Hosting. ▪ Applets			
	Desarrollo	100 min 250 min	• Elementos de una página web. • Estándares de diseño web (W3C)	9	13 – 17 de abril de 2026	17 de abril reunión de academia.
		350 min	HTML y CSS ▪ Estructura básica de una página web: html, head, title, body.	10	20 – 24 de abril de 2026	
		350 min	▪ Elementos básicos de una página web: ▪ Títulos. ▪ Párrafos. ▪ Imágenes ▪ Hipervínculos	11	27 de abril – 01 de mayo de 2026	Suspensión oficial: 1 de mayo de 2026.





SUBMÓDULO II. PROGRAMACIÓN WEB	Desarrollo	250 min	- Elementos básicos de una página web: - Vídeo. - Sonido. - Fuentes. - Tamaño.	12	04 – 08 de mayo de 2026	Suspensión oficial: 5 de mayo de 2026.
		100 min	Colores			
		100 min	- Estructura en una página web: - Tablas. - Divs. - Clases. - Menús. - Artículos.	13	11 – 15 de mayo de 2026	Suspensión oficial: 15 de mayo de 2026.
		250 min				
		350 min	- Estructura en una página web: - Secciones. - Formulario. - Márgenes. - Posición de los elementos en la página web.	14	18 – 22 de mayo de 2026	Evaluación intrasemestral.
		350 min	Desarrollo de páginas web con software de aplicación.	15	25 – 29 de mayo de 2026	
	Cierre	350 min	Publicación de páginas web. Situación didáctica	16	01 – 05 de junio de 2026	
				17	08 – 12 de junio 2026	Evaluación sumativa.
				18	15 – 19 de junio 2026	Reforzamiento académico. 18 de junio Reunión de academia.





Criterios de Evaluación

Situación didáctica:

GreenBytes - "Bytes verdes", tecnología al servicio del ambiente.



35%



Actividades:

25%

Actividad 1. Tabla comparativa: Elementos de la web.	3%
Actividad 2. Diseño de una página o sitio web.	3%
Actividad 3. Infografía "Estándares de Diseño Web (W3C)".	3%
Actividad 4. Estructurando mi Página Web.	3%
Actividad 5. Aprendo a identificar los elementos.	3%
Actividad 6. Clases, Menús y Formularios (parte 1 y 2).	4%
Actividad 7. ¿Qué es JavaScript?	3%
Actividad 8. Instrucciones en JavaScript.	3%

Prácticas:

25%

Práctica 1. Mi Primer Página Web.	2%
Práctica 2. Mi Página Web con multimedia.	2%
Práctica 3. Mejorando mi Página Web.	3%
Práctica 4. Enchulando mi Web.	3%
Práctica 5. Tablas en HTML	5%
Práctica 6. Clases, Menús y Formularios.	5%
Práctica 7. Publicando mi página web.	5%



Examen:

15%

¡Ánimo! ¡Y
muchísimo
éxito en este
submódulo!





Situación Didáctica

GREENBYTES – “BYTES VERDES”, TECNOLOGÍA AL SERVICIO DEL AMBIENTE

Propósito:

Diseñar páginas web en equipo de 5 integrantes, enfocadas a empresas de reciclado y protección ambiental. Utilizando software de edición de código, que permitan el uso de herramientas de programación en HTML y CSS; para ser presentados en un foro de emprendimiento.

Contexto:

Los avances tecnológicos han permitido que la sociedad pueda satisfacer ciertas necesidades obteniendo resultados de forma automática, lo que ha conllevado a consumir grandes cantidades de aparatos electrónicos; sin embargo, esto ha ocasionado que la contaminación de estos dispositivos vaya a la par en aumento.

Cuando un dispositivo se rompe o queda obsoleto, la forma más usual de deshacer de ellos es tirarlo al bote de la basura, teniendo como consecuencias que ciertos químicos del equipo estén expuestos al medio ambiente. En una parte esto se debe a que desconocen que lugares o empresas pueden brindar asesoría o tratamiento para los desechos tecnológicos. Por ello los alumnos de la capacitación de Desarrollo de software pueden diseñar páginas web y compartirlas en internet con el propósito de dar a conocer a las empresas que se dedican al reciclado y protección ambiental.

Conflicto cognitivo

¿Qué lenguajes son necesarios para la construcción de páginas y sitios web?
¿Cuál es la diferencia entre un sitio web y página web?
Menciona 10 software útiles para el diseño y desarrollo de páginas y sitios web.
¿En dónde se almacenan las páginas web?





Evaluación Diagnóstica



Instrucciones: Lee cuidadosamente los siguientes cuestionamientos y selecciona en base a tus conocimientos la respuesta que consideres correcta.



- ¿Es un documento electrónico multimedia capaz de incluir audio, video, texto y sus combinaciones, adaptado para la world wide web, que se visualizan mediante navegadores web? ()
A) Sitio web B) Internet C) Páginas Web D) URL
- ¿Es un lugar en la world wide web donde se almacenan documentos, entre ellos páginas web, organizados jerárquicamente? ()
A) Sitio web B) Dominio de Internet c) Applet D) Hipertexto
- ¿Es el nombre único con el que una página web o un sitio web se identifica en la red? ()
A) URL B) Hosting C) Dominio de internet D) HTML
- ¿Es la dirección única y específica que se asigna a cada uno de los recursos que hay en Internet (sitios web, páginas web, textos, fotos, vídeo...) para que puedan ser localizados por el navegador y visualizados por el usuario? ()
A) Dirección IP B) Hosting C) URL D) Dominio de internet
- ¿Es una etiqueta de HTML utilizada para dar prioridad a un título principal? ()
A) <p>Mi primera página web</p> B) <h2>Mi primera página web</h2> C) <body>mi primera página web</body> D) <h1>Mi primera página web</h1>
- ¿Cuál de las etiquetas se emplea para crear un hipervínculo a otra página web? ()
A) <a> B) <p> </p> C) <video> D) <h1> </h1>
- ¿Es un elemento que debe llevar toda página web? ()
A) Archivos B) Sitio C) Imágenes D) Dirección IP
- ¿Es la parte del desarrollo web que se encarga de la lógica de la misma y que funcione? ()
A) Programas B) Sitios C) Front end D) Back end





9. ¿Como se le llama al proceso de diseñar las páginas web de nuestro sitio web en el disco duro del equipo para luego enviarlos al servidor web junto con los archivos multimedia referenciados (imágenes, animaciones, audios, etc)? ()
A) Publicar el sitio web B) Usar un dominio C) Hosting D) Almacenamiento web
10. Significa Protocolo de Transferencia de Hipertexto, y es necesario nos permite realizar una petición de datos y recursos, como pueden ser documentos HTML. ()
A) HTML B) HTTP C) FTP D) PHP
11. Vínculo asociado a un elemento de un documento con hipertexto, que apunta a un elemento de otro texto u otro elemento multimedia. ()
A) Hipervínculo B) Referencia C) Enlace D) Dominio
12. Aparece en la parte superior de tu sitio y a lo largo de tus páginas. ()
A) Hipervínculo B) Imagen C) Body D) Encabezado
13. Es el servicio que provee a los usuarios de Internet un sistema para poder almacenar información, imágenes, vídeo, o cualquier contenido accesible vía web. ()
A) Dominio B) Protocolo C) Hosting D) Estructura
14. Es un programa informático que procesa una aplicación del lado del servidor, realizando conexiones bidireccionales o unidireccionales y síncronas o asíncronas con el cliente y generando o cediendo una respuesta en cualquier lenguaje o aplicación del lado del cliente: ()
A) Servidor HTTP B) Hosting C) Dominio D) Protocolo
15. Es un protocolo de comunicación de datos digitales clasificado funcionalmente en la capa de red según el modelo internacional OSI. ()
A) FTP B) IP C) HTTP D) TCP

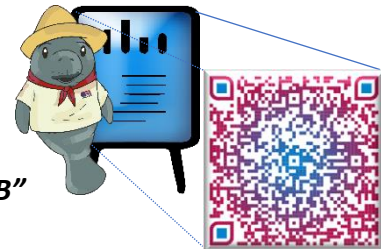


Lectura No 1. Conceptos básicos páginas web y sitios web

Instrucciones: Realiza el análisis de la siguiente información, posteriormente, completa la Actividad 1. Tabla comparativa: Elementos de la web.



LECTURA 1. "CONCEPTOS BÁSICOS PÁGINAS WEB Y SITIOS WEB"



Página Web

Se conoce como página web o electrónica a **un documento digital de carácter multimedia** (es decir, capaz de incluir audio, video, texto y sus combinaciones), adaptado a los estándares de la World Wide Web (WWW) y a la que se puede acceder a través de un navegador Web.

Las páginas Web **se encuentran programadas en un formato HTML o XHTML**, y se caracterizan por su relación entre unas y otras a través de hipervínculos.



El **hipertexto o hipervínculo**, es un mecanismo de comunicación que permite desde un documento navegar a otro documento. Este mecanismo se realiza insertando un enlace en algún texto del documento y es una facilidad que provee el lenguaje HTML.

¿Cuál es la diferencia?



https://youtu.be/vyq9_fda-ts?si=8xm-c8FerpBvap-9



Tipos de página web

Existen dos tipos de página web, conforme al modo en que se genera su contenido:

Páginas Web estáticas

- Operan mediante la descarga de un fichero programado en código HTML, en el que están todas las instrucciones para que el navegador reconstruya la página web. Esta compuesta generalmente por diferentes elementos como texto, imágenes, videos, etc. Este tipo de páginas son meramente informativas, documentales, no interactivas (su contenido siempre es el mismo).

Páginas Web dinámicas

- Las páginas web dinámicas se generan en el momento mismo del acceso del usuario, empleando para ello algún lenguaje interpretado (como el PHP), lo cual le permite recibir solicitudes del usuario, procesarlas en bases de datos y ofrecer una respuesta acorde a sus requerimientos. Es decir, su contenido se construye a partir de la información introducida por el usuario

Sitio web

Un sitio web es un lugar en la World Wide Web donde se almacenan documentos, entre ellos páginas web, organizados jerárquicamente. Es decir, un sitio web es un conjunto de varias páginas web y otros archivos.

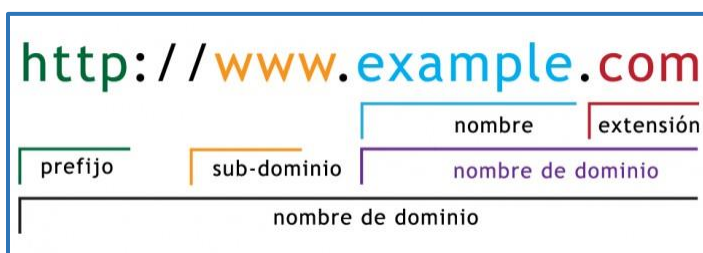
Al sitio web se ingresa usando el dominio que identifica la ubicación del sitio, y cada una de las páginas del sitio tiene una dirección URL única que la identifica y mediante la cual se puede acceder a ella.





Dominio de internet

Se conoce como dominio en Internet al nombre único con el que una página web o un sitio web se identifica en la red. Es una secuencia única de caracteres asociadas a las direcciones numéricas y que se muestra después del signo @ en las direcciones de correo electrónico y después de www. en las direcciones web. El utilizar un nombre de dominio en lugar de una dirección IP para identificar una ubicación en Internet, es mucho más fácil recordar y escribir direcciones web.



Los dominios se dividen en tres grupos:



➤ Dominios de Nivel Superior Genéricos, (gTLD, generic Top-Level Domain)

Este tipo de dominios son los de uso común y más utilizados a nivel mundial, no se ajustan a un país determinado y sus terminaciones definen el servicio para el cual se utilizan, por ejemplo:

.com	Sitio comercial	.info	Sitio informativo
.net	Empresa de servicios de Internet	.biz	Sitio de negocios
.org	Organización sin fines de lucro	.edu	Sitio educativo

➤ Dominios de Nivel Superior Geográfico, (ccTLD, country code Top-Level Domain)

Son conocidos también como dominios territoriales y son usados por países o territorios dependientes; se componen de 2 caracteres. Algunos ejemplos de dominios territoriales son:

.jp	Japón	.pe	Perú
.mx	México	.pr	Puerto Rico
.pa	Panamá	.uk	Reino Unido

➤ Dominios de Tercer Nivel

Los dominios de tercer nivel son aquellos que resultan tras combinar un dominio genérico (gTLD) y uno geográfico (ccTLD); los alcances del dominio genérico son delimitado por el dominio territorial. Por ejemplo, .com.mx y .com.co son dominios de tercer nivel de tipo comercial para México y Colombia respectivamente. Estos son algunos ejemplos de dominios de tercer nivel en México:

.com.mx	Entidades comerciales	.edu.mx	Instituciones Educativas
.net.mx	Proveedores de redes	.gob.mx	Entidades Gubernamentales
.org.mx	Organizaciones no lucrativas		

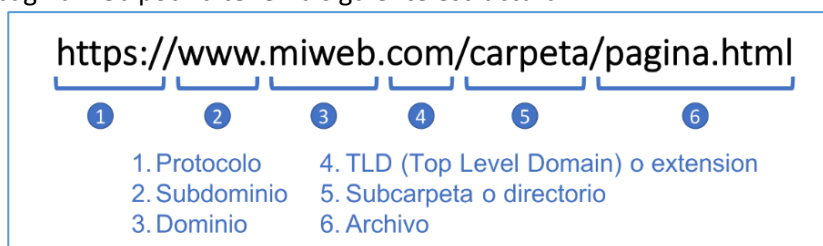


URL

Un Localizador de Recursos Uniforme, más conocido como URL (Uniform Resource Locator) es la dirección única y específica que se asigna a cada uno de los recursos que hay en Internet (sitios web, páginas web, textos, fotos, vídeo...) para que puedan ser localizados por el navegador y visualizados por el usuario.

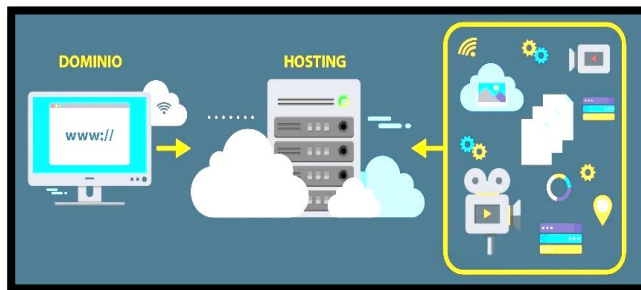
Cuando un usuario quiera navegar por la red y acudir a determinados sitios o páginas web, deberá hacer uso de este URL, que le da al navegador la información necesaria para identificar los documentos, que los encuentre y, de inmediato, se los muestre al navegante.

Una dirección url de una página web podría tener la siguiente estructura:



Hosting

El hosting o alojamiento web es un servicio para proporcionar espacio en línea para el almacenaje de páginas web. Estas páginas web estarán disponibles vía World Wide Web. Las compañías que ofrecen el servicio se suelen llamar empresas de hosting. Algunos servicios que pueden ofrecer los alojamientos son: FTP, cuentas de correo, control de acceso, estadísticas de rendimiento o visitas, etc.



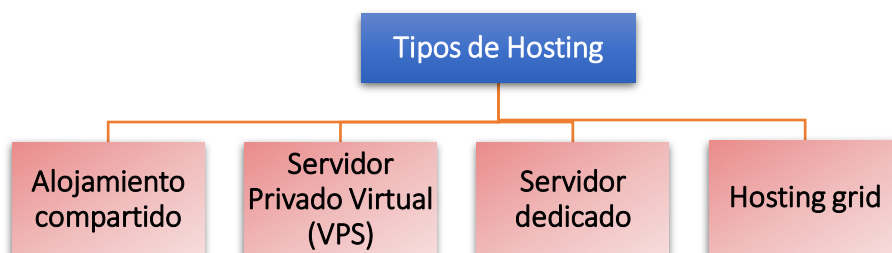
Los alojamientos tienen asociadas una serie características. Algunas son:

- **Espacio en disco o cuota:** es la cantidad de disco duro que tendremos para almacenar datos (correos, base de datos, código de la web, imágenes, archivos descargables, ...).
- **Transferencia:** es la capacidad de intercambio de datos, es decir, cada vez que alguien accede a nuestra página web, se consume una cantidad de bytes.
- **Tipo de software:** la mayor parte de los alojamientos tienen los servicios LAMP (Linux, Apache, MySQL y PHP). Es importante saber qué software y versión necesitará nuestra página web.



Tipos de hosting

Los diferentes tipos de alojamiento disponibles según las necesidades son:

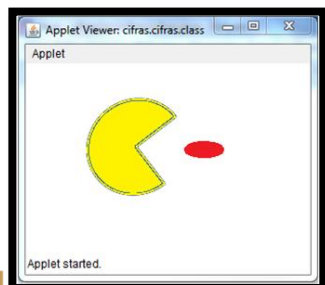


- **Alojamiento compartido:** la empresa pone muchos sitios web en el mismo servidor físico. Cada cliente tiene su propia asignación de espacio web físico y un conjunto de límites de ancho de banda.
- **Servidor Privado Virtual (VPS):** También se conoce como Servidor Dedicado Virtual. Es un servidor que se divide en servidores más pequeños lógicamente. Al cliente se le da su propia partición, que se instala con su propio sistema operativo.
- **Servidor dedicado:** En este tipo de alojamiento, un único servidor dedicado está configurado para un solo cliente. Es comúnmente utilizado por las empresas que necesitan la potencia, control y seguridad que ofrece un servidor dedicado.
- **Hosting grid:** Consiste fundamentalmente en utilizar clústeres de servidores para acercar las ventajas de una infraestructura de servidores compleja a usuarios que contratan servicios de alojamiento compartido y siempre a una fracción de coste.



Applets

Los Applet son programas desarrollados en Java que se integran en una página o sitio web y se emplean para brindar nuevas funciones que no se pueden desarrollar usando HTML. Estos se descargan directamente de la web y se ejecutan en el navegador de la máquina del usuario final (siempre y cuando dicho navegador tenga soporte para Java).



El Applet tiene una función específica, por ejemplo: calcular valores de figuras geométricas, velocidades u otros, mostrar imágenes con sonidos, presentar gráficos interactivos que reaccionen a acciones hechas con el mouse de la computadora, entre otros.





BIBLIOGRAFÍA

1. **Arguedas Matarrita, Carlos Alberto y Bejarano Salazar, Ana Gabriela (2015).** *USO DE APPLETS DE JAVA EN EL CURSO EN LÍNEA DE FÍSICA II, VALORACIÓN DEL ESTUDIANTE PARA SU APLICACIÓN EN SECUNDARIA*. Atenas, 2 (30), 109-122. [Fecha de Consulta 28 de Octubre de 2020]. ISSN:. Disponible en:
<https://www.redalyc.org/articulo.oa?id=4780/478047206009>
2. **Arume. (01 de Abril de 2019).** "Dominio y Alojamiento, ¿qué Son?". *Dominio y Alojamiento, ¿qué Son?*. www.arumeinformatica.es/blog/dominio-alojamiento/
3. **Desarrollo de Aplicaciones WEB.** Práctica Alojamiento Web. Universidad de Murcia. *Alojamiento web*. [www.Um.Es](http://www.um.es). <https://www.um.es/docencia/barzana/DAWEB/2017-18/daweb-practica-alojamiento-web.html>
4. **Estrada, s. (s.f.).** ¿CUÁL ES LA DIFERENCIA ENTRE UN SITIO WEB Y UNA PÁGINA WEB?. ingenioLab. <https://ingeniolab.co/cual-es-la-diferencia-entre-un-sitio-web-y-una-pagina-web/>
5. Glosario informático - Definición de términos informáticos. *Applet - Sección BD/ Programación*. [Www.Glosarioit.Com](http://www.Glosarioit.Com). Obtenido el 28 de octubre de 2020 de <https://www.glosarioit.com/Applet>
6. **Raffino, M. (5 de julio de 2020).** PÁGINA WEB. Concepto.de. Recuperado el 19 de octubre de 2020 de <https://concepto.de/pagina-web/>
7. **Raffino, M. (17 de julio de 2020).** DOMINIO EN INTERNET. Concepto de. Recuperado el 19 de octubre de 2020 de <https://concepto.de/dominio-en-internet/#ixzz6aqiRUeFd>
8. **Softwarelab.org. (s.f.).** ¿Qué es una URL y para que sirve?. 2020, octubre 28, de softwarelab.org Recuperado de <https://softwarelab.org/es/url/>
9. Tipos de Dominios - Conocimiento - Certerus.com . *Tipos de Dominios*. Mi.Certerus. Com. Recuperado el 28 de octubre de 2020 de <https://mi.certerus.com/knowledgebase/32/Tipos-de-Dominios.html>



Actividad 1. Tabla Comparativa: Elementos de la Web



Instrucciones: Después de haber analizado la lectura 1. “Conceptos Básicos de Páginas Web y Sitios Web”, completa correctamente la siguiente tabla.



En el mercado existe una enorme oferta de proveedores de Hosting y Nombres de dominio, para conocer un poco los precios de estos servicios, realiza una tabla comparativa en la que representes los siguientes datos referente a 5 proveedores diferentes.



HOSTING				
Nº	Nombre del proveedor	Nombre del plan económico	Precio del plan	Características del plan
1				
2				
3				
4				
5				



NOMBRES DE DOMINIO				
Nº	Nombre del proveedor	Dominio de alto nivel más barato	Precio de primera contratación / precio de renovación	Características adicionales que ofertan
1				
2				
3				
4				
5				





INSTRUMENTO DE EVALUACIÓN
LISTA DE COTEJO 1
Actividad 1: "Tabla Comparativa:
Elementos de la Web"

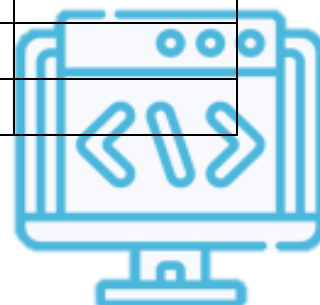


DATOS GENERALES

Nombre(s) del alumno(s)	Matrícula(s)
Producto: Documento electrónico y/o impreso	Fecha
Submódulo 2: Programación Web	Periodo: 2025 - 2026B
Nombre del docente	Firma del docente

CRITERIO	INDICADORES	VALOR OBTENIDO		PONDERACIÓN	CALIF	OBSERVACIONES Y/O SUGERENCIAS DE MEJORA
		SÍ	NO			
1	Respetar la estructura de la tabla indicada por el docente			1		
2	Presenta los 5 hosting con sus respectivas características.			2		
3	Presenta los 5 nombres de dominio con sus respectivas características.			2		
4	Presenta una actitud positiva y respetando los puntos de vista de sus compañeros.			1		
5	No presenta faltas ortográficas.			1		
6	Presenta el sustento de su investigación para responder a los campos en la tabla.			2		
7	Entrega en tiempo y forma.			1		

CALIFICACIÓN



Lectura No 2. Elementos de un Página Web

Instrucciones: Realiza el análisis de la siguiente información, posteriormente, realiza la Actividad 2.

Identifica los elementos de las páginas y los sitios web.



LECTURA 2. "ELEMENTOS DE UNA PÁGINA WEB"



Textos, imágenes, audio, video y otros contenidos

Cuando hablamos de las partes de una página web, generalmente nos referimos a su estructura. Es decir, todo el conjunto de elementos o de diseño o funcionalidades, tales como secciones o menús, que hacen la página web navegable y funcional. También es muy común referirnos a las partes de una página web cuando hablamos de contenido.

Hoy en día, la presencia digital es un factor muy importante para cualquier negocio. Seguramente alguna vez habrás escuchado el dicho popular de que si "no está en Google es porque no existe". Realmente no es un simple dicho, la Internet es parte de nuestras vidas, y ahora es prácticamente normal querer encontrar cualquier negocio o marca en el mundo digital.

La Página Web o Electrónica es un documento electrónico capaz de contener: texto, imágenes, sonido, videos, enlaces, programas, etc., que puede ser desplegado a través de un navegador como: Chrome, Firefox, Explorer, etc. Ejemplos:

<https://www.liverpool.com.mx>

<https://www.xbox.com/es-mx>

El Sitio Web es una colección de páginas web que se agrupan y conectan entre sí, regularmente en el mismo dominio o subdominio. Igual se accede a ellos por medio de un navegador. Estos se encuentran alojados en un servidor, ejemplos:

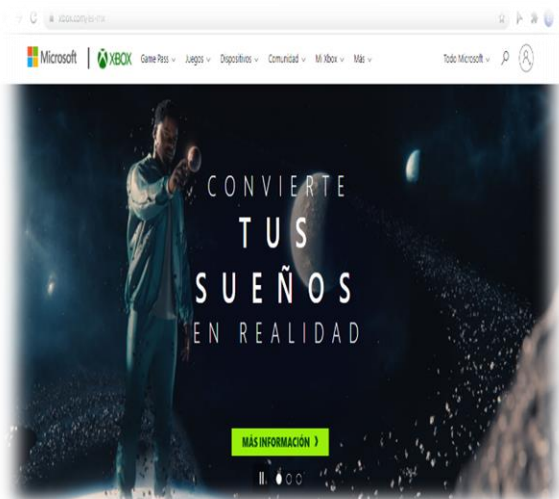
<https://www.xbox.com/es-MX/games/all-games?xr=shellnav>

<https://www.liverpool.com.mx/tienda/celulares/cat4450173>



Nadie puede dudar que Internet ha marcado un antes y un después en el mundo de las profesiones y que ha revolucionado la manera de entender incluso los negocios. Hoy en día una página web es una parte indispensable de la imagen de una persona, empresa o institución.

Principales Características que constituyen una página en Internet son:



Contenido

Una página web debe ser atractiva... Pero no solo a la vista, sino también en el contenido. Eso significa que debe tener contenido interesante que atraiga (y atrape) al público objetivo. De hecho, ninguna página puede considerarse exitosa si no cuenta con un contenido atractivo y actualizado con cierta frecuencia.

Eso servirá no solo para aumentar el número de nuevas visitas, sino para que los antiguos

visitantes consulten la página a menudo y la usen como referencia.



Tipografía

Es importante usar fuentes llamativas pero sencillas de leer. También es importante variar el tamaño de una misma fuente para resaltar secciones interesantes del contenido y del texto.



Imágenes

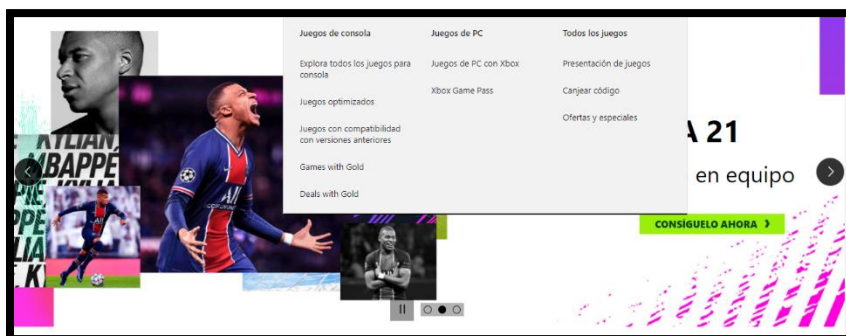
En el diseño web es muy importante las buenas fotografías, preferentemente tomadas por un profesional. Es importante que no estén pixeladas o desenfocadas, ya que puede causar una mala imagen del sitio en general.



Animaciones y movimiento

Para generar una sensación de dinamismo es bueno utilizar elementos animados. Pero atención: no abusos de ellos, ya que pueden sobrecargar la página.

Lo ideal es que no sobrepasen el 20% del total de la página ya que de otra forma pueden causar saturación, distraer la atención de cosas importantes que se quieren decir y por último sacar al usuario del sitio.



Botones e iconos

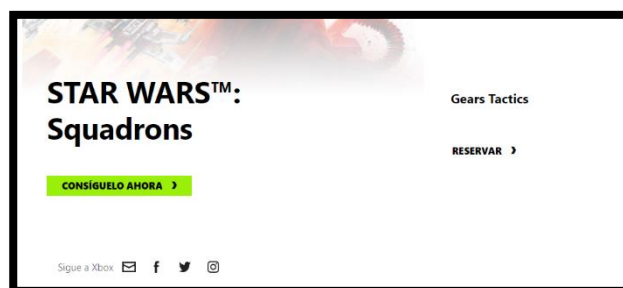


Utilizar botones prácticos y representativos, así como iconos grandes y vistosos aumenta la facilidad de navegación en la página. Pero no es necesario colocar accesorios que no brinden un valor real, por ejemplo, un reloj para clientes que se encuentran en el mismo uso horario.



Fondos claros y sencillos

Los fondos claros o totalmente blancos facilitan la visualización, pero también dan un aspecto despejado a la página resaltando las imágenes. Existe la tendencia a utilizar fondos negros para lecturas largas, para no cansar al lector con la luz del monitor.



Dominio

El nombre de dominio es la primera parte de una dirección Web que identifica al servidor que almacena y sirve su sitio.



Enlaces a redes

En la actualidad es de gran ayuda para los usuarios colocar enlaces a redes sociales con las cuales interactúan con facilidad. Esto ayuda además a crear comunidad.

Sigue a Xbox



Sobriedad

Una página debe permitir una fácil visualización de los contenidos (vídeos, imágenes, textos).



Texto

Elemento más significativo de cualquier sitio Web porque los usuarios navegan por la Web fundamentalmente en busca de información expresada en hipertexto.



Menú / Navegación

Un menú de navegación o mapa de sitio es el diagrama o estructura que organiza el contenido y dirige el flujo del usuario en tu sitio web.

Este es una sección fundamental en el desarrollo de tu sitio web, desde etapas iniciales previas al diseño, así como en mejora continua.



Actividad 2. Identifica los elementos de las páginas web



Instrucciones: Después de haber analizado la Lectura 2. "Elementos de una Página Web". Considera los siguientes conceptos y realiza un diseño de una página web, identificando además lo que es el sitio web:

Conceptos: Contenido, Tipografía, Imágenes, Animaciones y movimientos, Botones e Iconos, Fondos claros y sencillos, Dominio, Enlaces a redes, Sobriedad, Menú / Navegación. Considera el ejemplo para elaborar el tuyo:

Animaciones y movimientos

Sobriedad

Página web

Tipografía

Contenido

Sitio web

PARTE 1

PARTE
2

Considera un tema o una organización para la cual debes realizar el diseño de las páginas y el sitio web.





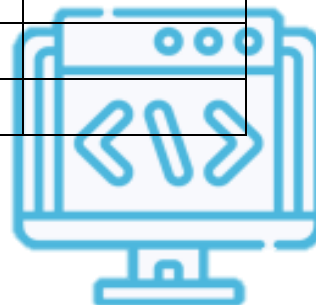
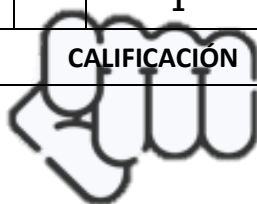
INSTRUMENTO DE EVALUACIÓN
LISTA DE COTEJO 2
Actividad 2: "Identifica los elementos de
las páginas y los sitios web"



DATOS GENERALES

Nombre(s) del alumno(s)	Matrícula(s)
Producto: Documento electrónico y/o impreso	Fecha
Submódulo 2: Programación Web	Periodo: 2025 - 2026B
Nombre del docente	Firma del docente

CRITERIO	INDICADORES	VALOR OBTENIDO		PONDERACIÓN	CALIF	OBSERVACIONES Y/O SUGERENCIAS DE MEJORA
		SÍ	NO			
1	Presenta un diseño de una página web sobria.			1		
2	Identifica el dominio, la URL y los enlaces a redes.			2		
3	Identifica muy bien el menú/navegación, los botones e iconos y presenta el contenido.			2		
4	Presenta imágenes, especifica las animaciones y movimientos.			1		
5	No presenta faltas ortográficas.			1		
6	Diferencia lo que es una página de un sitio web, además de elaborar una conclusión personal y entendible.			2		
7	Entrega en tiempo y forma.			1		
				CALIFICACIÓN		



Lectura No 3. Estándares de Diseño Web (W3C)

Instrucciones: Realiza el análisis de la siguiente información, posteriormente, contesta las preguntas y la evidencia que se solicita.



LECTURA 3. “ESTANDARES DE DISEÑO WEB (W3C)”



Hace unos años, el desarrollo de páginas web era caótico, había que realizar una versión de cada página web para prácticamente cada navegador. Cuando hablamos de estándares web, a lo que nos referimos son especificaciones formales a las que Internet debe cumplir. Por lo tanto, con frecuencia se trata menos de cómo aparece la interfaz de un sitio web y más sobre cómo está estructurado el back-end del mismo.

