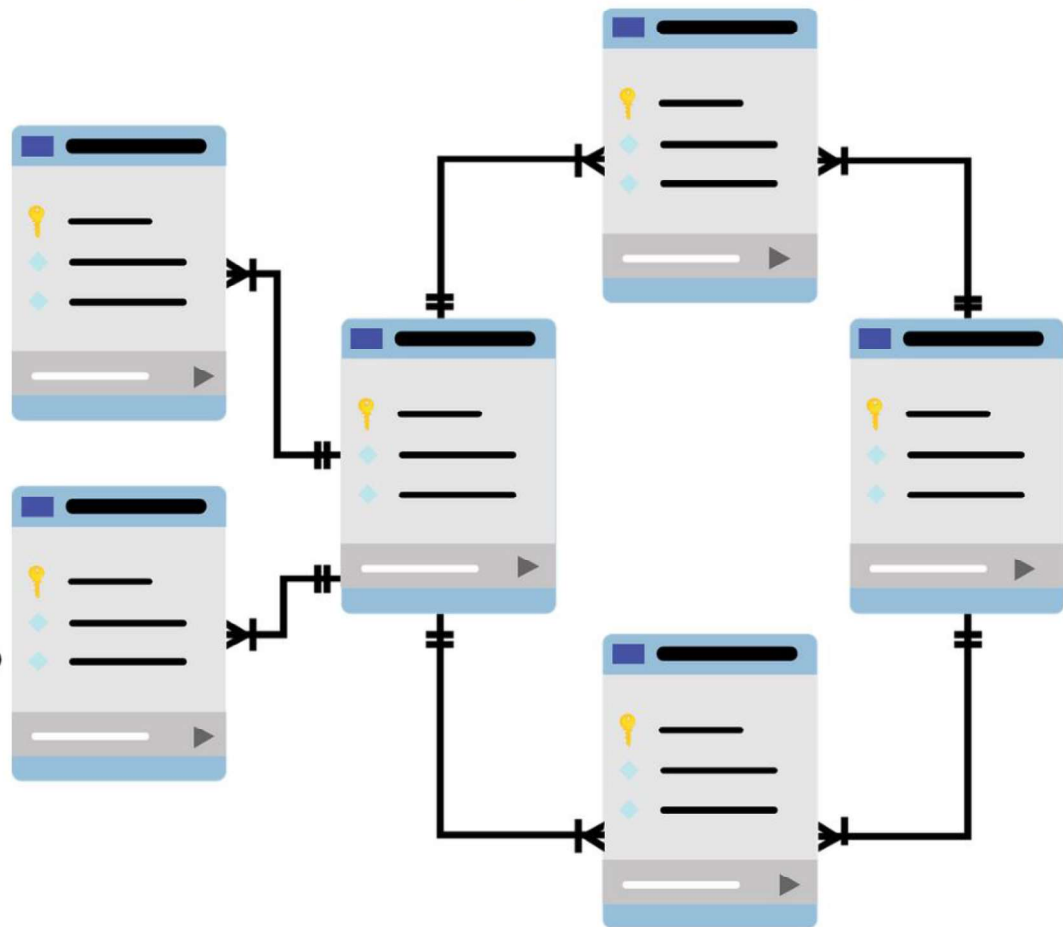


CARRERA TÉCNICA EN OFIMÁTICA

Módulo 4. Diseña y gestiona bases de datos ofimáticas

Quinto semestre



Submódulo 1

Dirección General de Educación Tecnológica Agropecuaria y Ciencias del Mar

Créditos

Desarrollo de Contenido

Christian Leonel González Rodríguez

Manuel Alejandro Moreno Raya

Tawny Nava García

Wilver Flores González

Revisión técnico – pedagógica y edición

Arit Furiati Orta

Itandehui García Flores

Judith Doris Bautista Velasco

México, 2021

Presentación

Actualmente los procesos de enseñanza y de aprendizaje se han diversificado en las formas, métodos y medios a través de los cuales se realizan para brindar una educación de calidad, por lo que cada día las instituciones educativas deben coadyuvar en dichos procesos a través de estrategias y acciones que favorezcan en los alumnos la adquisición de los aprendizajes tanto con la mediación de un docente de manera presencial como, en ocasiones singulares, a distancia.

Acorde con los principios de la Nueva Escuela Mexicana, los alumnos son sujetos activos y responsables de su propio aprendizaje, por lo que Dirección General de Educación Tecnológica Agropecuaria y Ciencias del Mar (**DGETAyCM**) pone a disposición de los estudiantes el presente material de apoyo que tiene el propósito de brindar elementos teóricos de los módulos profesionales de la **carreta técnica en ofimática**, así como el reforzamiento de estos a través de actividades de aprendizaje.

El material está organizado de modo progresivo para abordar los contenidos de la carrera Técnico en ofimática, en el presente material se trabajará el **Módulo IV “Diseña y gestiona bases de datos ofimáticas”**.

- Submódulo 1. Diseña bases de datos ofimáticas
- Submódulo 2. Gestiona información mediante el uso de sistemas manejadores de bases de datos ofimáticas

En este material se abordarán los contenidos del **submódulo 1**.

El primer apartado de cada lección denominado **“Contextualizando”** se muestra un primer acercamiento a los conceptos que se abordan, articulándolos con escenarios y situaciones de la vida cotidiana, con la intención de realizar asociaciones derivadas de los conocimientos previos de los estudiantes. En el apartado **“Vamos a aprender”** se integra información para analizar los conceptos y características de la temática. En la sección de **“Actividades de aprendizaje”** se proponen actividades para para asimilación de los **principales conceptos y características del tema**. En el apartado **“Autoevaluación”** se plantean una serie de indicadores de desempeño que buscan evaluar los aprendizajes e identificar los contenidos a reforzar. Finalmente, en la **sección “Para saber más”** se proporcionan recomendaciones para complementar los contenidos como videos y lecturas. En algunos casos, **se ha incorporado la sección “Ponlo en práctica”**, en se ofrecen algunos tutoriales o prácticas sugeridas para llevar a cabo con equipo de cómputo o dispositivo.

Deseamos que este material apoye la formación académica y sea una herramienta de utilidad en los procesos de aprendizaje para los estudiantes.

Índice

	Pág.
Submódulo 1. Diseña bases de datos ofimáticas	
Introducción a las bases de datos----- (Wilver Flores González)	7
Modelo relacional----- (Wilver Flores González)	13
Instalando nuestro entorno de trabajo: Laragon ----- (Christian Leonel González Rodríguez)	22
Crear, abrir y cerrar una base de datos----- (Tawny Nava García)	29
Diseño de tablas de datos ----- (Tawny Nava García)	42
Edición de estructura de tabla de datos----- (Manuel Alejandro Moreno Raya)	53

Estructura didáctica

Este material está dividido en submódulos y a lo largo de cada uno de ellos encontrarás diferentes secciones las cuales te facilitarán el abordaje de cada contenido.

En esta sección se delimitarán conceptos y características del tema a revisar, así como articulación de los contenidos con tus conocimientos previos relacionados con el tema y la relevancia de éstos en tu formación profesional/académica.

Emplearás los contenidos revisados para asimilar los principales conceptos y promover el desarrollo de las competencias profesionales.


En esta sección encontrarás información para analizar los conceptos y características del tema con énfasis en las competencias profesionales.

Evaluarás tus aprendizajes sobre los temas abordados e identificarás los contenidos que debes reforzar.

En este apartado se te proporcionan recomendaciones para profundizar en los contenidos.

Adicionalmente, en algunas lecciones podrás encontrar tutoriales o practicas sugeridas para que practiques lo que aprendiste, en algún dispositivo electrónico o equipo de cómputo.

Contextualizando



o precipita con esas grandes tormentas. ¿E otras cosas se pueden descargar o subir? ¿ el agua se almacena en las nubes?

¡Vamos a aprender!

En la actualidad has escuchado mencionar el *computación en la nube*, o has oído decir "col en la nube", "subelo en la nube", pero ¿sabes la nube?

La *nube* es un modelo de soporte tecnoló brinda acceso a un conjunto de recursos y informáticos compartidos, por ejemplo servidores, almacenamiento, aplicació servicios.

La fa
lo
La
in
us
difi
pu
En
7

Actividades de aprendizaje

Lee las siguientes oraciones y subraya la respuesta correcta.

1. Este tipo de nube se caracteriza por ofrecer estos servicios pueden ser gratuitos o pueden ser:
a) Encriptar b) Pública
2. Su uso es exclusivo de una persona o una ella son usuarios a los que la empresa les p
a) Híbrida b) Cifrar
3. Ofrece servicios donde se comparte inform música, vídeos, tutoriales, cocina, entre otro
a) Híbrida b) Pública
4. Ocultar el contenido de un mensaje a sim protegido y oculto.
a) Cifrar b) Pública
5. Si al conectarte a la red no te solicita una mejor es conectarse a redes que estén cifra tipo WPA2 (Acceso Wi-Fi protegido 2).
a) Cifrar b) Seguridad en la nube

Autoevaluación

Reflexiona y evalúa los conocimientos, habilidades y actitudes que adquiriste en esta lección.

Coloca una X en la columna que corresponda al desempeño que consideras que tienes para cada indicador.

Indicadores	Lo puedo hacer	Tengo dudas	Necesito trabajar más
Comprendo el concepto de computación en la nube.			
Conozco cuáles son las ventajas del uso de la computación en la nube.			
Entiendo cuáles son las desventajas de la			

Para saber más

- Capacitate para el empleo (2021). *Curso Fundamentos de cómputo en la nube*. Fundación Carlos Slim. <https://capacitateparaelempleo.org/pages.php?r=tema&tagID=8440>
- Surveillance. Self-defense (2018). *Qué debo saber sobre el cifrado*. <https://ssdloff.org/es/module/%C2%BEp/%C2%A7-es-el-cifrado>

15

Submódulo



Diseña bases de datos ofimáticas

Competencias profesionales

- Diseña bases de datos
- Instala software de bases de datos
- Crea y modifica estructura de bases de datos
- Gestiona información de bases de datos
- Relaciona bases de datos

Introducción a las bases de datos



¿Alguna vez te has preguntado qué hay detrás de todo el proceso de cobro de un supermercado, de una tienda de ropa o de un cine? ¿Cómo crees que lleven a cabo los registros para llevar un buen control y administración de todos los recursos?



Si tuvieras que organizar una fiesta y reunir el dinero de todos tus compañeros, hacer las compras, organizar los gastos... ¿cómo lo harías? ¿lo anotarías en un cuaderno o en un archivo electrónico? ¿si lo hicieras electrónico, qué tipo de archivo harías? ¿qué datos incluirías? ¿crees que sería fácil? ¿qué harías para facilitarte esta tarea?



¡Vamos a aprender!

Retomando los ejemplos anteriores, todos cuentan con algo muy importante en este tema de base de datos que es la **información** la cual cuenta con elementos que son los **datos**.

En la actualidad, las bases de datos se usan tan ampliamente que se pueden encontrar en organizaciones de todos los tamaños, desde grandes corporaciones y agencias gubernamentales, hasta pequeños negocios e incluso en hogares. Las actividades diarias con frecuencia te ponen en contacto con las bases de datos, ya sea directa o indirectamente.

En esta lección aprenderás los principales conceptos y elementos que conforman las bases de datos por lo que es importante que conozcas algunas definiciones.

Las **bases de datos** son colecciones de datos que contienen información relevante y sirven para almacenar grandes cantidades de información, que después puedas consultar de manera rápida. Puedes encontrar bases de datos en muchos lugares y situaciones de tu vida y uno de los propósitos principales es proporcionar a los usuarios una visión genérica de los datos.

A continuación se muestra un ejemplo de un caso sencillo.

En una Institución educativa se necesita tener la información de los estudiantes inscritos. Para ello se requiere de la siguiente información del estudiante:

- Matrícula
- Nombre
- Apellidos
- Correo electrónico
- Teléfono
- Carrera

Con estos datos, se organiza la información en una tabla y se le asigna un nombre, como se ve en la imagen.

El diagrama muestra una tabla con el título "Estudiantes". Las columnas están encabezadas por "Matrícula", "Nombre", "Apellidos", "Carrera", "Correo Electrónico" y "Teléfono". Las filas de datos son:

Matrícula	Nombre	Apellidos	Carrera	Correo Electrónico	Teléfono
2345561	José Luis	Pérez Jarquín	Tec. Agropecuario	luis@gmail.com	5543828923
2345543	Jacqueline	González Rodríguez	Tec. En Ofimática	jacky@gmail.com	5523423434
2342356	Andrea	Morales Torres	Tec. Agropecuario	andy@outlook.com	5552324223
2340988	Carlos	García Medrano	Tec. En Ofimática	karlo@hotmail.com	5552322312

Las etiquetas de diagrama indican: "Tabla" apunta al encabezado de la tabla; "Campo" apunta a una celda de encabezado; "Registro" apunta a una fila completa; "Dato" apunta a un valor dentro de una celda.

En la imagen anterior se muestra la forma de organizar los datos de acuerdo con la información en tablas, de esta manera la información se encuentra más ordenada.

Ahora se explica cada uno de los elementos que se muestran en la base de datos.

Tabla. Se utiliza para organizar y presentar información, se componen de filas y columnas (celdas) que se pueden rellenar con textos y gráficos.

Campo. Es el nombre que se le da a cada una de las columnas que forman la tabla. Contienen datos diferentes a los de otros campos. En el ejemplo anterior, un campo contendrá un tipo de datos único, como una dirección, o un número de teléfono, un nombre, etcétera.

Registro. Es cada una de las filas en que se divide la tabla, también es llamado fila o tupla y representa un objeto único de datos, en pocas palabras un registro es un conjunto de campos que contienen los datos que pertenecen a una misma entidad.

Dato. Es una representación simbólica (numérica, alfabética, algorítmica, espacial, etc.) de un atributo o variable cuantitativa o cualitativa. Los datos describen hechos empíricos, sucesos y entidades.

Ahora seguramente te preguntarás ¿y las tablas dónde o cómo las creo?

Para ello nos adentraremos en el tema de los **Sistemas Gestores de Base de Datos**, que es el Software que te ayudará a realizar tu base de datos organizando la información en tablas.

Un Sistema Gestor de Base de Datos (**SGBD**) es un conjunto de programas o software que permiten el almacenamiento, modificación y extracción de la información.

El objetivo principal de un **SGBD** es proporcionar una forma de almacenar y recuperar la información de una base de datos de manera que sea, tanto práctica como eficiente (Silberschatz et al., 2001).

Como se menciona anteriormente, los puedes acceder a la información usando este tipo de software, recuperar la información mediante en lenguaje específico de estos SGBD y generar informes.

Algunos ejemplos de SGBS son:



Ahora vas a conocer las características generales de algunos SGBD.

MySQL

Es un sistema de gestión de bases de datos relacional desarrollado bajo licencia dual GPL/Licencia comercial por Oracle Corporation y está considerado como Sistema Gestor de Base Datos open source, es decir de instalación libre. No necesita alguna clave de activación.

MySQL es el SGBD de algunos sitios web y aplicaciones basadas como: Airbnb, Uber, LinkedIn, Facebook, Twitter y YouTube.

MongoDB

MongoDB es una base de datos de documentos que ofrece una gran capacidad de adaptación y respuesta en rendimiento de la información además de ser mucho más flexible, también cuenta con un modelo de consultas y una gran ordenación de datos avanzado (MongoDB, 2021).

SQLite

SQLite es una herramienta de software libre (software que no tiene alguna licencia de uso), que permite almacenar información en dispositivos implantados de una forma sencilla, eficaz, potente, rápida y en equipos con pocas capacidades de hardware, como puede ser una tableta o un teléfono celular (Rómmel, F. 2020).