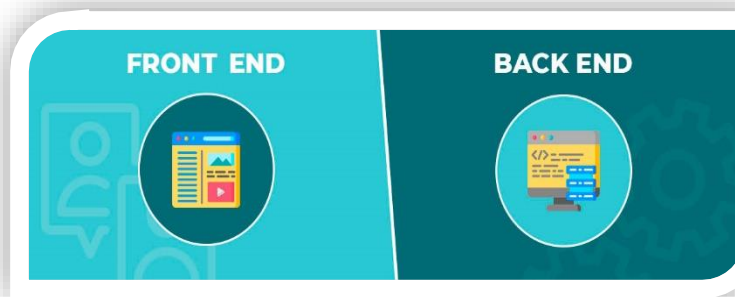


Back-End: “Es la parte del desarrollo web que se encarga de que toda la lógica de una página web funcione”.

Funciones de un Back-End:

- El Desarrollo de funciones que simplifiquen el proceso de desarrollo.
- Acciones de lógica.
- Conexión con bases de datos.
- Uso de librerías del servidor web (por ejemplo, para implementar temas de caché o para comprimir las imágenes de la web).





El desarrollador de Front-end se encarga de implementar todo lo relacionado con la parte visible, lo que «toca» el usuario cuando navega por la web. Resumiendo, al máximo el Front-end trabaja con HTML, CSS y JavaScript.

Los estándares web no se centran solo en el desarrollo web. Toman navegadores, HTTP, software de diseño y desarrollo, así como dispositivos de consumo. Esencialmente, los estándares web se desarrollan y formalizan para dar fuerza y consistencia al núcleo de la web. Cuanto más nos adherimos a estos estándares, más accesible será la web para todos. Para dar solución a este problema, el W3C lanzó una iniciativa en 1997 para lograr la accesibilidad web (Web Accessibility Initiative o WAI) y que se siguieran una serie de estándares o normas.

Los estándares web se introdujeron para proteger el ecosistema web, para mantenerlo abierto, gratuito y accesible para todos. Se perseguía poner la web en una burbuja protectora y no tener que construir sitios web adaptables a navegadores específicos. Cuando se introdujeron los estándares, se alentó a

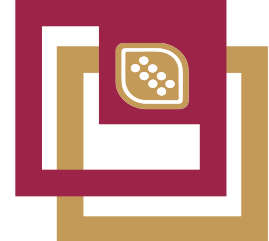


los creadores de navegadores a adherirse a una forma estandarizada de hacer las cosas, lo que resultó en una compatibilidad cruzada cada vez más fácil para los creadores de contenido y ya no hubo necesidad de crear múltiples versiones del mismo sitio web.

Algunos de los estándares Web más conocidos y ampliamente utilizados son:

- **HTML (HyperText Markup Language)**, para definir la estructura de los documentos.
- **XML (eXtensible Markup Language)**, que sirve de base para un gran número de tecnologías.
- **CSS (Cascading Style Sheets)**, que permite asignar estilos para la representación de los documentos.
- **JavaScript**, que permite otorgar dinamismo y funcionalidad.





Los estándares web, se basan en tres aspectos fundamentales:

- La interacción con el usuario, buscando una comunicación armoniosa entre el mismo y la aplicación.
- La optimización del modo en que se le presentan los datos al usuario, ayudando a que conozca en qué parte del ambiente de trabajo está situado y en qué momento.
- La estructura del sistema optimiza la ubicación de los componentes y las distintas secciones que se definen.

Describiendo los estándares.

HTML

El Lenguaje de Marcado de Hipertexto, en inglés, Hypertext Mark-Up Language, (HTML) es el lenguaje de marcado predominante para la construcción de páginas Web. Permite representar el contenido enriquecido en forma de texto, así como complementar el texto con objetos, como el caso de las imágenes. HTML describe la estructura del contenido, además, puede manejar la apariencia de un documento y también su comportamiento a través de un script, por ejemplo, JavaScript. El lenguaje HTML pretende continuamente proporcionar funcionalidades más avanzadas para crear páginas más ricas en contenido.



Asimismo, se ha definido una especificación compatible con HTML: el XHTML (Extensible Hypertext Markup Language), que suele definirse como una versión XML de validación de HTML, proporciona un esquema XML que permita validar el documento para comprobar si está bien estructurado.

El HTML dinámico (DHTML o Dynamic HTML) no es un estándar definido por el W3C, sino que es un término de mercado que utilizan Netscape y Microsoft para referirse al conjunto de nuevas tecnologías de Web. Dicho conjunto comprende: - HTML - Hojas de estilo (CSS) - JavaScript Esta conjunción de tecnologías permite ofrecer al usuario interfaces gráficas mucho más ricas y a la vez complejas, controlar formularios de forma más eficiente, brindar un número de facilidades al usuario y proporcionar un intercambio más interactivo.



XML

El Lenguaje de Marcas Extensible, en inglés, Extensible Markup Language, (XML) es un metalenguaje extensible de etiquetas desarrollado por el W3C. Es un estándar que supera los límites de la Web. Se trata de un formato de datos que se ha adoptado prácticamente de forma universal y que soluciona complejas necesidades. XML no es realmente un lenguaje en particular, sino una manera de definir lenguajes para diferentes necesidades. El Lenguaje de Marcado Extensible no ha nacido sólo para su aplicación en Internet, sino que se propone como un estándar para el intercambio de información estructurada entre diferentes plataformas. XML es una tecnología sencilla que tiene a su alrededor otras que la complementan. Tiene un papel muy importante en la actualidad ya que permite la compatibilidad entre sistemas para compartir la información de una manera segura, fiable y fácil.



CSS

Las Hojas de Estilo en Cascada, del inglés, Cascade StyleSheets (CSS) es un lenguaje de hojas de estilos creado para controlar la presentación de los documentos electrónicos definidos con HTML y XHTML. CSS es la mejor forma de separar contenido y presentación; es imprescindible para crear aplicaciones Web complejas. Separar contenido y presentación, brinda numerosas ventajas, ya que obliga a crear documentos HTML/ XHTML bien definidos, mejora la accesibilidad del documento, reduce la complejidad de su mantenimiento y permite visualizar el mismo documento en infinidad de dispositivos diferentes. El lenguaje CSS se utiliza para definir el aspecto de todos los contenidos, el formato de tablas, la separación, el color, tamaño y tipo de letra de titulares y/o textos, la tabulación con la que se muestran los elementos de una lista o menú.



Las ventajas de utilizar CSS son:

- Control centralizado de la presentación de un sitio Web completo con lo que se agiliza de forma considerable la actualización de este.
- Los Navegadores permiten a los usuarios especificar su propia hoja de estilo local que será empleada en una aplicación Web, con lo que aumenta considerablemente la accesibilidad, por ejemplo, personas con deficiencias visuales pueden configurar su propia hoja de estilo para aumentar el tamaño del texto o remarcar más los enlaces.
- Una página puede disponer de diferentes hojas de estilo según el dispositivo que la utilice.





JavaScript.

JavaScript tiene objetos predefinidos que puedes utilizar en tus proyectos. Es como si alguien ha estado programando objetos para que tu puedas aprovecharte de ellos, sin tener que programarlos tú de cero. Los objetos son como envoltorios de código. En JavaScript, prácticamente todo son objetos.

Por poner un ejemplo, los números y las cadenas de caracteres son transformados por el intérprete del lenguaje en objetos. Pero como los números y las cadenas tienen características diferentes, JavaScript ofrece de forma nativa objetos distintos para trabajar con ellos. Los objetos primitivos más utilizados son los siguientes:



- Números (number)
- Cadenas de texto (string)
- Valores booleanos (boolean)
- Arreglos o una variable que contiene otras variables (array)

WEBGRAFIA



- <https://revistadigital.inesem.es/informatica-y-tics/estandares-web/>
- <https://www.itconsultors.com/estandares-web>
- <https://rafarjonilla.com/que-es/backend/>
- <https://www.redalyc.org/pdf/1814/181421569009.pdf>
- <https://raulbocache.com/que-objetos-estandar-tiene-javascript/>
- <https://www.cualhost.com/marketing-en-linea/elementos-de-una-pagina-web-exitosa/>
- <https://es.godaddy.com/blog/elementos-para-una-pagina-web-de-exito/>
- <https://blog.virtualianet.com/disenio-web-los-8-elementos-imprescindibles/>
- <https://rockcontent.com/es/blog/elementos-de-un-sitio-web/>
- <https://www.lucushost.com/blog/partes-de-una-pagina-web-estructura-y-contenido/>
- <https://www.yeeply.com/blog/diferencia-entre-sitio-web-y-pagina-web/#:~:text=Un%20sitio%20web%20o%20cibersitio,a%20un%20dominio%20de%20internet.&text=Todas%20esas%20p%C3%A1ginas%20web%20vinculadas,a%20trav%C3%A9s%20de%20un%20navegador.>



Actividad 3. Infografía “Estándares de diseño web”



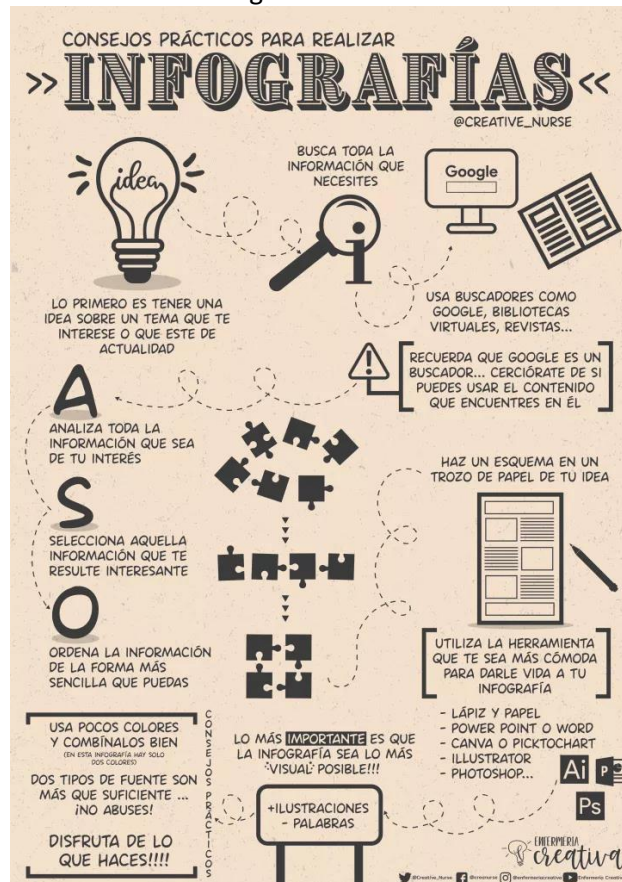
Instrucciones: Después de haber analizado la Lectura 3. “Estándares de diseño web (W3C)”. Considera los conceptos y realiza una infografía que te permita el dominio de estos, no olvides responder las preguntas de conclusión:



PREGUNTAS:

1. ¿Qué es un estándar de diseño web?
2. ¿Por qué son importante conocer los estándares?
3. ¿Menciona por lo menos 3 estándares utilizados para el diseño web?
4. ¿Qué sabía sobre el tema?
5. ¿Qué aprendí?

Considera las siguientes aplicaciones para la elaboración de tu infografía:



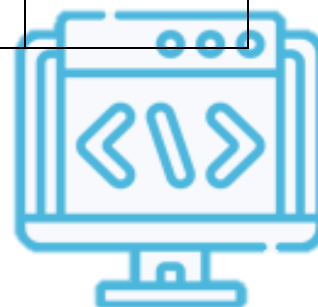


INSTRUMENTO DE EVALUACIÓN
LISTA DE COTEJO 3
Actividad 3: Infografía “Estándares de Diseño Web”



DATOS GENERALES

Nombre(s) del alumno(s)				Matrícula(s)		
Producto: Documento electrónico y/o impreso					Fecha	
Submódulo 2: Programación Web					Periodo: 2025 - 2026B	
Nombre del docente					Firma del docente	
CRITERIO	INDICADORES	VALOR OBTENIDO		PONDERACIÓN	CALIF	OBSERVACIONES Y/O SUGERENCIAS DE MEJORA
		SÍ	NO			
1	Presenta un diseño acorde al tema.			1		
2	Presenta los aspectos de XML, HTML, CSS y JavaScript.			2		
3	Presenta ilustraciones acordes a los aspectos del punto 2.			2		
4	Presenta el título o tema “Estándares de Diseño Web”.			1		
5	No presenta faltas ortográficas.			1		
6	Responde las preguntas de conclusión.			2		
7	Entrega en tiempo y forma.			1		
CALIFICACIÓN						



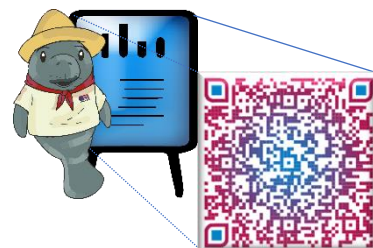
Lectura No 4. La Estructura de la Web

Instrucciones: Realiza el análisis de la siguiente información, posteriormente completa la Actividad

4. Estructurando la Web.



LECTURA 4. "LA ESTRUCTURA DE LA WEB"



Para crear una página web se necesita un documento HTML utilizando tres elementos o tags principales que cualquier sitio Web usa: html, head y body.

Un documento web (o página web) es, un conjunto de tags(etiquetas) HTML que se escriben en un editor de texto plano (sin formato) y se ejecutan en un navegador web.

Para crear un verdadero documento HTML comenzará con tres elementos contenedores:

- `<html>`
- `<head>`
- `<body>`

Estos tres se combinan para describir la estructura básica de la página:

- **`<html>`**: Este elemento envuelve todo el contenido de la página (excepto la **Document Type Definition**)
- **`<head>`**: Este elemento designa la parte de encabezado del documento. Puede incluir información opcional sobre la página Web, como puede ser el título (el navegador lo muestra en la barra de título), palabras clave de búsqueda opcionales y una hoja de estilo opcional.
- **`<body>`**: Este elemento alberga el contenido de su página Web, es decir, aquello que queremos que aparezca en el área de navegación del navegador.

Estructura básica





Sólo hay una manera correcta de combinar estos tres elementos. He aquí su colocación exacta, con el **!DOCTYPE** al comienzo de la página:

```
<!DOCTYPE html>
<html>
```

```
<head>
```

```
...
```

```
</head>
```

```
<body>
```

```
...
```

```
</body>
```

```
</html>
```

Toda página Web utiliza esta estructura básica. Los puntos suspensivos (...) muestran dónde insertaría la información adicional.

Una vez colocado el esqueleto XHTML, debe añadir dos conectores más a la mezcla. Toda página Web requiere un elemento **<title>** en la sección del encabezado. A continuación, deberá crear un contenedor para el texto en la sección del cuerpo de texto (**<body>**).

Un elemento contenedor de texto multiuso es **<p>**, que representa un párrafo. Veamos con más detalle los elementos que hay que agregar:

- **<title>**: Establece el título de la página Web, el cual tiene varias funciones. Primero, los navegadores lo muestran en la parte superior de la ventana. Segundo, cuando un visitante crea un marcador para la página, el navegador emplea el título para etiquetarlo en el menú Marcadores (o favoritos). tercero, cuando la página aparece en una búsqueda Web, el **motor de búsqueda** suele enseñar este título como primera línea en los resultados, seguido de un fragmento del contenido de la página.
- **<p>**: Indica un párrafo. Los navegadores no los sangran, pero añaden un pequeño espacio entre varios consecutivos para mantenerlos separados.



He aquí la página con estos dos nuevos ingredientes:

```
<!DOCTYPE html>
<html>
```

```
<head>
```

```
<title>Bienvenido a mi sitio Web</title>
```

```
</head>
```

```
<body>
```

```
<p>...</p>
```

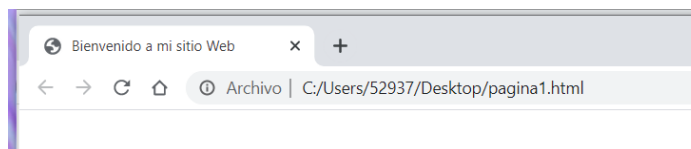
```
</body>
```

```
</html>
```





Si abre este documento en un navegador Web, verá que la página está vacía, pero ahora aparece el título.



Cuando un navegador muestra una página Web, el título se presenta en la parte superior de la ventana, con el texto al final. Si el suyo utiliza la navegación por pestañas, el título también aparece en ellas. Tal y como está ahora, este documento HTML es una buena plantilla para futuras páginas. La estructura básica está en su lugar; simplemente necesita cambiar el título y añadir algo de texto.

Lo primero que tenemos que saber es que en toda página web existen dos partes claramente diferenciadas: la cabecera, o **head**, y el cuerpo, o **body**. Es vital que la extensión del archivo sea **.html**, ya que sólo por la extensión el sistema operativo reconoce este archivo como un documento web, y no como un simple y archivo de texto. El nombre del archivo deberá escribirse de preferencia en minúsculas y sin espacios ni caracteres especiales.

Los únicos signos de puntuación admitidos son el punto (sólo uno), que lo usaremos para separar el nombre de la extensión y el guion bajo. Ejemplo: **"Pagina_1.html"**



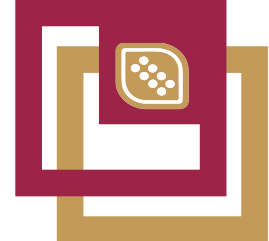
El nombre podrá contener letras y números, pero deberá empezar por una letra. Así mismo, nos abstendremos de meter en el nombre de un archivo letras acentuadas, eñes, cedillas, o cualquier otro carácter de algún alfabeto local.

Únicamente se admiten los caracteres del alfabeto internacional (ingles).

Observe los siguientes ejemplos:

INCORRECTO	CORRECTO
página1.htm, pues tiene una letra acentuada.	pagina1.htm
Pagina 1.htm, pues tiene un espacio en blanco y una mayúscula	
cañas.htm, pues tiene una eñe.	pagina_1.html
pagina/1.htm, pues tiene un carácter no permitido (la barra).	
pagina1-htm, porque la extensión no está separada del nombre por un punto y porque emplea un carácter no permitido. (el guion normal).	pagina-1.html
pagina.1.htm, porque hay más de un punto. Sólo debe usarse uno, para separar el nombre de la extensión.	

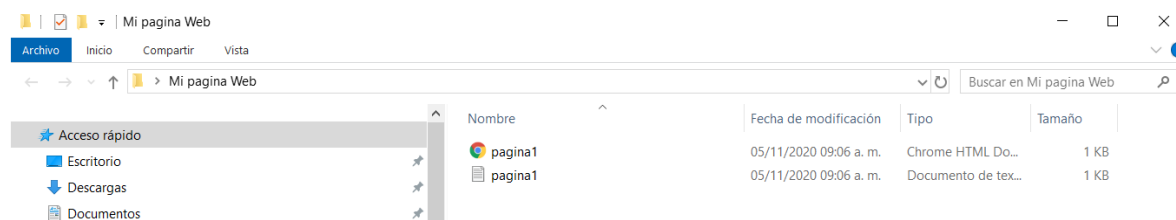




Puede que todas estas precauciones le parezcan excesivas ahora. La mayoría de ellas son realmente innecesarias en Windows, pero no olvide que está trabajando en la Red.

Los servidores web son mucho más sensibles a determinados aspectos del nombre de los ficheros que su máquina local. Otra norma importante es que **los archivos llevan una extensión, pero las carpetas en las que las guardamos no**. Así pues, en el nombre de una carpeta nunca incluiremos puntos.

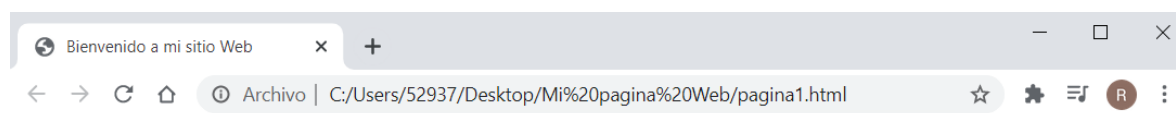
Una vez que haya guardado su código en su carpeta con el nombre adecuado, cierre el bloc de notas. En su carpeta, además del bloc de notas, verá el ícono que representa al archivo que acaba de grabar.



Observe que el ícono recuerda al de Explorer o Google Chrome (suponiendo que alguno de ellos sea su navegador por defecto). Esto es así porque al haber guardado el fichero con la extensión .htm el sistema operativo reconoce este fichero como documento web (también llamado documento html).



Si graba incorrectamente la extensión, el ícono será diferente y no podrá usar su archivo como documento web. **Para ejecutar** la página que acaba de crear, haga doble clic sobre el ícono. Automáticamente, se abrirá el navegador y se cargará la página. La ruta completa de acceso y el nombre del fichero aparecerán en la barra de dirección.



Ahora veamos en detalle cuál es el código de esta página que hemos creado y qué es lo que hace. En primer lugar, encontramos el tag **<html>**. Este tag se emplea siempre para iniciar el código html, es la forma de decirle al navegador que en este punto empieza el código. Por esta razón se pone esta línea siempre como la primera del código.





- No incluiremos ningún tag antes de este, con excepción del **DOCTYPE**. Por el contrario, al final del código tenemos el tag **</html>**, que le indica al navegador que el código finaliza en ese punto.
- No deberemos incluir ningún tag ni nada más después de esta línea. Observe que el tag de finalización es igual al de inicio, pero incluyendo la barra inclinada al principio.
- Dentro del código html encontraremos claramente diferenciados los dos bloques que mencionamos antes: el **head** (cabecera) y el **body** (cuerpo de la página).

Veamos el código de la cabecera:

<head>

<title>

Título de la página

</title>

</head>

Al igual que el código general, la cabecera tiene un principio indicado por **<head>** y un final indicado por **</head>**. Todo lo que queramos incluir en la cabecera irá colocado entre estas dos líneas. Como decíamos antes, en la cabecera incluiremos algunos datos generales de configuración de la página. En esta plantilla sólo hemos incluido el título de la página.

El título de la página se incluye entre los tags **<title>** y **</title>**, que indican, respectivamente, el comienzo y final de dicho título. El título es un **literal (una cadena de texto)** que aparecerá en la barra de título del navegador en tiempo de ejecución (cuando se ejecute la página). Este título es solamente eso: un texto libre para que aparezca en la página y no tiene nada que ver con el nombre que le dimos al fichero.

***“Una vez completado el head, pasamos a la segunda sección de toda la página web:
el body o cuerpo.”***

El **body** incluye, tal como lo mencionaba anteriormente, todo lo que queramos visualizar en el área de navegación. Se inicia con el tag **<body>** y se finaliza con el tag **</body>**. Como vemos en el archivo que hemos creado, de momento no hemos incluido nada. Por esta razón, al ejecutar la página, se ve el área de navegación vacía. Dentro del body iremos metiendo código e iremos visualizando los resultados.



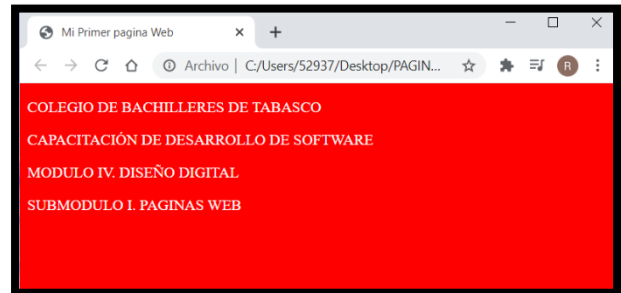
En la zona del código reservada para body de la página entre las líneas `<body>` y `</body>` podemos teclear lo siguiente: **Esto es una línea de texto.**

Por lo tanto, su código quedará como sigue:

```
<!DOCTYPE html>
<html>
  <head>
    <title>
      Mi Primer Página Web
    </title>
  </head>

  <body bgcolor="red">
    <P> <FONT COLOR="WHITE"> COLEGIO DE BACHILLERES DE TABASCO </P> <BR>
    <P> CAPACITACIÓN DE DESARROLLO DE SOFTWARE </P> <BR>
    <P> MODULO II. PROGRAMACIÓN WEB </P> <BR>
    <P> SITIOS Y PÁGINAS WEB </P> </FONT> <BR>
  </body>
</html>
```

El resultado en la pantalla será como el que se muestra:



Como ve para mostrar un texto en el área de navegación, basta con incluir dicho texto dentro del cuerpo de la página.

IMPORTANTE: Si te diste cuenta en el código dentro de la etiqueta de `<body>` agregamos un atributo que se llama **BGCOLOR= "red"**, este genera que el fondo de la página se torne en color rojo, o el color en ingles que decidas, también se pueden usar los códigos de colores que se verán más adelante. La etiqueta `<P>` permite escribir en líneas de párrafo cada que se apertura una de estas etiquetas le indica al lenguaje que se está iniciando un nuevo párrafo, se usa para darle un poco más de orden a la página, aunque si no se usa solo se obtendrán líneas de texto del ancho de la misma página. y por último la etiqueta `<BR>` que permite hacer un salto de línea esta no necesita tener una etiqueta de cierre.





Actividad 4. Estructurando Mi Página Web



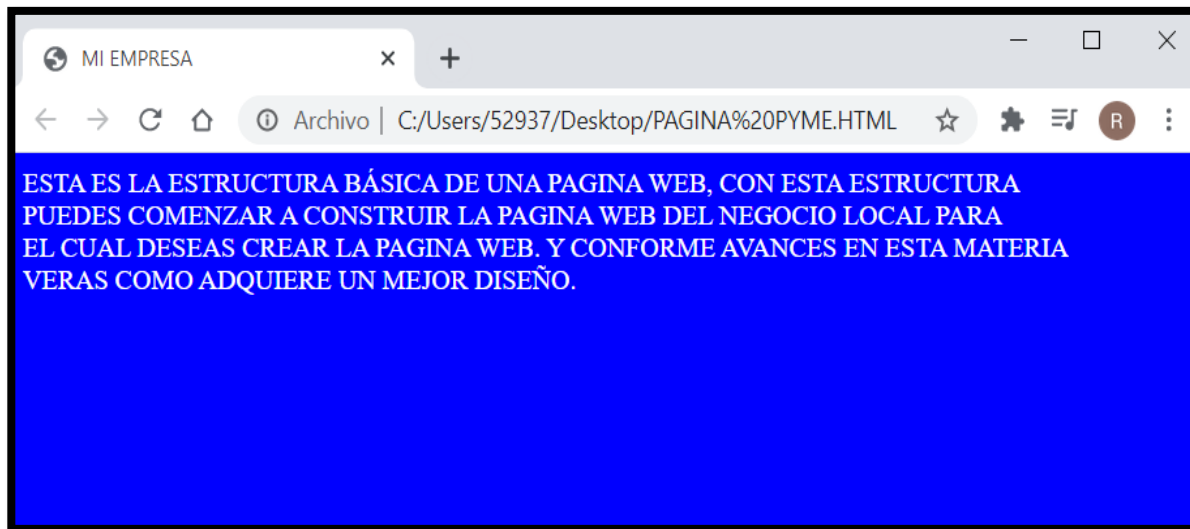
Instrucciones: Después de haber analizado la Lectura 4. “La Estructura de la Web”. Recorta las etiquetas correspondientes y pégalas en el recuadro para que formes la estructura básica de la página que se muestra en la figura.



CÓDIGO EN HTML

```
<!DOCTYPE HTML>
<HTML>
<HEAD>
    <TITLE>MI EMPRESA</TITLE>
</HEAD>
<BODY BGCOLOR="BLUE">
    <P><FONT COLOR="WHITE">
        ESTA ES LA ESTRUCTURA BÁSICA DE UNA PAGINA
        WEB, CON ESTA ESTRUCTURA<BR>
        PUEDES COMENZAR A CONSTRUIR LA PAGINA WEB
        DEL NEGOCIO LOCAL PARA<BR>
        EL CUAL DESEAS CREAR LA PAGINA WEB. Y
        CONFORME AVANCES EN ESTA MATERIA<BR>
        VERAS COMO ADQUIERE UN MEJOR DISEÑO.
    </FONT></P>
</BODY>
</HTML>
```





ESPACIO DE RECORTES



Práctica 1. Mi Primer Página Web

Instrucciones: Escribe la estructura básica de un programa de HTML en la aplicación SublimeText, o en la aplicación que tu docente te solicite, si **NO** puedes realizarlo en la aplicación puedes escribir a mano el código.



PRÁCTICA 1. "MI PRIMER PÁGINA WEB"



1. Investiga el nombre y el giro de una empresa pequeña o mediana de tu localidad que desee que se construya una página web básica para su negocio. Puede ser incluso una tiendita o si en tu casa tienen algún tipo de negocio familiar puedes empezar a construir su página web para colocarla en Internet.
2. Cuando tengas el nombre y giro del negocio o empresa, puedes empezar a construir la estructura básica de la página web, siguiendo el ejemplo y sustituyendo la información correspondiente como nombre del negocio:

NOTA: Lo que se encuentra subrayado y con color de fuente roja es información que puedes sustituir en base a la empresa, giro o colores que quieras utilizar para la página web.

```
<!DOCTYPE HTML>
<HTML>
<HEAD>
  <TITLE> "TIENDITA "DON DE DIOS"</TITLE>
</HEAD>
  <BODY BGCOLOR="BLUE">
    <P> <FONT COLOR="WHITE"> GIRO DE LA EMPRESA </FONT> </P>
    <P> <FONT COLOR="BLUE"> ESTE ESPACIO PUEDES EMPEZAR A
    UTILIZARLO PARA ESCRIBIR UN POCO ACERCA DEL GIRO DE LA EMPRESA O UN
    POCO DE SU HISTORIA PARA IRLA CONOCIENDO POCO A POCO RECUERDA
    QUE PUEDES USAR LA ETIQUETA <BR> PARA DAR UN SALTO DE LINEA EN LA
    INFORMACION <BR></FONT></P>
    <P> <FONT COLOR="BLACK"> PRODUCTOS QUE VENDE </FONT>
    </P>
    <P><FONT COLOR="WHITE"> ESTE ESPACIO
    PUEDES COLOCAR LOS PRODUCTOS QUE OFERTA Y ALGUNAS PROMOCIONES
    CON LAS QUE CUENTE <BR></FONT></P>
  </BODY>
</HTML>
```





3. Realiza una captura de pantalla de como quedo tu página hasta este punto imprímela y pégala en este espacio, de igual forma si no puedes imprimirla puedes dibujar como quedo.

CAPTURA DE PANTALLA O DIBUJO DE PRÁCTICA 1





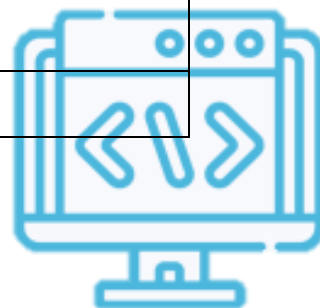
INSTRUMENTO DE EVALUACIÓN
LISTA DE COTEJO 4
Práctica 1: "Mi Primer Página Web"



DATOS GENERALES

Nombre(s) del alumno(s)	Matrícula(s)
Producto: Código y Captura o Dibujo Página Web	Fecha
Submódulo 2: Programación Web	Periodo: 2025 - 2026B
Nombre del docente	Firma del docente

CRITERIO	INDICADORES	VALOR OBTENIDO		PONDERACIÓN	CALIF	OBSERVACIONES Y/O SUGERENCIAS DE MEJORA
		SÍ	NO			
1	La página tiene un nombre en la pestaña del navegador.			1		
2	Contiene el Giro de la empresa en las etiquetas <P>			1		
3	Aplica las etiquetas para dar color al texto de forma correcta			2		
4	Contiene la información de los productos y promociones del comercio de la localidad.			2		
5	Sustituye el texto subrayado y en rojo por la información propia de la empresa.			1		
6	Aplica un color de fondo a la página web			1		
7	Se distingue la estructura de HTML aplicando las etiquetas básicas.			1		
8	Muestra la captura de pantalla o el dibujo del diseño final de la práctica.			1		
CALIFICACIÓN						

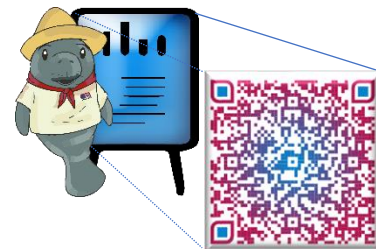


Lectura No 5. Elementos Básicos de las Páginas Web

Instrucciones: Realiza el análisis de la siguiente información, posteriormente, completa la Actividad 5. Aprendo a identificar los elementos básicos.



LECTURA 5. "ELEMENTOS BÁSICOS DE LAS PÁGINAS WEB: TÍTULOS, HIPERVINCULOS Y VIDEOS"



Etiquetas de Títulos

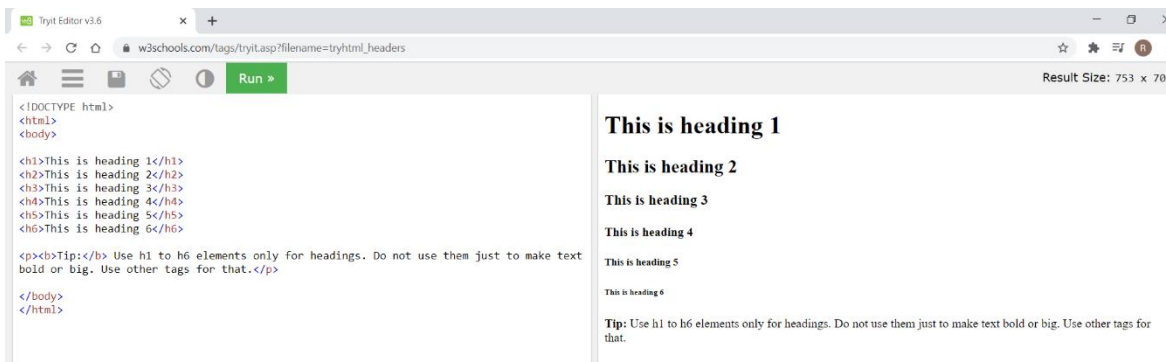
Hay hasta 6 etiquetas en HTML para definir títulos o secciones. Todas ellas llevan por defecto el texto en negrita, y son: **<h1>**, **<h2>**, **<h3>**, **<h4>**, **<h5>** y **<h6>**.

La etiqueta **<h1>** es la de mayor importancia y por tanto se utiliza para definir los titulares de la página. La importancia del resto de etiquetas es descendente, de forma que la etiqueta **<h6>** es la que se utiliza para delimitar las secciones menos importantes de la página.

Todas ellas tienen su respectiva etiqueta de cierre, y al igual que en los párrafos, el texto del título va encerrado entre las etiquetas de apertura y cierre:

```
<h1> Texto_del_título de nivel 1 </h1>
<h2> Texto_del_título de nivel 2 </h2>
.....
<h6> Texto_del_título de nivel 6 </h6>
```

Puedes probar tus códigos en HTML de forma online y observar el ejemplo con el uso de etiquetas de título escaneando el siguiente código.



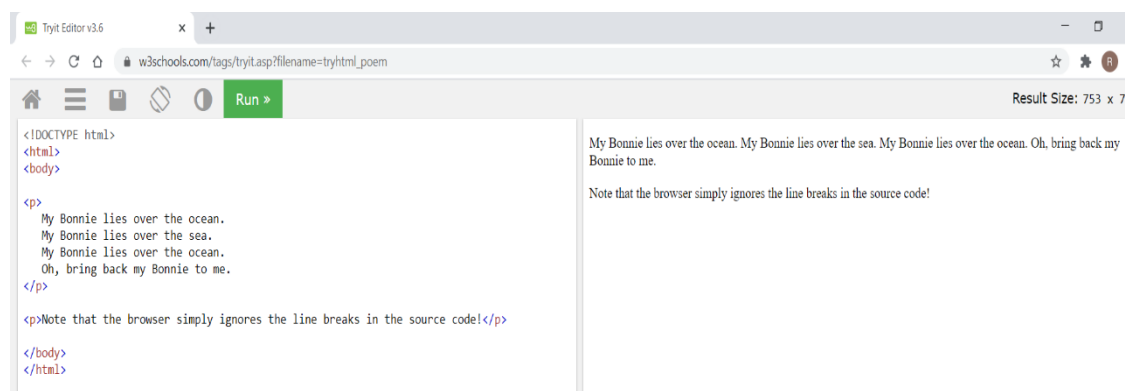
Etiquetas párrafo

Párrafos. Tal vez la etiqueta más utilizada en la composición de páginas sea la etiqueta que define los párrafos. esta es la etiqueta:

`<p> Texto_del_parrafo </p>`

Para crear un párrafo se encierra su texto entre las etiquetas `<p>` `</p>`. Los párrafos creados con HTML son elementos de bloque, por lo que siempre ocupan toda la anchura del elemento que los contiene (por defecto la ventana del navegador). La etiqueta crea un salto de línea de manera que el elemento siguiente se coloca siempre empezando una nueva línea. Observa el siguiente ejemplo:

https://www.w3schools.com/tags/tryit.asp?filename=tryhtml_poem



Etiqueta Imágenes

Cómo insertar una imagen:

``

La `<img>` etiqueta se utiliza para incrustar una imagen en una página HTML.

Las imágenes no se insertan técnicamente en una página web; las imágenes están vinculadas a páginas web. La `<img>` etiqueta crea un espacio de espera para la imagen referenciada.

La `<img>` etiqueta tiene dos atributos obligatorios:

- **src:** especifica la ruta a la imagen
- **alt:** especifica un texto alternativo para la imagen, si la imagen por alguna razón no se puede mostrar

Nota: Además, siempre especifique el ancho y el alto de una imagen. Si no se especifican el ancho y el alto, la página puede parpadear mientras se carga la imagen.



AQUÍ PUEDES VER UN EJEMPLO



Sugerencia: para vincular una imagen a otro documento, simplemente anide la `<img>` etiqueta dentro de una etiqueta `<a>` (vea el ejemplo a continuación).

Etiqueta Hipervínculos

La `<a>` etiqueta define un hipervínculo, que se utiliza para vincular de una página a otra.

El atributo más importante del `<a>` elemento es el atributo href, que indica el destino del enlace.

De forma predeterminada, los enlaces aparecerán de la siguiente manera en todos los navegadores:

- Un enlace no visitado está subrayado y en azul
- Un enlace visitado está subrayado y en violeta
- Un enlace activo está subrayado y en rojo

De forma predeterminada, la página vinculada se mostrará en la ventana del navegador actual. Para cambiar esto, debe especificar otro destino para el enlace.

El `target` atributo especifica dónde abrir el documento vinculado.

El `target` atributo puede tener uno de los siguientes valores:

- `_self` - Defecto. Abre el documento en la misma ventana / pestaña en la que se hizo clic
- `_blank` - Abre el documento en una nueva ventana o pestaña
- `_parent` - Abre el documento en el marco principal
- `_top` - Abre el documento en el cuerpo completo de la ventana.



Ejemplo

Utilice `target = "_blank"` para abrir el documento vinculado en una nueva ventana o pestaña del navegador:

```
<a href="https://www.w3schools.com/" target="_blank"> Visit W3Schools! </a>
```

Ejemplo

Cómo usar una imagen como enlace:

```
<a href="https://www.w3schools.com">
```

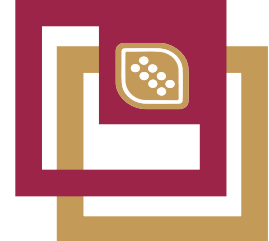
```
 </a>
```

Ejemplo

Cómo abrir un enlace en una nueva ventana del navegador:

```
<a href="https://www.w3schools.com" target="_blank"> Visit W3Schools.com! </a>
```





Ejemplo

Cómo vincular a una dirección de correo electrónico:

`<a href="mailto:someone@example.com"> Send email</a>`

Ejemplo

Cómo vincular a otra sección de la misma página:

`<a href="#section2"> Go to Section 2 </a>`

URL absolutas frente a URL relativas

Ambos ejemplos anteriores utilizan una **URL absoluta** (una dirección web completa) en el **href** atributo. Un enlace local (un enlace a una página dentro del mismo sitio web) se especifica con una **URL relativa** (sin la parte "https: // www"):

Ejemplo

`<h2> Absolute URLs </h2>`

`<p> <a href="https://www.w3.org/"> W3C </a> </p>`

`<p> <a href="https://www.google.com.mx/"> Google</a> </p>`

`<h2> Relative URLs </h2>`

`<p> <a href="html_images.asp"> HTML Images </a> </p>`

`<p> <a href="/css/default.asp"> CSS Tutorial </a> </p>`



Puedes probar tú mismo como se observan cada uno de los ejemplos:



Etiqueta Videos

El `<video>` elemento HTML se utiliza para mostrar un video en una página web.

El elemento HTML `<video>`

Para mostrar un video en HTML, use el `<video>` elemento:

FORMATO DE ARCHIVO	TIPO DE MEDIO
MP4	video / mp4
WebM	video / webm
Ogg	video / ogg

Ejemplo:

`<video width="320" height="240" controls>`

`<source src="movie.mp4"`

`type="video/mp4">`

`<source src="movie.ogg" type="video/ogg">`

`Your browser does not support the video tag.`

`</video>`





Cómo funciona

- El **controls** atributo agrega controles de video, como reproducción, pausa y volumen.
- Es una buena idea incluir siempre **width** y **height** atributos. Si no se configuran el alto y el ancho, la página puede parpadear mientras se carga el video.
- El **<source>** elemento le permite especificar archivos de video alternativos entre los que el navegador puede elegir. El navegador utilizará el primer formato reconocido.
- El texto entre el **<video>** y **</video>** etiquetas sólo se mostrará en los navegadores que no soportan el **<video>** elemento.

HTML **<video>** Reproducción automática

Para iniciar un video, use automáticamente el **autoplay** atributo:

Ejemplo

```
<video width="320" height="240" autoplay>
```

```
<source src="movie.mp4" type="video/mp4">
```

```
<source src="movie.ogg" type="video/ogg">
```

Your browser does not support the video tag.

```
</video>
```



Puedes probar tú mismo como se observan cada uno de los ejemplos



Video HTML: métodos, propiedades y eventos

El DOM HTML define métodos, propiedades y eventos para el **<video>** elemento. Esto le permite cargar, reproducir y pausar videos, así como configurar la duración y el volumen. También hay eventos DOM que pueden notificarle cuando un video comienza a reproducirse, se pausa, etc.

Etiqueta Audio

El elemento HTML **<audio>**

Para reproducir un archivo de audio en HTML, use el **<audio>** elemento:

Ejemplo

```
<audio controls>
```

```
<source src="horse.ogg" type="audio/ogg">
```

```
<source src="horse.mp3" type="audio/mpeg">
```

Your browser does not support the audio element.

```
</audio>
```





Audio HTML: cómo funciona

- El **controls** atributo agrega controles de audio, como reproducción, pausa y volumen.
- El **<source>** elemento le permite especificar archivos de audio alternativos entre los que puede elegir el navegador. El navegador utilizará el primer formato reconocido.
- El texto entre el **<audio>** y **</audio>** etiquetas sólo se mostrará en los navegadores que no soportan el **<audio>** elemento.

Formatos de audio HTML

Hay tres formatos de audio compatibles: **MP3, WAV y OGG**. El soporte del navegador para los diferentes formatos es:

NAVEGADOR	MP3	WAV	OGG
Internet Explorer	SÍ	NO	NO
Chrome	SÍ	SÍ	SÍ
Firefox	SÍ	SÍ	SÍ
Safari	SÍ	SÍ	NO
Ópera	SÍ	SÍ	SÍ

FORMATO DE ARCHIVO	TIPO DE MEDIO
MP3	AUDIO/MPEG
OGG	AUDIO/OGG
WAV	AUDIO/WAV



Actividad 5. Aprendo a Identificar Los Elementos Básicos: Títulos, Párrafos, Imágenes, Hipervínculos, Videos, Sonidos



Instrucciones: Después de haber analizado la Lectura 5. “Elementos Básicos De Las Páginas Web”. Observa la siguiente página web en desarrollo y coloca dentro del círculo el número de las etiquetas HTML que le corresponde a cada elemento. Después contesta las preguntas que se solicitan:



IDENTIFICA LAS ETIQUETAS

NOTA: En la siguiente página encontraras los números con las etiquetas correspondientes y recuerda que algunas etiquetas pueden repetirse. No olvides contestar las preguntas.

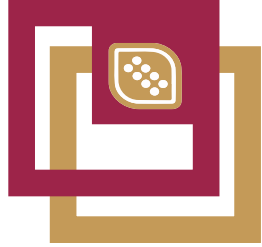




TABLA DE ETIQUETAS Y PREGUNTAS

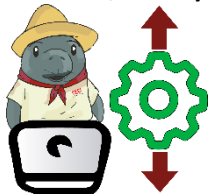
Etiquetas	Preguntas
1. <audio>	¿Qué atributos agrega controles para pausa, reproducción y volumen?
2. <h1>	¿Por qué de todas las etiquetas de encabezado esta se considera la más importante?
3. 	¿Cuál es la importancia de los atributos src y alt dentro de esta etiqueta?
4. <p>	¿Para qué sirve esta etiqueta?
5. <h2>	¿Cómo representarías el código en HTML usando esta etiqueta en la imagen anterior?
6. <video>	¿Cuál es la función del atributo source en esta etiqueta?
7. <title>	¿Como se escribiría en HTML para que aparezca el titulo como en la figura anterior?





Práctica 2. Mi Página Web con Multimedia

Instrucciones: Escribe la estructura de una Página Web utilizando los elementos básicos, en la cual describirás tu asignatura favorita en la aplicación SublimeText, o en la aplicación que tu docente te solicite, si **NO** puedes realizarlo en la aplicación puedes escribir a mano el código.

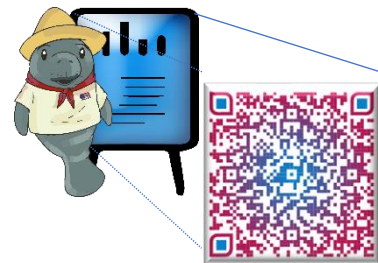


PRÁCTICA 2. "MI PÁGINA WEB CON MULTIMEDIA"

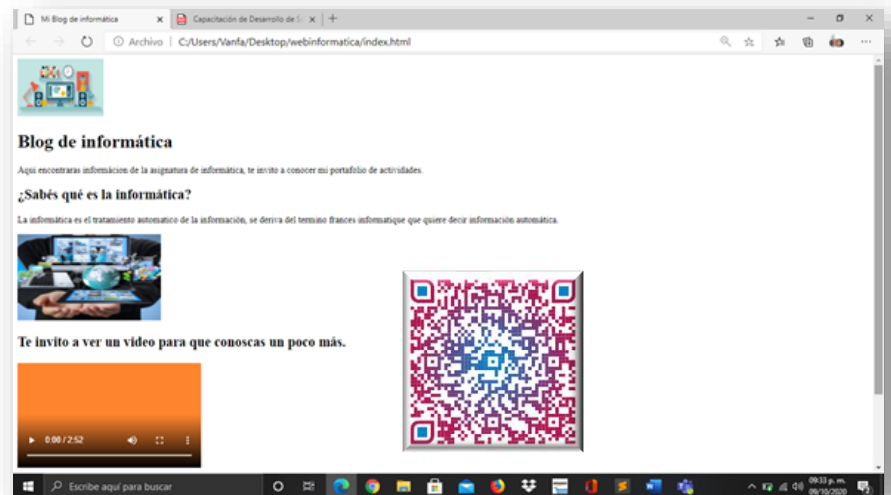


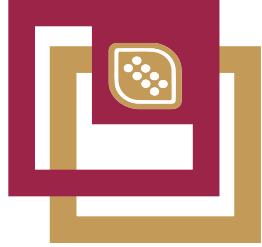
1. En tu editor de código crea un nuevo archivo llamado Practica_JAFP.html, donde las 4 letras corresponden a las iniciales de tu nombre.
2. Analiza y escribe el código para elaborar tu página web
3. Tu página web debe de llevar lo siguiente

- a) Una imagen como logotipo
- b) Un título con el nombre de tu página web
- c) Una descripción de que trata tu pagina
- d) Incluir un subtítulo de un tema
- e) Agregar una descripción de ese tema
- f) Incluir una imagen referente al tema
- g) Incluir un subtítulo para visualizar un video
- h) Incluir un video



4. Observa cómo debe de quedar y guíate de este prototipo adáptalo a tu asignatura preferida.





Código HTML de la pagina

NOTA: LAS ETIQUETAS SUBRYADAS Y EN COLOR AZUL TU DOCENTE PODRÁ EXPLICARTE SU FUNCIONAMIENTO DENTRO DEL CODIGO HTML.

```
<!DOCTYPE HTML>
<HTML>
<head>
  <meta charset="utf-8">
  <title> Mi Blog de Informática </title>
  <link rel="stylesheet" type="text/css" href="estilos.css">
</head>
<body>
  <div id="contenedor">
    <header>
      
      <h1> Blog de informática </h1>
      <p id = "ptitulo">Aquí encontraras información de la asignatura de
      informática, te invito a conocer mi portafolio de actividades. </p>
    </header>

    <section id="uno">
      <h2> ¿Sabes qué es la informática? </h2>
      <p> La informática es el tratamiento automático de la información, se
      deriva del término francés informatique que quiere decir información
      automática. </p>
      

      <h2> Te invito a ver un video para que conozcas un poco más. </h2>
      <video width="320" height="185" controls>
        <source src="pagweb.mp4" type="video/mp4">
        Tu navegador no soporta video.
      </video>
    </section>
  </div>
</body>
</html>
```

<meta charset="utf-8">

Está destinado a indicar la codificación de caracteres utilizada para HTML 5



<link rel="stylesheet" type="text/css" href="estilos.css">

El atributo REL se usa para definir la relación entre el archivo enlazado y el documento HTML. REL=StyleSheet especifica un estilo persistente o preferido mientras que REL="Alternate StyleSheet" define un estilo alterno.





INSTRUMENTO DE EVALUACIÓN
LISTA DE COTEJO 5
Práctica 2: "Mi Página Web con Multimedia"



DATOS GENERALES

Nombre(s) del alumno(s)	Matrícula(s)
Producto: Código y Captura o Dibujo Página Web	Fecha
Submódulo 2: Programación Web	Periodo: 2025 - 2026B
Nombre del docente	Firma del docente

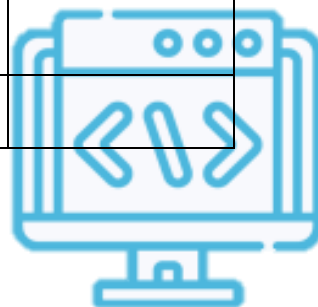
CRITERIO	INDICADORES	VALOR OBTENIDO		PONDERACIÓN	CALIF	OBSERVACIONES Y/O SUGERENCIAS DE MEJORA
		SÍ	NO			
1	El código se encuentra estructurado de forma correcta.			1		
2	Incluye las etiquetas HTML correspondientes para los títulos.			1		
3	Utiliza las etiquetas para inserción de videos.			2		
4	Utiliza las etiquetas para inserción de imágenes.			2		
5	Entrega en tiempo y forma.			1		
6	Muestra los controles del video.			1		
7	Utiliza diversas etiquetas de títulos.			2		
CALIFICACIÓN						

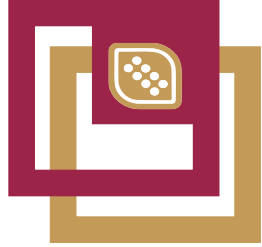
CONCLUYE:

¿Qué aprendí?

¿Cuál es la etiqueta para insertar imágenes?

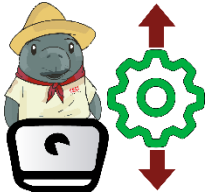
¿Con que finalidad puedo aplicar la inserción de imágenes en una página web?





Práctica 3. Mejorando Mi Página Web

Instrucciones: Escribe la estructura de una Página Web utilizando los elementos básicos, en la cual describirás tu asignatura favorita y podrás ir mejorando con nuevas etiquetas y elementos en la aplicación SublimeText, o en la aplicación que tu docente te solicite, **si NO puedes realizarlo en la aplicación puedes escribir a mano el código.**

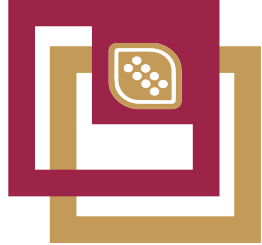


PRÁCTICA 3. "MEJORANDO MI PÁGINA WEB"



1. Abre el archivo Practica_JAFP.html de la práctica 2
2. Colócate al **final de tu página debajo donde agregaste el video**
3. Agrégale tus datos personales utilizando las etiquetas correspondientes
4. Utiliza hipervínculos para ir a tus redes sociales (Facebook, Instagram, YouTube) los hipervínculos deben de abrirse en otra pestaña del navegador
5. Añade un audio de podcast
6. Observa cómo debe de quedar y guíate de este prototipo adáptalo a tu asignatura preferida.





Código HTML de la página

```
<video width="320" height="185" controls>
<source src="pagweb.mp4" type="video/mp4">
  Tu navegador no soporta video.
</video>
</section>

<section id="dos">
  <h3> Esta página esta elaborada por: </h3>
  <p> Nombre del alumnos del COBATAB </p>
  <p> Te invito a seguirme en mis redes sociales </p>
  <br>
  <a href="https://www.facebook.com" target="_blank"> Mi Facebook </a>
  <a href="https://www.twitter.com" target="_blank"> Mi Twitter </a>
  <h3> Escucha el mensaje del día que tengo para ti </h3>
  <audio controls>
    <source src="horse.ogg" type="audio/ogg">
    <source src="podcast.mp3" type="audio/mp3">
    Tu navegador no soporta el audio.
  </audio>
</section>
</div>
</body>
</html>
```

**RECUERDA ESTA SECCIÓN NO TIENES QUE VOLVER A ESCRIBIRLA
YA LA CREAMOS EN LA PRACTICA 2 EN EL ARCHIVO
Practica_JAFP.html SOLO LA MUESTRO COMO REFERENCIA**





INSTRUMENTO DE EVALUACIÓN
LISTA DE COTEJO 6
Práctica 3: “Mejorando Mi Página Web”



DATOS GENERALES

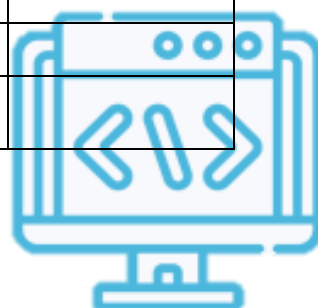
Nombre(s) del alumno(s)	Matrícula(s)
Producto: Código y Captura o Dibujo Página Web	Fecha
Submódulo 2: Programación Web	Periodo: 2025 - 2026B
Nombre del docente	Firma del docente

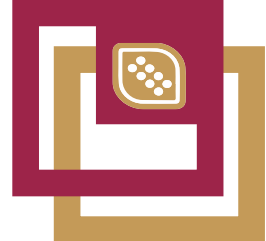
CRITERIO	INDICADORES	VALOR OBTENIDO		PONDERACIÓN	CALIF	OBSERVACIONES Y/O SUGERENCIAS DE MEJORA
		SÍ	NO			
1	El código muestra la estructura correcta de una página web en HTML.			1		
2	Incluye las etiquetas HTML correspondientes al tema de la lectura.			2		
3	Se aplican correctamente las etiquetas para inserción de audio.			2		
4	Se aplican correctamente las etiquetas para el uso de hipervínculos			2		
5	Al realizar el uso de hipervínculos estos se apertura en otra ventana o pestaña			1		
6	Se muestran los controles de audio y video.			1		
7	Entrega en tiempo y forma			1		
				CALIFICACIÓN		

CONCLUYE:

¿Qué aprendí? ¿Cuál es la etiqueta para la alineación de párrafos?

¿Para qué sirven los hipervínculos?





Lectura No 6. Estilos: Fuente, Tamaño y Colores

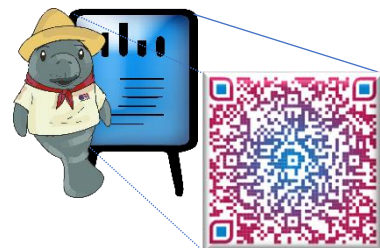
Instrucciones: Realiza el análisis de la siguiente información, posteriormente, realiza la Práctica 4. Enchulando Mi Web.



LECTURA 6. "ESTILOS: FUENTE, TAMAÑO Y COLORES"

¿Qué es CSS?

- **CSS son las siglas de Cascading Style Sheets**
- CSS describe **cómo se mostrarán los elementos HTML en pantalla, papel o en otros medios.**
- CSS **ahorra mucho trabajo.** Puede controlar el diseño de varias páginas web a la vez
- Las hojas de estilo externas se almacenan en **archivos CSS**
- CSS es el lenguaje que usamos para diseñar un documento HTML.
- CSS describe cómo se deben mostrar los elementos HTML.



¿Por qué utilizar CSS?

CSS se utiliza para definir estilos para sus páginas web, incluido el diseño y las variaciones en la visualización para diferentes dispositivos y tamaños de pantalla.



Sintaxis CSS

Un conjunto de reglas CSS consta de un selector y un bloque de declaración:

- El selector apunta al elemento HTML al que desea aplicar estilo.
- El bloque de declaración contiene una o más declaraciones separadas por punto y coma.
- Cada declaración incluye un nombre de propiedad CSS y un valor, separados por dos puntos.
- Varias declaraciones de CSS se separan con punto y coma y los bloques de declaración están rodeados por llaves.

Ejemplo

En este ejemplo, todos los elementos <p> estarán alineados al centro, con un color de texto rojo:

```
p {
  color: red;
  text-align: center;
}
```





Ejemplo explicado

- **P** es un **selector** en CSS (apunta al elemento HTML que desea diseñar: <p>).
- **color** es una propiedad y **red** es el valor de la propiedad
- **text-align** es una propiedad y **center** es el valor de la propiedad

Selectores CSS

Los selectores CSS se utilizan para "buscar" (o seleccionar) los elementos HTML que desea aplicar estilo.

Podemos dividir los selectores de CSS en cinco categorías:

- Selectores simples (seleccione elementos basados en nombre, identificación, clase)
- Selectores de combinador (seleccione elementos basados en una relación específica entre ellos)
- Selectores de pseudo-clase (seleccione elementos basados en un cierto estado)
- Selectores de pseudoelementos (seleccionar y aplicar estilo a una parte de un elemento)
- Selectores de atributos (seleccione elementos basados en un atributo o valor de atributo)

El selector de elementos CSS

El selector de elementos selecciona elementos HTML según el nombre del elemento.

Ejemplo

Aquí, todos los elementos <p> de la página estarán alineados al centro, con un color de texto rojo:

```
p {
  text-align: center;
  color: red;
}
```

Altura y ancho de CSS

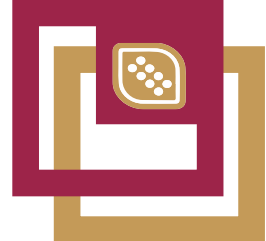
- Las propiedades **height** y **width** se utilizan para establecer la altura y el ancho de un elemento.
- Las propiedades de alto y ancho no incluyen relleno, bordes ni márgenes. Establece la altura / ancho del área dentro del relleno, borde y margen del elemento.

Ejemplo

Establezca la altura y el ancho de un elemento <div>:

```
div {
  height: 200px;
  width: 50%;
  background-color: powderblue;
}
```





Colores CSS

Los colores se especifican utilizando nombres de colores predefinidos o valores RGB, HEX, HSL, RGBA, HSLA.

- **Color de fondo CSS**

Puede establecer el color de fondo de los elementos HTML:

Ejemplo

```
<h1 style="background-color:DodgerBlue;"> Hello World </h1>
```

```
<p style="background-color:Tomato;"> Lorem ipsum... </p>
```



AQUÍ PODRÁS
ENCONTRAR LOS
CÓDIGOS DE COLORES

- **Color de texto CSS**

Puede establecer el color del texto:

Ejemplo

```
<h1 style="color:Tomato;">Hello World</h1>
```

```
<p style="color:DodgerBlue;">Lorem ipsum...</p>
```

```
<p style="color:MediumSeaGreen;">Ut wisi enim...</p>
```



- **Color de texto**

Para **color** esta propiedad se utiliza para establecer el color del texto. El color viene especificado por:

- un nombre de color, como "rojo"
- un valor HEX, como "# ff0000"
- un valor RGB, como "rgb (255,0,0)"

Consulta el código para obtener una lista completa de posibles valores de color.

El color de texto predeterminado para una página se define en el selector de cuerpo.



```
body {  
  color: blue;  
}
```

```
h1 {  
  color: green;  
}
```





- **Color del borde CSS**

Puede establecer el color de los bordes:

Ejemplo

```
<h1 style="border:2px solid Tomato;"> Hello World </h1>
```

```
<h1 style="border:2px solid DodgerBlue;"> Hello World </h1>
```

```
<h1 style="border:2px solid Violet;"> Hello World </h1>
```

Fuentes CSS

Las propiedades de fuente CSS definen la familia de fuentes, la negrita, el tamaño y el estilo de un texto.

Familias de fuentes CSS

En CSS, hay dos tipos de nombres de familias de fuentes:

- **familia genérica:** un grupo de familias de fuentes con un aspecto similar (como "Serif" o "Monospace").
- **familia de fuentes:** una familia de fuentes específica (como "Times New Roman" o "Arial").

Familia tipográfica

- La familia de fuentes de un texto se establece con la **font-family** propiedad.
- La **font-family** propiedad debe contener varios nombres de fuentes como sistema de "reserva". Si el navegador no admite la primera fuente, intenta la siguiente fuente y así sucesivamente.
- Comience con la fuente que desee y termine con una familia genérica, para permitir que el navegador elija una fuente similar en la familia genérica, si no hay otras fuentes disponibles.

Estilo de fuente

La **font-style** propiedad se usa principalmente para especificar texto en cursiva.

Esta propiedad tiene tres valores:

- normal: el texto se muestra normalmente.
- cursiva: el texto se muestra en cursiva.
- oblicuo: el texto es "inclinado" (oblicuo es muy similar a la cursiva, pero menos compatible).





Peso de fuente

La **font-weight** propiedad especifica el peso de una fuente:

```
p.normal {  
    font-weight: normal;  
}  
  
p.thick {  
    font-weight: bold;  
}
```



AQUÍ PODRÁS ENCONTRAR
LOS ESTILOS PARA APLICAR
A LAS FUENTES

Tamaño de fuente

- La **font-size** propiedad establece el tamaño del texto.
- Ser capaz de administrar el tamaño del texto es importante en el diseño web. Sin embargo, no debe utilizar ajustes de tamaño de fuente para que los párrafos se vean como encabezados o los encabezados se vean como párrafos.
- Utilice siempre las etiquetas HTML adecuadas, como <h1> - <h6> para los títulos y <p> para los párrafos.
- El valor del tamaño de fuente puede ser un tamaño absoluto o relativo.

Tamaño absoluto:

- Establece el texto en un tamaño especificado.
- No permite que un usuario cambie el tamaño del texto en todos los navegadores (malo por razones de accesibilidad).
- El tamaño absoluto es útil cuando se conoce el tamaño físico de la salida.

Tamaño relativo:

- Establece el tamaño relativo a los elementos circundantes.
- Permite a un usuario cambiar el tamaño del texto en los navegadores.

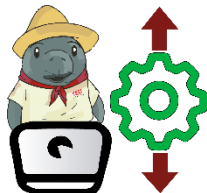


AQUÍ PODRÁS
ENCONTRAR UN CURSO
COMPLETO SOBRE CSS



Práctica 4. Enchulando Mi Web

Instrucciones: Escribe la estructura de una Página Web utilizando los elementos básicos, y podrás ir mejorando con nuevas etiquetas aplicando Estilos CSS y elementos en la aplicación SublimeText, o en la aplicación que tu docente te solicite, **si NO puedes realizarlo en la aplicación puedes escribir a mano el código.**



PRÁCTICA 4. "ENCHULANDO MI WEB"



1. Abre el archivo Practica_JAFP.html de la practica 3
2. Abre también un archivo nuevo en tu editor de código favorito crea un archivo llamado estilos.css
3. A tu **página web**, es decir, en el archivo **Practica_JAFP.html** en la etiqueta meta colócale el código de enlace a la hoja de estilos de CSS, asegúrate sea el nombre de estilos.css, si ya lo tienen no lo modifiques.

```
<link rel="stylesheet" type="text/css" href="estilos.css">
```



4. Ahora dentro del archivo CSS escribe el código correspondiente para que tu página web tenga un estilo único:
 - a. Cambia tamaño de letra
 - b. Colores de letras
 - c. Posiciones
 - d. Tipo de letra

NOTA: Puedes seguir el ejemplo del código que escribirás en el archivo estilos.css como se muestra a continuación, Y RECUERDA TU PUEDES CAMBIAR EL ESTILO, color, fuente, tipo de fuente, en base a lo que tú quieras mostrar





Código HTML de la Página "estilos.css"

```
body{
  background: white;
}
```

```
#contenedor{
  width: 960px;
  margin: auto;
}
```

```
header{
  height: 200px;
  background: violet;
}
```

```
header img{
  margin-top: 20px;
  margin-left: 20px;
}
```

```
header h1{
  margin-right: 170px;
  float: right;
  color: violet;
  font-style: bold;
  font-size: 50px;
  color: white;
}
```

```
#ptitulo{
  text-align: center;
  font-size: 15px;
  color: white;
  font-style: bold;
}
```

```
#uno{
  background: pink;
}
```

```
section h2{
  font-style: bold;
  font-size: 30px;
  margin: auto;
}
```

RECUERDA QUE ESTE
CODIGO TIENES QUE
COLOCARLO EN EL ARCHIVO
estilos.css y vincularlo a la
página Practica_JAFP.html



PUEDES USAR LA
APLICACIÓN SUBLIME TEXT
E INCLUSO EN EL BLOC DE
NOTAS PREGUNTALE A TU
PROFESOR COMO HACERLO





```
p{
  margin: auto;
  font-size: 20px;
}

.sec{
  margin-left: 320px;
}

video{
  margin-left: 300px;
}

#dos{
  background: violet;
}

a{
  margin-top: 30px;
  text-decoration: none;
  text-align: center;
  padding: 10px;
  font-size: 20px;
  background: white;
  border-radius: 6px;
  margin-left: 20px;
}

audio{
  margin-left: 300px;
}
```



AQUÍ PODRAS
ENCONTRAR UN CURSO
COMPLETO SOBRE CSS



- Guarda los cambios en tu sitio web y una vez completado enséñaselo a tu profesor.