Postgres

PostgreSQL es un sistema gestor de bases de datos relacionales, está orientado a objetos, es decir, todos los elementos de la base de datos van a poder tratarse como objetos.

Es multiplataforma lo que significa que puede ser instalado en Microsoft Windows, GNU/Linux, MacOS.

En la práctica, los lenguajes de definición y manipulación de datos no son dos lenguajes separados; en su lugar simplemente forman partes de un único lenguaje de bases de datos, es el SQL (Structured Query Language) en español es un Lenguaje de Consultas Estructurado (Silberschatz et al., 2001).

Este lenguaje es un lenguaje de programación diseñado para actualizar, obtener, y calcular información en bases de datos.

Referencias

- Silberschatz, A., Korth, H. F., & Sudarshan, S. (2001). Fundamento de Base de Datos (4.a ed.). Mcgraw-Hill College.
- Gonzalez, G.J. (2018). Que es Postgres, retomado el día 01 de julio de 2021 de <https://openwebinars.net/blog/que-es-postgresql/>
- Rómmel, F(2020), SQLite: La Base de Datos Embebida. SoftwareGurú, Herramientas y Tecnologías (17). Retomado el 01 de julio de 2021 de <https://sg.com.mx/revista/17/sqlite-la-base-datos-embebida>
- MongoDB(2021) Retomado el día 01 de julio de 2021 de <https://www.mongodb.com/es/what-is-mongodb>



Actividad de aprendizaje

Identifica la cantidad de Campos, Datos, Registros y Tablas que existen en la siguiente tabla.

CODIGO	DEPENDENCIA	CÉDULA	NOMBRE	APELLIDO	EDAD
230087003	REPUESTOS	11233685	MARIA	FERNANDEZ	46
230087004	ADMINISTRACION	12768399	PEDRO	GONZALES	38
230087005	ALMACEN	8905722	GERARDO	GIL	57
230087006	REPUESTOS	7622503	LUIS	HERNANDEZ	56
230087007	ALMACEN	12334778	FERNANDA	BRITO	39
230087008	ADMINISTRACION	15097882	ESTELA	MARIN	32
230087009	ADMINISTRACION	17028553	OMAIRA	RODRIGUEZ	29
230087010	MERCADEO	4078298	RICHARD	PEREZ	62
230087011	REPUESTOS	11978233	MANUEL	SEPULVEDA	47
230087012	MERCADEO	14092773	BLANCA	ARIAS	37
230087013	ALMACEN	6986001	TOMAS	BENITEZ	54

Campos=	Datos=	Registros=	Tablas=
---------	--------	------------	---------

Relaciona la definición con el concepto que corresponde.

Son las partes de una Tabla	Tabla
Este sistema es un gestor de bases de datos relacionales, está orientado a objetos, es decir, todos los elementos de nuestra base de datos van a poder tratarse como objetos.	Dato
Es un conjunto de programas o software que permiten el almacenamiento, modificación y extracción de la información.	Postgrest
Se utiliza para organizar y presentar información, se componen de filas y columnas (celdas) que se pueden rellenar con textos y gráficos.	Registro, Campo, Dato
Es una representación simbólica (numérica, alfabética, algorítmica, espacial, etc.) de un atributo o variable cuantitativa o cualitativa. Los datos describen hechos empíricos, sucesos y entidades.	Campo
Es el nombre que se le da a cada una de las columnas que forman la tabla. Contienen datos de diferente tipo a los de otros campos. En el ejemplo anterior, un campo contendrá un tipo de datos único, como una dirección, o un número de teléfono, un nombre, etc.	SGBD



Autoevaluación

Indicadores	Lo puedo hacer	Tengo dudas	Necesito trabajar más
Entiendo que es una base de datos			
Conozco los elementos de una tabla			
Soy capaz de Identificar claramente la diferencia entre un campo y un dato			
Identifico claramente lo que es un Software Gestor de Base de Datos			
Reconozco las diferencias de cada uno de los gestores de base de datos			
Reconozco la importancia del lenguaje de los SGBD para la realización de las bases de datos			

Modelo relacional



Contextualizando

Existen muchas tiendas que venden artículos de diferentes marcas, por ejemplo, en un supermercado ¿cuántas marcas de refresco, papel higiénico, detergente o yogur puedes observar? ¿te imaginas cómo se organiza la tienda para poder llevar un registro de cada uno de sus proveedores?

Además, cada uno de los productos tiene un precio diferente de compra y de venta por lo que la tienda debe llevar un control de ellos para exhibirlos al público.

También deben contar con algún sistema para identificar cuántos productos se venden de cada tipo y marca para que con base en ello puedan volver a surtir.

¡Es una gran cantidad de información! ¿cómo crees que se realizan todos estos procesos?



¡Vamos a aprender!

En la lección anterior se realizó una introducción a las bases de datos y seguramente te estarás preguntando ¿cómo hago una base de datos? ¿Qué elementos contiene una base de datos?

Antes de comenzar a diseñar el Modelo Relacional, lo primero que debes de hacer es **determinar el propósito de la base de datos.**

Es una buena idea anotar el propósito de la base de datos en un papel: su propósito, cómo espera usarla y quién la usará. Por ejemplo, para una base de datos pequeña para un negocio familiar, escriba algo como: "La base de datos de clientes es una lista con información de los clientes cuya finalidad es el envío de correo y la creación de informes". Si la base de datos es más compleja o la usan muchas personas, el propósito podría constar fácilmente de uno o varios párrafos, y debería incluir cuándo y cómo cada persona usará la base de datos. Tener ese resumen te ayudará a centrarse en tus objetivos cuando tomes decisiones (Microsoft, 2021).

Algunas definiciones básicas:

Un **modelo** es una representación gráfica de nuestra base de datos. A este modelo le llamaremos modelo relacional.

El **modelo relacional** es un diagrama que ayuda a entender los datos y observar cómo se relacionan entre ellos.

Un **modelo de datos** es una colección de herramientas gráficas que usualmente incluyen un tipo de diagrama y vocabulario especializado para describir la estructura de la base de datos. Este modelo representa una descripción de la estructura de la base de datos, incluidos los datos, las relaciones dentro de los datos, las restricciones sobre los datos y a veces semántica o significados de los datos (Catherine, 2009)

El modelo entidad-relación lo desarrolló P. P. Chen en 1976 para facilitar el diseño de bases de datos al permitir expresar las propiedades conceptuales de la base de datos en un esquema, es decir el funcionamiento de un negocio, empresa, escuela entre otros.

Vamos a adentrarnos en el modelo entidad-relación y conocer cómo se realiza utilizando este ejemplo para explicar los conceptos básicos.

En una universidad las facultades son dirigidas por un director es decir cada facultad tendrá su director; así la facultad de Derecho tendrá su director, la facultad de Medicina tendrá su director, etc. Pero a su vez, un director dirige una facultad, es decir un director no puede dirigir más de una facultad.

Cada facultad cuenta con una serie de docentes; estos docentes sólo pueden pertenecer a una facultad es decir "Si el Mtro. José López Méndez pertenece a la facultad de Derecho, no puede dar clases en la facultad de Medicina". Los docentes imparten varias asignaturas; y cada asignatura pueden ser impartidas por diferentes docentes.

Los estudiantes se inscriben a las asignaturas que corresponden al semestre que están cursando.

Para que comprendas la estructura del modelo entidad-relación debes conocer los principales elementos que lo conforman.

Una **entidad** es un elemento del mundo real, este puede ser cualquier tipo de sujeto, objeto, suceso o concepto sobre el que se recoge información, por ejemplo: coches, casas, empleados, clientes, empresas, oficios, diseños de productos, etc. Las entidades se representan gráficamente en el modelo, mediante rectángulos y su nombre debe aparecer en el interior.

Una entidad sólo puede aparecer una vez en el esquema conceptual.

FACULTAD

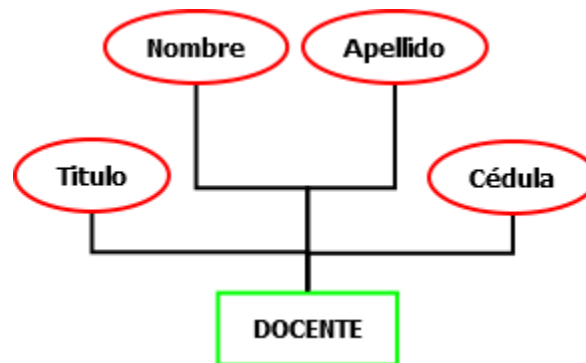
DIRECTOR

ESTUDIANTE

ASIGNATURA

DOCENTE

Un **atributo o campo** define o identifica las características de la entidad. Cada entidad contiene distintos atributos, que dan información sobre esta entidad. Por ejemplo, si asignara los atributos de la entidad “coche” algunos serian: número de chasis, matrícula, marca, modelo, por mencionar algunos. Gráficamente, se representan mediante óvalos que derivan de las entidades o relaciones a las que pertenecen.

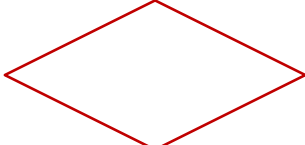


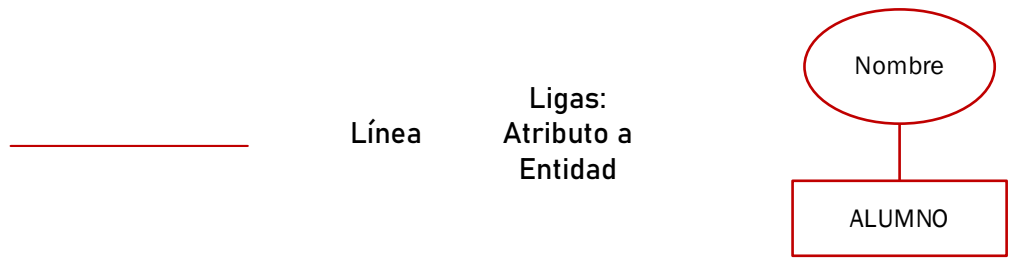
La **relación** es un vínculo que nos permite definir una dependencia entre varias entidades, es decir, nos permite exigir que varias entidades compartan ciertos atributos de forma indispensable.

Las relaciones se muestran en los diagramas como rombos, que se unen a las entidades mediante líneas.



A continuación, se presenta una tabla con los símbolos básicos para el desarrollo de los diagramas Entidad - Relación.

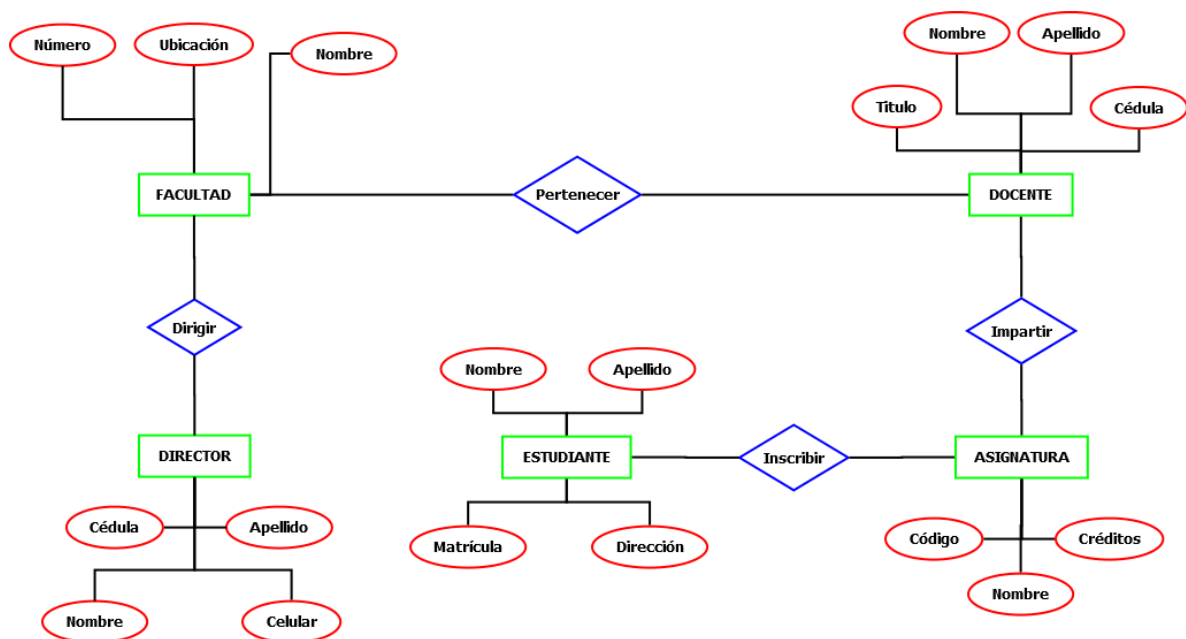
Símbolo	Nombre	Significado	Ejemplo
	Rectángulo	Entidad	
	Óvalo	Atributo	
	Diamante	Relación	



Ya que conoces los elementos del modelo relacional podrás elaborar uno. Tomando en cuenta el ejemplo de la Universidad y las relaciones que existen entre las diferentes figuras, podrás determinar qué elementos requieres conocer de cada uno para poder elaborar el modelo de entidad-relación:

- De las facultades necesitas conocer su nombre, ubicación y número de bloque.
- De los rectores se requiere el número de cédula profesional, nombre, apellidos y número celular.
- De los docentes se debe registrar el número de cédula profesional, nombre, apellidos y título de sus estudios profesionales.
- Cada asignatura debe identificarse con un código, nombre y número de créditos.
- De los estudiantes se debe registrar el número de matrícula, nombre, apellidos y dirección de residencia.

El modelo relacional del ejemplo anterior quedaría de la siguiente manera:



Hasta aquí hemos terminado con el diagrama, pero existe un punto muy importante en la relación, para ello continuaremos con el siguiente tema: cardinalidad.

La **cardinalidad** es el tipo de relaciones que se da entre entidades, es decir cómo se relacionan.

La correspondencia de **cardinalidades** o razón de cardinalidad expresa el número de entidades a las que otra entidad puede estar asociada vía un conjunto de relaciones. (Silberschatz, F. Korth, & S. Sudarshan, 2002).

Los tipos de relaciones que conocerás son:

- Relación uno a uno (1-1).
- Relación uno a varios(1-M).
- Relación varios a uno(M-1).
- Relación varios a varios(N-M).

Uno a uno (1 – 1). Una entidad en A se asocia con al menos una entidad en B, y una entidad en B se asocia con al menos una entidad en A. Por ejemplo: Una facultad puede ser dirigida por un solo director y un director solo puede dirigir a una Facultad.



Uno a varios (1 – M). Una entidad en A se asocia con cualquier número de entidades en B (ninguna o varias). Una entidad en B, sin embargo, se puede asociar con al menos una entidad en A. Por ejemplo: Una facultad cuenta con muchos docentes, pero un docente solo puede pertenecer a una facultad.



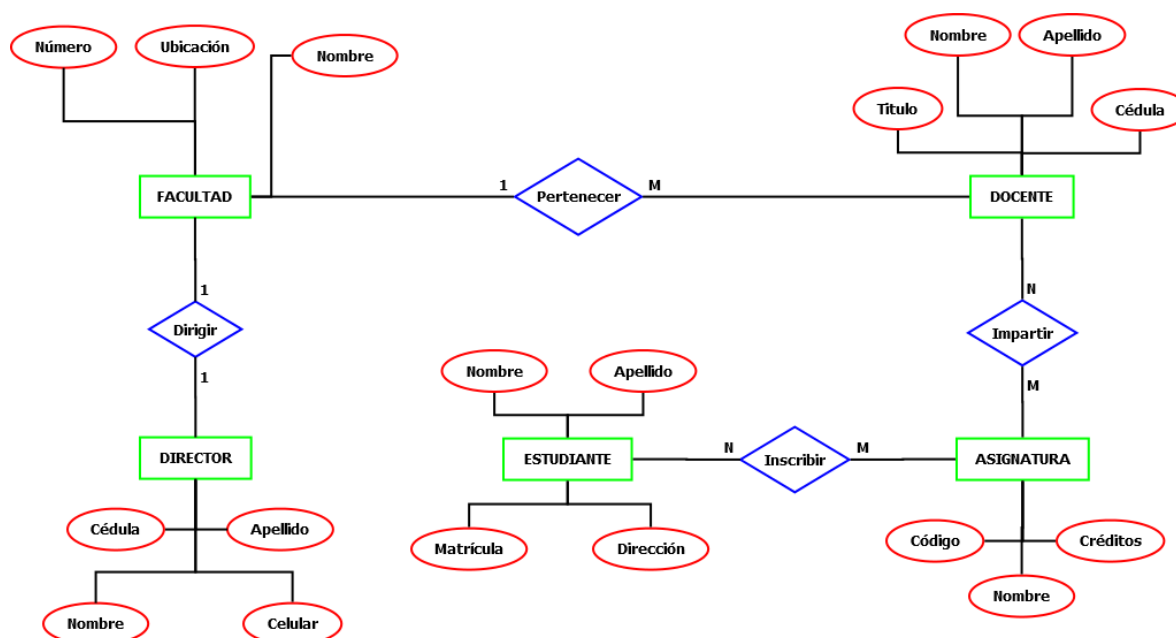
Varios a Uno (N – 1). Al menos una entidad A se asocia con una entidad B. Una entidad B se asocia se asocia con cualquier número de entidades de A. Por ejemplo: Un Docente solo puede pertenecer a una sola Facultad, sin embargo, una facultad tiene muchos docentes.



Varios a varios (N – M). Una entidad en A se asocia con cualquier número de entidades (ninguna o varias) en B, y una entidad en B se asocia con cualquier número de entidades (ninguna o varias) en A. Por ejemplo: Un docente puede impartir varias asignaturas, así mismo una asignatura puede ser impartida por varios docentes.



Al aplicar las relaciones de cardinalidad a tu proyecto, el modelo de entidad relación queda de la siguiente manera:



Para saber más

Para reforzar el conocimiento sobre el tema **relaciones de cardinalidad** revisa el siguiente video escaneando el código QR o dando clic al link.



<https://bit.ly/2TzGEXT>

Referencias

- Avilés, G. (2020). Base de Datos. <https://www.youtube.com/watch?v=wx7XohWT-w>
- Catherine M., R. (2009). Base de Datos. McGRAW-HILL Interamericana editores.
- Courcuera, P. (2016). Base de Datos: Modelo Relacional. Modelo Relacional.
- Microsoft (2021), Conceptos básicos del diseño de una base de datos, Disponible en <https://support.microsoft.com/es-es/office/conceptos-b%C3%A1sicos-del-dise%C3%B1o-de-una-base-de-datos-eb2159cf-1e30-401a-8084-bd4f9c9ca1f5>
- Silberschatz, A., F. Korth, H., & S. Sudarshan. (2002). Fundamentos de Base de Datos. Madrid: McGRAW-HILL/INTERAMERICANA DE ESPAÑA, S. A. U.



Actividad de aprendizaje

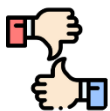
Lee la situación y realiza el modelo entidad-relación con las relaciones correspondientes entre las entidades.

En una biblioteca se requiere hacer una base de datos de los libros que se tienen, contemplando algunos datos como: título, ISBN, editorial y número de página y autor(es).

En la misma base de datos se desean registrar el préstamo de libros de los usuarios, el cual se lleva a cabo a través de un formato que contiene los siguientes datos: número de usuario, nombre, dirección, teléfono y correo electrónico. Además, se debe contemplar los datos del préstamo: libro, nombre del usuario, fecha de préstamo, fecha de devolución.

Relaciona las columnas según corresponda el enunciado.

Es un diagrama que ayuda a entender los datos y observar cómo se relacionan entre ellos.	El modelo relacional
Una entidad en A se asocia con cualquier número de entidades en B (ninguna o varias). Una entidad en B, sin embargo, se puede asociar con al menos una entidad en A.	Uno a uno
Es un vínculo que nos permite definir una dependencia entre varias entidades	Relación
Una entidad en A se asocia con al menos una entidad en B, y una entidad en B se asocia con al menos una entidad en A.	Muchos a muchos
Este elemento define o identifica las características de la entidad.	Cardinalidad
Una entidad en A se asocia con cualquier número de entidades (ninguna o varias) en B, y una entidad en B se asocia con cualquier número de entidades (ninguna o varias) en A.	Entidad
Es un elemento del mundo real, este puede ser cualquier tipo de sujeto, objeto, suceso o concepto sobre el que se recoge información.	Atributo o campo
Es el tipo de relaciones que se da entre entidades, es decir cómo se relacionan.	Uno a varios



Autoevaluación

Indicadores	Lo puedo hacer	Tengo dudas	Necesito trabajar más
Entiendo qué es el modelo relacional			
Comprendo para qué sirve el modelo de entidad-relación			
Conozco los elementos que debe tener el modelo relacional			
Soy capaz de identificar las entidades de un ejemplo concreto			
Sé qué son los atributos y cómo se relacionan con las entidades			
Comprendo la importancia de las relaciones del modelo relacional			
Conozco la simbología que debo utilizar al crear un modelo de entidad-relación			
Puedo explicar qué es la cardinalidad			
Comprendo los tipos de relaciones que se pueden dar en el modelo relacional.			

Instalando nuestro entorno de trabajo: Laragon



Contextualizando

¿Te imaginas cómo funciona internet? ¿Dónde se almacena toda la información de internet?



El internet es una red de redes, en donde se conectan miles de ordenadores ya sean a través de cables telefónicos, satélites, por cualquier medio como fibra óptica, coaxial u ondas electromagnéticas.

Estas conexiones son llevadas a través de servidores que funcionan como “grandes almacenes” que resguardan y dan “contestación” a cada una de las peticiones que realizas en el mundo de internet; por ejemplo, cuando accedes a tu correo electrónico de Gmail.com o Hotmail.com, esta petición es enviada a los servidores de dichas empresas, se autentican con tu contraseña y envían los correos electrónicos que tienes almacenados.

Por consiguiente, cualquier información que guardes en la Internet, se almacena en servidores, servidores descentralizados como la nube, e inclusive en la blockchain, considera por algunos como la web 3.0 y que puede marcar el futuro de la Internet.



¡Vamos a aprender!

Estas iniciando nuevo tema y necesitas poner manos a la obra en el desarrollo de una base de datos.

En las lecciones anteriores aprendiste qué es un Sistema Gestor de Base de datos y cómo diseñar tu base de datos de acuerdo con el Modelo Relacional, ahora debes instalar un sistema que te ayudará a crear la base de datos de manera local en tu computadora y así tener en orden la información deseada.

Por lo que, en esta lección aprenderás a instalar el entorno de trabajo denominado **Laragon**, que te ayudará a continuar con el desarrollo de las prácticas y sobre todo con tu aprendizaje.

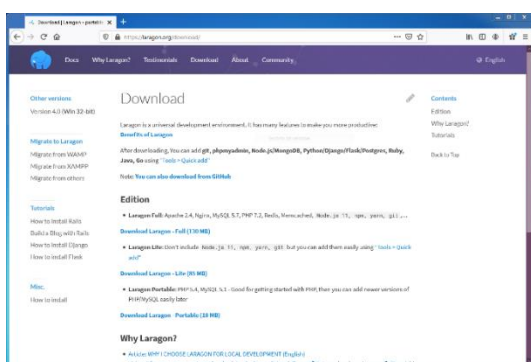
Laragon es un software que integra Apache como servidor web; que nos ayude a mostrar nuestras consultas en una página web, MySQL como servidor de bases de datos; que te

ayudará a crear tablas, crear consultas e insertar registros. Por su parte PHP como intérprete entre la página web y tu base de datos, ya que la información que insertes en tu base de datos, la podrás tener accesible directamente desde una página web.

Para instalar Laragon tienes que tomar en cuenta algunos requisitos mínimos de tu computadora, como son:

1. Sistema Operativo Windows 10 de preferencia y Windows 7 para entornos de 32bits.
2. Memoria RAM 2Gb mínimo
3. Espacio en disco duro disponible: 300Mb

Ahora aprenderás cómo instalar Laragon.

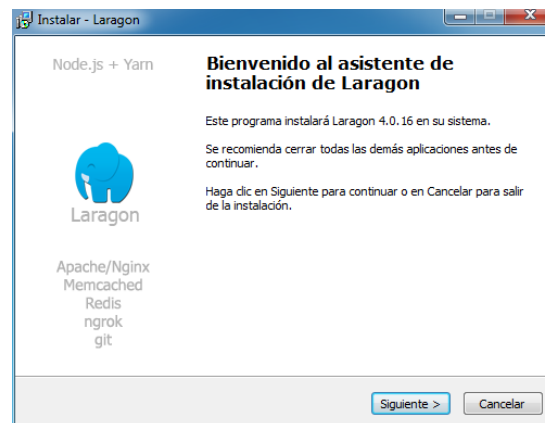


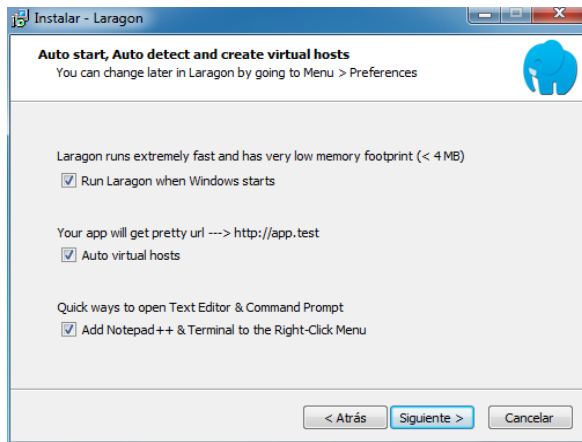
Entra al sitio web:
<https://laragon.org/download/>

Desde este sitio web, busca el archivo: “Download Laragon - Lite” que te permite iniciar la instalación. Una vez descargado lo vas a abrir.

Seleccionar el idioma “Español” e inicia el asistente.

Determina la ubicación en la que quieres que se instale. La ubicación por default es: C:\Laragon.

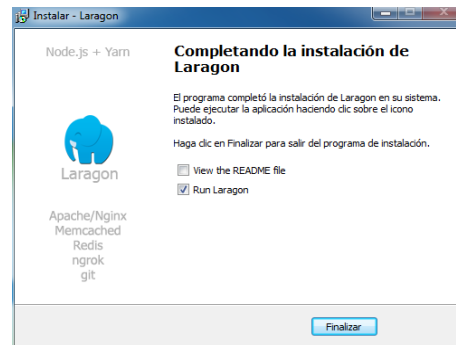




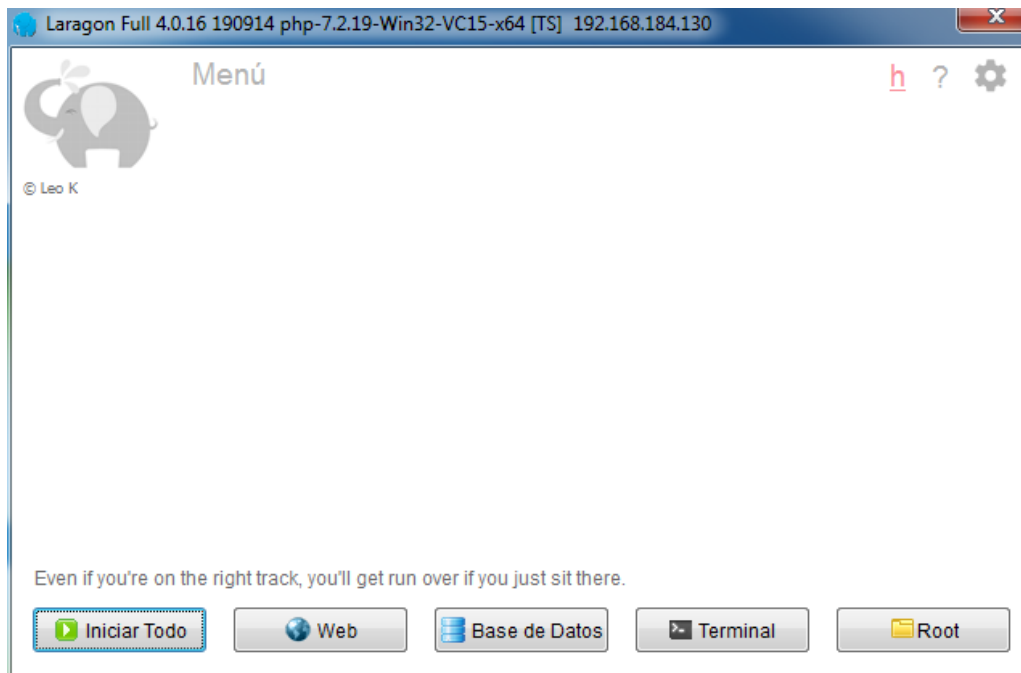
Selecciona y confirma las opciones que te muestra:

1. Iniciar automáticamente Laragon al encender nuestra computadora.
2. Permitir sitios web virtuales
3. Agregar comandos básicos a nuestra terminal del sistema.

Una vez terminada la instalación da clic en **Finalizar**.



Interfaz de Laragon

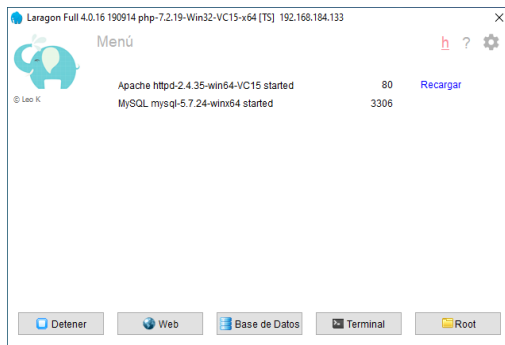
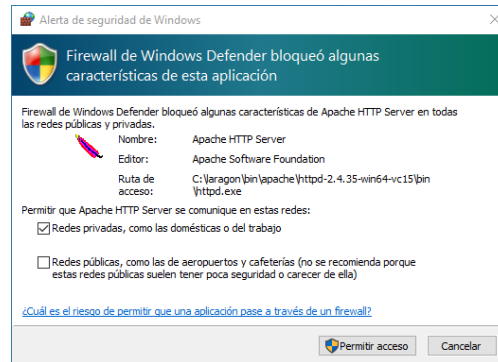


Cuando utilizas por primera vez Laragon, debes iniciar los servicios (Apache y MySQL). Da clic en **Iniciar todo**.

- **Web:** Te permite iniciar el navegador web, mostrando sitio web en blanco.
- **Base de Datos:** Te permite iniciar el motor de bases de datos.
- **Terminal:** Te da acceso a la línea de comandos de MySQL
- **Root:** Abre la carpeta donde se almacenan tus archivos del sitio web.
-

Al dar clic en botón **Iniciar Todo**, si Windows tiene habilitado el Firewall te notificará que el programa desea tener acceso. En este caso, Laragon necesita acceso a los puertos específicos para realizar las conexiones pertinentes.