INSTRUMENTO DE EVALUACIÓN
LISTA DE COTEJO 7
Práctica 4: "Enchulando Mi Web"



DATOS GENERALES

Nombre(s) del alumno(s)	Matrícula(s)
Producto: Código y Captura o Dibujo Página Web	Fecha
Submódulo 2: Programación Web	Periodo: 2025 - 2026B
Nombre del docente	Firma del docente

CRITERIO	INDICADORES	VALOR OBTENIDO		PONDERACIÓN	CALIF	OBSERVACIONES Y/O SUGERENCIAS DE MEJORA
		SÍ	NO			
1	Se aplican los atributos necesarios para insertar los colores de fondo.			1		
2	La página web muestra un diseño llamativo.			2		
3	Utilizo atributos y etiquetas propias del lenguaje CSS.			2		
4	Cambio el tamaño de los textos.			2		
5	Los hipervínculos se abren en otra ventana o pestaña.			1		
6	Entrega en tiempo y forma.			2		
CALIFICACIÓN						

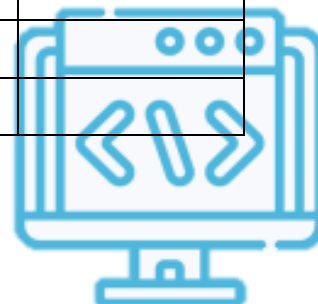
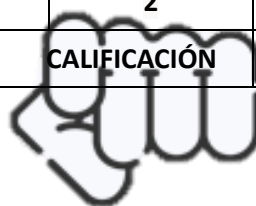


CONCLUYE:

¿Qué aprendí?

¿Cuál es la etiqueta para aplicar estilos a las páginas web?

¿Qué es una hoja de estilo en cascada?





Lectura No 7. Uso de Tablas en HTML

Instrucciones: Realiza el análisis de la siguiente información, posteriormente, realiza la Práctica 5.

Tablas en HTML.



LECTURA 7. "USO DE TABLAS EN HTML"

Una **tabla** es un conjunto de celdas organizadas dentro de las cuales podemos alojar distintos contenidos. En un principio nos podría parecer que las tablas son raramente útiles y que pueden ser utilizadas principalmente para listar datos como agendas, resultados y otros

Planes	Basic	Medium	Premium
◀ Lorem	Element	Element	Element
✓ Lorem	Element	Element	Element
◀ Lorem	Element	Element	Element
	Buy	Buy	Buy

datos de una forma organizada. Nada más lejos de la realidad. Hoy, gran parte de los diseñadores de páginas basan su maquetación en este tipo de artilugios. En efecto, una tabla nos permite organizar y distribuir los espacios de la manera más óptima. Nos puede ayudar a generar texto en columnas como los periódicos, prefijar los tamaños ocupados por distintas secciones de la página o poner de una manera sencilla un pie de foto a una imagen. Puede que en un principio nos resulte un poco complicado trabajar con estas estructuras, pero, si deseamos crear una página de calidad, tarde o temprano tendremos que vernoslas con ellas y nos daremos cuenta de las posibilidades que nos ofrecen. Para empezar, nada más sencillo que por el principio: las tablas son definidas por las etiquetas.

<table> y </table>

Dentro de estas dos etiquetas colocaremos todas las otras etiquetas, textos e imágenes que darán forma y contenido a la tabla. Las tablas son descritas por líneas de izquierda a derecha. Cada una de estas líneas es definida por otra etiqueta y su cierre:

<tr> y </tr>





<table>

<td></td>	<td></td>	<td></td>	</tr>
<td></td>	<td></td>	<td></td>	</tr>
<td></td>	<td></td>	<td></td>	</tr>
<td></td>	<td></td>	<td></td>	</tr>

</table>

Asimismo, dentro de cada línea, habrá diferentes celdas. Cada una de estas celdas será definida por otro par de etiquetas.

Dentro de estas etiquetas será donde coloquemos nuestro contenido. Las etiquetas son:

<td> y </td>

Aquí se tiene un ejemplo de estructura de tabla:

<table>

<tr>

<td> Celda 1, Linea1 </td>

<td> Celda 2, Linea1 </td>

</tr>

<tr>

<td> Celda 1, Linea2 </td>

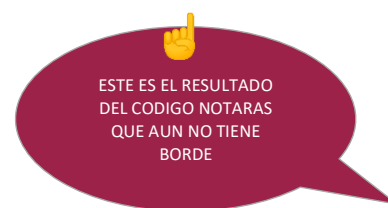
<td> Celda 2, Linea2 </td>

</tr>

</table>

Celda 1, línea 1 Celda 2, línea 1

Celda 1, línea 2 Celda 2, línea 2



Nota: Hasta aquí hemos visto todas las etiquetas que necesitamos conocer para crear tablas. Existen otras etiquetas, pero lo que podemos conseguir con ellas se puede conseguir también usando las que hemos visto. Por poner un ejemplo, señalamos la etiqueta, que sirve para crear una celda cuyo contenido esté formateado como un título o cabecera de la tabla. En la práctica, lo que hace es poner en negrita y centrado el contenido de esa celda, lo que se puede conseguir aplicando las correspondientes etiquetas dentro de la celda. Así:

<td align="center"> contenido de la celda </td>

A partir de esta idea simple y sencilla, las tablas adquieren otra magnitud cuando les incorporamos toda una batería de atributos aplicados sobre cada tipo de etiquetas que las componen. A lo largo de esta lectura nos adentraremos en el estudio de estos atributos de manera a proporcionaros los útiles indispensables para una buena puesta en página.



Tablas en HTML. Atributos para filas y celdas.

Hemos visto que las tablas están compuestas de líneas que, a su vez, contienen celdas. Las celdas son delimitadas por las etiquetas <td> o por las etiquetas <th> (si queremos texto en negrita y centrado) y constituyen un entorno independiente del resto del documento. Esto quiere decir que:

- Podemos usar prácticamente cualquier tipo de etiqueta dentro de la etiqueta para, de esta forma, dar forma a su contenido.
- Las etiquetas situadas en el interior de la celda no modifican el resto del documento.
- Las etiquetas de fuera de la celda no son tenidas en cuenta por ésta.

Así pues, podemos especificar el formato de nuestras celdas a partir de etiquetas introducidas en su interior o mediante atributos colocados dentro de la etiqueta de celda o bien, en algunos casos, dentro de la etiqueta, si deseamos que el atributo sea válido para toda la línea. La forma más útil y actual de dar forma a las celdas es a partir de las hojas de estilo en cascada que ya tendréis la oportunidad de abordar más adelante. Veamos a continuación algunos atributos útiles para la construcción de nuestras tablas. Empecemos viendo atributos que nos permiten modificar una celda en concreto o toda una línea:

- **align** Justifica el texto de la celda del mismo modo que si fuese el de un párrafo.
- **valign** Podemos elegir si queremos que el texto aparezca arriba (top), en el centro (middle) o abajo (bottom) de la celda.
- **bgcolor** Da color a la celda o línea elegida.
- **bordercolor** Define el color del borde.

Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
Public Sees 9:30 - 10:30	Public Sees 10:30 - 11:30	Public Sees 10:30 - 11:30	Public Sees 10:30 - 11:30	Public Sees 10:30 - 11:30	Public Sees 10:30 - 11:30	Public Sees 10:30 - 11:30
Apexing 12:30 - 13:30	Apexing 12:30 - 13:30	Education 12:30 - 13:30	Apexing 12:30 - 13:30	Education 12:30 - 13:30	Oldham Active 12:30 - 13:30	Public Sees 12:30 - 13:30
Public Sees 14:30 - 15:30	Parent & Baby 14:30 - 15:30	Public Sees 14:30 - 15:30	Public Sees 14:30 - 15:30	Public Sees 14:30 - 15:30	Parent and Baby 14:30 - 15:30	Apexing 14:30 - 15:30
Education 16:30 - 17:30	Public Sees 16:30 - 17:30	Public Sees 16:30 - 17:30	Education 16:30 - 17:30	Oldham Active 16:30 - 17:30	Public Sees 16:30 - 17:30	Public Sees 16:30 - 17:30

Otros atributos que pueden ser únicamente asignados a una celda y no al conjunto de celdas de una línea son:

- **background** Nos permite colocar un fondo para la celda a partir de un enlace a una imagen.
- **height** Define la altura de la celda en pixels o porcentaje.
- **width** Define la anchura de la celda en pixels o porcentaje.
- **colspan** Expande una celda horizontalmente.
- **rowspan** Expande una celda verticalmente.





Nota: El atributo height no funciona en todos los navegadores, además, su uso no está muy extendido. Las celdas por lo general tienen el alto que necesitan para que quepa todo el contenido que se le haya insertado, es decir, crecen lo suficiente para que quepa lo que hemos colocado dentro. El atributo width sí que funciona en todos los navegadores y lo tendréis que utilizar constantemente. Si le asignamos un ancho a la celda, el ancho será respetado y si dicha celda tiene mucho texto o cualquier otro contenido, la celda crecerá hacia abajo todo lo necesario para que quepa lo que hemos colocado.

Un matiz al último párrafo. Se trata de que, si definimos una celda de un ancho 100 por ejemplo, y colocamos en la celda un contenido como una imagen que mida más de 100 píxeles, la celda crecerá en horizontal todo lo necesario para que la imagen quepa. Si el elemento, aunque más ancho, fuera divisible (como un texto) el ancho sería respetado y el texto crecería hacia abajo o lo que es lo mismo, en altura, como señalábamos en el anterior párrafo.

Estos últimos cuatro atributos descritos son de gran utilidad. Concretamente, **height y width** nos ayudan a definir las dimensiones de nuestras celdas de una forma absoluta (en píxeles o puntos de pantalla) o de una forma relativa, es decir por porcentajes referidos al tamaño total de la tabla.



Ejemplo:

`<table width="80">` Dará una anchura de 80 píxeles a la celda. Sin embargo,

`<table width="80%">` Dará una anchura a la celda del 80% de la anchura de la tabla.

Hay que tener en cuenta que, definidas las dimensiones de las celdas, el navegador va a hacer lo que buenamente pueda para satisfacer al programador. Esto quiere decir que puede que en algunas ocasiones el resultado que obtengamos no sea el esperado. Concretamente, si el texto presenta una palabra excesivamente larga, puede que la anchura de la celda se vea aumentada para mantener la palabra en la misma línea. Por otra parte, si el texto resulta muy largo, la celda aumentará su altura para poder mostrar todo su contenido. Análogamente, si por ejemplo definimos dos anchuras distintas a celdas de una misma columna, el navegador no sabrá a cuál hacer caso. Es por ello por lo que resulta conveniente tener bien claro desde un principio como es la tabla que queremos diseñar. No está de más si la prediseñados en papel si la complejidad es importante.



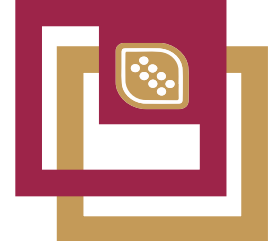


TABLE USING COLSPAN, ROWSPAN

A	B	C	D	E	F	AH	H	I	J	K	L	M
						G						
						GH						
						N						
						O						
						P						
						Q						
		R	S	T	U	V	W	X				
					Y							
Z	1	2	3	4	5	6	7	8	9	10		

El HTML resulta en general fácil pero las tablas pueden convertirse en un verdadero quebradero de cabeza si no llegamos a comprenderlas debidamente. Los atributos **rowspan** y **colspan** son también utilizados frecuentemente. Gracias a ellos es posible expandir celdas fusionando éstas con sus vecinas. El valor que pueden tomar

estas etiquetas es numérico. El número representa la cantidad de celdas fusionadas.

Así, `<td colspan="2">` Fusionara la celda en cuestión con su vecina a la derecha.

Esta celda tiene un colspan="2"	
Celda normal	Otra celda

Esta celda tiene rowspan="2", por eso tiene fusionada la celda de abajo.	Celda Normal
	Otra celda normal

Del mismo modo `<td rowspan="2">` Expandirá la celda hacia abajo fusionándose con la celda inferior.

El resto de los atributos presentados presentan una utilidad y uso bastante obvios. Los dejamos para que ustedes los exploren.



Atributos de la tabla

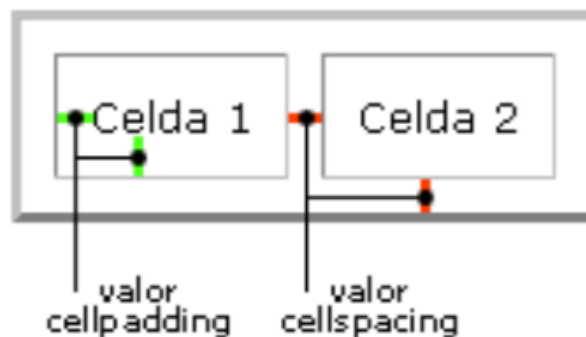
Además de los atributos específicos de cada celda o línea, las tablas pueden ser adicionalmente formateadas a partir de los atributos que nos ofrece la propia etiqueta.

- **Align.** Alinea horizontalmente la tabla con respecto a su entorno.
- **Background.** Nos permite colocar un fondo para la tabla a partir de un enlace a una imagen.
- **Bgcolor.** Da color de fondo a la tabla.
- **Border.** Define el número de pixels del borde principal.
- **Bordercolor.** Define el color del borde.
- **Cellpadding.** Define, en pixels, el espacio entre los bordes de la celda y el contenido de la misma.
- **Cellspacing.** Define el espacio entre los bordes (en pixels).
- **Height.** Define la altura de la tabla en pixels o porcentaje. width Define la anchura de la tabla en pixels o porcentaje.



Los atributos que definen las dimensiones, height y width, funcionan de una manera análoga a la de las celdas tal y como hemos visto. Contrariamente, el atributo align no nos permite justificar el texto de cada una de las celdas que componen la tabla, sino más bien, justificar la propia tabla con respecto a su entorno.

Los atributos **cellpadding** y **cellspacing** nos ayudaran a dar a nuestra tabla un aspecto más estético. En un principio puede parecernos un poco confuso su uso, pero un poco de practica será suficiente para hacerse con ellos. En la siguiente imagen podemos ver gráficamente el significado de estos atributos.



Puedes comprobar por ti mismo que los atributos definidos para una celda tienen prioridad con respecto a los definidos para una tabla. Podemos definir, por ejemplo, una tabla con color de fondo rojo y una de las celdas de color de fondo verde y se verá toda la tabla de color rojo menos la celda verde. Del mismo modo, podemos definir un color azul para los bordes de la tabla y hacer que una celda particular sea mostrada con un borde rojo. (Aunque esto no funcionará en todos los navegadores debido a que algunos no reconocen el atributo bordercolor)



Tabla de color rojo de fondo	El atributo bgcolor de la tabla está en rojo.
Celda normal	Esta celda está en verde. tiene el atributo bgcolor en color verde



Tablas anidadas

Muy útil también es el uso de tablas anidadas. De la misma forma que podíamos incluir listas dentro de otras listas, las tablas pueden ser incluidas dentro de otras. Así, podemos incluir una tabla dentro de la celda de otra. El modo de funcionamiento sigue siendo el mismo, aunque la situación puede complicarse si el número de tablas embebidas dentro de otras es elevado. Vamos a ver un código de anidación de tablas. Veamos primero el resultado y luego el código, así conseguiremos entenderlo mejor. Recuerda que para que el borde una tabla se vea debes usar el atributo border.

Celda de la tabla principal	<table> <tr> <td>Tabla anidada, celda 1</td><td>Tabla anidada, celda 2</td></tr> <tr> <td>Tabla anidada, celda 3</td><td>Tabla anidada, celda 4</td></tr> </table>	Tabla anidada, celda 1	Tabla anidada, celda 2	Tabla anidada, celda 3	Tabla anidada, celda 4
Tabla anidada, celda 1	Tabla anidada, celda 2				
Tabla anidada, celda 3	Tabla anidada, celda 4				

```
<table cellspacing="10" cellpadding="10" border="3">
<tr>
  <td align="center">
    Celda de la tabla principal
  </td>
  <td align="center">
    <table cellspacing="2" cellpadding="2" border="1">
      <tr>
        <td> Tabla anidada, celda 1</td>
        <td> Tabla anidada, celda 2</td>
      </tr>
      <tr>
        <td> Tabla anidada, celda 3</td>
        <td> Tabla anidada, celda 4</td>
      </tr>
    </table>
  </td>
</tr>
</table>
```

HTML encabezados de la tabla

Encabezados de la tabla se definen con la **<th>** tag.

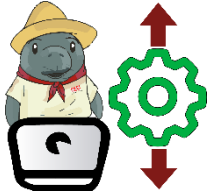
Por defecto, todos los principales navegadores muestran títulos de la tabla como negrita y centrado:

```
<table style="width:100%">
<tr>
  <th> Nombre (s) </th>
  <th> Apellidos </th>
  <th> Puntos </th>
</tr>
<tr>
  <td> Fernando </td>
  <td> Zapata </td>
  <td> 17 </td>
</tr>
</table>
```



Práctica 5. Tablas en HTML

Instrucciones: Escribe la estructura de una Página Web utilizando los elementos básicos, y agrega Tablas para dar una mejor distribución a la página web, no olvides si gustas agrega estilos CSS puedes utilizar SublimeText, o en la aplicación que tu docente te solicite, **si NO puedes realizarlo en la aplicación puedes escribir a mano el código.**



PRÁCTICA 5. "TABLAS EN HTML"



1. Crea una carpeta que se llame PRACTICA 5, recuerda aquí deberás guardar todos los archivos generados e incluso las imágenes y archivos de estilos.
2. Descarga la imagen de **COBACHITO.JPG** si es posible y guárdala en la carpeta creada, o puedes utilizar otra imagen de tu preferencia toma en cuenta el nombre de la imagen ya que debe ser idéntico en el código.
3. Apertura un archivo nuevo que le nombraremos **PRACTICAS_JAFP.HTML**, donde JAFP son las iniciales de tu nombre; en el software que estés utilizando para HTML guárdalo en la carpeta creada del mismo nombre, puede incluso utilizar bloc de notas. Vamos a ir creando una tabla con la forma del horario de clases para ello copia el siguiente código.

```
<!DOCTYPE html>
<html>
<head>
<title>TABLAS EN HTML</title>
</head>
<body>

<table cellpadding="10" cellspacing="10" border="3" bordercolor="blue">
<tr>
<td align="center"></td><td colspan="4"><h1
align="center">Horario de Clases</h1></td><td align="center">
</td>
<tr>
<td width="20%" align="center" bgcolor="gray">HORAS</td>
<td width="20%" align="center" bgcolor="gray">LUNES</td>
<td width="20%" align="center" bgcolor="gray">MARTES</td>
<td width="20%" align="center" bgcolor="gray">MIÉRCOLES</td>
<td width="20%" align="center" bgcolor="gray">JUEVES</td>
<td width="20%" align="center" bgcolor="gray">VIERNES</td>
</tr>
<tr>
<td width="20%" align="center"><font color="red">6:50-7:40</font></td>
<td width="20%" align="center"><font color="blue">INGLES</font></td>
<td width="20%" align="center"><font color="blue">HISTORIA</font></td>
<td width="20%" align="center"><font color="blue">GEOGRAFIA</font></td>
<td width="20%" align="center"><font color="blue">TALLER</font></td>
<td width="20%" align="center"><font color="blue">DIBUJO</font></td>
</tr>
</body>
</html>
```



4. Hasta este punto tendremos creado lo que se muestra en la siguiente imagen.

HORAS	LUNES	MARTES	MIERCOLES	JUEVES	VIERNES
6:50-7:40	INGLÉS	HISTORIA	GEOGRAFÍA	TALLER	DIBUJO

5. Muy bien hecho hasta este punto te hemos mostrado como crear la estructura. Ahora es momento en que TU continúes el código desde la última etiqueta de cierre `</TR>` que se muestra encerrada y con una mano en la página anterior, para completar el horario de clases. Puedes usar las mismas materias o puedes sustituirlas por las que tu llevas.

6. Tu producto final deberá quedar algo como se muestra en la figura.

HORAS	LUNES	MARTES	MIERCOLES	JUEVES	VIERNES
6:50-7:40	INGLÉS	HISTORIA	GEOGRAFÍA	TALLER	DIBUJO
7:40-8:30	INGLÉS	HISTORIA	GEOGRAFÍA	TALLER	DIBUJO
8:30-9:20	INGLÉS	HISTORIA	GEOGRAFÍA	TALLER	DIBUJO
RECESO					
9:50-10:40	INGLÉS	HISTORIA	GEOGRAFÍA	TALLER	DIBUJO
10:40-11:30	INGLÉS	HISTORIA	GEOGRAFÍA	TALLER	DIBUJO
11:30-12:20	INGLÉS	HISTORIA	GEOGRAFÍA	TALLER	DIBUJO
12:20-13:10	INGLÉS	HISTORIA	GEOGRAFÍA	TALLER	DIBUJO

7. Bien ahora CAMBIA y aplica los TUS COLORES Y ESTILOS PARA QUE TENGA UNA FORMA MAS LLAMATIVA MIENTRAS MAS CREATIVO MEJOR.

8. Recuerda ir guardando los cambios y todos los archivos generados tenerlos guardado en la carpeta nombrada PRACTICA 5.

9. Para ir finalizando ahora en la siguiente página copia o toma una captura de tu código completo y pégalo en el recuadro correspondiente.

10. También comparte una captura de pantalla o dibuja como quedo tu horario de clases con los colores y estilos que aplicaste.

CODIGO COMPLETO DEL HORARIO DE CLASES

```
<!DOCTYPE html>
<html>
<head>
<title>TABLAS EN HTML </title>
</head>
<body>
<table cellpadding="10" cellspacing="10" border="3" bordercolor="blue">
<tr>
<td align="center"></td><td colspan="4"><h1 align="center">Horario de Clases</h1></td><td align="center"></td>
<tr>
<td width="20%" align="center" bgcolor="gray">HORAS</td>
<td width="20%" align="center" bgcolor="gray">LUNES</td>
<td width="20%" align="center" bgcolor="gray">MARTES</td>
<td width="20%" align="center" bgcolor="gray">MIÉRCOLES</td>
<td width="20%" align="center" bgcolor="gray">JUEVES</td>
<td width="20%" align="center" bgcolor="gray">VIERNES</td>
</tr>
<tr>
<td width="20%" align="center"><font color="red">6:50-7:40</font></td>
<td width="20%" align="center"><font color="blue">INGLES</font></td>
<td width="20%" align="center"><font color="blue">HISTORIA</font></td>
<td width="20%" align="center"><font color="blue">GEOGRAFIA</font></td>
<td width="20%" align="center"><font color="blue">TALLER</font></td>
<td width="20%" align="center"><font color="blue">DIBUJO</font></td>
</tr>
<tr>
<td width="20%" align="center"><font color="red">7:40-8:30</font></td>
<td width="20%" align="center"><font color="blue">INGLES</font></td>
<td width="20%" align="center"><font color="blue">HISTORIA</font></td>
<td width="20%" align="center"><font color="blue">GEOGRAFIA</font></td>
<td width="20%" align="center"><font color="blue">TALLER</font></td>
<td width="20%" align="center"><font color="blue">DIBUJO</font></td>
</tr>
<tr>
<td width="20%" align="center"><font color="red">8:30-9:20</font></td>
<td width="20%" align="center"><font color="blue">INGLES</font></td>
<td width="20%" align="center"><font color="blue">HISTORIA</font></td>
<td width="20%" align="center"><font color="blue">GEOGRAFIA</font></td>
<td width="20%" align="center"><font color="blue">TALLER</font></td>
<td width="20%" align="center"><font color="blue">DIBUJO</font></td>
</tr>
<tr>
<td width="100%" align="center" bgcolor="blue" colspan="6"><font color="red">R E C E S O</font></td>
<tr>
<td width="20%" align="center"><font color="red">9:50-10:40</font></td>
<td width="20%" align="center"><font color="blue">INGLES</font></td>
<td width="20%" align="center"><font color="blue">HISTORIA</font></td>
<td width="20%" align="center"><font color="blue">GEOGRAFIA</font></td>
<td width="20%" align="center"><font color="blue">TALLER</font></td>
<td width="20%" align="center"><font color="blue">DIBUJO</font></td>
</tr>
<tr>
<td width="20%" align="center"><font color="red">10:40-11:30</font></td>
<td width="20%" align="center"><font color="blue">INGLES</font></td>
<td width="20%" align="center"><font color="blue">HISTORIA</font></td>
<td width="20%" align="center"><font color="blue">GEOGRAFIA</font></td>
<td width="20%" align="center"><font color="blue">TALLER</font></td>
<td width="20%" align="center"><font color="blue">DIBUJO</font></td>
</tr>
<tr>
<td width="20%" align="center"><font color="red">11:30-12:20</font></td>
<td width="20%" align="center"><font color="blue">INGLES</font></td>
<td width="20%" align="center"><font color="blue">HISTORIA</font></td>
<td width="20%" align="center"><font color="blue">GEOGRAFIA</font></td>
<td width="20%" align="center"><font color="blue">TALLER</font></td>
<td width="20%" align="center"><font color="blue">DIBUJO</font></td>
</tr>
<tr>
<td width="20%" align="center"><font color="red">12:20-13:10</font></td>
<td width="20%" align="center"><font color="blue">INGLES</font></td>
<td width="20%" align="center"><font color="blue">HISTORIA</font></td>
<td width="20%" align="center"><font color="blue">GEOGRAFIA</font></td>
<td width="20%" align="center"><font color="blue">TALLER</font></td>
<td width="20%" align="center"><font color="blue">DIBUJO</font></td>
</tr>
</body>
</html>
```





CAPTURA O DIBUJO DEL HORARIO DE CLASES

	Horario de Clases					
HORAS	LUNES	MARTES	MIÉRCOLES	JUEVES	VIERNES	
6:50-7:40	INGLES	HISTORIA	GEOGRAFIA	TALLER	DIBUJO	
7:40-8:30	INGLES	HISTORIA	GEOGRAFIA	TALLER	DIBUJO	
8:30-9:20	INGLES	HISTORIA	GEOGRAFIA	TALLER	DIBUJO	
RECESO						
9:50-10:40	INGLES	HISTORIA	GEOGRAFIA	TALLER	DIBUJO	
10:40-11:30	INGLES	HISTORIA	GEOGRAFIA	TALLER	DIBUJO	
11:30-12:20	INGLES	HISTORIA	GEOGRAFIA	TALLER	DIBUJO	
12:20-13:10	INGLES	HISTORIA	GEOGRAFIA	TALLER	DIBUJO	





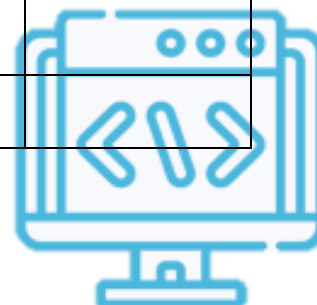
INSTRUMENTO DE EVALUACIÓN
LISTA DE COTEJO 8
Práctica 5: "Tablas en HTML"



DATOS GENERALES

Nombre(s) del alumno(s)	Matrícula(s)
Producto: Código y Captura o Dibujo Página Web	Fecha
Submódulo 2: Programación Web	Periodo: 2025 - 2026B
Nombre del docente	Firma del docente

CRITERIO	INDICADORES	VALOR OBTENIDO		PONDERACIÓN	CALIF	OBSERVACIONES Y/O SUGERENCIAS DE MEJORA
		SÍ	NO			
1	Se muestra el uso de tablas en la estructura HTML.			1		
2	Se observa el uso de colores dentro de las celdas.			2		
3	Se aprecia el uso de colores en los bordes de la celda.			1		
4	Se observa el uso de imágenes dentro de las tablas.			2		
5	Se muestra el uso de los atributos de centrado dentro de la tabla.			1		
6	Se observa el uso del atributo Colspan.			2		
7	Se muestra de forma estructura la tabla.			1		
CALIFICACIÓN						



Lectura No 8. Uso de Clases, Menús y Formularios en HTML

Instrucciones: Realiza el análisis de la siguiente información, posteriormente, realiza la Actividad 6.

Clases y Menús Parte 1 y 2



LECTURA 8. "USO DE CLASES, MENÚS Y FORMULARIOS EN HTML"



El atributo **class**

Dado que el número de etiquetas del HTML es reducido, es probable que en una página web se quiera dar un formato distinto a elementos con la misma etiqueta. Se puede sortear esta limitación utilizando clases, que se pueden asignar a cualquier elemento de una página web.

Las clases se asignan con el atributo **class**. El valor del atributo **class** (el nombre de la clase) sólo pueden contener caracteres ingleses, números, guiones y subrayados, aunque el primer carácter no puede ser ni un número ni un guion.

A continuación, observamos un ejemplo aplicando el atributo class en un archivo de estilos CSS.



RECUERDA ESTE ES EL ARCHIVO DE
TU PAGINA.HTML

```
<p>- Haga el favor de poner atención en la primera cláusula porque es muy importante. Dice que ... la parte contratante de la primera parte será considerada como la parte contratante de la primera parte. ¿Qué tal, está muy bien, eh?</p>
```

```
<p class="aviso">- No, eso no está bien. Quisiera volver a oírlo.</p>
```

```
p.aviso {  
  color: red;  
}
```

RECUERDA ESTE ES EL ARCHIVO
ESTILOS.CSS

- Haga el favor de poner atención en la primera cláusula porque es muy importante. Dice que ... la parte contratante de la primera parte será considerada como la parte contratante de la primera parte. ¿Qué tal, está muy bien, eh?

- No, eso no está bien. Quisiera volver a oírlo.

Y ASI ES COMO SE VE

Los navegadores distinguen los nombres de clases escritos en mayúsculas o minúsculas. En estos apuntes se recomienda utilizar únicamente minúsculas. Observemos aquí un **Ejemplo claro**:

```
<p class="aviso">- Haga el favor de poner atención en la primera cláusula porque es muy importante. Dice que ... la parte contratante de la primera parte será considerada como la parte contratante de la primera parte. ¿Qué tal, está muy bien, eh?</p>
```

```
<p class="Aviso">- No, eso no está bien. Quisiera volver a oírlo.</p>
```

```
p.aviso {  
  color: red;  
}
```

```
p.Aviso {  
  color: blue;  
}
```

OBSEVA EL USO DE MAYUSCULAS Y
MINUSCULAS EN EL NOMBRE DE LAS
CLASES

- Haga el favor de poner atención en la primera cláusula porque es muy importante. Dice que ... la parte contratante de la primera parte será considerada como la parte contratante de la primera parte. ¿Qué tal, está muy bien, eh?

- No, eso no está bien. Quisiera volver a oírlo.

Y ASI ES COMO SE VE



A un elemento con clase se le aplica tanto la regla de la clase como la regla de la etiqueta general (esta es una característica del mecanismo de cascada de las hojas de estilo). En el ejemplo siguiente, en la hoja de estilo se ha escrito una regla para párrafos `<p>` en general y otra regla para los párrafos de clase `aviso`. El elemento con clase hereda las propiedades de los elementos sin clase.

```
<p>- Haga el favor de poner atención en la primera cláusula porque es muy importante. Dice que ... la parte contratante de la primera parte será considerada como la parte contratante de la primera parte. ¿Qué tal, está muy bien, eh?</p>
```

```
<p class="aviso">- No, eso no está bien. Quisiera volver a oírlo.</p>
```

```
p {
  font-style: oblique;
}

p.aviso {
  color: red;
}
```

- Haga el favor de poner atención en la primera cláusula porque es muy importante. Dice que ... la parte contratante de la primera parte será considerada como la parte contratante de la primera parte. ¿Qué tal, está muy bien, eh?

- No, eso no está bien. Quisiera volver a oírlo.