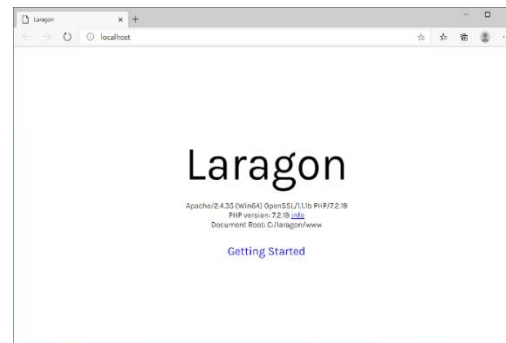
Da clic en **Permitir acceso**



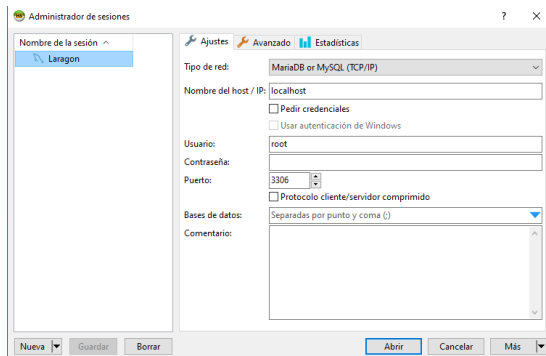
Observa la notificación de que los servidores se han cargado

1. Apache httpd
2. MySQL

Para probar que todo funcione correctamente da clic en **Web**



La ventana del explorador web mostrará el título del programa Laragon.



Para visualizar MySQL, da clic en **Bases de Datos** y te abrirá la ventana de conexión. Deja los valores por default y da clic en **Abrir**.



Actividad de aprendizaje

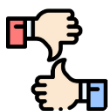
1. ¿Qué es Laragon?

2. ¿Qué puedes crear con Laragon?

3. ¿Qué acción lleva a cabo el botón root?

4. ¿En qué puerto corre Apache y MySQL?

5. Menciona 3 herramientas adicionales que trae Laragon para los servicios de MySQL y PHP.



Autoevaluación

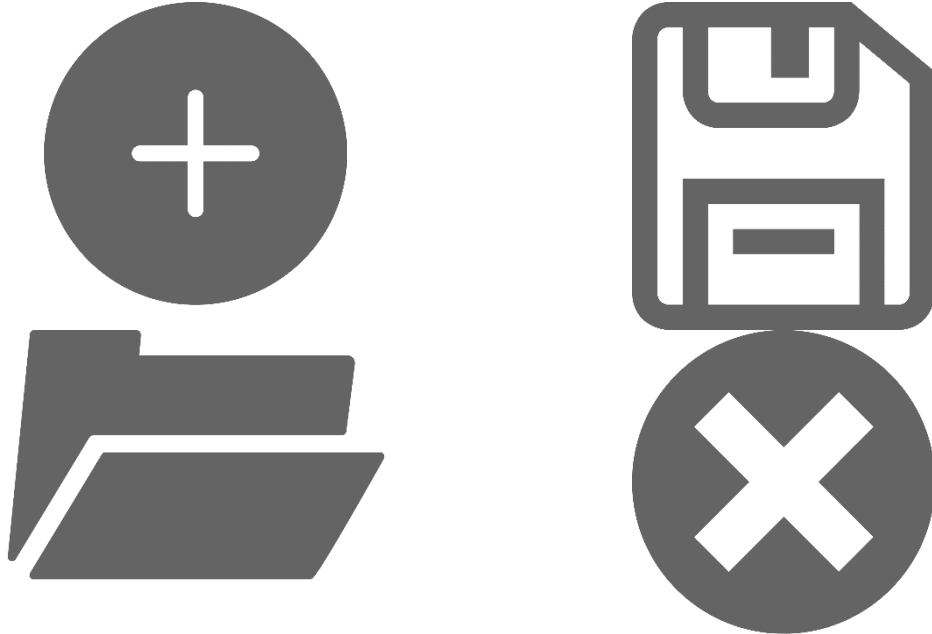
Indicadores	Lo puedo hacer	Tengo dudas	Necesito trabajar más
Entiendo qué es Laragon			
Comprendo para qué sirve Laragon			
Conozco el procedimiento para instalar Laragon en mi equipo de cómputo			
Sé cuáles son las características mínimas que debe tener mi equipo de cómputo para instalar Laragon			
Soy capaz de instalar Laragon en mi computadora.			

Crear, abrir y cerrar una base de datos



Contextualizando

Desde hace más de treinta años, el funcionamiento básico de las herramientas informáticas como los procesadores de texto, las hojas de cálculo y los presentadores de diapositivas, funcionan a partir de cuatro principios básicos que son: crear (nuevo), guardar, abrir y cerrar.



¿Reconoces los iconos? ¿Sabes para qué sirve cada uno? ¿Los has utilizado?



¡Vamos a aprender!

La opción crear, permite un nuevo tipo de archivo, que podrá ser utilizado posteriormente según el uso que convenga y la aplicación correspondiente para su utilización. Por otro lado, la opción abrir permite la apertura del archivo para agregar, modificar o borrar cierta información que el usuario considere pertinente en función del tiempo y momento en que esté utilizando dicha información. La opción cerrar, permite que la información quede a resguardo protegida en su integridad informática a sabiendas de que cuenta con la última versión del usuario con los cambios y adecuaciones necesarias en el uso que se les está dando.

La importancia de las acciones crear, abrir y cerrar, cuando se maneja una base de datos, cobra relevancia cuando analizas que forman parte de los comandos (instrucciones) básicos a emplear al hacer uso de un Sistema Gestor de Base de Datos (SGBD) como lo es MySQL a través de la aplicación Laragon. Esta importancia estriba en que precisamente, son esas acciones (crear, abrir y cerrar) las que te permitirán operar de forma asertiva la administración de una base de datos para los objetivos curriculares de este submódulo profesional.

Un DBMS (por sus siglas en inglés Data Base Management System), significa "Sistema Gestor de Bases de Datos" y permite la administración de bases de datos, como su nombre propiamente lo dice.

Una base de datos relacional se refiere a una base de datos que almacena datos en un formato estructurado, utilizando dentro de cada tabla (puede haber más de una tabla) filas y columnas.

NC	Nombre	Apaterno	Amaterno	Semestre	Grupo	Direccion	Telefono
20DTA001	Pedro	Zapata	Pérez	1	A	Ignacio Mendoza #45	777-135-XXXX
20DTA002	Juan	Arce	Fernández	3	B	Benito Juárez #110	777-149-XXXX
20DTA003	Laura	Molina	Vázquez	5	A	Av. Perseverancia #23	777-091-XXXX
20DTA004	Araceli	Gómez	Sánchez	3	B	José Ma. Morelos #101	777-234-XXXX
20DTA005	José	López	García	1	A	Revolución Agraria #54	777-395-XXXX
20DTA006	Julia	Márquez	Gonzalez	5	B	Miguel Hidalgo #77	777-108-XXXX

Esto facilita la localización y el acceso a valores específicos dentro de la base de datos. Es "relacional" porque los valores dentro de cada tabla están relacionados entre sí. La estructura relacional permite ejecutar consultas en varias tablas a la vez.



El SQL (por sus siglas en inglés Structured Query Language) es un lenguaje estándar (mundial) de consulta estructurado para el acceso a bases de datos.

El lenguaje SQL se utiliza para acceder y manipular datos en cualquier base de datos del mercado, como, por ejemplo, para las bases de datos MySQL, Oracle, DB2, SQL Server, Access.

El SQL se compone de sentencias, cada una con una utilidad diferente, por ejemplo:

- Creación de una base de datos (**CREATE DATABASE**)
- Creación de una tabla (**CREATE TABLE**)
- Creación de una vista (**CREATE VIEW**)
- Creación de un índice de una tabla (**CREATE INDEX**)
- Creación de procedimientos almacenados (**CREATE PROCEDURE**)

- Creación de disparadores (**CREATE TRIGGER**)
- Consultar los datos almacenados en una tabla (**SELECT**)
- Insertar datos en una tabla (**INSERT**)
- Modificar datos ya existentes en una tabla (**UPDATE**)
- Borrar datos almacenados en una tabla (**DELETE**)
- Dar permiso de acceso a los datos de una tabla (**GRANT**)
- Eliminar permisos de acceso a datos de una tabla (**REVOKE**)
- Finalizar la transacción de una sentencia SQL (**COMMIT**)
- Retroceder la transacción de una sentencia SQL (**ROLLBACK**).

Estas sentencias se pueden agrupar por funcionalidades:

1.- Lenguaje de definición de datos (DDL).

DDL está compuesto por sentencias para la creación (**CREATE**), modificación (**ALTER**) y borrado (**DROP**) de los componentes principales de una base de datos:

- base de datos (**DATABASE**)
- tablas (**TABLE**)
- vistas (**VIEW**)
- índices (**INDEX**)
- procedimientos almacenados (**PROCEDURE**)
- disparadores (**TRIGGER**).

2.- Lenguaje de manipulación de datos (DML).

DML está compuesto por sentencias que sirven para:

- consultar (**SELECT**)
- insertar (**INSERT**)
- modificar (**UPDATE**)
- borrar (**DELETE**)

3.- Lenguaje de control de datos (DCL).

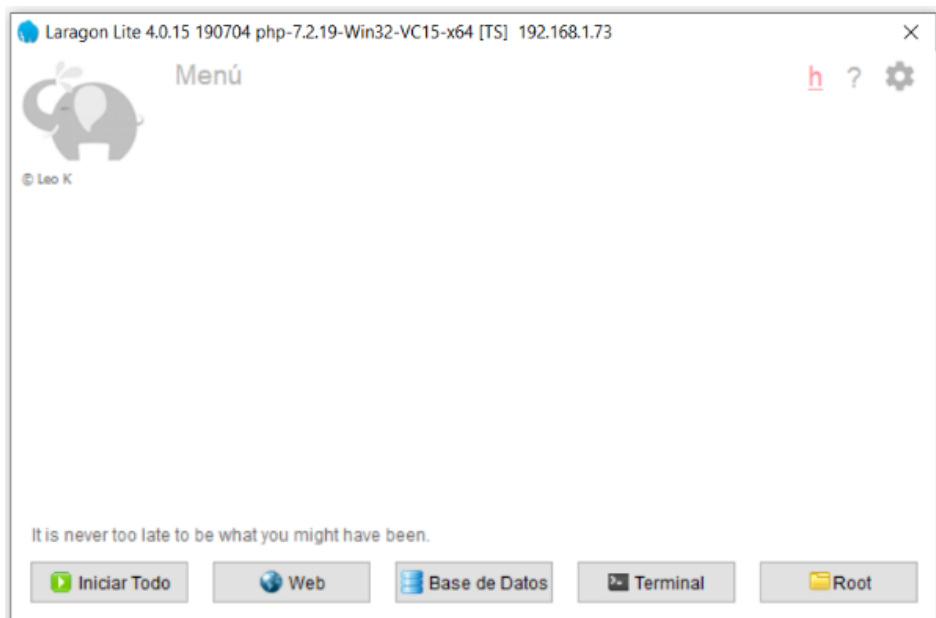
DCL está compuesto por sentencias SQL para controlar las funciones de administración:

- Confirmar la operación (**COMMIT**)
- Retroceder la operación (**ROLLBACK**)
- Dar permisos (**GRANT**)
- Quitar permisos (**REVOKE**)

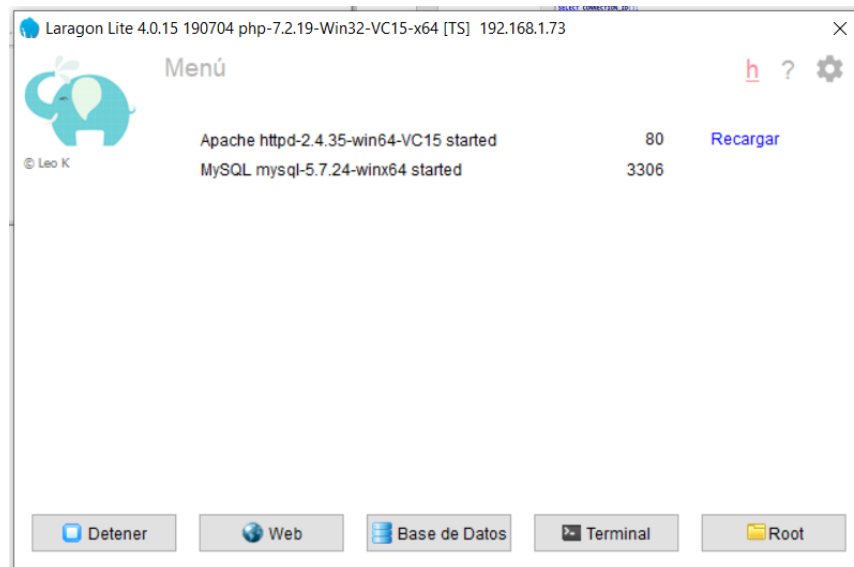
Hasta este momento, hemos revisado la parte teórica de los fundamentos, sentencias SQL y operaciones básicas dentro de una base de datos, procedamos a la parte práctica de las mismas. A continuación, se presentan ejemplos prácticos para que movilicemos el conocimiento adquirido hasta ahora.

Crear base datos en Laragon.

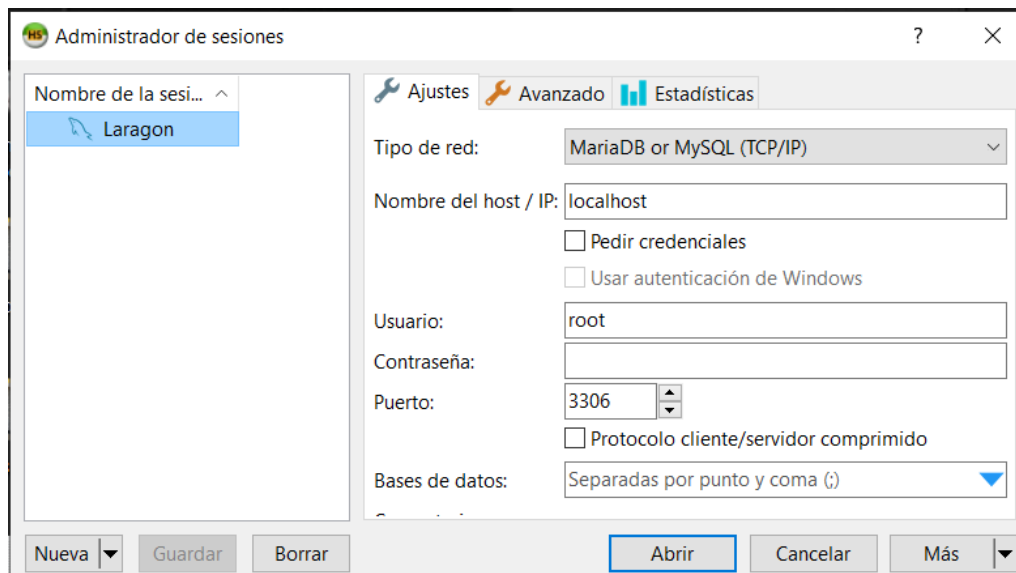
Ejecuta la herramienta Laragon, que previamente has instalado.



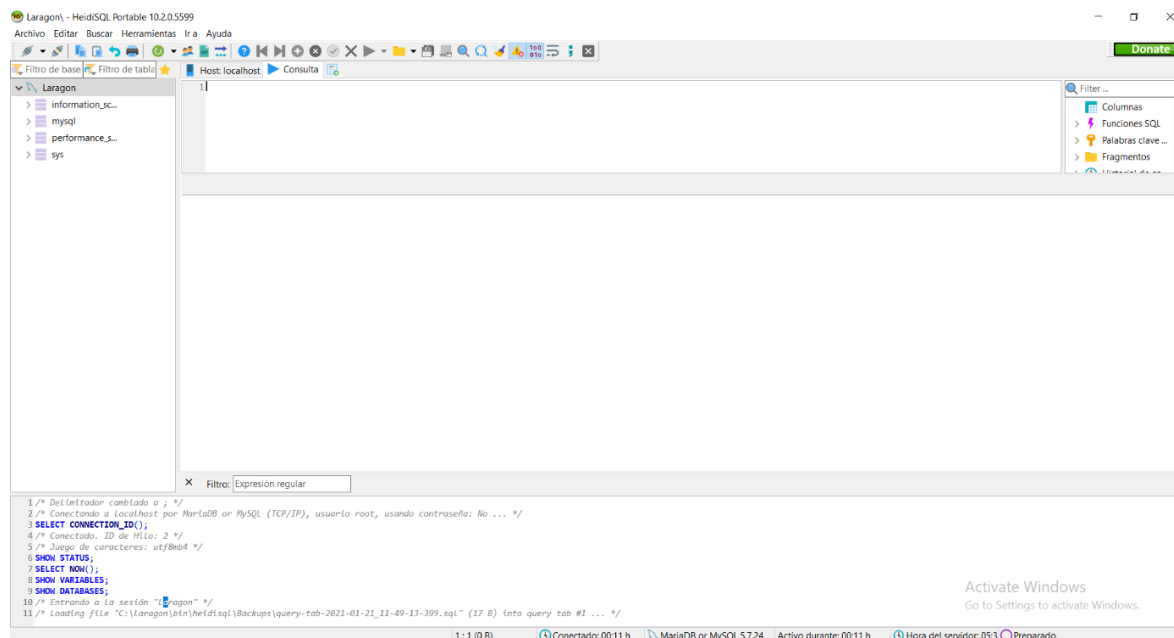
Inicializa los servicios de Apache y MySQL presionando en el botón **Iniciar Todo**.



Presiona el botón de **Base Datos** para comenzar con el proceso de crear una nueva base de datos, el cual despliega la siguiente pantalla:



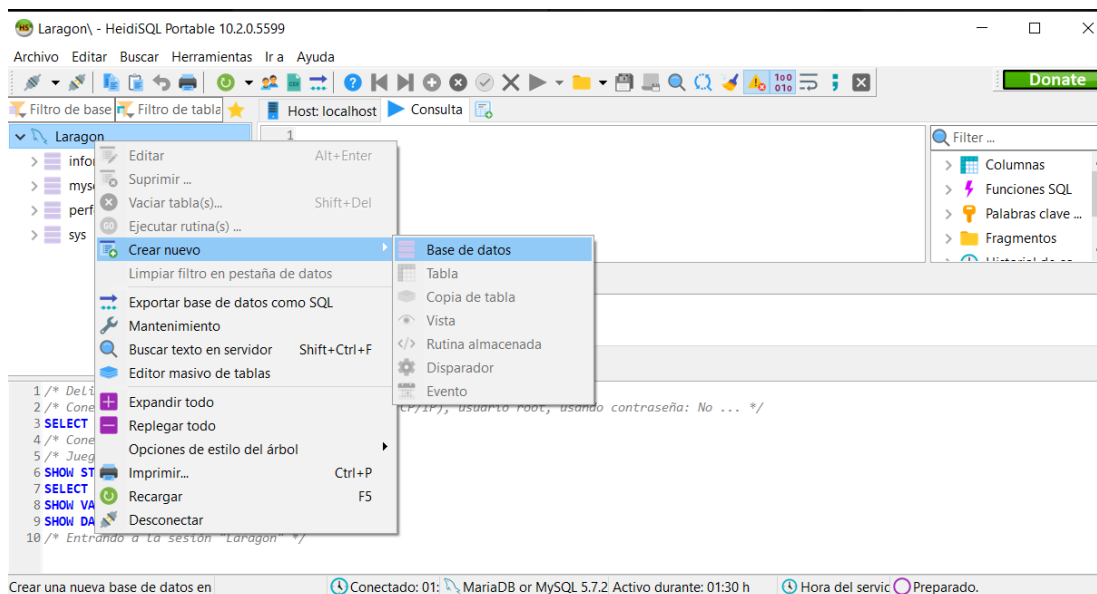
Para continuar con el procedimiento, presiona el botón **Abrir**. Enseguida aparece la interfaz de trabajo de para la creación de bases de datos, como puedes observar en la siguiente figura:



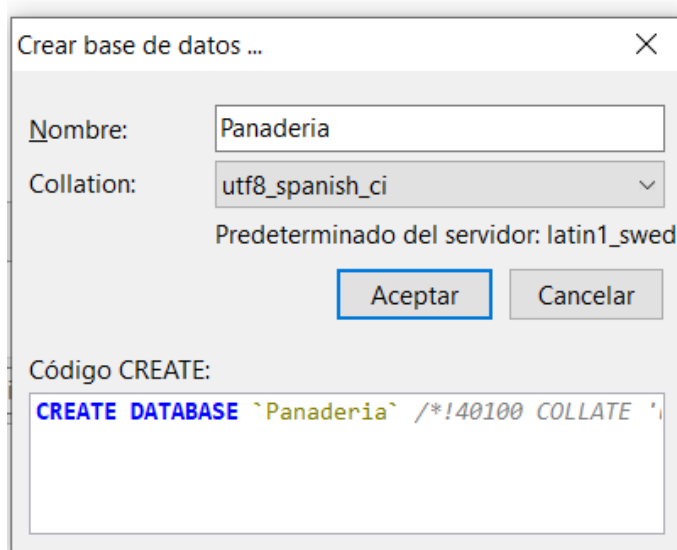
Puedes crear una nueva base de datos en Laragon de dos maneras:

1. Utilizando el entorno gráfico.
2. Utilizando la pestaña de instrucciones y consultas.

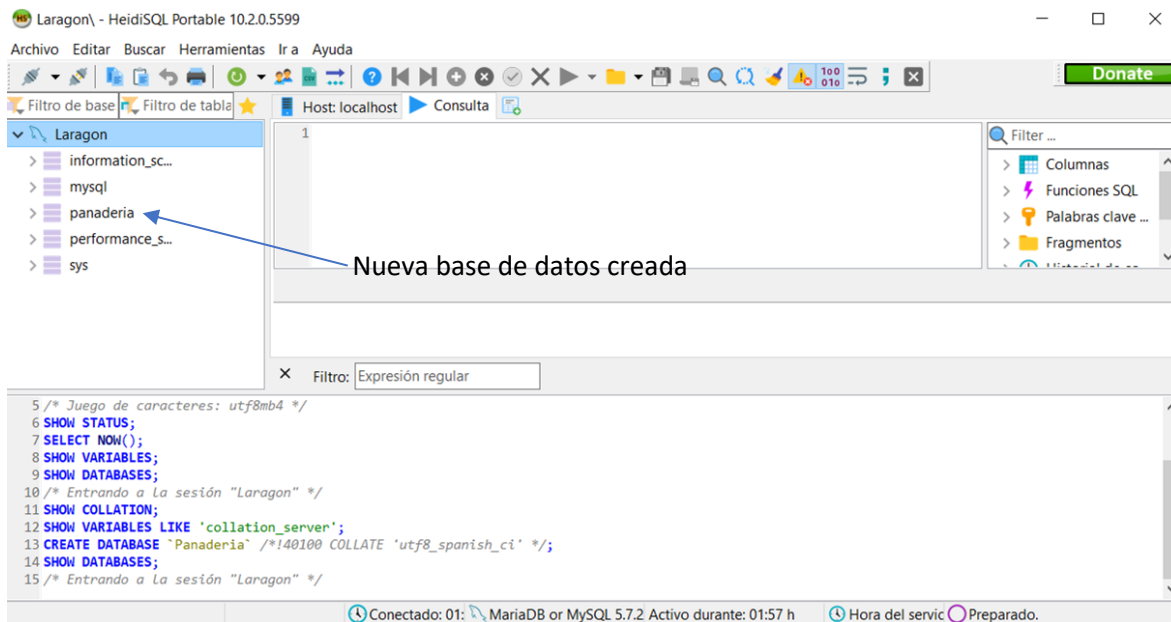
1. Utilizando el entorno gráfico



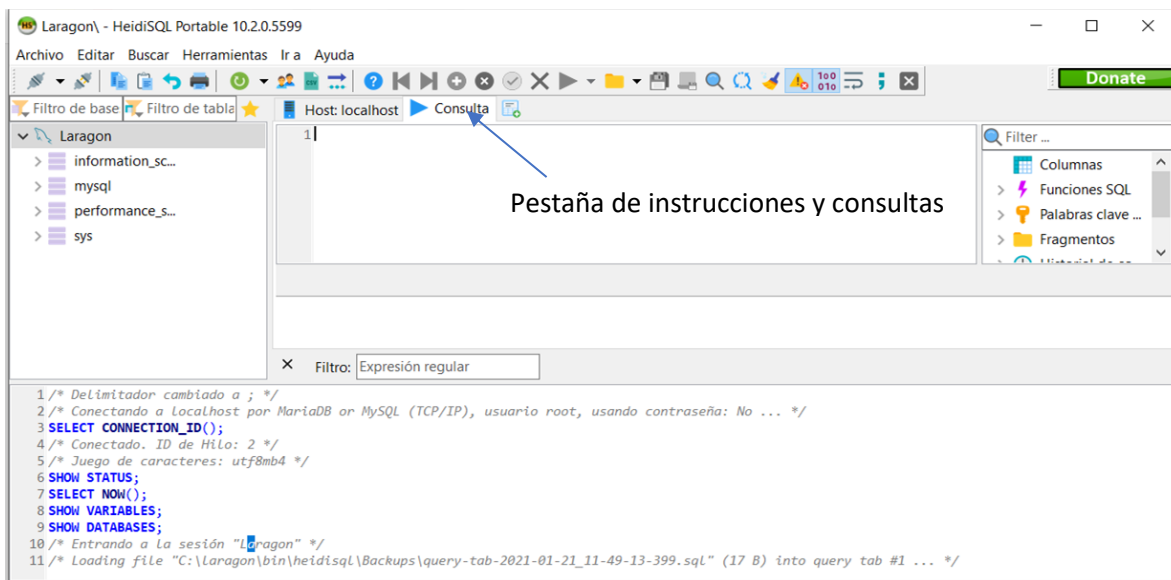
Posiciónate en la zona de conexiones de las bases de datos y presiona el botón derecho del ratón sobre la palabra "Laragon" para desplegar el menú y posteriormente selecciona "Crear nuevo" y elegir la opción "Base de Datos".

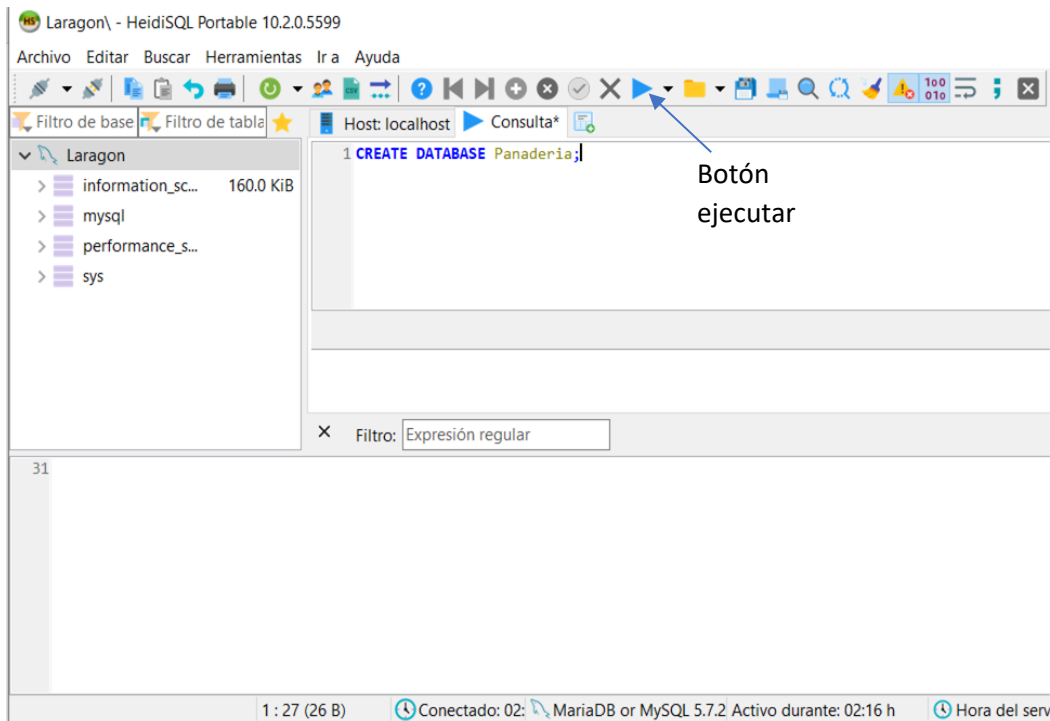


En esta ventana escribe el nombre que deseas dar a la nueva base de datos y presiona **Aceptar** para crear la base de datos.



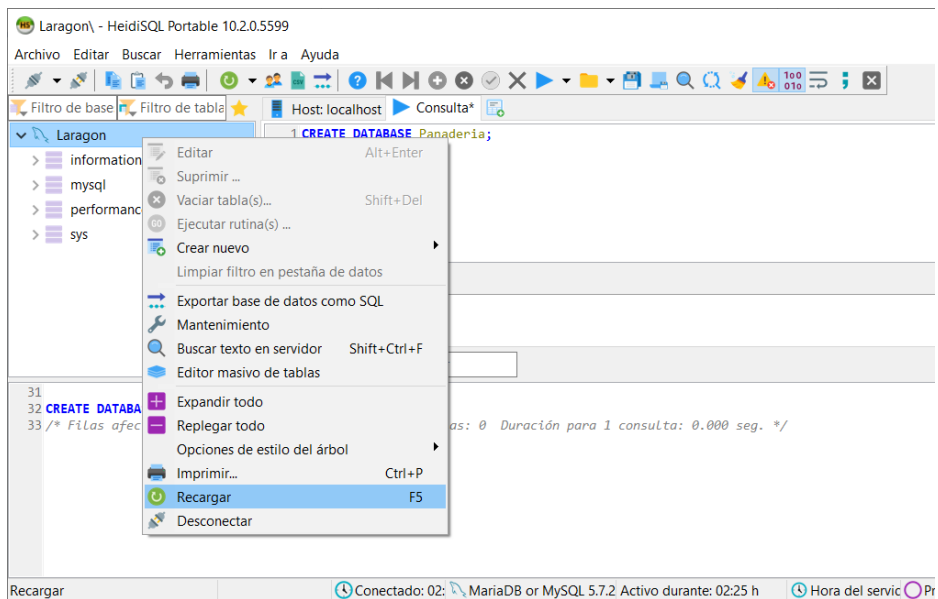
2. Utilizando la pestaña de instrucciones y consultas



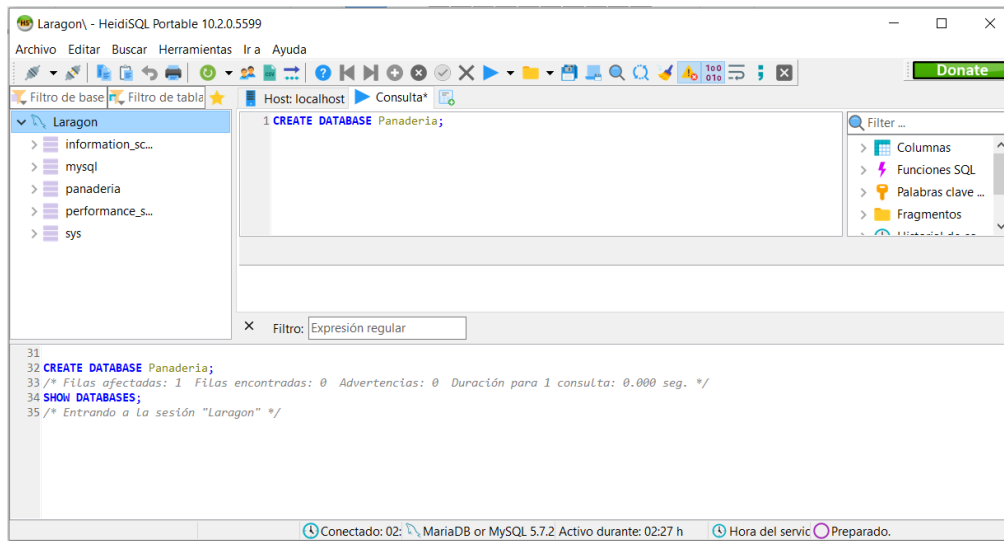


Da clic sobre la pestaña de consulta y en el área en blanco escribe la sentencia SQL, que en este caso corresponde a **CREATE DATABASE**. La sintaxis debe quedar de la siguiente manera: **CREATE DATABASE Panaderia;**

Posteriormente, ejecuta la sentencia presionando en el botón **Ejecutar** o presionando la tecla **F9**.