OBSERVEMOS QUE EL PARRAFO EN COLOR ROJO HEREDO EL ATRIBUTO DE ESTILO DE LETRA ASIGNADO AL PRIMERO

Combinar varias clases simultáneamente

Un elemento puede tener asignado varias clases. Para asignar varias clases a un elemento, hay que escribir en el atributo `class` los nombres de todas las clases que se quieren aplicar, separados por espacios.

```
<p class="groucho">- Haga el favor de poner atención en la primera cláusula porque es muy importante. Dice que ... la parte contratante de la primera parte será considerada como la parte contratante de la primera parte. ¿Qué tal, está muy bien, eh?</p>
```

```
<p>- No, eso no está bien. Quisiera volver a oírlo.</p>
```

```
<p class="groucho aviso"> Dice que... la parte contratante de la primera parte será considerada como la parte contratante de la primera parte.</p>
```

```
p.groucho {
  font-style: oblique;
}

p.aviso {
  color: red;
}
```

- Haga el favor de poner atención en la primera cláusula porque es muy importante. Dice que ... la parte contratante de la primera parte será considerada como la parte contratante de la primera parte. ¿Qué tal, está muy bien, eh?

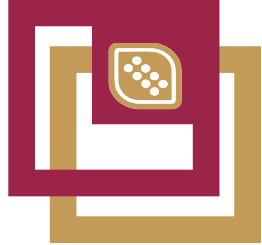
- No, eso no está bien. Quisiera volver a oírlo.

- Dice que... la parte contratante de la primera parte será considerada como la parte contratante de la primera parte.



El orden en que se escriban los nombres de las clases en el atributo `class` no es importante, aunque sí que es importante el orden de las reglas en la hoja de estilo. Como muestran los dos ejemplos siguientes, si la misma propiedad está definida en varias clases y a un elemento tiene asignado varias clases, se aplica la definición de la clase que aparece después en la hoja de estilo (esta es una característica del mecanismo de cascada de las hojas de estilo).





```
<p class="borde-1 borde-2">Este párrafo tiene como clase "borde-1
borde-2".</p>

<p class="borde-2 borde-1">Este párrafo tiene como clase "borde-2
borde-1".</p>
```

```
p.borde-1 {
  border: red 3px solid;
}

p.borde-2 {
  border: blue 3px dotted;
}
```

Este párrafo tiene como clase "borde-1 borde-2".

Este párrafo tiene como clase "borde-2 borde-1".

```
<p class="borde-1 borde-2">Este párrafo tiene como clase "borde-1
borde-2".</p>

<p class="borde-2 borde-1">Este párrafo tiene como clase "borde-2
borde-1".</p>
```

```
p.borde-2 {
  border: blue 3px dotted;
}

p.borde-1 {
  border: red 3px solid;
}
```

Este párrafo tiene como clase "borde-1 borde-2".

Este párrafo tiene como clase "borde-2 borde-1".

Se puede escribir un selector que se aplique únicamente a los elementos que tienen definidas varias clases.

```
<p class="borde-1">Este párrafo tiene como clase "borde-1".</p>

<p class="borde-2">Este párrafo tiene como clase "borde-2".</p>

<p class="borde-1 borde-2">Este párrafo tiene como clase "borde-1
borde-2".</p>
```

```
p.borde-1.borde-2 {
  border: red 3px dotted;
}

p.borde-1 {
  border: red 3px solid;
}

p.borde-2 {
  border: blue 3px dotted;
}
```

Este párrafo tiene como clase "borde-1".

Este párrafo tiene como clase "borde-2".

Este párrafo tiene como clase "borde-1 borde-2".

Asignar la misma clase a distintos elementos

Se pueden asignar la misma clase a elementos distintos:

- Definiendo la clase para cada elemento, en cuyo caso cada elemento puede tener propiedades distintas, aunque el nombre de la clase sea el mismo.

```
<p class="borde-1">Este párrafo tiene como clase "borde-1".</p>

<p>Este párrafo tiene esta <span class="borde-1">palabra</span> con
clase "borde-1".</p>
```

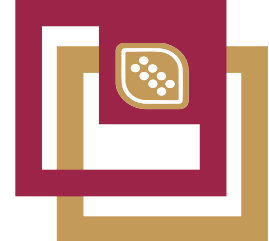
```
p.borde-1 {
  border: red 3px solid;
}

span.borde-1 {
  border: red 3px dotted;
}
```

Este párrafo tiene como clase "borde-1".

Este párrafo tiene esta palabra con clase "borde-1".





- Definiendo una clase universal, en cuyo caso todos los elementos tienen las mismas propiedades.

```
<p class="borde-1">Este párrafo tiene como clase "borde-1".</p>
<p>Este párrafo tiene esta <span class="borde-1">palabra</span> con
clase "borde-1".</p>
```

```
.borde-1 {
  border: red 3px solid;
}

/* La clase universal también se puede escribir así:
*.borde-1 {
  border: red 3px solid;
}

*/
```

Este párrafo tiene como clase "borde-1".

Este párrafo tiene esta palabra con clase "borde-1".

Número de clases

En principio, se suelen definir tantas clases como tipos de elementos distintos queremos mostrar, pero se podría definir una clase menos aprovechando que también podemos definir en la hoja de estilo la etiqueta sin clase. Los dos ejemplos siguientes darían el mismo resultado.

En este primer ejemplo, para distinguir las intervenciones de los dos personajes en un diálogo, se da clase a todos los párrafos. Los nombres de la clase son los nombres de los personajes.



```
<p class="groucho">- Haga el favor de poner atención en la primera
cláusula porque es muy importante. Dice que ...<span class="aviso">la
parte contratante de la primera parte será considerada como la parte
contratante de la primera parte</span>. ¿Qué tal, está muy bien, eh?
</p>
```

```
<p class="chico">- No, eso no está bien. Quisiera volver a oírlo.</p>
```

```
<p class="groucho">- Dice que... la parte contratante de la primera parte
será considerada como la parte contratante de la primera parte.</p>
```

```
<p class="chico">- Esta vez creo que suena mejor.</p>
```

```
<p class="groucho">- Si quiere se lo leo otra vez.</p>
```

```
<p class="chico">- Tan solo la primera parte.</p>
```

```
<p class="groucho">- ¿Sobre la parte contratante de la primera parte?
</p>
```

```
<p class="chico">- No, solo la parte de la parte contratante de la
primera parte.</p>
```

```
p.groucho {
  color: red;
}

p.chico {
  color: blue;
}
```

- Haga el favor de poner atención en la primera cláusula porque es muy importante. Dice que ... la parte contratante de la primera parte será considerada como la parte contratante de la primera parte. ¿Qué tal, está muy bien, eh?

- No, eso no está bien. Quisiera volver a oírlo.

- Dice que... la parte contratante de la primera parte será considerada como la parte contratante de la primera parte.

- Esta vez creo que suena mejor.

- Si quiere se lo leo otra vez.

- Tan solo la primera parte.

- ¿Sobre la parte contratante de la primera parte?

- No, solo la parte de la parte contratante de la primera parte.





En este segundo ejemplo, para distinguir las intervenciones de los dos personajes en un diálogo, sólo se da clase a los párrafos de uno de ellos. El nombre de la clase es el nombre del personaje. La propiedad que antes se establecía en la clase del segundo personaje, esta vez se establece para la etiqueta general (sin clase).

```
<p class="groucho">- Haga el favor de poner atención en la primera
cláusula porque es muy importante. Dice que ...<span class="aviso">la
parte contratante de la primera parte será considerada como la parte
contratante de la primera parte</span>. ¿Qué tal, está muy bien, eh?
</p>

<p>- No, eso no está bien. Quisiera volver a oírlo.</p>

<p class="groucho">- Dice que... la parte contratante de la primera parte
será considerada como la parte contratante de la primera parte.</p>

<p>- Esta vez creo que suena mejor.</p>

<p class="groucho">- Si quiere se lo leo otra vez.</p>

<p>- Tan solo la primera parte.</p>

<p class="groucho">- ¿Sobre la parte contratante de la primera parte?
</p>

<p>- No, solo la parte de la parte contratante de la primera parte.</p>
```

```
p.groucho {
  color: red;
}

p {
  color: blue;
}
```

- Haga el favor de poner atención en la primera cláusula porque es muy importante. Dice que ... la parte contratante de la primera parte será considerada como la parte contratante de la primera parte. ¿Qué tal, está muy bien, eh?

- No, eso no está bien. Quisiera volver a oírlo.

- Dice que... la parte contratante de la primera parte será considerada como la parte contratante de la primera parte.

- Esta vez creo que suena mejor.

- Si quiere se lo leo otra vez.

- Tan solo la primera parte.

- ¿Sobre la parte contratante de la primera parte?

- No, solo la parte de la parte contratante de la primera parte.

La ventaja del segundo método es que hay que asignar menos clases. El inconveniente es que las propiedades de la clase no definida hay que asignarla a la etiqueta general y se aplican a todos los elementos. En general, suele ser más conveniente definir tantas clases como elementos distintos, salvo que una de las clases se tenga que aplicar a casi todos los elementos (más que el resto de las clases juntas, por ejemplo) en cuyo caso se puede omitir esta clase y dejarla como etiqueta general. Pero este es un aspecto para valorar en cada página en concreto.

¿Cómo crear un Menú en HTML?

Por lo general, para crear un menú utilizamos una lista desordenada. Se puede utilizar **div**, **span**, **p** u otras etiquetas, pero la lista desordenada le da más flexibilidad.

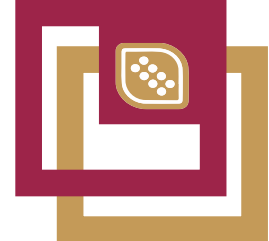
En primer lugar, abra la página HTML por medio de su editor.

La lista desordenada es creada con ayuda de las etiquetas `<ul>`. Abra la etiqueta `ul` y cierre ella:

```
<ul> </ul>
```

Luego hay que añadir los elementos de la lista a la lista. Cada botón del menú será un elemento de la lista. Apliquemos las etiquetas de `<li>` y `</li>`





Vamos a crear un menú con cinco elementos:

```
1 <ul>
2   <li> Inicio </li>
3   <li> Quines somos </li>
4   <li> Servicios </li>
5   <li> Productos </li>
6   <li> Contacto </li>
7 </ul>
```

Lo último que necesitamos hacer es añadir enlaces a los botones de nuestro menú. En HTML enlaces son creados mediante la etiqueta <a>.

```
1 <ul>
2   <li> <a href="#"> Inicio </a> </li>
3   <li> <a href="#"> Quines Somos </a> </li>
4   <li> <a href="#"> Servicios </a> </li>
5   <li> <a href="#"> Productos </a> </li>
6   <li> <a href="#"> Contacto </a> </li>
7 </ul>
```

Como puede notar cada etiqueta <a> tiene un parámetro href. Parámetro href define la dirección URL o el fichero vinculado con un elemento del menú. **Reemplace el símbolo # con el nombre de fichero o la dirección URL para activar los enlaces. Esto se refiere si existe alguna página o parte de la página que esté vinculada a ese botón del menú.**



¿Cómo crear un Menú Vertical y Horizontal en HTML/CSS?

Los siguientes ejemplos te ayudarán a crear un menú vertical y horizontal en HTML usando estilos CSS. Antes de continuar asegúrate de que sabes las etiquetas HTML de la lista desordenada.

Ok, en primer lugar, hay que crear un menú HTML usando la lista desordenada:

Luego hay que crear un nuevo fichero CSS y adjuntarlo a la página HTML:

```
1 <link href="style.css" rel="stylesheet" type="text/css" />
```

```
1 <ul>
2   <li> <a href="#"> Inicio </a> </li>
3   <li> <a href="#"> Quienes Somos </a> </li>
4   <li> <a href="#"> Servicios </a> </li>
5   <li> <a href="#"> Productos </a> </li>
6   <li> <a href="#"> Contacto </a> </li>
7 </ul>
```

Asegúrate de poner el código CSS en la etiqueta <head> de tu página html.

También puede utilizar los estilos en línea (online).

```
1 <style type="text/css">
2   ...aquí debes colocar el CSS...
3 </style>
```





Como resultado debe obtener el siguiente código HTML:

```
1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
2 "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
3 <html xmlns="http://www.w3.org/1999/xhtml">
4
5 <head>
6   <link href="style.css" rel="stylesheet" type="text/css" />
7 </head>
8
9 <body>
10
11 <ul>
12   <li> <a href="#"> Inicio </a> </li>
13   <li> <a href="#"> Quienes Somos </a> </li>
14   <li> <a href="#"> Servicios </a> </li>
15   <li> <a href="#"> Productos </a> </li>
16   <li> <a href="#"> Contacto </a> </li>
17 </ul>
18
19 </body>
</html>
```



La lista desordenada tiene sus propios estilos, así que sin los cambios adicionales obtiene un menú vertical. Para obtener el menú horizontal es necesario hacer algunos cambios en HTML y CSS.

En primer lugar, hay que añadir una nueva clase a la lista, reemplace `<ul>` con `<ul class="horizontal">`

Ahora en el fichero CSS vamos a crear el menú horizontal. La lista desordenada tiene valores de margen y relleno (margin and padding) asignados por defecto. Tenemos que limpiar ellos:

Luego haga los cambios para que sus elementos de la lista se muestren de manera horizontal:

```
1 ul.horizontal li {
2   display:block;
3   float:left;
4   padding:0 10px;
5 }
```

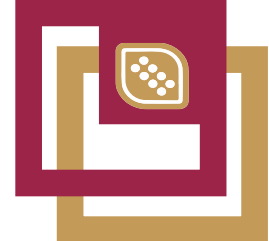
Hemos añadido el valor de relleno horizontal (padding) a los elementos de la lista para que luego no se peguen unas a otras. Ahora su menú está listo, asigne enlaces, añada un poco de colores y fondo y ya.

```
1 ul.horizontal {
2   margin:0;
3   padding:0;
4 }
```



AQUÍ PUEDES VER UN
VIDEOTUTORIAL PARA
ENTENDER MEJOR LOS MENÚS





FORMULARIOS

Un formulario es un conjunto de controles (botones, cajas de texto, casillas de verificación, botones radio, etc) que permiten al usuario introducir datos y enviarlos al servidor web para su procesamiento.

```
<form action="ejemplo.php" method="get">
  <p>Nombre: <input type="text" name="nombre" size="40"></p>
  <p>Año de nacimiento: <input type="number" name="nacido" min="1900">
</p>
  <p>Sexo:
    <input type="radio" name="hm" value="h"> Hombre
    <input type="radio" name="hm" value="m"> Mujer
  </p>
  <p>
    <input type="submit" value="Enviar">
    <input type="reset" value="Borrar">
  </p>
</form>
```

Nombre:

Año de nacimiento:

Sexo: ☐ Hombre ☐ Mujer

La etiqueta que delimita un formulario es la etiqueta `<form> ...</form>`. Los atributos más importantes de la etiqueta `<form>` son:

- **action**: contiene el nombre del agente que procesará los datos remitidos al servidor (por ejemplo, un script de PHP)
- **method**: define la manera de enviar los datos al servidor. Los valores posibles son:
 - **get**: los valores enviados se añaden a la dirección indicada en el atributo **action**.
 - **post**: los valores se envían de forma separada.



Si el atributo **method** no está establecido, el formulario se comporta como si el valor fuera **get**.

La etiqueta `<form>` es un elemento de bloque. En su interior puede haber cualquier elemento típico de una página web (párrafos, imágenes, divisiones, listas, tablas, etc.), además de las etiquetas que crean los controles.

Las etiquetas que crean los controles en los formularios son `<input>`, `<button>`, `<select>`, `<optgroup>`, `<option>` y `<textarea>`. Además, se pueden estructurar los controles con las etiquetas `<fieldset>` y `<legend>`. Por último, la etiqueta `<label>` permite mejorar la accesibilidad de los controles. El navegador envía únicamente los datos de los controles contenidos en el formulario. En una misma página puede haber varios formularios que envíen datos al mismo o a diferentes agentes.





Atributos comunes de los controles.

El atributo **type**

El atributo **type** de la etiqueta `<input>` indica el tipo de control de que se trata.

- En HTML 4.01, los tipos de `<input>` eran: botón para enviar submit, botón para reiniciar reset, texto text, contraseña password, archivo file, imagen image y oculto hidden.
- En HTML 5 se añadieron los tipos de `<input>`: botón **button**, número **number**, búsqueda **search**, teléfono **tel**, dirección web **url**, dirección de correo **email**, fecha **date**, hora **time**, color **color** y rango **range**.
- En HTML 5.1 se añadieron los tipos de `<input>`: mes **month**, semana **week** y fecha y hora local **datetime-local**.

El atributo **name**

El atributo **name** identifica al control.

Todos los controles de un formulario deben tener el atributo **name** porque, en general, sólo se envían los controles que lo tienen (aunque hay excepciones, como los controles de tipo imagen o los botones, que no los necesitan). En general, los atributos **name** no se deben repetir, porque no se recibiría toda la información al enviarla. La única excepción es el botón radio, como se explica en el apartado correspondiente.

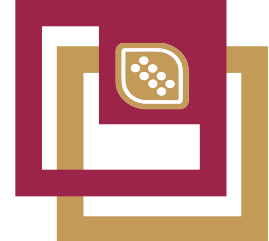
El atributo **value**

El atributo **value** permite establecer el valor inicial de un control, aunque cada control lo utiliza de una forma ligeramente distinta y algunos controles no lo admiten (área de texto, archivo o imagen). En algunos casos (botones) el valor se muestra al usuario y el usuario no puede modificarlo.

En otros casos (cajas de texto, contraseña) el valor se muestra en la página y el usuario puede modificarlo. En otros casos (casillas de verificación, botones radio, opciones de menú, oculto) el valor no se muestra en la página y el usuario no puede modificarlo.

En algunos casos (botones, oculto, botones radio) el atributo **value** es necesario. En otros casos (casilla de verificación, opciones de menú) es conveniente.





En otros casos (cajas de texto, contraseña, imagen) puede omitirse. En el caso del selector de archivo, los navegadores no lo admiten por motivos de seguridad.

El atributo **required**

El atributo **required** permite indicar qué controles es obligatorio rellenar para enviar el formulario.

<pre><p>Nombre: <input type="text" name="nombre" required></p> <p>Sexo: <input type="radio" name="hm" value="h" required> Hombre <input type="radio" name="hm" value="m" required> Mujer </p></pre>	Nombre: Sexo: ☐ Hombre ☐ Mujer Enviar Borrar
--	--

Si alguna de las opciones de un botón radio tiene establecido el atributo **required**, el control es obligatorio.

<pre><p>Sexo: <input type="radio" name="hm" value="h" required> Hombre <input type="radio" name="hm" value="m"> Mujer </p> <p>Edad: <input type="radio" name="edad" value="menor"> Menor <input type="radio" name="edad" value="adulto" required> Adulto </p></pre>	Sexo: ☐ Hombre ☐ Mujer Edad: ☐ Menor ☐ Adulto Enviar Borrar
--	--



El atributo **placeholder**

El atributo **placeholder**, permite mostrar en los controles de texto un texto que desaparece al escribir en el control.

<pre><p>Apellidos: <input type="text" name="apellidos" size="40" placeholder="Escriba sus dos apellidos"> </p></pre>	Apellidos: Escriba sus dos apellidos
--	--

El atributo **size**

El atributo **size** permite establecer la longitud de los controles de texto (text, password, search, etc.). Por omisión, las cajas suelen tener 20 caracteres de longitud.

<pre><p>Nombre: <input type="text" name="nombre"></p> <p>Apellidos: <input type="text" name="apellidos" size="40"></p> <p>Nacionalidad: <input type="text" name="nacionalidad" size="20"></p></pre>	Nombre: Apellidos: Nacionalidad:
---	--





Los atributos **maxlength** y **minlength**

Los atributos **maxlength** y **minlength** permiten establecer la longitud máxima y mínima, respectivamente, que puede escribir el usuario en un control de texto (text, password, search, etc.).

```
<p>Usuario: <input type="text" name="usuario" maxlength="6"></p>
<p>Contraseña: <input type="password" name="password" minlength="6"></p>
```

Usuario:

Contraseña:

El atributo **autofocus**

El atributo **autofocus** permite indicar el control que debe tener el foco al cargarse el formulario. Este atributo facilita la accesibilidad del formulario pues permite que el usuario pueda empezar a rellenar el formulario sin necesidad de hacer clic en el elemento. Como el navegador desplaza automáticamente la página hasta la posición del elemento, se recomienda que el elemento con atributo autofocus se encuentren al principio de la página, para que el desplazamiento no confunda al usuario.

Nota: En estos apuntes (lecciones y ejercicios) no se utiliza este atributo porque como en las páginas se incluyen varios formularios en distintas posiciones de la página, el desplazamiento automático a un elemento con atributo autofocus puede confundir al lector.



El atributo **disabled**

El atributo **disabled** permite deshabilitar el control. Una vez deshabilitado, el control ni siquiera puede coger el foco.

```
<input type="submit" value="Enviar" disabled>
```

```
<input type="text" name="texto" disabled>
```

El atributo **readonly**

El atributo **readonly** hace que el control no sea modificable, aunque el control puede coger el foco.

```
<input type="text" name="texto" value="¡A que no me cambias!" readonly>
<input type="text" name="texto" placeholder="¡A que no me cambias!"
readonly>
```



El atributo **tabindex**

El atributo `tabindex` permite controlar el orden en que el foco pasa de un elemento a otro mediante el tabulador (Tab para avanzar y Shift+Tab para retroceder). Los valores de `tabindex` pueden ser números naturales (incluido el cero), no necesariamente consecutivos. Si no está presente, los controles se visitan en el orden en que aparecen en el texto. Si está presente, los controles se visitan de menor a mayor.

```
<input type="text" name="texto1" placeholder="Texto 1">
<input type="text" name="texto2" placeholder="Texto 2">
<input type="text" name="texto3" placeholder="Texto 3">
<input type="text" name="texto4" placeholder="Texto 4">
```

Texto 1 Texto 2 Texto 3
Texto 4

```
<input type="text" name="texto1" placeholder="Texto 1" tabindex="5">
<input type="text" name="texto2" placeholder="Texto 2" tabindex="8">
<input type="text" name="texto3" placeholder="Texto 3" tabindex="7">
<input type="text" name="texto4" placeholder="Texto 4" tabindex="6">
```

Texto 1 Texto 2 Texto 3
Texto 4

**"SI DESEAS CONOCER MAS ATRIBUTOS DE LOS
FORMULARIOS DA CLIC EN ESTE CODIGO"**



BIBLIOGRAFÍA

- Marco, B. S. (Octubre de 2016). *Páginas Web HTML y hojas estilos CSS*. Obtenido de <https://www.mclibre.org/consultar/htmlcss/html/html-clases.html>
- Marco, B. S. (Junio de 2020). *Páginas web HTML y hojas de estilo CSS*. Obtenido de <https://www.mclibre.org/consultar/htmlcss/html/html-formularios.html#>
- Theme Technologies LLC. Operated by Jetimpex Inc. (Marzo de 2020). *MonsterHelp*. Obtenido de <https://www.templatemonster.com/help/es/htmlcss-how-to-create-vertical-and-horizontal-menu.html>



Actividad 6. Clases, Menús y Formularios



Instrucciones: Después de haber analizado la Lectura 8. “Uso de Clases, Menús y Formulario en HTML”. Completa la siguiente tabla con el concepto o nombre de etiqueta o atributo faltante.



Definición	Etiqueta o Atributo
1. Sólo pueden contener caracteres ingleses, números, guiones y subrayados, aunque el primer carácter no puede ser ni un número ni un guion.	
2.	<i>required</i>
3. Es un conjunto de controles (botones, cajas de texto, casillas de verificación, botones radio, etc) que permiten al usuario introducir datos y enviarlos al servidor web para su procesamiento.	
4.	<code></code>
5. Sirve para añadir los elementos de la lista a la lista.	
6. Define la dirección URL o el fichero vinculado con un elemento del menú	
7.	<i>method</i>
8. Las etiquetas que crean los controles en los formularios son:	
9.	<code><label></code>
10. Hace que el control de un formulario no sea modificable, aunque el control puede coger el foco.	



Parte 2.



HTML



Instrucciones: Escribe el código para la siguiente captura de pantalla que se te presenta.

Nombre:
Año de nacimiento:
Dirección:
Sexo: ☐ Hombre ☐ Mujer
Semestre: ☐ 1er ☐ 2do ☐ 3er ☐ 4to ☐ 5to ☐ 6to

```

<!DOCTYPE HTML>
<HTML>
<head>
  <meta charset="utf-8">
  <title> MI EMPRESA </title>
  <link rel="stylesheet" type="text/css" href="estilos.css">
</head>
<body>
<form action="ejemplo.php" method="get">
  <p> Nombre: <input type="text" name="nombre" size="40"> </p>
  <p> Año de nacimiento: <input type="number" name="nacido"
min="1900"> </p>
  <p> Dirección: <input type="text" name="nombre" size="80"> </p>
  <p> Sexo:
    <input type="radio" name="hm" value="h"> Hombre
    <input type="radio" name="hm" value="m"> Mujer
  </p>
  <p> <font color="blue"> Semestre:
    <input type="radio" name="hm" value="1"> 1er
    <input type="radio" name="hm" value="2"> 2do
    <input type="radio" name="hm" value="3"> 3er
    <input type="radio" name="hm" value="4"> 4to
    <input type="radio" name="hm" value="5"> 5to
    <input type="radio" name="hm" value="6"> 6to
  </p> </font>
  <p>
    <input type="submit" value="Enviar">
    <input type="reset" value="Borrar">
  </p>
</form>
</BODY>
</HTML>

```





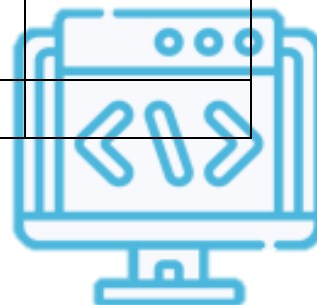
INSTRUMENTO DE EVALUACIÓN
LISTA DE COTEJO 9
Actividad 6: "Clases, Menús y Formularios Parte 1"



DATOS GENERALES

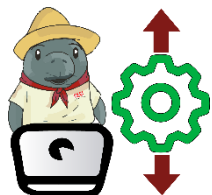
Nombre(s) del alumno(s)	Matrícula(s)
Producto: Código y Captura o Dibujo Página Web	Fecha
Submódulo 2: Programación Web	Periodo: 2025 - 2026B
Nombre del docente	Firma del docente

CRITERIO	INDICADORES	VALOR OBTENIDO		PONDERACIÓN	CALIF	OBSERVACIONES Y/O SUGERENCIAS DE MEJORA
		SÍ	NO			
1	Se observa el uso de la etiqueta <form>			1		
2	Aplica espacio proporcional a los campos de nombre y dirección.			1		
3	Se muestra el uso de los botones de Enviar y Borrar			2		
4	Se muestra el uso de los atributos type, name, value			2		
5	Se observa el uso del color azul dentro de las opciones.			1		
6	Se muestra la estructura de la página web en HTML			1		
CALIFICACIÓN						



Práctica 6. Clases, Menús y Formularios

Instrucciones: Escribe la estructura de una Página Web utilizando los elementos básicos y ahora te ayudaremos a generar un menú que puedes utilizar para solucionar el problema de la Situación Didáctica. No olvides si gustas agrega estilos CSS puedes utilizar SublimeText, o en la aplicación que tu docente te solicite, **si NO puedes realizarlo en la aplicación puedes escribir a mano el código.**



PRÁCTICA 6. "USO DE CLASES, MENÚS Y FORMULARIOS"



1. Abre Sublime Text, o la aplicación donde realizaras la codificación en HTML.
2. Empieza a escribir el siguiente código:

```

1  <html>
2      <head>
3          <title>Menu Desplegable</title>
4          <style type="text/css">
5
6              * {
7                  margin:0px;
8                  padding:0px;
9              }
10
11             #header {
12                 margin:auto;
13                 width:500px;
14                 font-family:Arial, Helvetica, sans-serif;
15             }
16
17             ul, ol {
18                 list-style:none;
19             }
20
21             .nav > li {
22                 float:left;
23             }
24
25             .nav li a {
26                 background-color:#000;
27                 color:#fff;
28                 text-decoration:none;
29                 padding:10px 12px;
30                 display:block;
31             }
32
33             .nav li a:hover {
34                 background-color:#434343;
35             }

```



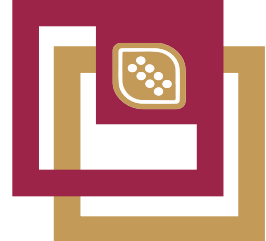


```

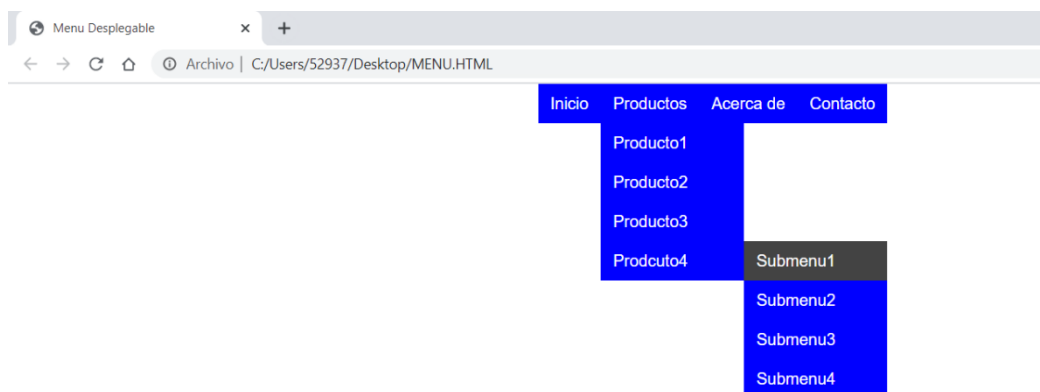
36
37     .nav li ul {
38         display:none;
39         position:absolute;
40         min-width:140px;
41     }
42
43     .nav li:hover > ul {
44         display:block;
45     }
46
47     .nav li ul li {
48         position:relative;
49     }
50
51     .nav li ul li ul {
52         right:-140px;
53         top:0px;
54     }
55
56 </style>
57 </head>
58 <body>
59     <div id="header">
60         <ul class="nav">
61             <li><a href="">Inicio</a></li>
62             <li><a href="">Servicios</a>
63                 <ul>
64                     <li><a href="">Submenu1</a></li>
65                     <li><a href="">Submenu2</a></li>
66                     <li><a href="">Submenu3</a></li>
67                     <li><a href="">Submenu4</a>
68                         <ul>
69                             <li><a href="">Submenu1</a></li>
70                             <li><a href="">Submenu2</a></li>
71                             <li><a href="">Submenu3</a></li>
72                             <li><a href="">Submenu4</a></li>
73                         </ul>
74                     </li>
75                 </ul>
76             </li>
77             <li><a href="">Acerca de</a>
78                 <ul>
79                     <li><a href="">Submenu1</a></li>
80                     <li><a href="">Submenu2</a></li>
81                     <li><a href="">Submenu3</a></li>
82                     <li><a href="">Submenu4</a></li>
83                 </ul>
84             </li>
85             <li><a href="">Contacto</a></li>
86         </ul>
87     </div>
88 </body>
89 </html>

```





3. Guarda tu archivo como MenuDesplegable.txt o la extensión de tu aplicación en la que programas.
4. Ahora **Guarda como** nuevamente MenuDesplegable.HTML si lo realizaste en el bloc de notas para poder ver el resultado, si lo realizaste en otra aplicación que permite previsualizar el código encontraras el mismo resultado.
5. Muy bien ya codificaste tu Menú debes obtener una pantalla como la siguiente.



6. Recuerda que puedes cambiar los nombres si te das cuenta se sustituyó la palabra Submenú por la de Producto de igual forma puedes cambiar los estilos o colores que desees utilizar.
7. Guarda los cambios que realizaste en la programación y actualiza el sitio HTML para que se reflejen los cambios.
8. Entrega tu práctica a tu docente para su revisión



Observa el siguiente video en el código QR para entender mejor cada sección de la práctica.



IMPORTANTE:

EMPIEZA A DISEÑAR LA PAGINA WEB DE LA SITUACIÓN DIDACTICA APLICANDO LO APRENDIDO PUEDES USAR ESTE MENU PARA AGREGARLO O TOMARLO COMO BASE, VE AVANZANDO EN EL DISEÑO Y CONSULTA CON TU DOCENTE SOBRE LOS AVANCES O DUDAS QUE TENGAS.





INSTRUMENTO DE EVALUACIÓN
LISTA DE COTEJO 10
Práctica 6: "Clases, Menús y Formularios"



DATOS GENERALES						
Nombre(s) del alumno(s)				Matrícula(s)		
Producto: Código y Captura o Dibujo Página Web				Fecha		
Submódulo 2: Programación Web				Periodo: 2025 - 2026B		
Nombre del docente				Firma del docente		
CRITERIO	INDICADORES	VALOR OBTENIDO		PONDERACIÓN	CALIF	OBSERVACIONES Y/O SUGERENCIAS DE MEJORA
		SÍ	NO			
1	Se observa la estructura de la página web en HTML.			1		
2	Se aplica los estilos para el menú.			2		
3	Cambia el color del fondo del menú en base a sus necesidades.			2		
4	Aplica el uso de las listas de manera correcta y en forma de cascada.			2		
5	Genera los submenús dentro de los otros menús.			2		
6	Entiende la forma en que los estilos se aplican a las listas.			1		
CALIFICACIÓN						

CONCLUYE:

¿Para qué se utilizan los menús y qué importancia tiene el uso de las clases en las páginas web?

¿Para qué sirven los formularios dentro de una página web?





¿Qué es JavaScript?



<https://shre.ink/JScrip>

Lectura No 9. ¿Qué es JavaScript?



Instrucciones: Realiza el análisis de la siguiente información, posteriormente, realiza la Actividad 7.



LECTURA 9. "QUE ES JAVASCRIPT"

¿Qué es JavaScript?



JavaScript es uno de los más potentes e importantes lenguajes de programación en la actualidad, por tres enfoques claros: es útil, práctico y está disponible en cualquier navegador web. JavaScript fue creado por Brendan Eich y vio la luz en el año 1995 con el nombre de LiveScript que luego fue nombrado JavaScript, nace como un lenguaje sencillo destinado a añadir algunas características interactivas a las páginas web. Sin embargo, hoy en día ha crecido de manera acelerada y es el lenguaje de programación que se utiliza en casi todos los sitios web en el mundo.





El poder de JavaScript está disponible principalmente en el lado frontend, agregando mayor interactividad a la web, también puedes usar las librerías y framework como: jquery, angular, backbone, react y demás escritas sobre JavaScript y que ayudan a crear una mejor experiencia de usuario en nuestros sitios web.

Las características de JavaScript que lo hacen uno de los más populares en la actualidad son:

1. Es liviano.
2. Multiplataforma, ya que se puede utilizar en Windows, Linux o Mac y en el navegador de tu preferencia.
3. Es imperativo y estructurado, mediante un conjunto de instrucciones indica a la computadora que tarea debe realizar.
4. Prototipado, debido a que usa prototipos en vez de clases para el uso de herencia.
5. Orientado a objetos y eventos.
6. Es interpretado, no se compila para poder ejecutarse.



Estas son las características que hacen de JavaScript un lenguaje que permite desarrollar aplicaciones gigantes y potentes, como lo es: google doc, facebook, twitter e incluso capaz de ejecutarse en el servidor como un servidor web muy rápido.

JavaScript es un lenguaje de programación o de secuencias de comandos que permite implementar funciones complejas en páginas web, cuando en una página web se muestran actualizaciones de contenido, mapas interactivos, animación de gráficos 2D y 3D, desplazamiento de máquinas reproductoras de video, etc., podemos decir que es probable que JavaScript está involucrado. Es la tercera capa de las tecnologías web estándar, dos de las cuales HTML y CSS se ha revisado en apartados anteriores:





- ❖ HTML es el lenguaje de marcado que se usa para estructurar y dar significado al contenido web, esto puede ser párrafos, encabezados, tablas de datos o insertando imágenes y videos en la página.
- ❖ CSS es un lenguaje de reglas de estilo que se usa para aplicar estilo al contenido HTML, para dar color al fondo y tipos de letra, así como distribuir el contenido en múltiples columnas.
- ❖ JavaScript es un lenguaje de secuencias de comandos que permite crear contenido de actualización dinámica, controlar multimedia, animar imágenes y prácticamente todo lo demás. Es sorprendente lo que se puede lograr con unas pocas líneas de código JavaScript.

Las tres capas se superponen muy bien. Se tiene una etiqueta de texto simple. Usamos HTML para darle estructura y propósito.

```
1 | <p>Player 1: Chris</p>
```

Se mostraría lo siguiente:

Player 1: Chris



Podemos agregar algo de CSS a la mezcla para que se vea bien:

```
1 | p {
2 |   font-family: 'helvetica neue', helvetica, sans-serif;
3 |   letter-spacing: 1px;
4 |   text-transform: uppercase;
5 |   text-align: center;
6 |   border: 2px solid rgba(0,0,200,0.6);
7 |   background: rgba(0,0,200,0.3);
8 |   color: rgba(0,0,200,0.6);
9 |   box-shadow: 1px 1px 2px rgba(0,0,200,0.4);
10 |  border-radius: 10px;
11 |  padding: 3px 10px;
12 |  display: inline-block;
13 |  cursor: pointer;
14 | }
```

PLAYER 1: CHRIS

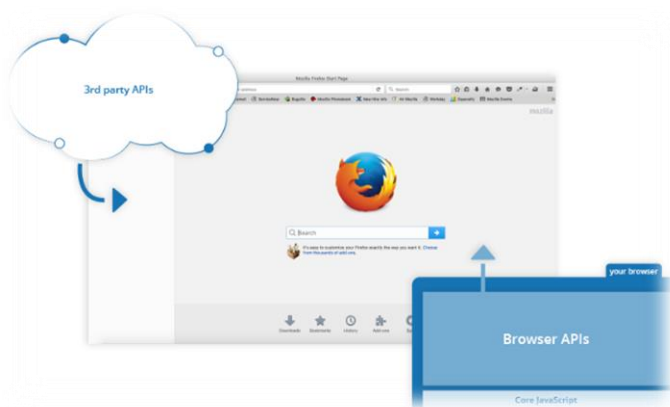


Finalmente se puede agregar algo de JavaScript para implementar un comportamiento dinámico:

```
1 const para = document.querySelector('p');
2
3 para.addEventListener('click', updateName);
4
5 function updateName() {
6   let name = prompt('Enter a new name');
7   para.textContent = 'Player 1: ' + name;
8 }
```

El núcleo del lenguaje JavaScript del lado del cliente consta de algunas características de programación comunes que permiten hacer cosas como:

- ❖ Almacenar valores útiles dentro de variables. En el ejemplo anterior, se pide que ingreses un nuevo nombre y luego almacenamos ese nombre en una variable llamada name.
- ❖ Operaciones sobre fragmentos de texto conocidas como “cadenas” (strings) en programación. En el ejemplo anterior se toma la cadena “Player 1” y se une a la variable name para crear la etiqueta de texto completa. Ejemplo: “Player 1: Chris”.
- ❖ Se ejecuta el código en respuesta a ciertos eventos que ocurren en una página web. Se puede usar un evento clic en nuestro ejemplo anterior para detectar cuándo se hace clic en el botón y luego ejecutar el código que actualiza la etiqueta de texto.



Lo que es más emocionante es la funcionalidad construida sobre el lenguaje JavaScript de lado del cliente. Las denominadas interfaces de programación de aplicaciones (API) las cuales proporcionan herramientas adicionales para utilizar en el código JavaScript.

Las API son conjuntos de bloques de construcción de código listos para usar que permiten a un desarrollador implementar programas que de otra forma sería difícil o imposible implementar. Generalmente se dividen en dos categorías:



Las API's del navegador están integradas en dicho navegador web y pueden exponer datos del entorno informático circundante o realizar tareas complejas y útiles. Algunas son:

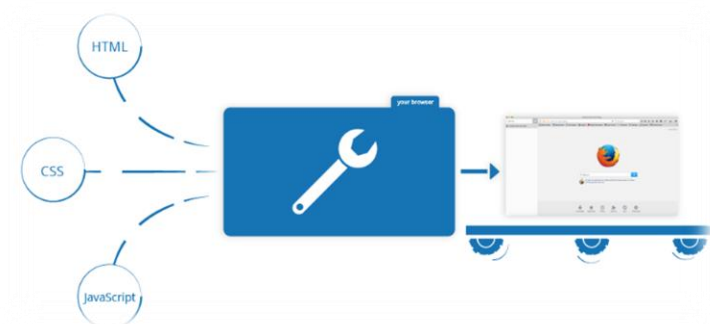
- ❖ La API del DOM (Documente Object Model) permite manipular HTML y CSS, crear, eliminar y cambiar el HTML, aplicar dinámicamente nuevos estilos a la página. Cada vez que aparece una ventana emergente en una página, o se muestra algún nuevo contenido, ese es el DOM en acción.
- ❖ La API de Geolocalización recupera información geográfica. Así es como Google Maps puede encontrar tu ubicación y trazarla en un mapa.
- ❖ Las APIs Canvas y WebGL permiten crear gráficos animados en 2D y 3D.
- ❖ APIs de audio y video como HTMLMediaElement y WebRTC te permite hacer cosas realmente interesantes con multimedia como reproducir audio y video directamente en una página web, o tomar video de tu cámara web y mostrarlo en la computadora de otra persona.

Las APIs de terceros no están integradas en el navegador de forma predeterminada y por lo general se debe obtener un código e información de algún lugar de la web. Estos pueden ser:

- ❖ La API de Twitter te permite hacer cosas como mostrar tus últimos tweets en tu sitio web.
- ❖ La API de Google Maps y la API de OpenStreetMap te permiten insertar mapas personalizados en tu sitio web y otras funciones similares.



Cuando cargas una página web en tu navegador, estás ejecutando tu código HTML, CSS y JavaScript dentro de un entorno de ejecución que es la pestaña del navegador.



Un uso muy común de JavaScript es modificar dinámicamente HTML y CSS para actualizar una interfaz de usuario a través de la API del modelo de objetos del documento. Hay que tener en cuenta que el código de los documentos web generalmente se cargan y ejecutan en el orden en que aparecen en la página. Si JavaScript se carga e intenta ejecutarse antes de que se haya cargado el HTML y el CSS al que afecta, pueden producirse errores.



Cuando el navegador encuentra un bloque de JavaScript, generalmente lo ejecuta en orden, de arriba abajo. Por ello hay que tener cuidado con el orden en el que colocas las cosas.

Veamos el orden de nuestro ejemplo.

```
1  const para = document.querySelector('p');
2
3  para.addEventListener('click', updateName);
4
5  function updateName() {
6    let name = prompt('Enter a new name');
7    para.textContent = 'Player 1: ' + name;
8  }
```

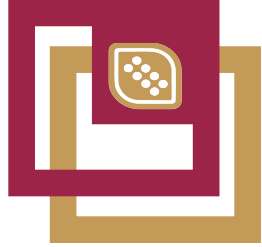
Aquí se selecciona un párrafo de texto (línea 1), luego adjuntamos un detector de eventos (línea 3) de modo que cuando se hace clic en el párrafo, el bloque de código updateName() (líneas 5 – 8) se ejecuta. El bloque de código updateName() estos tipos de bloques de código reutilizables se denominan funciones, la cual pide al usuario un nuevo nombre y luego inserta ese nombre en el párrafo para actualizar la pantalla.



Si se cambia el orden de las dos primeras líneas de código ya no funcionaría, en su lugar se obtendría un error en la consola del desarrollador del navegador TypeError: para is undefined. Lo cual significa que el objeto para aún no existe, por ello no se puede agregar un detector de eventos. JavaScript es un lenguaje de programación interpretado ligero.

El navegador web recibe el código JavaScript en su forma de texto original y ejecuta el script a partir de ahí. La mayoría de los intérpretes de JavaScript modernos utilizan una técnica llamada compilación en tiempo real para mejorar el rendimiento; el código fuente de JavaScript se compila en un formato binario más rápido mientras se usa el script, de esta forma se puede ejecutar lo más rápido posible. Sin embargo, JavaScript todavía se considera un lenguaje interpretado, ya que la compilación se maneja en el entorno de ejecución, en lugar de antes.





JavaScript se aplica a la página HTML de manera similar a CSS. Mientras que CSS usa elementos `<link>` para aplicar hojas de estilo externas y elementos `<style>` para aplicar hojas de estilo internas a HTML, JavaScript solo necesita un amigo en el mundo de HTML el elemento `{html<element("script")}`

BIBLIOGRAFÍA.

Álvarez, Miguel Ángel, Gutiérrez, Manu. (2007). Manual de JavaScript. Desarrolloweb.com
Recuperado de:

<https://blogsaverroes.juntadeandalucia.es/plataformaeiv/files/2016/09/manual-javascript.pdf>

Gauchat, J. D. (2012). El gran libro de HTML5, CSS3 y JavaScript. Primera edición. Editorial Marcombo. España. ISBN: 9788426717825.



¿Qué es JavaScript? - Aprende Desarrollo web | MDN. (s. f.). MDN Web Docs.
https://developer.mozilla.org/es/docs/Learn/JavaScript/First_steps/What_is_JavaScript

Manual de JavaScript. (s. f.).

DesarrolloWeb.com. <https://desarrolloweb.com/manuales/manual-javascript.html>



Actividad 7. ¿Qué es JavaScript?



Instrucciones: Con lo que lograste comprender de los conceptos tras realizar la Lectura No 9. Elige una de las dos actividades que se te presentan a continuación.

Actividad A

Elabora un juego didáctico interactivo siguiendo los siguientes pasos:

1.- Selecciona 10 conceptos mencionados en la lectura, considera que los conceptos deben ser cortos y entendibles.

2.- Ingresa a la herramienta EducaPlay (o similar).



3.- Si aún no tienes cuenta en la plataforma es necesario que realices un registro de un usuario nuevo. Puedes registrarte utilizando tu cuenta institucional en la opción de Microsoft.

Crea y comparte tu mundo de juegos educativos

¿Aún no tienes cuenta? [Regístrate gratis](#), es fácil y rápido.

Inicia sesión

Correo electrónico

Contraseña

☐ Mantener sesión iniciada en este navegador

Entrar

¿Has olvidado tu contraseña?

Acceder con Google

¿No estás registrado? [Obtén tu cuenta gratis](#)

4.- Después de ingresar a la plataforma con tu cuenta, debes dar clic en la opción "Crear Actividad".



5.- De las opciones que se te muestran en pantalla selecciona entre las opciones de Crucigrama y Sopa de Letras.



Sopa de letras

Esconde palabras entre un cuadrado lleno de letras para que las encuentren.



Crucigrama

Forma filas y columnas de palabras entrecruzadas para que rellenen sus letras.

7.- Comparte con tu maestro y compañeros tu nuevo juego.

6.- Siguiendo las instrucciones crea un Crucigrama o Sopa de Letras agregando los conceptos y definiciones identificados previamente.





Actividad B

Instrucciones: Después de realizar la Lectura 9. ¿Qué es JavaScript? Señalando lo más importante resuelve la siguiente sopa de letras.

J	A	V	A	S	C	R	I	P	T	E	S	M	U	U
Y	P	D	I	V	S	A	V	N	A	C	E	R	T	P
I	I	D	O	L	S	O	N	J	Q	U	E	R	Y	D
M	U	L	T	I	P	L	A	T	A	F	O	R	M	A
C	O	D	I	G	O	G	M	L	W	E	B	G	I	T
B	R	E	N	D	A	N	E	I	C	H	R	A	R	E
P	O	P	U	L	A	R	E	V	E	N	T	O	S	N
L	I	M	P	E	R	A	T	I	V	O	O	I	M	A
P	R	O	T	O	T	I	P	A	D	O	P	O	S	M
C	O	M	A	N	D	O	S	N	U	C	L	E	O	E
O	D	A	T	N	E	I	R	O	B	J	E	T	O	S
Q	U	E	R	Y	S	E	L	E	C	T	O	R	I	B
E	S	T	R	U	C	T	U	R	A	D	O	L	E	Y
I	N	T	E	R	P	R	E	T	A	D	O	H	A	C
E	R	L	I	V	E	S	C	R	I	P	T	L	O	C
O	N	T	O	S	T	Y	L	E	D	O	E	L	I	M
S	C	R	I	P	T	P	T	Y	P	E	E	T	U	D
E	L	U	N	D	E	F	I	N	E	D	H	O	Y	.

JAVASCRIPT	UPDATENAME	QUERYSELECTOR	API
CSS	NUCLEO	BRENDAN EICH	LIVESCRIPT
LIVIANO	PROTOTIPADO	MULTIPLATAFORMA	IMPERATIVO
OBJETOS	ORIENTADO	ESTRUCTURADO	EVENTOS
NAME	CANVAS	INTERPRETADO	WEBGI
CODIGO	POPULAR	COMANDOS	JQUERY
STYLE	SCRIPT	UNDEFINED	TYPE

Escribe la frase que esconde las letras restantes.





INSTRUMENTO DE EVALUACIÓN
LISTA DE COTEJO 11
Actividad 7: "¿Qué es JavaScript?"



Actividad A

DATOS GENERALES

Nombre(s) del alumno(s)	Matrícula(s)
Producto: Juego didáctico interactivo	Fecha
Submódulo 2: Programación Web	Periodo: 2025 - 2026B
Nombre del docente	Firma del docente

CRITERIO	INDICADORES	VALOR OBTENIDO		PONDERACIÓN	CALIF	OBSERVACIONES Y/O SUGERENCIAS DE MEJORA
		SÍ	NO			
1	Elige de 6 a 10 conceptos para la elaboración de las preguntas.			1		
2	Presenta un excelente uso del software para la creación de su juego didáctico.			2		
3	Presenta al menos 5 iconos o imágenes para mejorar la calidad de su juego didáctico.			2		
4	Presenta una actitud positiva en la realización y aplicación de su juego didáctico.			1		
5	No presenta faltas ortográficas.			1		
6	Considera las opiniones de sus compañeros para mejorar el juego didáctico.			2		
7	Entrega en tiempo y forma.			1		
				CALIFICACIÓN		

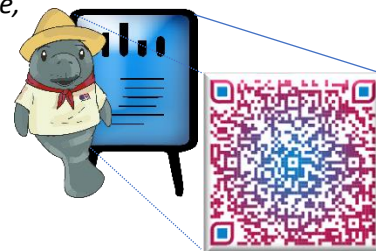


Lectura No 10. Instrucciones en JavaScript

Instrucciones: Realiza el análisis de la siguiente información, posteriormente, realiza la Actividad 8. Instrucciones en JavaScript



LECTURA 10. "INSTRUCCIONES EN JAVASCRIPT"



Como cualquier otro lenguaje de programación, JavaScript tiene algunas características especiales: *sintaxis*, *modelo de objetos*, etc. Claramente, cualquier cosa que diferencia un lenguaje de otro. Además, JavaScript es un lenguaje relativamente especial en su acercamiento a las cosas. Esta parte es esencial para cualquier principiante de programación e incluso para aquellos que ya conocen un lenguaje de programación debido a que las diferencias con otros lenguajes de programación son numerosas.

JavaScript es un lenguaje de programación de **scripts (secuencia de comandos)** orientado a objetos. Esta descripción es un poco rudimentaria, hay varios elementos que vamos a explicar.

- ❖ Es un lenguaje de programación En primer lugar, un lenguaje de programación es un lenguaje que permite a los desarrolladores escribir código fuente que será analizado por un ordenador.
- ❖ Un desarrollador o programador es una persona que desarrolla programas. Puede ser un profesional (un ingeniero, programador informático o analista) o un aficionado.
- ❖ El código fuente está escrito por el desarrollador. Este es un conjunto de acciones, llamadas instrucciones, lo que permitirá dar órdenes al ordenador para operar el programa. El código fuente es algo oculto, como un motor en un automóvil está oculto, pero está ahí, y es quien asegura que el coche puede ser conducido. En el caso de un programa, es lo mismo, el código fuente rige el funcionamiento del programa. Dependiendo del código fuente, el ordenador realiza varias acciones, como abrir un menú, iniciar una aplicación, efectuar búsquedas, en fin, todo lo que el equipo es capaz de hacer.

JavaScript te permite programar **scripts**. Como se mencionó anteriormente, un lenguaje de programación es utilizado para escribir código fuente a ser analizada por un ordenador.



Hay tres formas de usar el código fuente:

- ❖ **Lenguaje compilado** como: El código fuente se da a un programa llamado compilador que lee el código fuente y lo convierte en un lenguaje que el equipo será capaz de interpretar: el lenguaje binario, es de 0 y 1. Lenguajes como C o C ++ son lenguajes compilados muy conocidos.
- ❖ **Lenguaje precompilado:** aquí, el código fuente se compila en parte, por lo general en un código más fácil de leer para el ordenador, pero que todavía no es binario. Este código intermedio es para ser leído por lo que se llama una "Máquina Virtual", que ejecutará el código. Lenguajes como C # o Java se llaman precompilados.
- ❖ **Lenguaje interpretado:** en este caso, no hay compilación. El código fuente se mantiene sin cambios, y si desea ejecutar este código, debemos proporcionar un intérprete que va a leer y realizar las acciones solicitadas.

Los scripts son en su mayoría interpretados. Y cuando decimos que JavaScript es un lenguaje interpretado, lo que significa que es un lenguaje interpretado. Por tanto, es necesario contar con un intérprete para ejecutar código Javascript, y el intérprete que se utiliza una frecuencia: se incluye en tu navegador de internet. Cada navegador tiene un intérprete Javascript, que varía en función del navegador. Si está utilizando Internet Explorer, el intérprete es llamado JScript (versión 9 intérprete llamado Chakra), en Mozilla Firefox se llama SpiderMonkey y el motor V8 es el de Google Chrome.



Queda un aspecto a analizar: **orientado a objetos**. Este concepto es bastante complicado de configurar. Sin embargo, un lenguaje de programación orientado a objetos es un lenguaje que contiene elementos, llamados objetos y los objetos diferentes tienen características específicas y formas de uso diferente.

El lenguaje proporciona objetos básicos, como imágenes, fechas, cadenas de caracteres... También es posible crear tus propios objetos para hacer la vida más fácil y obtener un código fuente más claro (fácil de leer) y una forma de programar mucho más intuitivo (lógica).



JavaScript es un lenguaje utilizado principalmente con el lenguaje HTML, se aprenderá cómo integrar este lenguaje en tus páginas web, descubrir su sintaxis básica y mostrar un mensaje en la pantalla del usuario.

¡Hola Mundo! ¡No se deroga la regla tradicional de que todos los tutoriales de programación comenzarán mostrando el texto "Hello World!", ("¡Hola Mundo!" en español) al usuario. A continuación, se muestra un programa HTML simple que contiene la instrucción Javascript, situada dentro de un elemento

Al ejecutarlo aparece lo siguiente:

```
<!DOCTYPE html>
<html>
<head>
<title> ¡Hola Mundo! </title>
</head>
<body>
  <script>
    Alert('¡Hola Mundo!');
  </script>
</body>
</html>
```



En el código HTML indicado anteriormente, vemos algunas nuevas características. En primer lugar, un elemento **<script>** está presente: es él quien tiene el código JavaScript de la siguiente manera:

Alert(¡Hola Mundo!)

Es una declaración, es decir, una orden, o más bien una acción que el equipo tendrá que hacer. los lenguajes de programación consisten en una secuencia de instrucciones que, colocados de extremo a extremo, permiten obtener un programa o un script completo.

En este ejemplo, no es una instrucción: se llama a la función alert. El cuadro de diálogo de alert() alert() es una declaración simple, llamada función, que muestra un cuadro de diálogo que contiene un mensaje. este mensaje se coloca entre comillas, entre los paréntesis de la function alert().

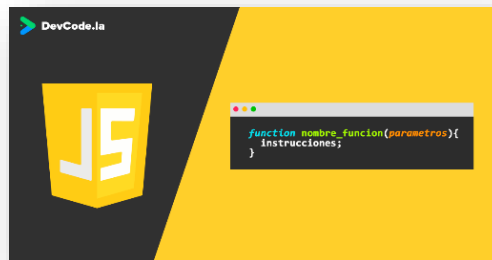


Sintaxis de JavaScript.

La sintaxis de Javascript no es complicada. Generalmente, las instrucciones deben estar separadas por un punto y coma que se coloca al final de cada instrucción:

Código: JavaScript

```
sentencia_1;  
sentencia_2;  
sentencia_3;
```



Algunas secuencias de comandos están disponibles en un formato comprimido, es decir, todo el código se escribe como secuencia, no hay retornos de línea. Esto reduce considerablemente el tamaño de una secuencia de comandos y sirve para asegurar que la página se carga más rápidamente. Existen programas para "comprimir" el código JavaScript. No hay que olvidar un solo punto y coma el código comprimido no va a funcionar porque las instrucciones no están debidamente separadas. También es una de las razones por las que siempre se pone el punto y coma al final de la instrucción.

Javascript no es sensible a los espacios. Esto significa que puedes alinear las instrucciones que quieras, siempre que no interfiera con la secuencia de comandos. Por ejemplo, esto es correcto:

Código: JavaScript

```
instruccion_1;  
instruccion_1_1;  
instruccion_1_2;  
instruccion_2; instruccion_3;
```

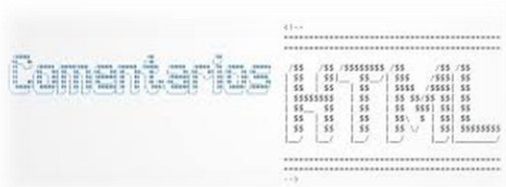
La sangría en la programación informática es una manera de estructurar el código para hacerlo más legible. Las instrucciones son priorizadas en varios niveles y espacios de usos o lengüetas para desplazar a la derecha y crear una jerarquía. Un ejemplo de código sangrado:

Código: JavaScript

```
function interruptor(elemID) {  
    var elem = document.getElementById(elemID);  
    if (elem.style.display == 'block') {  
        elem.style.display = 'none';  
    } else { elem.style.display = 'block';  
    }  
}
```



Comentarios.



Los comentarios son anotaciones realizadas por el desarrollador para explicar el funcionamiento de un script, una instrucción o incluso un grupo de instrucciones. Los comentarios no interfieren con la ejecución de un script.

Hay dos tipos de comentarios: los de fin de línea y las multilíneas.

Comentarios de fin de línea. Se utilizan para comentar una instrucción. Comienza con dos barras de división:

Código: JavaScript

sentencia_1 // Esta es mi primera instrucción

sentencia_2; // La tercera declaración es la siguiente:

sentencia_3;

El texto colocado en un comentario se ignora cuando se ejecuta un script, lo que significa que puedes poner un comentario, incluso en una instrucción (que, obviamente, no se ejecutará):

Código: JavaScript

sentencia_1 // Esta es mi primera instrucción

sentencia_2; // La tercera declaración da problemas, la cancelo temporalmente //

sentencia_3;



Comentarios multilínea. Este tipo permite saltos de línea. Un comentario multilínea comienza con /* y termina con */:

Código: JavaScript

/* Este script consta de tres pasos:

- Instrucción uno está haciendo algo

- Instrucción dos para otra cosa

- Instrucción tres que pone fin a la secuencia de comandos */

sentencia_1;

sentencia_2;

sentencia_3 // Fin del script

<!--	Código HTML o Contenido	-->
------	-------------------------	-----

Inicio del comentario COMENTARIO Fin del comentario

Inicio del comentario COMENTARIO Fin del comentario

Ten en cuenta que un comentario de varias líneas se puede mostrar en una sola línea:

Código: JavaScript

sentencia_1 /* Esta es mi primera instrucción */

sentencia_2;



Funciones.

En el ejemplo de ¡Hola mundo!, Se utilizó la función alert (). Una función consiste en dos partes: su nombre, seguido por un par de paréntesis (una apertura y un cierre): **Código: JavaScript**

myFunction () // "function" quiere decir "función" en inglés

Entre paréntesis se indican los argumentos que también se llaman parámetros. Estos contienen los valores que se pasan a la función. En el caso de ¡Hola mundo!, Son las palabras "¡Hola, mundo! " lo que se transfieren como parámetros:

Código: JavaScript

alert ('Hola mundo!');

Colocar el código JavaScript en la página HTML.

Los códigos JavaScript son insertados a través del elemento <script>. Este elemento tiene un atributo de tipo que se utiliza para indicar el tipo de lenguaje que vamos a utilizar. En nuestro caso, es JavaScript.

Para situar el código JavaScript directamente en una página web, nada más simple, siguiendo el ejemplo de ¡Hola, mundo!: se coloca el código en el elemento <script>

Código: HTML

```
<!DOCTYPE html>
<html>
<head>
<title> ¡Hola Mundo! </title>
</head>
<body>
<script>
Alert('¡Hola Mundo!');
</script>
</body>
</html>
```

Los comentarios de encuadramiento se utilizan para aislar el código Javascript para que el **Validator W3C (World Wide Web Consortium)** no los interprete.



Código: HTML

```
<body>
<script>
<!--
Valor 1> valor_2;
//-->
<script>
</body>
```





Es posible, y conveniente escribir el código JavaScript en un archivo externo con la extensión. Js. Este archivo se llama desde la página web mediante el elemento `<script>` y su atributo `src` que contiene la dirección URL del archivo. Js.

He aquí un pequeño ejemplo:

Código: JavaScript - contenido de ficheros hola.js

```
alert ('¡Hola mundo!');
```

El archivo hola.js se encuentra en el mismo directorio que el programa en HTML.

La mayoría de los cursos de Javascript, y ejemplos, muestran la necesidad de colocar el elemento `<script>` dentro de `<head>` cuando se utiliza para cargar un archivo JavaScript. Eso es correcto, sí, pero hay mejoras.

Una página web es leída por el navegador de forma lineal, es decir, en primer lugar, lee `<head>`, después los elementos de `<body>` uno después del otro.

Código: HTML

```
<!DOCTYPE html>
<html>
<head>
  <title>¡Hola Mundo!</title>
</head>
<body>
  <p>
    <!--
    Contenido de la página Web
    ...
    -->
  </p>
</script>
// Un poco de código JavaScript...
</script>
<script src="hola.js"></script>
</body>
</html>
```

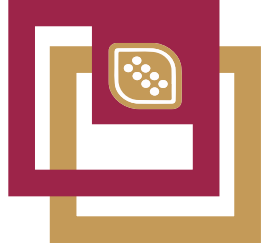
Código: HTML - Página Web

```
<! DOCTYPE html>
<html>
  <head>
    <title>¡Hola
    Mundo!</title>
  </head>
  <body>
    <script src="hola.js">
  </script>
</body>
</html>
```



Si se llama a un archivo JavaScript desde el principio de la carga de la página, el navegador por lo tanto cargará este archivo, y si es grande, la carga de la página se desacelerará. Esto es normal, ya que el navegador cargará el archivo antes de empezar a mostrar el contenido de la página. Para superar este problema, es conveniente colocar los elementos `<script>` justo antes de cerrar `<body>`.





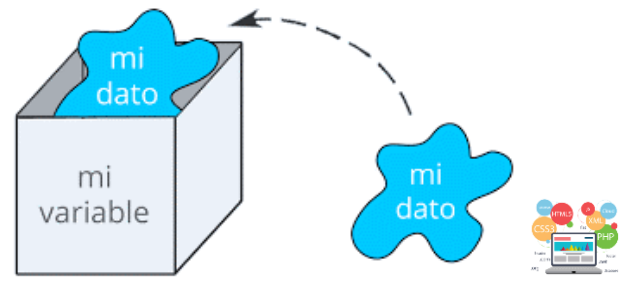
Acorde a ello podemos decir lo siguiente:

- ❖ Las instrucciones deben estar separadas por un punto y coma.
- ❖ Un código JavaScript bien presentado es más legible y fácil de editar.
- ❖ Es posible incluir comentarios con los caracteres `//`, `/*` y `*/`.
- ❖ Los códigos Javascript son colocados en un elemento de script.
- ❖ Es posible incluir un archivo JavaScript con el atributo src del elemento `<script>`.



Variables

Una variable es un espacio de almacenamiento en un ordenador para grabar cualquier tipo de datos como una cadena de caracteres, un valor numérico o estructuras un poco más específicas.



Declarar una variable en primer lugar, ¿qué significa "declarar una variable"?

Se trata simplemente de espacio de almacenamiento de reserva en memoria, nada más. Una vez que se declara la variable, puedes comenzar a almacenar datos sin problema. Para declarar una variable, **primero debes encontrar un nombre**. Es importante destacar que el nombre de una variable puede contener sólo caracteres alfanuméricos, es decir, letras de la A a la Z y números del 0 al 9, guion bajo (`_`) y dólar (`$`) también son aceptados.

El nombre de la variable no puede comenzar con un número y no puede consistir únicamente de palabras clave utilizadas por Javascript. Por ejemplo, no se puede crear una variable llamada `var` porque encontrarás que esta palabra clave ya está en uso, sin embargo, puedes crear una variable llamada `var_`. En las palabras clave utilizadas por JavaScript, se pueden llamar "palabras reservadas", simplemente porque no tienes el derecho de usarlos como nombres de variables.