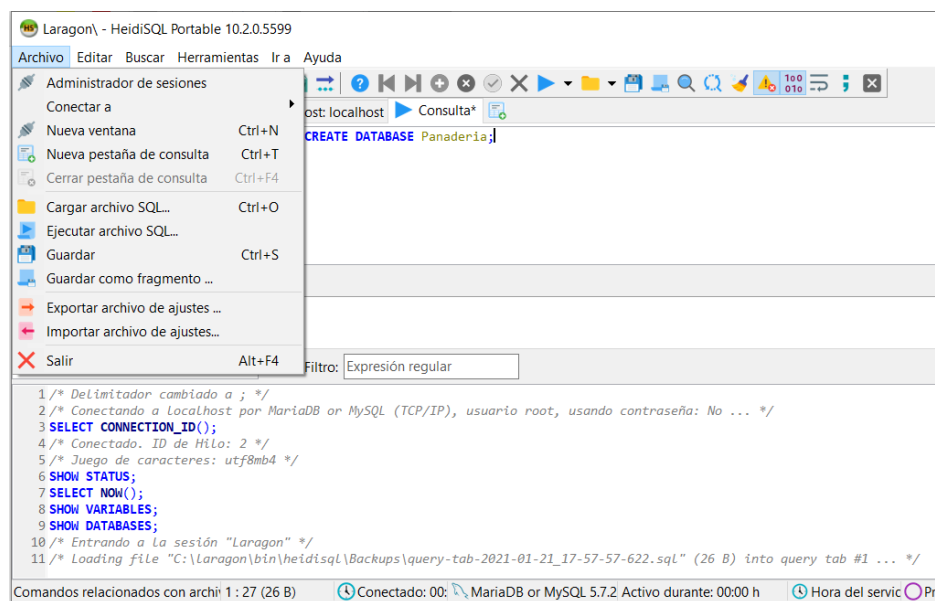


Una vez que se ejecutó la sentencia, tienes que recargar para que se actualice la interfaz y te aparezca la nueva base de datos.



Salir de una base de datos

Propiamente no se cierra una base de datos, solo se deja de utilizar. Cuando abres la aplicación, se establece una conexión a la base de datos y esta se mantiene mientras se selecciona la base de datos a utilizar. Para cada base de datos que se abre se habilita una conexión distinta.



Para salir de la base de datos es necesario desplegar el menú **Archivo** y elegir la opción **Salir** o presionar las teclas **Alt + F4**.



Para saber más

Para reforzar el conocimiento sobre el tema **Crear, abrir y cerrar una base de datos en Laragon** revisa el siguiente vídeo escaneando el código QR o dando clic en el enlace.



<https://youtu.be/KyKic5jTZkQ>

Referencias

- Gilfillan, Ian, La Biblia de MySQL (2018), Anaya Multimedia
- 11sql.com (2012). Tutorial de SQL: Curso de SQL <http://sql.11sql.com/>
- Tech Lib (s.f.) Definición RDBMS. <https://techlib.net/definition/rdbms.html>



Actividad de aprendizaje

Relaciona los conceptos con su respectiva función en MySQL.

- | | | |
|--|---|----------------------|
| Creación de disparadores | • | • (INSERT) |
| Insertar datos en una tabla | • | • (COMMIT) |
| Retroceder la transacción de una sentencia SQL | • | • (UPDATE) |
| Finalizar la transacción de una sentencia SQL | • | • (GRANT) |
| Consultar los datos almacenados en una tabla | • | • (CREATE TRIGGER) |
| Creación de una base de datos | • | • (CREATE DATABASE) |
| Modificar datos ya existentes en una tabla | • | • (CREATE TABLE) |
| Eliminar permisos de acceso a datos de una tabla | • | • (SELECT) |
| Creación de una tabla | • | • (ROLLBACK) |
| Creación de un índice de una tabla | • | • (CREATE VIEW) |
| Dar permiso de acceso a los datos de una tabla | • | • (CREATE PROCEDURE) |
| Creación de procedimientos almacenados | • | • (CREATE INDEX) |
| Creación de una vista | • | • (REVOKE) |
| Borrar datos almacenados en una tabla | • | • (DELETE) |

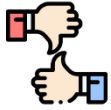
Implementa las instrucciones para realizar la acción solicitada.

- a) **Escribe las instrucciones para crear la base de datos Escuela en lenguaje MySQL.**

- b) **Escribe las instrucciones para crear la base de datos Negocio en lenguaje MySQL.**

- c) **Escribe las instrucciones para crear la base de datos Juguetería en lenguaje MySQL.**

- d) **Escribe las instrucciones para crear la base de datos Tienda en lenguaje MySQL.**



Autoevaluación

Indicadores	Lo puedo hacer	Tengo dudas	Necesito trabajar más
Comprendo que es el lenguaje SQL			
Entiendo la importancia en la creación y gestión de una base de datos.			
Soy capaz de agrupar las sentencias por funcionalidad.			
Comprendo cómo crear una base de datos por medio de la interfaz gráfica.			
Soy capaz de crear una base de datos por medio de las sentencias SQL.			
Comprendo la importancia de las bases de datos y las tablas que almacenan información relacional.			

Diseño de tablas de datos



Una **tabla** es una cuadrícula que nos sirve para presentar la información de forma organizada o para mejorar la administración de la información. Está conformada por filas y columnas que, propiamente en el desarrollo de esta lección, llamaremos campos y registros. Cada tabla tiene “n” campos (verticales), que sirven para especificar las características de algo en particular, por ejemplo, las características para identificar a un **alumno**: su número de control, nombre, apellido paterno, apellido materno, semestre, grupo, dirección, teléfono, entre otros.

Por otro lado, también se cuenta con “m” registros (horizontales) que, a su vez, incorporan de forma secuencial los datos de cada alumno (como en el ejemplo descrito) en la tabla. Una tabla pertenece a una **Base de Datos (BD)**. Es preciso mencionar que cada BD puede contener más de una tabla y que, en su debido momento, se puede interactuar entre las tablas de una misma BD o bien, entre tablas de diferentes Bases de Datos.

A continuación se muestra un ejemplo de la tabla **Alumno**:

NC	Nombre	Apaterno	Amaterno	Semestre	Grupo	Dirección	Telefono
20DTA001	Pedro	Zapata	Pérez	1	A	Ignacio Mendoza #45	777-135-XXXX
20DTA002	Juan	Arce	Fernández	3	B	Benito Juárez #110	777-149-XXXX
20DTA003	Laura	Molina	Vázquez	5	A	Av. Perseverancia #23	777-091-XXXX
20DTA004	Araceli	Gómez	Sánchez	3	B	José Ma. Morelos #101	777-234-XXXX
20DTA005	José	López	García	1	A	Revolución Agraria #54	777-395-XXXX
20DTA006	Julia	Márquez	González	5	B	Miguel Hidalgo #77	777-108-XXXX



¡Vamos a aprender!

Hasta el momento has aprendido que para **crear una base de datos** requieres de un **análisis de la información**, aplicar un **modelo entidad - relación** que permita **crear distintas tablas de datos**.

Las tablas de datos son parte fundamental en una base de datos, en ellas puedes almacenar información de un objeto y/o una persona.

Las tablas son de gran utilidad para llevar una buena organización y control de la información, posteriormente al utilizar un programa como un manejador de datos te podrán ayudar a consultar o generar reportes.

En esta lección, como en la anterior, utilizarás el gestor de bases de datos MySQL y Laragon para la creación de las tablas de datos, pero para iniciar debes de recordar ¿qué elementos tiene una tabla de datos y cómo se representan? ¿cuál es la función de la llave

principal y la llave foránea en una tabla de datos? ¿cuáles son los tipos de datos que guarda un campo?

¿Qué es una tabla? Es una herramienta de organización de información que se utiliza en bases de datos en la informática. Las tablas hacen referencia al modelado o recopilación de datos por parte de una aplicación de un programa que permite operar con los mismos organizándose y poniéndolos en relación de diversas maneras. Los elementos que la componen son campos que son las columnas y los registros que son las filas, existe un campo que es asignado como una llave primaria y/o una llave secundaria.



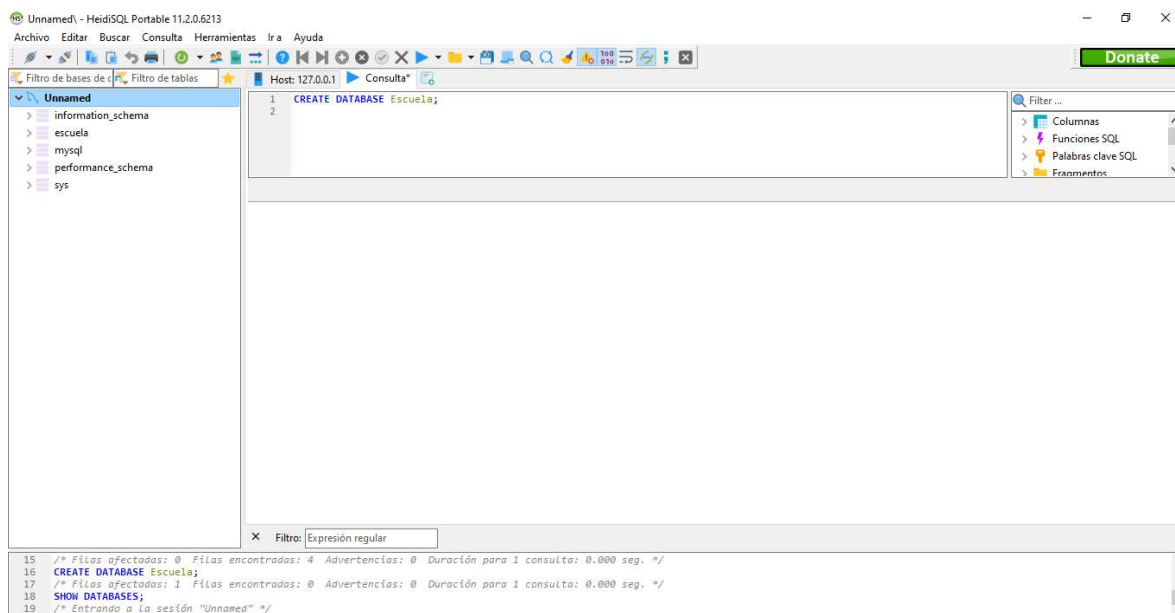
Crear tablas de datos

Para poder crear una tabla de datos es necesario haber creado una base de datos. Tomemos como ejemplo la creación de la BD **Escuela**.

Recordemos, una BD se crea desde la herramienta **Laragon** utilizando una sentencia en lenguaje **SQL**, como se muestra a continuación:

```
CREATE DATABASE Escuela;
```

Si, ejecutamos dicho código en Laragon (presionando Ejecutar SQL ► o bien la tecla F9), obtendrás lo siguiente:



Continuamos utilizando como ejemplo a la base de datos **Escuela** y, ahora, vas a crear una tabla de datos que se llamará **Alumno**, misma que estará conformada por ocho campos que se describen en la siguiente tabla:

Estructura de la tabla **Alumno**:

Nombre del Campo	Tipo de datos	Descripción
NC (PRIMARY KEY)	VARCHAR (10)	Número de Control
Nombre	VARCHAR (40)	Nombre del Alumno
APaterno	VARCHAR (40)	Apellido Paterno
AMaterno	VARCHAR (40)	Apellido Materno
Semestre	INT	Semestre que cursa
Grupo	VARCHAR (1)	Grupo en la Escuela
Direccion	VARCHAR (50)	Dirección del Alumno
Telefono	VARCHAR (15)	Teléfono del Alumno

Es tiempo de hacer algunas precisiones.

En la estructura propuesta de la tabla **Alumno**, los ocho **Campos** van escritos con mayúsculas y minúsculas (salvo NC) por convención internacional, ya que es el estándar para programar en innumerables lenguajes de programación.

Como podrás observar, no llevan signos de acentuación ni espacios, por ello se abrevió Apellido Paterno por **APaterno** y lo propio con **AMaterno**.

Otra situación que considerar es la llave primaria o **PRIMARY KEY**, dicho campo permite que solamente exista en la tabla un número único e irreplicable que lo identifique de todos los registros de la tabla, de forma que si alguien intenta insertar un registro con un número de control (NC) que ya existe, no podrá hacerlo porque la tabla tiene como llave primaria precisamente al NC.

También, es importante que sepas que hay dos tipos de datos dentro de esta tabla:

1. **INT**. Permite que registres valores enteros, es decir números, que van desde el 0 hasta el 4,294,967,295.
2. **VARCHAR**. Se define el número de letras o números a utilizar, y acepta texto y dígitos aunque al final, todo lo considera como texto.