Para declarar una variable, simplemente hay que escribir la siguiente línea: **Código: JavaScript**

`var miVariable;`

JavaScript es un lenguaje sensible en las denominaciones, ten cuidado de no confundir las mayúsculas y minúsculas. En el siguiente ejemplo, tenemos tres variables diferentes declaradas:

La palabra clave **var** está presente para indicar que se declara una variable. Una vez que se declara, se puede almacenar lo que quieras:

Código: JavaScript

`var miVariable;
miVariable = 2;`

Código: JavaScript

`var miVariable;
var mivariable;
var MIVARIABLE;`

El signo = se utiliza para asignar un valor a la variable, aquí le hemos asignado el número 2. Cuando das un valor a una variable, decimos que se trata de una asignación, ya que asigna un valor a la variable. Es posible simplificar el código en una sola línea: **Código: JavaScript**

`var miVariable = 5.5 // Como puedes ver, los números decimales se separan con un punto`

Del mismo modo, puedes declarar y asignar variables en una sola línea:

Código: JavaScript

`var miVariable1, miVariable2 = 4, miVariable3;`

Aquí hemos declarado tres variables en una fila, pero sólo la segunda tiene valor asignado. Y la última posibilidad, que puede ser útil de vez en cuando:

Código: JavaScript

`var miVariable1, miVariable2;
miVariable1 = miVariable2 = 2;`

Ambas variables ahora contienen el mismo número 2. Puedes hacer lo mismo con tantas variables como desees.

A diferencia de muchos lenguajes, JavaScript es un lenguaje de tipado dinámicamente. Esto significa, generalmente, que cualquier declaración de variables se hace con la palabra clave var independientemente de su contenido, mientras que, en otros lenguajes, como el C, es necesario especificar el tipo de contenido que tendrá que contener la variable.

En JavaScript, nuestras variables son tipeadas dinámicamente, lo que significa que puedes asignarle texto primero y luego borrarlo y poner un número cualquiera y sin restricciones. Vamos a empezar por ver cuáles son los tres tipos principales de JavaScript.





- ❖ **El tipo numérico (número):** representa cualquier número, ya sea un entero, un número negativo, en notación científica, etc. Este es el tipo de números. Para asignar un tipo numérico a una variable, sólo tienes que escribir el número `var numero = 5;` Al igual que muchos lenguajes, JavaScript reconoce varios formatos para los números, como por ejemplo escribir `var numero = 5.5` o en notación científica `var numero = 3.65 e5`, o escribir el número hexadecimal, `var numero = 0x391;` En resumen, hay diferentes maneras de escribir los valores numéricos.

- ❖ **Cadenas de caracteres (alias string):** Este tipo representa texto. Puede asignarse de dos maneras diferentes.

Código: JavaScript

`var text1 = "Mi primer texto" //`

Con comillas `var text2 = 'Mi segundo mensaje' //` Con apóstrofes Es importante tener en cuenta que si escribes `miVariable var = '2 '`, el tipo de esta variable es cadena de caracteres y no un tipo numérico. Otro punto importante, si usas apóstrofes para "enmarcar" el texto y deseas utilizar apóstrofes en el texto mismo, entonces tienes

Código: JavaScript

que "escapar" de los apóstrofes como se indica seguidamente:

`var text = 'Esto \' es algo ';`

¿Por qué esto? Porque si no cancelas tu apóstrofo, Javascript cree que el texto se detiene en el apóstrofo contenido en la palabra "es". Ten en cuenta que este problema es idéntico al de las comillas. En nuestro caso, solemos utilizar apóstrofes, pero cuando el texto los contiene también entonces las comillas pueden ser muy útiles.



- ❖ **Booleanos (booleano):** un tipo booleano permite dos estados verdadero o falso. Estos dos estados se pueden escribir como sigue:

Código: JavaScript

`var EsVerdadero = true; var EsFalso = false;`

Puede ser que tengas en alguna ocasión la necesidad de probar la existencia de una variable o comprobar su tipo. En tales situaciones, la instrucción `typeof` es muy útil.

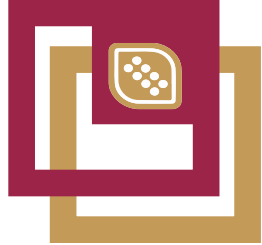
Operadores aritméticos.

Ahora que se declara una variable y se asigna un valor, podemos comenzar con los operadores aritméticos.

El último operador, módulo, es simplemente el resto de una división. Por ejemplo, si se divide 5 entre 2 se tiene resto 1, que es el módulo.

Operador	Símbolo
Suma	+
Resta	-
Multiplicación	*
División	/
Módulo	%





Programar el cálculo es casi tan fácil como en una calculadora, por ejemplo:

Código: JavaScript

```
var resultado = 3 + 2;  
alert (resultado) // Muestra « 5 »
```



Así que puedes hacer cálculos con dos números, es bueno, pero con dos variables que contienen números en sí es más útil:

Código: JavaScript

```
var número1=3, número2 = 2, resultado;  
resultado = numero1 * numero2;  
alert (resultado) // Muestra: « 6 »
```

El operador **+**, además de las sumas, se puede hacer lo que se conoce como concatenación entre cadenas. La concatenación es añadir una cadena al final de otra, como en este ejemplo:

Código: JavaScript

```
var hola= 'Hola', nombre = 'tu', resultado;  
resultado = hola + nombre;  
alert (resultado) // muestra: « Holatu »
```



La concatenación es un buen momento para introducir la primera interacción con el usuario a través de la función prompt (). He aquí cómo se usa:

Código : JavaScript

```
var NombreUsuario = prompt('Introduce nombre:');  
alert(NombreUsuario); //
```

Muestra el nombre introducido. La función prompt () se utiliza como alert (), pero tiene una pequeña particularidad. Devuelve lo que el usuario escribió bajo una cadena de caracteres, es por eso que escribió esto:

Código: JavaScript

```
var texto = prompt ('Entra algo:');
```

Así, el texto escrito por el usuario terminará almacenado directamente en el texto variable.

Código: JavaScript

```
var inicio = 'Hola', nombre, final = '!', resultado;  
nombre= prompt ("¿Cuál es tu nombre? ");  
resultado = inicio + nombre + final  
alert (resultado);
```



Ten en cuenta que en nuestro caso concatenamos cadenas de caracteres, pero se pueden concatenar una cadena y un número de la misma manera:

Código: JavaScript

```
var text o= 'Un nombre : ', numero= 42, resultado;
resultado= texto + numero;
alert(resultado); // Muestra: « Un nombre : 42 »
```

Ahora trataremos de hacer una adición con números proporcionados por el usuario:

Todo lo que se escribe en el campo de texto prompt () se recupera como una cadena de caracteres, aunque sea un número. Por lo tanto, si se utiliza el operador +, no será una suma sino una concatenación. Se ha de efectuar una conversión. El concepto es simple: convertir la cadena en un número. Para ello, necesitarás la función parseInt () que se utiliza de esta manera:

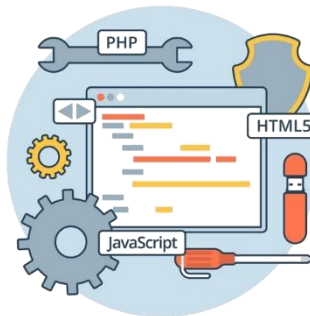
Código: JavaScript

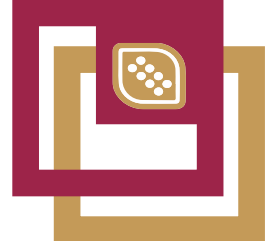
```
var texto = '1313', numero;
numero = parseInt(texto);
alert(typeof numero); // Muestra : « numero »
alert(numero); // Muestra : « 1313 »
```

Ahora que ya sabes cómo usarlo, vamos a ser capaces de adaptar nuestro código: **Código: JavaScript**

```
var primero, segundo, resultado;
primero = prompt ('Ingrese el primer número: ');
segundo = prompt ('Ingrese el segundo número:');
resultado = parseInt (primero) + parseInt (segundo);
alert (resultado);
```

Ahora, si escribes dos veces la cifra 1, el resultado es 2.

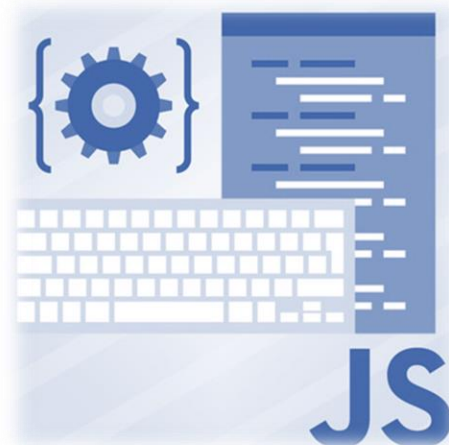




Eventos.

Una de las características principales de JavaScript es la posibilidad que ejecute código cuando ocurre un evento. Un ejemplo de algunos eventos en JavaScript es:

- ❖ onLoad: Cuando carga un documento HTML.
- ❖ onUnload: cuando se abandona un documento HTML.
- ❖ onMouseOver: el ratón pasa por encima de un enlace.
- ❖ onClick: se hace click sobre un enlace o un campo de un formulario.
- ❖ onSubmit: al enviar un formulario.
- ❖ onChange: al cambiar el contenido de un campo de un formulario.
- ❖ KeyDown: cuando el usuario presiona una tecla
- ❖ KeyUp: cuando el usuario suelta una tecla.



BIBLIOGRAFÍA.



Álvarez, Miguel Ángel, Gutiérrez, Manu. (2007). Manual de JavaScript. Desarrolloweb.com
Recuperado de:

<https://blogsaverroes.juntadeandalucia.es/plataformaeiv/files/2016/09/manual-javascript.pdf>

Gauchat, J. D. (2012). El gran libro de HTML5, CSS3 y JavaScript. Primera edición. Editorial Marcombo. España. ISBN: 9788426717825.

Ramos, R. (2016). *¿Qué es JavaScript y para qué sirve?* Agencia de Marketing Rafa Ramos.

<https://soyrafaramos.com/que-es-javascript-para-que-sirve/>



Actividad 8. Instrucciones en JavaScript



Instrucciones: Después de realizar la Lectura 10. Instrucciones en JavaScript señalando lo más importante relaciona las siguientes columnas.



1. Deben estar separadas por un punto y coma (;) que se coloca al final.	Funciones ()
2. Está escrito por el desarrollador.	Sangría ()
3. Es un lenguaje de programación de scripts.	Lenguaje interpretado ()
4. Consiste en dos partes: su nombre seguido por un par de paréntesis.	Instrucciones ()
5. Es un espacio de almacenamiento en un ordenador.	Orientado a objetos ()
6. El código fuente se da a un programa llamado compilador.	Código fuente ()
7. En la programación es una manera de estructurar el código.	Comentarios ()
8. En este caso no hay compilación y el código fuente se mantiene sin cambios.	<script> ()
9. En su mayoría son interpretados.	Variables ()
10. Son anotaciones realizadas por el desarrollador para dar explicaciones.	JavaScript ()
11. Los códigos son insertados a través de este elemento.	Scripts ()
12. Es un concepto bastante complicado de configurar.	Lenguaje compilado ()



Realiza la evidencia en línea

https://es.educaplay.com/recursos-educativos/11144783-instrucciones_en_javascript.html



Ejemplo Guiado



Instrucciones: Realiza el siguiente ejemplo paso a paso.



❖ Ingresa a Sublime Text o Bloc de notas.

❖ Empieza escribir la primera parte de la página HTML.

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>Formularios en HTML y JavaScript</title>
```

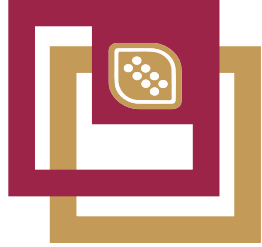
❖ Escribiremos el elemento <script>.

```
<script language="javascript">
```

❖ Escribimos la función dentro del elemento <script>.

```
function calculos(formulario)
{
    var n1,n2,n3,n4,n5,n6,n7;
    n1=parseInt (formulario.n1.value);
    n2=parseInt (formulario.n2.value);
    n3=n1+n2;
    n4=n1-n2;
    n5=n1*n2;
    n6=n1/n2;
    n7=n1%n2;
    formulario.n3.value=n3;
    formulario.n4.value=n4;
    formulario.n5.value=n5;
    formulario.n6.value=n6;
    formulario.n7.value=n7;
}
```





- ❖ Cerramos el elemento <script>.

```
</script>
```

- ❖ Cerramos el head.

```
</head>
```

- ❖ Escribimos las siguientes instrucciones.

```
<body bgcolor="silver">
```

```
<font color="red">
```

```
<marquee>
```

```
<h1>Operaciones básicas</h1>
```

```
</marquee>
```

```
<center>
```

```
<h2>Introduce el valor de dos números</h2>
```

```
<hr size=10 color="pink">
```

- ❖ Insertamos el formulario.

```
<form name="form1">
```

- ❖ Se crean los primeros elementos del formulario.

```
Número 1:<input type="text" name="n1" size="3"><br>
```

```
Número 2:<input type="text" name="n2" size="3"> <br>
```

- ❖ Se inserta el botón para el formulario.

```
<input type="button" value="¡Calcular!" onclick="calculos(form1)"> <br>
```

- ❖ Se insertan los elementos del formulario para mostrar los resultados.

```
La suma de los números es:<input type="text" name="n3" size="3"><br>
```

```
La resta de los números es:<input type="text" name="n4" size="3"><br>
```

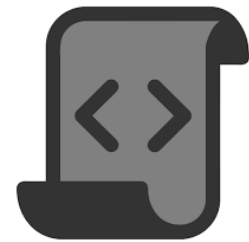
```
La multiplicación de los números es:<input type="text" name="n5" size="3"><br>
```

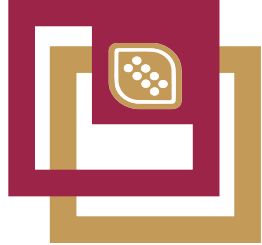
```
La división de los números es:<input type="text" name="n6" size="3"><br>
```

```
El residuo de los números es:<input type="text" name="n7" size="3"><br>
```

- ❖ Cerrar el formulario.

```
</form>
```





- ❖ Colocar el resto de las instrucciones.