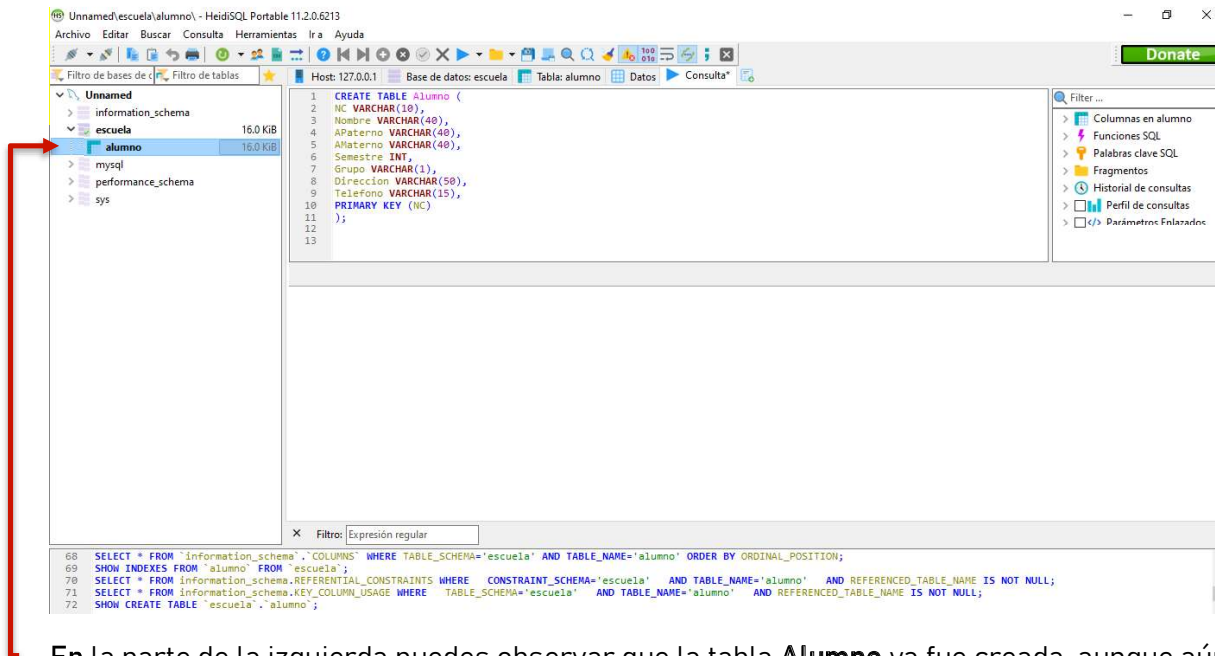
En el **lenguaje MySQL**, se utiliza la instrucción **CREATE TABLE** para especificar la creación de la estructura de la tabla. Observa cómo se crearía la tabla **Alumno**, utilizando lenguaje SQL.

```
CREATE TABLE Alumno (  
  NC VARCHAR(10),  
  Nombre VARCHAR(40),  
  APaterno VARCHAR(40),  
  AMaterno VARCHAR(40),  
  Semestre INT,  
  Grupo VARCHAR(1),  
  Direccion VARCHAR(50),  
  Telefono VARCHAR(15),  
  PRIMARY KEY (NC)  
);
```

Dentro de la sintaxis de la instrucción **CREATE TABLE**, se especifica el nombre de la tabla a crear, cada uno de los campos indicados, el tipo de datos y su tamaño, así como **PRIMARY KEY** con el campo que será la llave primaria en la tabla de datos.

Observa el resultado de ingresar el código SQL para crear la tabla **Alumno**, aunque recuerda que primero debes seleccionar la BD **Escuela**, misma que ya creaste.

La siguiente imagen ilustra el resultado de ejecutar el código SQL mostrado previamente.



En la parte de la izquierda puedes observar que la tabla **Alumno** ya fue creada, aunque aún no se le ha ingresado ningún registro.

Ahora, ya sabiendo el procedimiento para crear una tabla en Laragon, vas a diseñar la estructura de la nueva tabla a utilizar, que llevará el nombre de **Profesor** y tendrá la siguiente estructura:

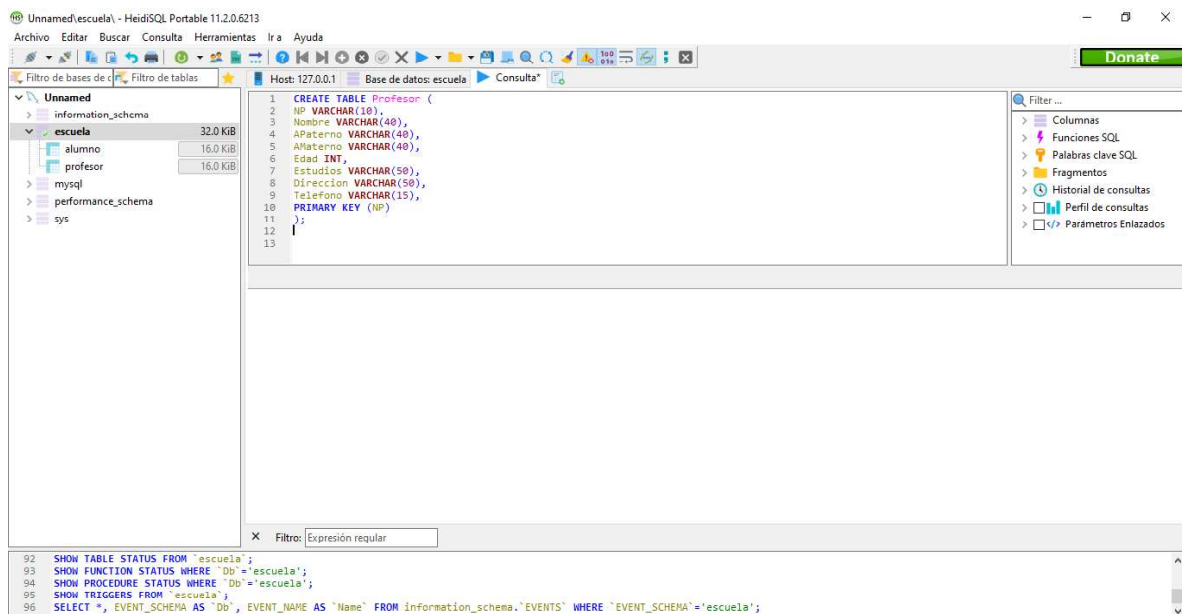
Nombre del Campo	Tipo de datos	Descripción
NP (PRIMARY KEY)	VARCHAR (10)	Número de Profesor
Nombre	VARCHAR (40)	Nombre del Profesor
APaterno	VARCHAR (40)	Apellido Paterno
AMaterno	VARCHAR (40)	Apellido Materno
Edad	INT	Edad del Profesor
Estudios	VARCHAR (50)	Grado máximo de estudios
Direccion	VARCHAR (50)	Dirección del Alumno
Telefono	VARCHAR (15)	Teléfono del Alumno

La sentencia/instrucción SQL a utilizar para la creación de la tabla Profesor, es la siguiente:

```
CREATE TABLE Profesor (  
NP VARCHAR(10),  
Nombre VARCHAR(40),  
APaterno VARCHAR(40),  
AMaterno VARCHAR(40),  
Edad INT,  
Estudios VARCHAR(50),  
Direccion VARCHAR(50),  
Telefono VARCHAR(15),  
PRIMARY KEY (NP)  
);
```

Observa el resultado de ingresar el código SQL para crear la tabla **Alumno**, aunque recuerda que primero debes seleccionar la BD **Escuela**, misma que ya creaste.

La siguiente imagen ilustra el resultado de ejecutar el código SQL mostrado previamente.



Con esta acción, has culminado la creación de dos tablas, que llevan por nombre **Alumno** y **Profesor**, dentro de la Base de Datos **Escuela**.

Ahora, procederás a revisar el comando **SHOW TABLES**.

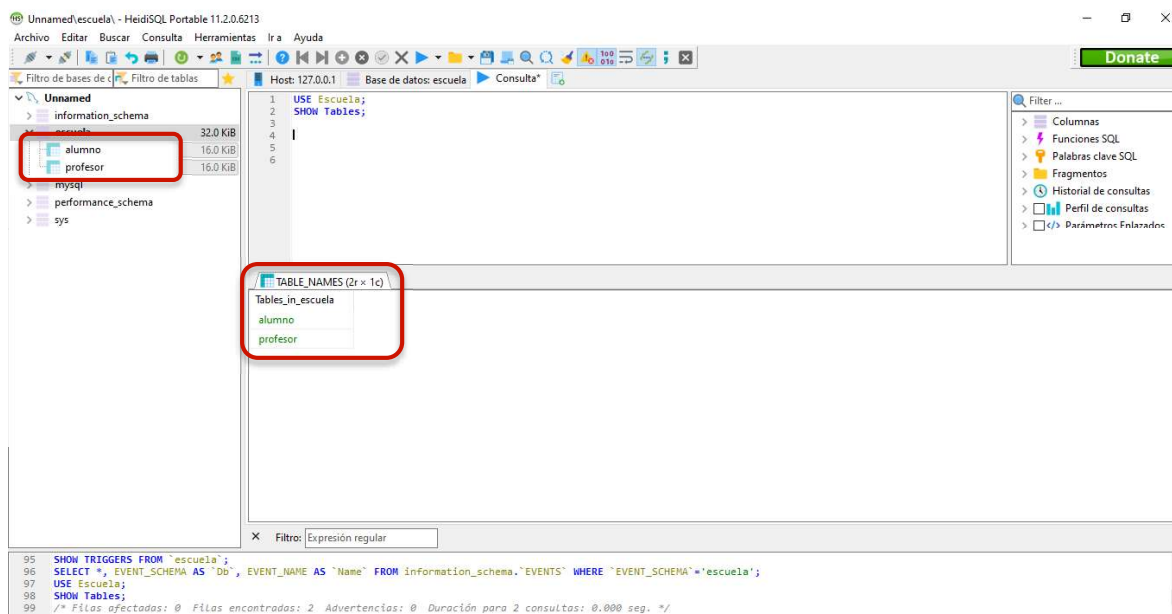
Mostrar las tablas de datos creadas

En el **lenguaje SQL**, dentro del Sistema Gestor de Bases de Datos (SGBD) MySQL, se utiliza la instrucción **SHOW TABLES** para mostrar el nombre de las tablas que contiene la base de datos.

Por ejemplo, si quieres revisar el nombre de las tablas de la base de datos **Escuela**, que has creado dentro de esta lección, el código o sentencia SQL a utilizar es la siguiente:

```
USE Escuela;  
SHOW Tables;
```

Con lo que obtendrás el siguiente resultado al ejecutar el código en Laragon.



Es posible observar, que como resultado, en la parte inferior se encuentran las dos tablas creadas, Alumno y Profesor al igual que en la parte izquierda superior de la pantalla. Por lo anterior, se confirma que las tablas fueron creadas con éxito y ya se encuentran listas para la captura de registros.

Otro comando que debes revisar es **DESCRIBE**, te muestra la estructura completa de cada tabla de datos. Para ello, se ejemplifica únicamente la tabla **Alumno**.

Muestra la estructura de una tabla de datos

También en el **lenguaje MySQL** se utiliza la instrucción **DESCRIBE** para mostrar la estructura de la tabla de datos.

Por ejemplo, si quieres revisar la estructura de la tabla Alumno, debes escribir lo siguiente:

```
DESCRIBE Alumno;
```

Con lo que obtendrás el siguiente resultado al ejecutar el código en Laragon.

Field	Type	Null	Key	Default	Extra
NC	varchar(10)	NO	PRI	(NULL)	
Nombre	varchar(40)	YES		(NULL)	
APaterno	varchar(40)	YES		(NULL)	
AMaterno	varchar(40)	YES		(NULL)	
Semestre	int(11)	YES		(NULL)	
Grupo	varchar(1)	YES		(NULL)	
Direccion	varchar(50)	YES		(NULL)	
Telefono	varchar(15)	YES		(NULL)	

Como puedes observar, en la parte inferior se muestran los ocho campos creados para la tabla Alumno. Por lo anterior, se confirma que dicha tabla fue creada con éxito y ya se encuentra lista para la captura de registros.



Para saber más



Para reforzar el conocimiento sobre el tema **Diseño de tablas de datos en Laragon** revisa el siguiente vídeo escaneando el código QR o dando clic al link.

<https://youtu.be/MwcvpT27ugs>

Referencias

- Victoria Bembibre. (2009). Definición de Base de datos. <https://www.definicionabc.com/tecnologia/base-de-datos.php#:~:text=Se%20le%20llama%20base%20de,ordenarlos%20y%20clasificarlos%20en%20conjunto>
- MySQL. (2020). Manual de referencia del MySQL 5.0. <https://downloads.mysql.com/docs/refman-5.0-es.pdf>
- Desarrollo web (s.f.) Listado y descripción de los distintos tipos de datos de MySQL. www.desarrolloweb.com



¡Ponlo en práctica!

Crea tablas de datos siguiendo el procedimiento (no es necesario equipo de cómputo)

Nota: Si cuentas con equipo de cómputo y acceso a internet, prueba lograr el reto de hacer funcionar los códigos utilizando Laragon.

Recuerda que para crear tablas de datos necesitas generar la base de datos, en este caso utilizarás la BD de **Biblioteca**. Dentro de esa **BD**, vas a crear dos tablas: **Libro** y **Autor**.

- a) Escribe las instrucciones para **crear la base de datos Biblioteca en lenguaje MySQL**.

Para crear la tabla de **Libro** utiliza la siguiente estructura:

Nombre del Campo	Tipo de datos	Descripción
NL (PRIMARIA KEY)	INT,	Identificación del libro
Título	VARCHAR (50),	Título del libro
Edición	VARCHAR (30),	Número de Edición
Fecha	VARCHAR (30),	Fecha de publicación

- b) Escribe las instrucciones para **crear la tabla de Libro en lenguaje MySQL**.

Para crear la tabla de **Autor** utiliza la siguiente estructura:

Nombre del Campo	Tipo de datos	Descripción
NA (PRIMARIA KEY)	INT,	Identificación del libro
Nombre	VARCHAR (50),	Nombre del Autor
PaisOrigen	VARCHAR (30),	País de Origen
Género	VARCHAR (30),	Género Literario

c) Escribe las instrucciones para **crear la tabla de Autor en lenguaje MySQL**.

Muestra la estructura de las tablas de datos

Ahora practica con los códigos vistos en esta lección, escribiendo las instrucciones **en lenguaje MySQL**, para mostrar la estructura de la tabla **Libro** en el primer recuadro y en el segundo a la tabla **Autor**.

Muestra las tablas de datos creadas en la base de datos

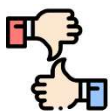
Escribe las instrucciones para **mostrar las tablas de datos incluidas en la BD Biblioteca** en lenguaje MySQL.



Actividad de aprendizaje

Lee las siguientes oraciones y subraya la respuesta correcta

1. Es una instrucción en lenguaje MySQL que muestra la estructura de una tabla de datos.
a) show b) describe c) create d) update
2. Es una instrucción en lenguaje MySQL que permite crear una tabla de datos.
a) references b) show c) database d) create
3. Es una instrucción en lenguaje MySQL que permite enlistar todas las tablas que contiene una base de datos.
a) show b) describe c) create d) update
4. En una tabla de datos las columnas representan:
a) tipos de datos b) registros c) campos d) datos
5. En una tabla de datos las filas representan:
a) llaves b) registros c) campos d) datos
6. En el programa Laragon para configurar la llave principal en la propiedad del campo se utiliza la instrucción.
a) primary key b) primary c) references d) update
7. ¿Qué tipo de dato permite almacenar valores enteros o numéricos?
a) VARCHAR b) INT c) PRIMARY d) KEY
8. ¿Qué tipo de dato permite almacenar valores en modo texto, incluyendo letras y números?
a) VARCHAR b) INT c) PRIMARY d) KEY



Autoevaluación

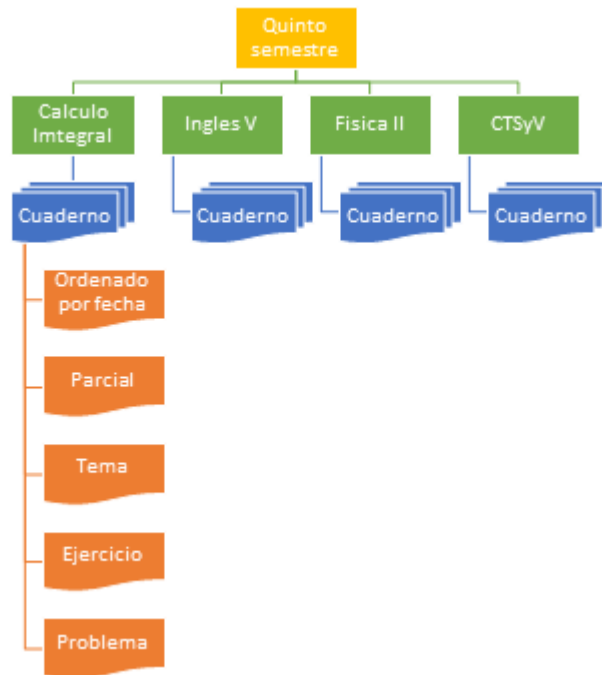
Indicadores	Lo puedo hacer	Tengo dudas	Necesito trabajar más
Comprendo qué es una tabla de datos.			
Puedo crear una base de datos			
Reconozco lo que representa el campo y los registros en una tabla de datos.			
Soy capaz de escribir las instrucciones en el Lenguaje MySQL CREATE TABLES para crear una tabla de datos.			
Soy capaz de escribir las instrucciones en el Lenguaje MySQL SHOW TABLES para mostrar las tablas de datos.			
Soy capaz de escribir las instrucciones en el Lenguaje MySQL DESCRIBE para mostrar la estructura de datos de una tabla.			
Puedo manejar el programa Laragon para crear una tabla de datos.			
Puedo manejar el programa Laragon para mostrar la estructura de la tabla de datos.			
Puedo manejar el programa Laragon para enlistar las tablas de datos.			