```
</center>
<hr size=10 color="pink">
</font></body>
</html>
```



- ❖ Guardar el archivo. Das clic en archivo, Guardar como, escribes el nombre OperacionesBasicas.html y das clic en guardar.
- ❖ No cerrar el Sublime texto o bloc de notas por cualquier error que se presente para corregirlo.
- ❖ Buscas el archivo en tu carpeta y lo abres.
- ❖ Se abrirá el explorador y mostrará la siguiente pantalla.

- ❖ Escribir el valor en Número 1 y Número 2.
- ❖ Dar clic en el botón calcular.

¡Calcular!

- ❖ Mostrará la siguiente pantalla.

- ❖ Ahora puedes cambiar el estilo de la página con CSS.





Lectura No 11. ¿Cómo Publico Mi Página Web?

Instrucciones: Realizar la lectura subrayando lo más relevante para resolver la Práctica 7. Publicando la Página Web Dispositivos Móviles.



LECTURA 11. "¿CÓMO PUBLICO MI PÁGINA WEB?"



Determinando la mejor Opción para mi Pagina Web

Para **poder publicar tu web en Internet** y que todo el mundo pueda verla necesitas una serie de recursos. No basta con planear la página web y desarrollar la web. Una vez la tienes creada hay que hacerla accesible a cualquiera que quiera visitarla. Para ello, los siguientes pasos pueden ayudarte. Debes empezar con conocer dos conceptos indispensables para lograr este propósito, hosting y dominio, vamos a identificar cada uno.

¿Qué es un dominio de internet y cuál es su importancia para un sitio web?

A grandes rasgos, **un dominio web es el nombre único que se le asigna a un sitio web**. Haciendo una analogía: si un sitio web es una casa, el dominio es su dirección. Cada sitio web tiene un nombre de dominio, y este nombre es único para cada sitio.



En términos prácticos, el dominio es una dirección virtual que identifica la presencia en internet de una persona, empresa u organización, y que sirve para dirigir a los visitantes hacia un sitio web, así como para crear cuentas personalizadas de correo electrónico. Estrictamente hablando, el dominio es una **cadena de caracteres única** que un usuario debe ingresar a un navegador web para acceder a un sitio web específico.

Por ejemplo, para visitar el sitio de GoDaddy no basta con escribir "GoDaddy" en la barra de direcciones, ya que esto solamente le indicará al buscador que se ponga a buscar todo lo relacionado con GoDaddy, como se ve en este ejemplo:

"Así, para acceder directamente al dominio de GoDaddy lo correcto sería escribir: godaddy.com."

Ahora, una breve explicación técnica...





Internet es una inmensa red de computadoras conectadas entre sí a través de una infraestructura global. Para identificarlas, **a cada computadora se le asigna una dirección Protocolo de Internet (IP en inglés)**, que es una compleja cadena de números y letras que las computadoras usan para reconocer un sitio web. Una dirección IP típica se ve así:



172.16.254.1

¡Imagínate que tuvieras que recordar todos estos números para visitar tus páginas web favoritas!



Pues precisamente los dominios de internet fueron inventados para resolver este problema. De ahí la importancia de tener un nombre de dominio simple y fácil de recordar, dado que esto ayuda a que las personas **puedan encontrar tu sitio web sencillamente**.

Ahora, los nombres posibles para tener un sitio web son finitos, así como la competencia por estar en la mente de los usuarios, así que existe una alta demanda en el mercado de dominios de internet. Como resultado, cada nombre de dominio tiene un valor monetario y se puede comprar, vender e intercambiar a través de varios distribuidores en línea.



Elementos que forman la estructura de un dominio de internet

Muchas veces se habla de diferentes tipos de dominios, pero en realidad todo se reduce a los dos componentes principales del dominio web: un nombre y una extensión de dominio.

www.misitioweb.com

dominio extensión

Partes del dominio

El nombre de dominio es básicamente la parte que va después del “**www.**” (o, en el caso de tu correo, lo que sigue al signo de @) y la extensión de dominio es la parte final de la dirección web (“**.com**” en el ejemplo).

Ahora entiendo la importancia del dominio y hosting





Tipos de dominios web

Utilizando un lenguaje más técnico, el nombre y la extensión representan a los dos niveles que construyen un dominio web, y existen términos específicos para identificarlos: la extensión se conoce como **dominio de nivel superior** (TLD por sus siglas en inglés) y el nombre se identifica como **dominio de segundo nivel** (SLD). En el ejemplo que mostramos arriba, el nombre de dominio “misitioweb” es el SLD y la extensión “.com” es el TLD.

Estos términos suenan complejos, pero podemos aclararlos al explicar los tipos de dominios que existen:

Dominios de nivel superior (TLD). También denominados sufijos o extensiones de dominio, los TLD son el segmento final del nombre de un dominio. Como ya lo mencionamos, el TLD en nuestro ejemplo sería “.com”.

La **Corporación de Internet para la Asignación de Nombres y Números (ICANN)**, que es una organización sin fines de lucro con sede en Estados Unidos, es la encargada de crear e implementar las políticas para los nombres de dominio, así como de verificar los registros de sitios en las diferentes extensiones disponibles. Para ICANN, hay dos tipos principales de extensiones de dominio:

- **Dominios genéricos o gTLD (generic Top-Level Domain)**

Son el tipo más común de extensión. Las 5 más utilizadas son: **.com**, **.net**, **.org**, **.edu** y **.gob**. Su función es definir el propósito de un sitio web, por ejemplo para fines comerciales (.com) o educativos (.edu).

Y últimamente han aparecido muchos dominios genéricos nuevos. Algunos ejemplos son:

[.xyz](#) [.online](#) [.app](#) [.vip](#) [.club](#) [.space](#) [.live](#) [.site](#) [.dev](#)
[.shop](#)

- **Dominios geográficos o ccTLD (country code Top-Level Domain)**

Se utilizan para localizar un dominio en un territorio o país determinado. Se conforman por **dos letras asociadas con el país de origen**, por ejemplo: **.mx es la extensión para México**, **.us** para Estados Unidos y **.uk** para el Reino Unido. Muchas compañías u organizaciones prefieren registrar su dominio con estas extensiones para señalar que están ubicadas en dicho país o que hacen negocios en cierta región.



Dominios de segundo nivel (SLD). Es la sección después del www. y antes de la extensión o TLD, y es la parte más importante del dominio porque **representa la identidad principal del sitio web ante los usuarios**. Así, el papel del SLD es reforzar la identidad de la marca o del sitio web. En nuestro ejemplo, justamente, la marca y el dominio son uno mismo: **misitioweb**.

Dominios de tercer nivel. Existe una subcategoría de los TLD, que consiste en combinar un dominio genérico de primer nivel con una extensión de país. El ejemplo más claro es **“.com.mx”**.

Esta es la opción a la que acuden muchos negocios nuevos que quieren registrar un dominio que ya está ocupado en el TLD de primer nivel, además de que suelen ser más económicos.

Diferencia entre dominio y URL

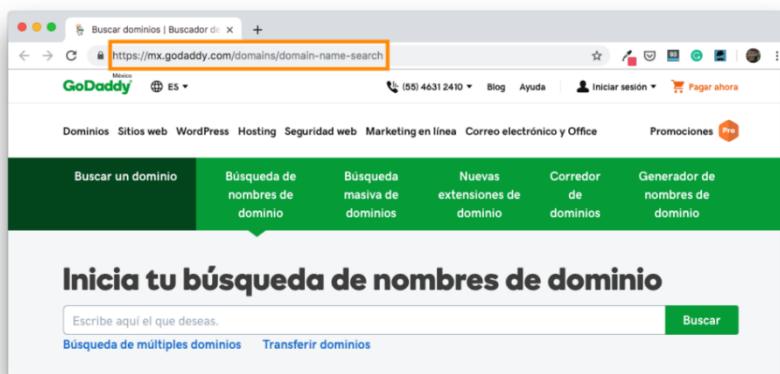
Antes hablamos sobre los caracteres que conforman al nombre de dominio, y quisimos incluir este punto particular porque sabemos que algunas personas pueden **confundir al dominio con la URL**, pese a que existen diferencias sustanciales.

¿Qué es una URL?

URL son las siglas en inglés de *Uniform Resource Locator* o Localizador Uniforme de Recursos, que es la **dirección específica asignada a cada uno de los recursos disponibles en la red**, con la finalidad de que estos puedan ser localizados o identificados. Por ejemplo, la siguiente URL:

<https://mx.godaddy.com/domains/domain-name-search>

Te lleva directamente al sitio de GoDaddy donde se encuentra nuestro buscador de nombres de dominio.



Como puedes ver, esta URL contiene el dominio principal de GoDaddy (godaddy.com) y otros elementos:

- **El protocolo y el certificado de seguridad.** El inicio de una URL es el protocolo de entrada, que indica al navegador cómo acceder a un recurso web específico. La mayoría de las direcciones web utilizan HTTP (Protocolo de transferencia de hipertexto) o el estándar recomendado HTTPS (HTTP con certificado de seguridad SSL). En el ejemplo anterior, “https://” es el protocolo de seguridad.

Cabe mencionar que, en los inicios de internet, los certificados de seguridad sólo los utilizaban los sitios de e-Commerce para proteger los datos financieros de sus clientes. Hoy en día, hasta los sitios web informativos los utilizan para darle seguridad al usuario que navega por sus páginas. La forma más fácil de identificarlo es por el símbolo del candado junto a la URL:



Aprende que es el Certificado de seguridad SSL



- **El subdominio.** Es un subgrupo de los tipos de dominio ya mencionados y sirve para crear secciones diferenciadas dentro del dominio principal. Si la URL incluye un subdominio, éste viene antes del nombre del dominio, separado por un punto. En nuestro ejemplo, <https://mx.godaddy.com/domains/domain-name-search>, mx. es el subdominio. **(Ojo: no confundir con una extensión .mx, que en este caso sería otro dominio totalmente distinto al principal).**

Como podrás notar, no todos los subdominios incluyen las siglas www. de World Wide Web antes del nombre de dominio, y tú como usuario quizá ya te acostumbraste a escribirlo sin las iniciales “www” porque los navegadores completan la información por ti. Las URLs sin subdominio también son comunes.





- **La ruta.** Esta sección de la URL define el recurso exacto que se va a mostrar en el navegador web. Es todo lo que aparece después del nombre de dominio. En el ejemplo, la ruta sería: **/domains/domain-name-search**. Es importante notar que la ruta siempre empieza con una diagonal (/) y distingue entre mayúsculas y minúsculas.
- **El directorio o subcarpeta.** Esencialmente, esta sección de la URL es una carpeta dentro del sitio web principal que alberga el recurso específico. En nuestro ejemplo, **/domains/** es el directorio. Algunas URL tienen subcarpetas dentro de los directorios.
- **El nombre del archivo.** Es la última sección de la URL. Le indica al servidor web el documento exacto que debe mostrar al usuario final. En nuestro ejemplo, el archivo o página es **/domain-name-search**. Las extensiones de archivo comunes incluyen .pdf, .png y .html.

Con todo lo visto hasta ahora, en el siguiente gráfico podrás ver claramente las partes que conforman una URL:



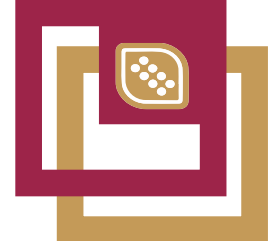
En conclusión, la principal diferencia entre dominio y URL es que el primero es el nombre del sitio web, mientras que el segundo se utiliza para acceder a una página específica del sitio web. Así, cada página, imagen o medio tiene una URL única.

¿Cómo funciona un dominio y cuál es el rol del Servidor de Nombres de Dominio?

Ahora sabes que el dominio web es el nombre de tu sitio en internet, que tiene dos secciones principales –digamos que es como el nombre y apellido de una persona– y que los elementos “extra” que agregues a ese dominio servirán para encontrar una página o archivo específico en internet.

Ahora llega el momento de explicar cómo funciona todo en conjunto.

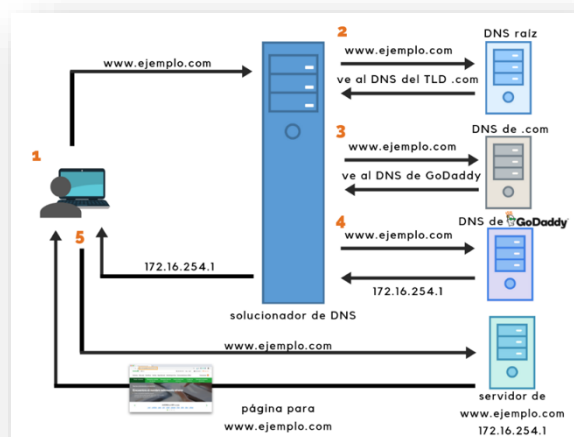




A grandes rasgos, un *Domain Name System* –también denominado Servidor de Nombres de Dominio o DNS por las siglas del inglés– es una **gran base de datos que contiene muchas direcciones IP** (¿recuerdas que hablamos de ellas en el primer punto?), así como el listado de personas, entidades o empresas que reclaman la propiedad sobre dichos sitios.

El proceso que te permite ingresar a un sitio web a través de un sistema DNS ocurre en milisegundos y de forma imperceptible para el ojo humano. A continuación, lo resumimos:

1. Cuando escribes el dominio `www.ejemplo.com` en tu navegador web y das clic al botón Enter, inmediatamente se envía una solicitud al proveedor de servicios de internet (ISP en inglés, es decir, tu compañía telefónica o proveedor de banda ancha) que cuenta con servidores DNS propios, también llamados solucionadores de DNS.
2. El solucionador de DNS de tu ISP reenvía la solicitud a un servidor de nombres de raíz, que revisa si en los servidores del ISP existe alguna referencia almacenada en caché (temporalmente) sobre la IP o la fuente directa del dominio `www.ejemplo.com`. Si es así, el solucionador recibe los datos de la página; si no, acude a un nivel superior.
3. Entonces, el solucionador de DNS reenvía la solicitud a un servidor de nombres de dominio superior, donde se ubican todos los dominios `.com`. Este DNS responde a la solicitud con la información sobre algún servidor de nombres asociado con el dominio `www.ejemplo.com` (por ejemplo, algún DNS de GoDaddy).
4. El solucionador acude al DNS de GoDaddy asociado con `www.ejemplo.com`, obtiene la dirección IP (por ejemplo: `172.16.254.1`) asociada y la transmite de regreso al usuario.
5. Finalmente, el navegador web obtiene la IP, recupera los archivos del sitio web y te muestra la página web en tu pantalla.



Ejemplo de consulta de dominio de internet



¿Y por qué es importante el DNS?

Como mencionamos anteriormente, un DNS se utiliza comúnmente para asignar nombres de dominio a direcciones IP, particularmente la dirección IP del hosting donde se aloja tu dominio. Saber esto es importante en caso de tener que redirigir tu dominio a otro servicio de hosting.

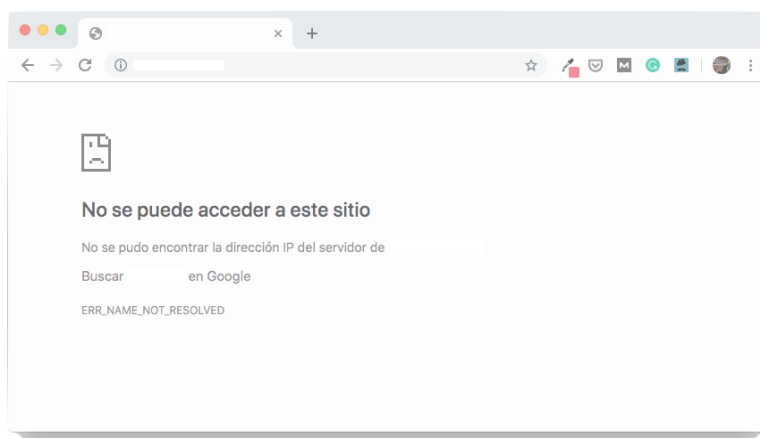
Por eso es fundamental que, al adquirir un dominio, registres el DNS del hosting que vayas a utilizar para alojar tu página web. A este proceso se le conoce como “apuntar el dominio a los DNS del hosting”.

En general, los DNS de un dominio utilizan dos direcciones IP: una para el DNS primario y otra para el DNS secundario. Esto se hace en caso de que la primera IP tenga problemas o deje de funcionar. Normalmente se parecen a esto:

ns1.ejemplo.com

ns2.ejemplo.com

Así, si tu dominio no está correctamente configurado o tus DNS no apuntan a un hosting particular, los usuarios de internet no podrán acceder a tu sitio y verán un error como este:



¿Cómo puedo registrar un dominio web?



Antes de comprar tu dominio, es importante que conozcas la estructura a través de la cual se realiza el registro en internet. En este proceso participan tres actores esenciales que, ordenados de forma jerárquica, se llaman **registro, registrador y registrante**:

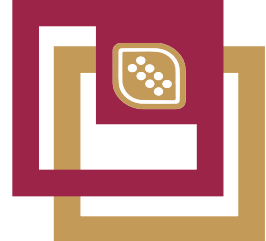
CONSULTA EL QR
PARA VER COMO LA
EMPRESA GODADDY
TE AYUDA A
REGISTRAR UN SITIO



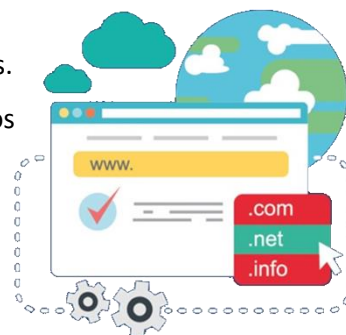
Registro. Es la organización que **posee los derechos de las extensiones de dominio**. El registro se encarga de crear dominios de nivel superior, así como de establecer las pautas para usar los TLD y distribuir los derechos para vender esos dominios a los registradores. Algunos registros populares son:

1. Donuts
2. VeriSign
3. Afiliadas
4. Amazon Registry Services
5. Global Registry Services





Registrador. Es la plataforma mediante la cual se pueden adquirir dominios. Es el **intermediario ante el registro para vender extensiones** a los usuarios que quieren adquirirlas.



Siempre que quieras comprar un dominio deberás acudir con un registrador.

Esta estructura permite que varios registradores ofrezcan las mismas extensiones de dominio, y explica también por qué es posible encontrar un mismo nombre de dominio en varios registradores.

Además de la compra - venta, otras funciones del registrador son: **administrar, renovar o transferir** los registros de nombres de dominio a otros registradores, y ofrecer un sistema para gestionar los registros de los usuarios.

Registrante. Es la persona, entidad o empresa que registra un dominio en un registrador. Es decir, **es el dueño del dominio.**



En palabras más simples, tú como registrante podrías comprarle “ejemplo.com” al registrador GoDaddy, pero a su vez GoDaddy trabajó con el registro de VeriSign para poder ofrecerte ese dominio.

Y una vez que hayas encontrado tu dominio ideal, debes seguir estos sencillos pasos para registrarlo:

1. **Ir con un registrador y verificar la cuota de registro del dominio que quieres.**
2. **Crear una cuenta con el proveedor de tu elección.**
3. **Pagar el dominio, lo cual te dará derecho a ser propietario de ese dominio por una cantidad determinada de tiempo.**
4. **Si lo deseas, puedes obtener servicios adicionales como seguridad SSL si tu proveedor los ofrece.**



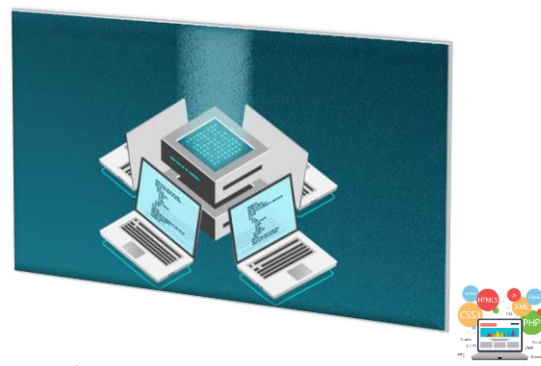
¿Cómo hacer que mi dominio aparezca en internet?

Para llevar tu propuesta a internet, además de un dominio necesitarás tener bajo control los elementos básicos de la web, que son sitio, página y hosting.

El sitio y la página los conoces bien: uno es el **conjunto de páginas y archivos** (como imágenes o videos) que conforman todo (“ejemplo.com”) y la otra es **lo que ve el usuario en una sola URL** dentro del sitio (“ejemplo.com/yy” y “ejemplo.com/xx”). El objetivo de ambos es producir una experiencia a los potenciales visitantes de tu sitio.

¿Qué es el hosting?

Por su parte, el hosting o alojamiento web **son las computadoras donde se almacenan los datos de tu sitio en línea**. A estas computadoras se les denomina servidores, y las empresas de hosting ofrecen servidores web como un servicio.



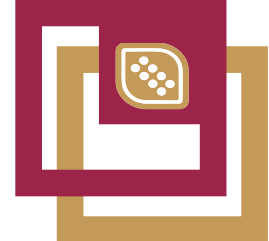
Retomando la analogía que escribimos al principio, el dominio es la dirección, el sitio web es la casa, una página web puede ser uno de los ladrillos y **el hosting es el terreno** donde está construida. Para que tengas un mayor control sobre los datos de tu sitio, así como de la configuración de sus servidores, **lo más recomendable es comprar tu dominio y hosting de una misma compañía**.

Así también podrás administrarlos con una sola cuenta y aprovechar los descuentos que suelen ofrecer por adquirir un paquete con ambos servicios. Si acaso ya contrataste estos elementos por separado, ¡no te preocupes! Solamente necesitarás enviar a tu proveedor de hosting la información proporcionada por el sistema de denominación de dominio, o DNS.

Protección de marca y otras consideraciones sobre registro de dominios

Debes considerar que puedes usar tu nombre con varias extensiones (o sea, con diferentes dominios de nivel superior), pero **sólo se pueden registrar una vez**. Es decir, puedes conseguir “spaclinic.com” y “spaclinic.mx”, pero si alguien más ya compró el dominio “spaclinic.com” ya no podrás usarlo.





Así que es importante registrar tu dominio lo antes posible para evitar perder el nombre perfecto para tu página, y evitar que otras personas o compañías puedan realizar acciones especulativas con tu nombre. Y algo todavía más importante es que, antes de efectuar el registro, te asegures de **verificar que el nombre elegido de dominio no pertenezca a una marca registrada**, ya que podrías tener problemas judiciales por apropiación indebida de propiedad intelectual.

Estas son otras cuestiones que debes tomar en cuenta antes de registrar un dominio de internet:

- **Opciones a la medida.** En el mercado hay varias promociones de paquetes donde puedes adquirir hosting web con dominio incluido. Aquí sólo debes asegurarte de elegir el plan que más te convenga, para que no pagues un espacio de almacenamiento o una cantidad de dominios que no vas a usar.
- **Validez del registro.** Entre los varios datos del contrato de registro debe aparecer tu nombre como registrante (propietario) y como contacto administrativo. Aunque también puedes registrarte como anónimo.
- **Herramientas de administración.** El registrador debe ofrecerte una plataforma para hacer modificaciones básicas en los datos de tu registro de dominio como son: cambiar y actualizar el contacto administrativo y de facturación; redirigir el dominio a una URL diferente; modificar las DNS del dominio; revisar la fecha de expiración del registro para renovación.
- **Estabilidad y soporte técnico.** Un registrador serio siempre estará disponible para resolver tus dudas y responder en caso de problemas técnicos. Asimismo, los registradores de prestigio no desaparecerán de la noche a la mañana, por lo que tendrás certeza de que tu dominio siempre estará ahí.



Ahora que estoy listo para comprar un dominio, ¿qué sigue?

Está más que claro que el marketing digital es imprescindible en esta época de tecnologías móviles y redes sociales. Tan sólo por mencionar un dato, los expertos estimaron en el año 2019 que más de 155 millones de personas en América Latina adquirirían productos y servicios en línea, un 18% más comparado con las 126.8 millones de compras registradas en la región dos años antes.



Considerando este dato, debes pensar que tener un dominio web te da la posibilidad de **crear una presencia digital atractiva y accesible** en todo momento, dirigida a los consumidores de hoy que usan internet para cualquier consulta: desde decidir qué desayunar hasta planear sus vacaciones o una boda.

Ahora que aprendiste el funcionamiento de los dominios de internet, sus características y las particularidades del proceso de registro, estás listo para dar el primer paso hacia la creación de tu sitio web. Para eso te recomendamos el creador de páginas web, una herramienta de edición y publicación en línea que te ayuda a crear una página web fácilmente y sin necesidad de contar con habilidades técnicas.

Los 10 mejores hostings gratuitos

Si a pesar de las desventajas que te acabamos de mencionar sigue siendo tu idea probar un hosting gratuito, aquí tienes un listado con los mejores proveedores de este tipo y sus características principales.



Lucushost

De entre todas las páginas de hostings gratuitos, Lucushost es posiblemente la mejor de todas.

Se trata de una empresa española que **cuenta con una plataforma estable** para ofrecer alojamiento sin coste a sitios web de todo tipo.

Entre las características más destacadas de Lucushost están la instalación en un solo clic de distintos CMS incluidos por supuesto WordPress y PrestaShop, certificado de seguridad SSL gratis y soporte personalizado las 24 horas del día. Todo lo que ofrece este hosting hace dudar que sea completamente gratuito, pero en realidad lo es.

Características principales

Idioma: Español

Espacio web: 1Gb

Transferencia mensual: 5Gb

Sitios web: 1

Panel de control: cPanel

Soporte: Sí



000WebHost

Ampliamente usada por clientes de todo el mundo, 000WebHost es otro de esos **hostings gratuitos sin publicidad más populares**.

Destaca por contar con el panel de control más popular: cPanel, con el cual podrás gestionar tu sitio completamente, lo que incluye instalar un CMS, crear bases de datos, usuarios y otras tareas. Además, con tu cuenta gratuita podrás tener hasta un máximo de dos páginas web.

Características principales

Idioma: Inglés

Espacio web: 1Gb

Transferencia mensual: 10Gb

Sitios web: 2

Panel de control: cPanel

Soporte: No



Un pequeño detalle de 000WebHost: tu sitio estará en línea 23 horas al día.

Award Space

Award Space **ofrece servicio de alojamiento gratuito desde hace más de 15 años** así que sin dudas sabe lo que hace.

A pesar de ser un servicio sin costo, la empresa busca brindar lo mejor a sus clientes, por eso todos sus planes incluido el gratuito incluyen discos SSD, por lo que tu página podrá cargar muy rápido.



Características principales

Idioma: Inglés

Espacio web: 1Gb

Transferencia mensual: 5Gb

Sitios web: 1

Panel de control: Propio

Soporte: Sí (limitado)



X10Hosting

Por lo general un hosting gratuito es un alojamiento muy limitado, pero esto no ocurre con X10Hosting, **un proveedor que ofrece espacio web y transferencia mensual de datos ilimitados**.

Incluye cPanel desde el cual podrás instalar WordPress y otros gestores de contenidos en pocos pasos.

Características principales

Idioma: Inglés

Espacio web: Ilimitado

Transferencia mensual: ilimitado

Sitios web: 2

Panel de control: cPanel

Soporte: Sí





5GBFree

En este proveedor gratuito puedes tener tu sitio web con la comodidad de disponer de 5Gb de espacio en disco y nada menos que 20 Gb de transferencia mensual, lo que implica que podrás recibir miles de visitantes.

Al crear tu cuenta **puedes alojar un máximo de tres páginas web diferentes** y cada una de ellas con su base de datos independiente.

No cuenta con soporte personalizado, pero tiene un foro donde puedes recibir y brindar ayuda a otros usuarios de este servicio de hosting gratuito.

Características principales

Idioma: Inglés

Espacio web: 5Gb

Transferencia mensual: 20Gb

Sitios web: 1

Panel de control: cPanel

Soporte: No

5GBfree

FreeHostia

FreeHostia es una empresa estadounidense bastante conocida que **ofrece un modesto plan gratuito**

Características principales

Idioma: Inglés

Espacio web: 250mb

Transferencia mensual: 6Gb

Sitios web: 1

Panel de control: Propio

Soporte: Sí

de alojamiento web. Por ejemplo, el espacio se limita a 250Mb por lo que solo puede usarse con proyectos pequeños.

Sin embargo, si dejamos de lado el espacio, tiene otras características que lo convierten en una buena opción, hablamos de configurar hasta 5 dominios, instalar CMS con un solo clic, y tener una base de datos MySQL.



RunHosting

Con RunHosting tendrás **tu propio sitio web en línea sin pagar absolutamente nada**, pero con la limitación de 1Gb de espacio en disco y 5Gb de tráfico mensual.

Características principales

Idioma: Inglés

Espacio web: 1Gb

Transferencia mensual: 5Gb

Sitios web: 1

Panel de control: Propio

Soporte: Sí (limitado)





Free Hosting

A diferencia de las páginas mencionadas hasta ahora, Free Hosting es menos conocida en el mercado hispano, pues es una empresa checa, pero esto no quiere decir que no merezca una oportunidad.

Por mencionarte algunos aspectos que la convierten en una buena opción tenemos que permite crear una página web con su respectiva base de datos y **no tiene límites de transferencia mensual** de datos.

FREEHOSTING

Características principales

Idioma: Inglés

Espacio web: 10Gb

Transferencia mensual: Ilimitada

Sitios web: 1

Panel de control: cPanel

Soporte: No

Características principales

Idioma: Inglés

Espacio web: 1Gb

Transferencia mensual: 5Gb

Sitios web: 1

Panel de control: cPanel

Soporte: Sí (limitado)



Batcave

Además de su

particular nombre, Batcave tiene lo que te hace falta para montar un proyecto web sencillo y ponerlo en línea sin necesidad de pagar ni un centavo. Como es de esperarse sus funcionalidades y opciones son limitadas pero **suficientes para un proyecto amateur**.



Dinahosting

Dinahosting es lo que podemos considerar un servidor de prueba, **no apto para proyectos grandes y mucho menos serios**. Lo decimos porque con su plan gratuito solo tendrás 10 Mb de almacenamiento, es decir, tu web no podrá pesar más que eso. Así que debe tratarse de un sitio muy pequeño.

Aunque es bastante limitado lo mencionamos porque te ayudará a experimentar y montar algún sitio de prueba además su interfaz está en español.

Características principales

Idioma: Español

Espacio web: 10Mb

Transferencia mensual: 1.5Gb

Sitios web: 1

Panel de control: Propio

Soporte: No

Estos son tan solo **10 sitios en los que puedes publicar tu sitio web y comenzar a tener tu presencia online**. Estamos seguros de que conoces otros proveedores de hosting gratuito.



GREENBYTES – “GYTES VERDES”, TECNOLOGÍA AL SERVICIO DEL AMBIENTE



Propósito:

Diseñar páginas web en equipo de 5 integrantes, enfocadas a empresas de reciclado y protección ambiental. Utilizando software de edición de código, que permitan el uso de herramientas de programación en HTML y CSS; para ser presentados en un foro de emprendimiento.



Instrucciones: Con los conocimientos que has adquirido a lo largo del submódulo, es momento de diseñar en equipo, un sitio web enfocado a una empresa de reciclado y/o protección ambiental. Desarrollando las páginas web que presenten el contenido multimedia necesario; no olvides aplicar los estilos suficientes para dar un atractivo visual a las páginas y finalmente llevar el sitio web a un servidor web.

¿Qué debemos hacer?

Organizados en equipo, es necesario realizar una investigación de las organizaciones recicladoras de desechos tecnológicos en el país. Considera el siguiente enlace:

<https://mxcity.mx/2018/10/5-centros-de-reciclaje-de-basura-electronica-en-la-ciudad-de-mexico/>



Consideren la información, los elementos multimedia y el diseño de las páginas y el sitio web.



Seguidamente utiliza el software proporcionado o el que consideres pertinente para crear las páginas web.





Es necesario considerar la estructura básica en `<html>`, además de considerar las etiquetas e instrucciones para organizar la multimedia. Utilizar las estructuras para diseñar un menú de navegación en el sitio web.



No olvides aplicar `CSS` que les permitirá presentar un estilo a las páginas, armonizando la presentación del sitio web.

En última instancia, consideren implementar un formulario de contacto a través del uso de `JS` que les permitirá interactuar en la página web.

Para finalmente, utilizar un hosting donde puedan alojar los documentos y el contenido multimedia para que funcione correctamente.



Es momento de diseñar, desarrollar las páginas web para crear el sitio web y enviarlo al hosting.

diviértete
APRENDIENDO



Práctica 7. Publicando Mi Página Web



SIGA

Instrucciones: Con el equipo de 5 integrantes formado, y tu Pagina Web que da solución a la situación didáctica, lista y totalmente funcional. Realiza:



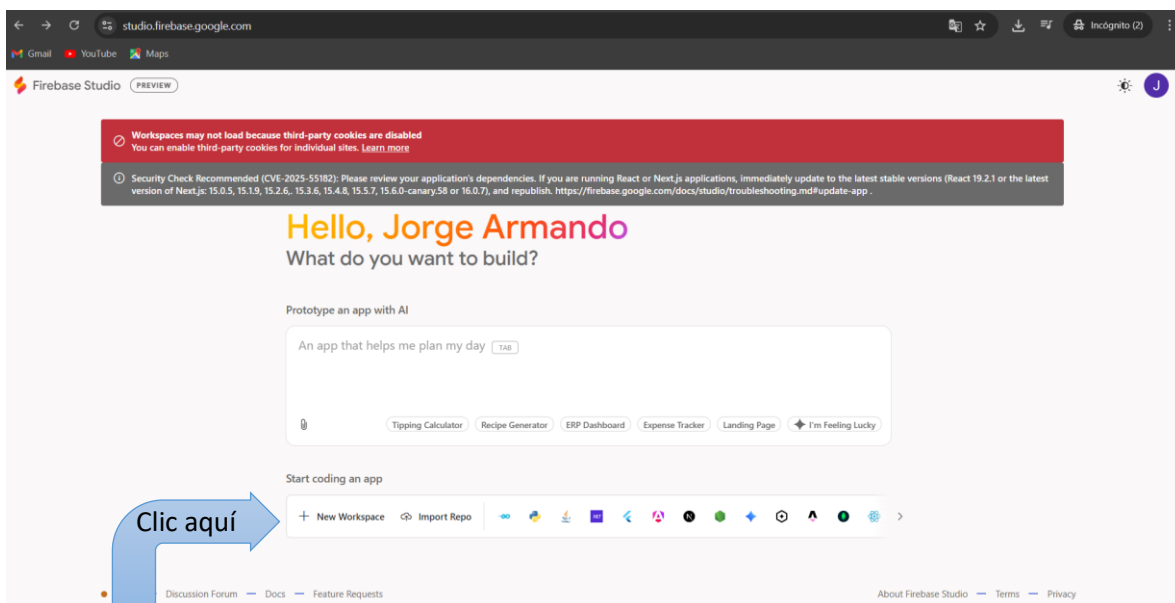
PRÁCTICA 7. "PUBLICANDO MI PAGINA WEB"

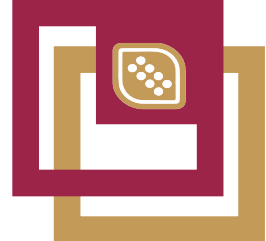


Instrucciones

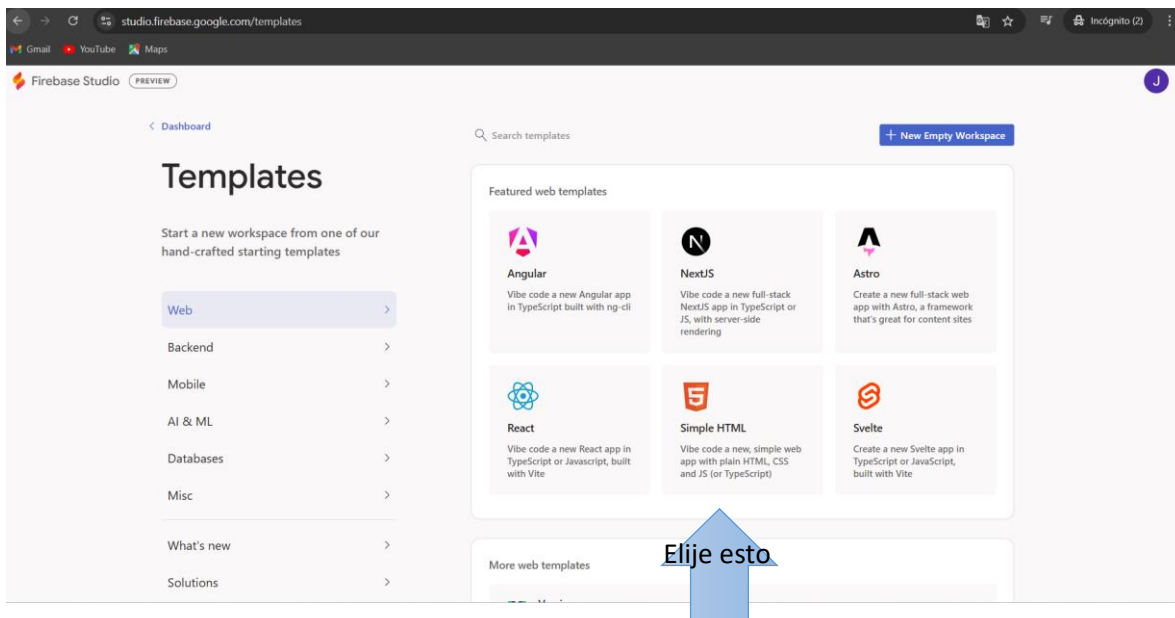
Accesa a la siguiente página de Hosting gratuito <https://studio.firebase.google.com/>

1. Da clic en New Workspace – Web / Simple HTML
2. Da clic en Registro Gratuito en la página puedes hacer uso de tu correo electrónico o el correo de Gmail del equipo.

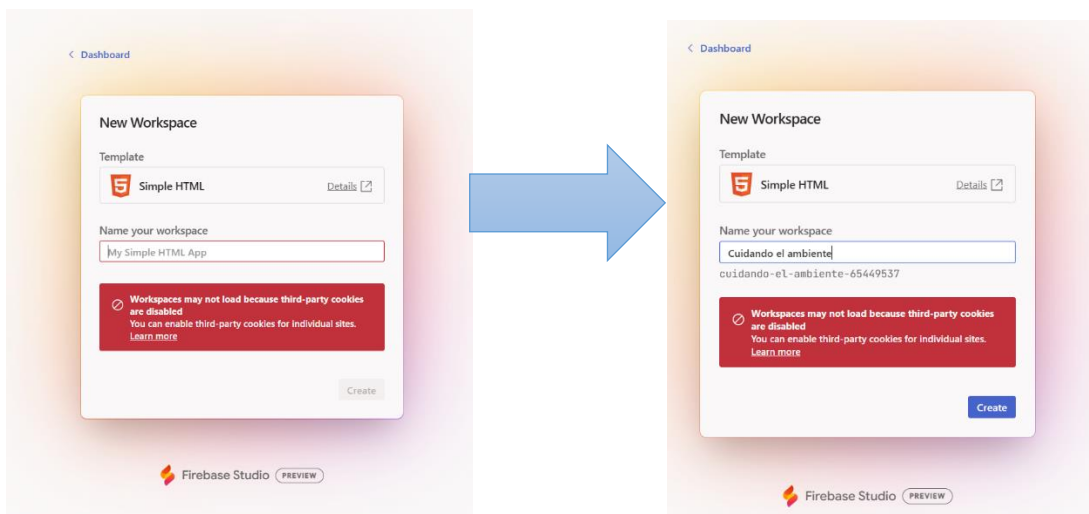


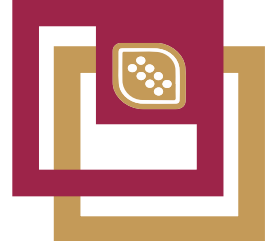


3. Puedes dar clic en el link para trabajar tu sitio web.

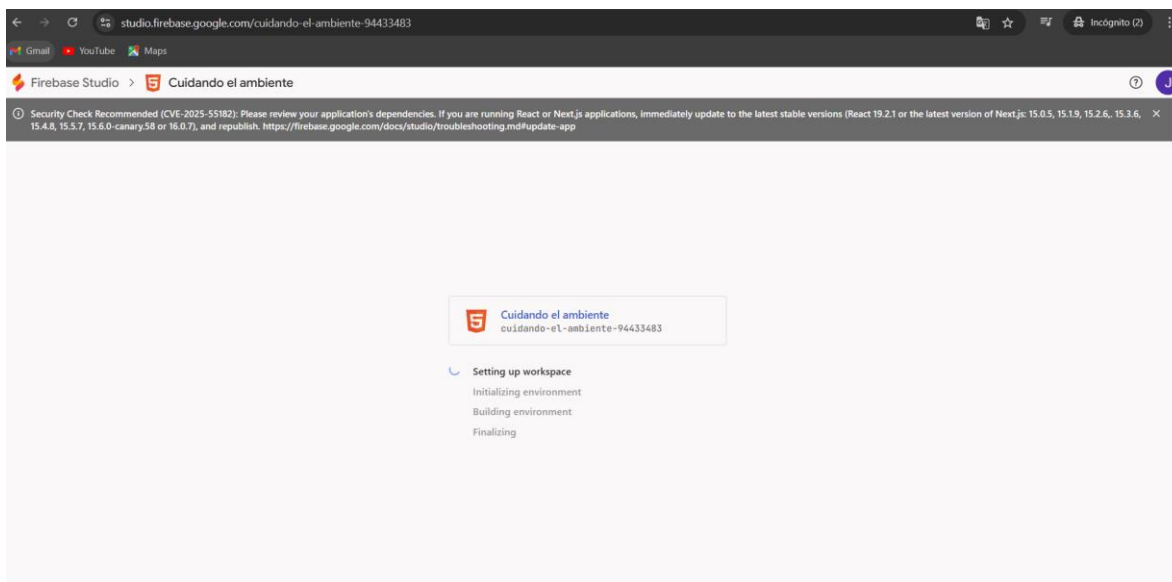


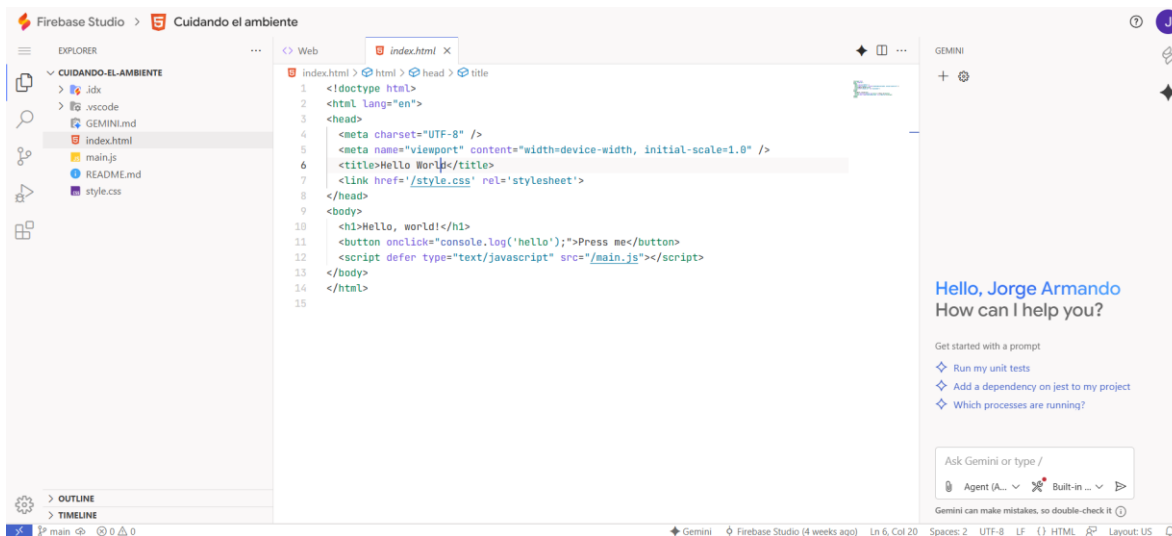
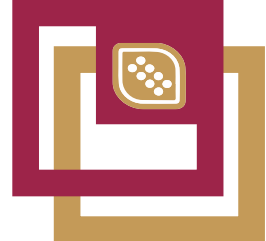
4. Dale un nombre





5. En este caso solo se verá la dirección y podrán trabajar en su sitio web.





SI QUIERES VER EL PROCESO PASO A PASO ACCESA AL CÓDIGO QR

6. ENTREGA EN EL FORMATO QUE TE SOLICITA TU DOCENTE LA DIRECCION URL PARA SU EVALUACIÓN FINAL.
7. ESTA ACTIVIDAD SE EVALUA CON EL INSTRUMENTO DE EVALUACIÓN A CONTINUACIÓN.





COLEGIO DE BACHILLERES DE TABASCO PLANTEL No ____

RÚBRICA DE EVALUACIÓN GREEN BYTES – “BYTES VERDES”, TECNOLOGÍA AL SERVICIO DEL AMBIENTE



Capacitación Desarrollo de Software	Módulo II. Base de Datos y Desarrollo Web	Submódulo 2. Programación Web
Situación didáctica. Green Bytes – Bytes Verdes, Tecnología al Servicio del Ambiente.		Nombre de los estudiantes:
Docente:		
Semestre. 4to Grupo.	Turno.	Fecha.
Competencia profesional		
CLBDS4. Construye sitios web creativos y funcionales mediante software de diseño web, para transmitir información a gran escala de forma responsable y empática en diversos contextos.		
Evidencia de aprendizaje. Documento electrónico / Sitio Web		

Instrucciones: Marque con una (X) para resaltar si cumplió con los criterios de la actividad. Sume los puntos para obtener la calificación.

INDICADORES	%	CRITERIOS			
		EXCELENTE	BUENO	SUFICIENTE	INSUFICIENTE
Sobriedad, homogeneidad y contenido.	20%	Presenta una página web por cada uno de los integrantes del equipo. Además, que el contenido es acorde a la organización investigada y presenta sobriedad todas las páginas web. 20	Presenta las páginas web de al menos el 80% de los integrantes del equipo. Además, que el contenido es acorde a la organización investigada y presenta sobriedad todas las páginas web. 15	Presenta las páginas web de al menos el 50% de los integrantes del equipo. Además, que el contenido es acorde a la organización investigada y presenta sobriedad todas las páginas web. 10	Presenta las páginas web de al menos el 30% de los integrantes del equipo. Además, que el contenido es acorde a la organización investigada y presenta sobriedad todas las páginas web. 5
HTML y multimedia insertada	30%	Utiliza correctamente la estructura de la página web y presenta 10 o más etiquetas HTML para insertar multimedia, aplicar formato y organizar el contenido de la página web. 30	Utiliza correctamente la estructura de la página web y presenta entre 7 y 9 etiquetas HTML para insertar multimedia, aplicar formato y organizar el contenido de la página web. 25	Utiliza correctamente la estructura de la página web y presenta entre 3 y 6 etiquetas HTML para insertar multimedia, aplicar formato y organizar el contenido de la página web. 15	Utiliza correctamente la estructura de la página web y presenta menos de 3 etiquetas HTML para insertar multimedia, aplicar formato y organizar el contenido de la página web. 5





INDICADORES	%	CRITERIOS			
		EXCELENTE	BUENO	SUFICIENTE	INSUFICIENTE
Uso de CSS	20%	Aplica correctamente el código en cascada a todas las páginas web presentadas. 20	Aplica correctamente el código en cascada al 80% de las páginas web presentadas. 15	Aplica correctamente el código en cascada al 50% de las páginas web presentadas. 10	Aplica correctamente el código en cascada a al menos el 30% de las páginas web presentadas. 5
Interacción y comunicación	20%	Presenta un menú por el cual es posible moverse entre todas las páginas, además de utilizar código en JavaScript. 20	Presenta un menú por el cual es posible moverse en al menos el 80% de las páginas, además de utilizar código en JavaScript. 15	Presenta un menú por el cual es posible moverse en al menos el 50% de las páginas, además de utilizar código en JavaScript. 10	Presenta un menú por el cual es posible moverse a al menos el 30% de las páginas y no utiliza código en JavaScript. 5
Publicación y entrega	10%	Publica la página web en un hosting y entrega la URL junto con la conclusión de su participación (una argumentación de su experiencia). 10	Publica la página web en un hosting y entrega la URL pero no presenta la conclusión de su participación (una argumentación de su experiencia). 8	Publica la página web en un hosting pero no entrega la URL ni la conclusión de su participación (una argumentación de su experiencia). 5	No publica la página web en un hosting pero entrega el sitio local junto con la conclusión de su participación (una argumentación de su experiencia). 3
Total	100%	100	83	50	23



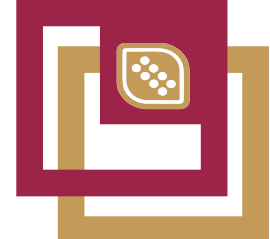
Firma del evaluador. _____



Conflicto cognitivo

¿Qué lenguajes son necesarios para la construcción de páginas y sitios web?
 ¿Cuál es la diferencia entre un sitio web y página web?
 Menciona 10 software útiles para el diseño y desarrollo de páginas y sitios web.
 ¿En dónde se almacenan las páginas web?





Anexos

HIMNO DEL COBATAB



¡Oh!, Colegio de Bachilleres, impetuosa y querida
institución casa fiel del conocimiento,
hoy te canto este himno con amor,
eres rayo de esperanza del mañana,
eres la voz de la verdad.

¡Oh!, Colegio de Bachilleres, eres
luz en medio de la oscuridad.
//Colegio de bachilleres conducta
clara y firme decisión
Colegio de bachilleres tu misión
para siempre es ser mejor.//

En Tabasco se ha sembrado
la semilla que un día germinara,
el impulso de la vida modernista
en progreso de toda la sociedad,
es tu memorable historia gran
orgullo para toda la región,
educación que genera cambio,
ejemplo digno en cada generación.



PORRA INSTITUCIONAL

¡Somos!
¡Somos!

Jóvenes Bachilleres
Jóvenes Bachilleres

Con Valor y Lealtad
De Norte a Sur
De Este a Oeste

Somos líderes Bachilleres del Sureste
COBATAB Unido, COBATAB Fortalecido.
Este encuentro lo gano porque lo gano
Como dijo el Peje, me canso ganso.

Jóvenes Bachilleres
Jóvenes Bachilleres

¡Somos!
¡Somos!

Jóvenes Bachilleres
Jóvenes Bachilleres

COBATAB Unido, COBATAB Fortalecido.

¡Somos!
¡Somos!



COBACHITO



Colegio de Bachilleres,
 Está de fiesta señores
 Pues todos sus estudiantes
 Hoy celebran con honores.

Que ya llegó la alegría
 Es hora de motivar
 Bailemos con algarabía
 Cobachito nos guiará.

Allá por el acahual
 En los ríos de Tabasco
 Aconchado en unas ramas
 O nadando sin parar.

Un manatí se ha ganado
 El cariño de la gente
 Cobachito le han llamado
 Y no para de bailar.

Cobachito, con él vamos a ganar
 Cobachito, eres espectacular
 Cobachito, respetamos tu hábitat
 Cobachito, mascota del COBATAB.

Mientras la orquesta se escucha
 Y la porra se emociona
 Los jóvenes bachilleres
 A una voz ovacionan.

Con orgullo representan
 A una gran institución
 COBATAB está presente
 Y Cobachito ya llegó.

Cobachito...