Edición de estructura de tabla de datos



Contextualizando

Como se aprendiste en la lección anterior, no es suficiente tener una tabla de datos, si no especificar qué información debe contener, cómo la actualizarlas, como la vas a mostrar y cómo se va a clasificar. Es como si tuvieras una libreta de tus apuntes, pero con las hojas vacías y cuando en tus estudios escribes información de forma ordenada de acuerdo a la fecha.



Si tomas como ejemplo la asignatura de Calculo Integral, tienes un cuaderno donde escribes tus progresos y apuntes que el docente te va guiando (se puede asemejar a una tabla), esta información se va conteniendo y ordenando de diferentes formas (registros). Una vez que el docente te pide que escribas en tu cuaderno tu estarás “Insertando” más registros en tu libreta, si por alguna razón tú te das cuenta de que el ejercicio que has resuelto no es correcto y decides corregirlo, lo que estarás haciendo es “Actualizar” este registro, pero más sin embargo, detectas que el apunte que has escrito pertenece a otra libreta podrías “Eliminar” el registro para escribirlo en la libreta correspondiente, si algún compañero te pregunta por un apunte tu podrás hacer una búsqueda tomando en cuenta diferentes criterios y determinarás una “Selección” de la información de acuerdo a la fecha, parcial, tema, ejercicio o problema que estés buscando.



¡Vamos a aprender!

Para que comprendas mejor el desarrollo de una base de datos trabajarás con el siguiente ejemplo:

El dueño de una veterinaria te solicita realizar una base de datos, para ello crearás una base de datos en Laragon llamada **Veterinaria** por medio de las sentencias formadas en lenguaje SQL.

```
CREATE DATABASE `Veterinaria`
```

Posteriormente, el cliente desea contener la información básica de las mascotas que van ingresando, para ello se genera una tabla en Laragon llamada **"Mascotas"**

Campo	Tipo y tamaño
Nombre	String(15)
Dueño	String(15)
Especie	String(10)
Sexo	String(1)
Nacimiento	Date
Fallecimiento	Date

Para crear una tabla llamada Mascotas, con los campos, Nombre de tipo cadena y tamaño 15, el campo Dueño de tipo cadena y tamaño 15, el campo Especie de tipo cadena y tamaño 10, el campo Sexo de tipo cadena y tamaño 1, el campo Nacimiento de tipo fecha y Fallecimiento de tipo fecha se requiere la siguiente sentencia:

```
CREATE TABLE `Mascotas` (  
  `Nombre` VARCHAR(15) NULL DEFAULT NULL,  
  `Dueño` VARCHAR(15) NULL DEFAULT NULL,  
  `Especie` VARCHAR(10) NULL DEFAULT NULL,  
  `Sexo` VARCHAR(1) NULL DEFAULT NULL,  
  `Nacimiento` DATE NULL DEFAULT NULL,  
  `Fallecimiento` DATE NULL DEFAULT NULL  
);
```

Incluir información nueva

Al momento de crear una tabla, esta se encuentra vacía y una de las formas directas de ingresar información es usando la sentencia **INSERT**, esta sentencia ayuda a incorporar información de nuevos registros uno por vez, que a continuación se te muestra.

De esta sencilla manera, se suministran valores para cada columna, dispuestos en el orden en el cual se especificaron las columnas en la sentencia **CREATE TABLE**.

La sentencia **INSERT** sentencia tiene algunas variantes de sintaxis según el caso en que se desee aplicar:

1. Escribiendo valores en todas las columnas de una tabla
2. Escribiendo valores en determinados campos
3. Escribiendo valores en forma no ordenada
4. Escribiendo varios registros a la vez

1. **Escribiendo valores en todas las columnas de una tabla:** A cada campo de la tabla se le asigna un valor según su tipo de dato.

La siguiente forma se usa para insertar valores en todos los campos:

```
INSERT INTO nom_tabla VALUES (dato1, dato2, dato3,...);
```

Supón que el registro con el nombre Diane obtiene un nuevo hamster llamado "Puffball". Se podría agregar un nuevo registro, usando la sentencia **INSERT** de este modo:

```
INSERT INTO mascotas  
VALUES ('Puffball','Diane','hamster','f','1999-03-30',NULL);
```



Si observas que las cadenas alfanuméricas y las fechas son representados como cadenas delimitadas por apóstrofes. También, con **INSERT**, se pueden insertar valores **NULL** directamente, para indicar un valor ausente.

2. **Escribiendo valores en determinados campos:** Se escriben valores solo en algunos campos de la tabla, esto depende de las reglas de negocio que se están modelando

en la base de datos. Obviamente los valores deben corresponder según su tipo de dato de los campos.

La siguiente forma es usada cuando se va a agregar datos en todos los campos:

```
INSERT INTO nom_tabla (Campo1, Campo2, Campo3,...) VALUES (dato1, dato2, dato3,...);
```

Suponga que solamente desea agregar información en algunos campos.

```
INSERT INTO mascotas (nombre, dueño, sexo, nacimiento)
VALUES ('Puffball','Diane', 'f','1999-03-30');
```



Si observas se ha agregado información de solo algunos campos, al momento de ejecutar la sentencia, se integrará la información nueva y dejará sin información los campos no mencionados en la sentencia.

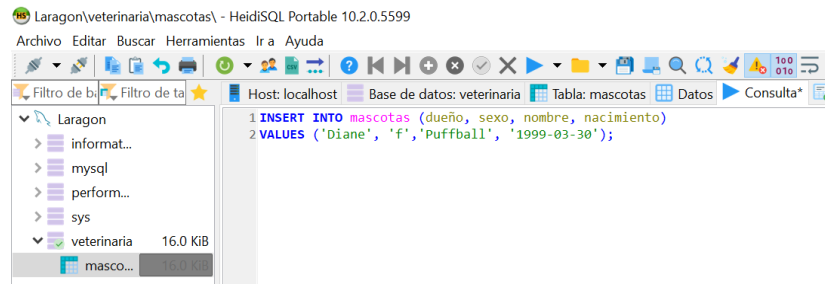
3. **Escribiendo valores en forma no ordenada:** Esto nos dice que si los campos aparecen en el orden: campo 1, campo 2 y campo 3 en la tabla, nosotros podemos escribir mediante la sentencia **INSERT**, primero el campo 2, luego el campo 3 y por último el campo 1, o de forma desordenada.

En el caso de la sentencia con los campos desordenados, se usa la siguiente estructura:

```
INSERT INTO nom_tabla (Campo2, Campo3, Campo1,...) VALUES (dato2, dato3, dato1,...);
```

Esta forma es usada para determinar el orden de la inserción de datos en algún orden determinado.

```
INSERT INTO mascotas (dueño, sexo, nombre, nacimiento)
VALUES ('Diane', 'f','Puffball', '1999-03-30');
```

4. **Escribiendo varios registros a la vez:** con esta sentencia se registraron varios registros en una misma instrucción, para ello se debe incluir en la sección de los valores los registros entre paréntesis y separando cada registro con una coma, como se muestra a continuación:

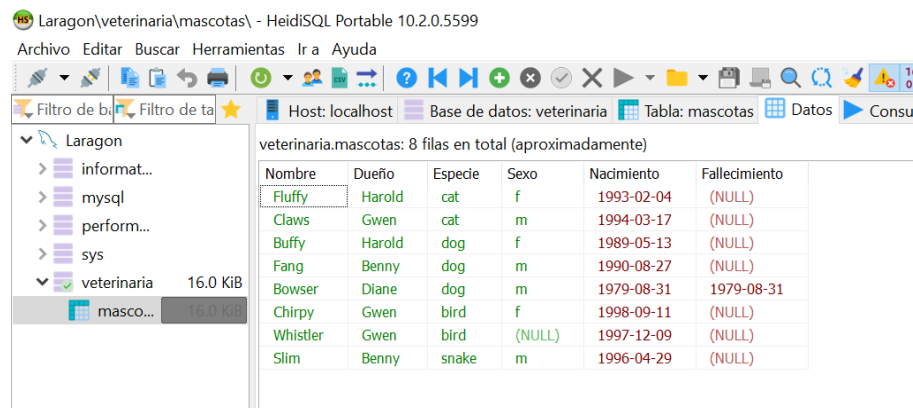
```

INSERT INTO mascotas
VALUES ('Fluffy','Harold','cat','f','1993-02-04',NULL),
('Claws','Gwen','cat','m','1994-03-17',NULL),
('Buffy','Harold','dog','f','1989-05-13',NULL),
('Fang','Benny','dog','m','1990-08-27',NULL),
('Bowser','Diane','dog','m','1979-08-31','1979-08-31'),
('Chirpy','Gwen','bird','f','1998-09-11',NULL),
('Whistler','Gwen','bird',NULL,'1997-12-09',NULL),
('Slim','Benny','snake','m','1996-04-29',NULL);

```



Con esta sentencia se ingresarán todos los registros que se desean.



Mostrar datos

Puede ocurrir que una vez que se haya cargado información en la tabla de datos es necesario mostrar la información incorporada, para ello se usa la sentencia **SELECT**.

La sintaxis de la sentencia **SELECT** es la siguiente:

```
SELECT campos FROM tabla_de_datos WHERE condiciones;
```

Un ejemplo para mostrar todos los campos registrados en una tabla de datos.

```
SELECT * FROM mascotas;
```

Y se mostrarán los datos de la siguiente manera:

mascotas (6x11)					
Nombre	Dueño	Especie	Sexo	Nacimiento	Fallecimiento
Puffball	Diane	hamster	f	1999-03-30	(NULL)
Puffball	Diane	(NULL)	f	1999-03-30	(NULL)
Puffball	Diane	(NULL)	f	1999-03-30	(NULL)
Fluffy	Harold	cat	f	1993-02-04	(NULL)
Claws	Gwen	cat	m	1994-03-17	(NULL)
Buffy	Harold	dog	f	1989-05-13	(NULL)
Fang	Benny	dog	m	1990-08-27	(NULL)
Bowser	Diane	dog	m	1979-08-31	1979-08-31
Chirpy	Gwen	bird	f	1998-09-11	(NULL)
Whistler	Gwen	bird	(NULL)	1997-12-09	(NULL)
Slim	Benny	snake	m	1996-04-29	(NULL)

Actualizar información

Si existe el caso de que algún dato deba ser modificado, para ello se utiliza la sentencia **UPDATE**, esta sentencia puede modificar información contenida en uno o varios campos de acuerdo con una condicionante, su estructura es la siguiente:

```
UPDATE tabla SET campo1='Nuevo valor', campo2='Nuevo valor'... WHERE campo1 = 'valor';
```

Un ejemplo de cómo actualizar información de que la mascota “Slim”; cambie su sexo a “f”, la sentencia sería:

```
UPDATE mascotas SET sexo='f' WHERE nombre = 'Slim';
```

En el caso de que se desee modificar para el caso de la mascota “Fang”, su dueño es “Alex” y es un “bird”, la sentencia sería:

```
UPDATE mascotas SET dueño='Alex', especie='bird' WHERE nombre = 'Fang';
```

Eliminar registros

Para eliminar registros específicos, es necesario usar la sentencia **DELETE**, esta sentencia elimina determinados registros por medio de una condicionante.

```
DELETE FROM tabla WHERE campo1 = 'valor';
```

Para ejemplificar la eliminación de algún registro, se ha considerado que el registro de la mascota “Buffy” ya no es necesario en la tabla de datos, se debe escribir la sentencia **DELETE** en el programa Laragon por medio de la siguiente sentencia:

```
DELETE FROM mascotas WHERE nombre = 'Buffy';
```

Uso de Laragon

Para utilizar el entorno gráfico realizar las operaciones anteriores se usan de la siguiente manera:

Partes de la pantalla de Laragon

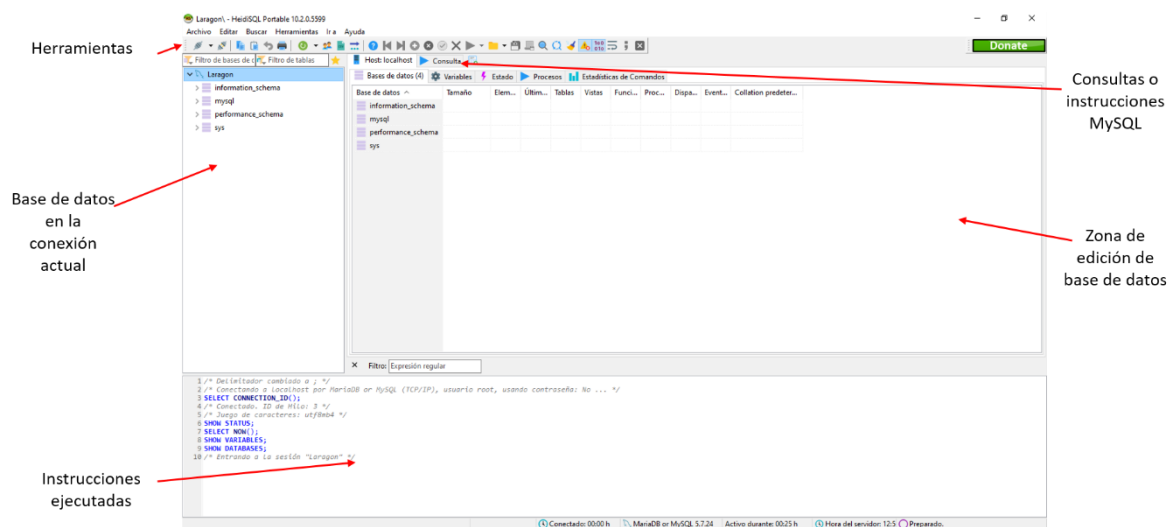
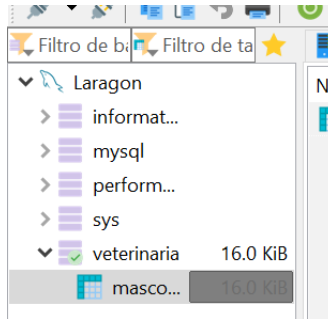


Ilustración 38 Captura de pantalla de Laragon que muestra la interfase básica del programa

- **Herramientas:** En esta barra encontrarás las herramientas de manipulación de conexión, herramientas de edición, de ejecución de instrucciones o consultas, de búsqueda y entre otras herramientas.



- **Base de datos en la conexión actual:** Aquí se encuentran las bases de datos que se encuentran en el dominio actual.



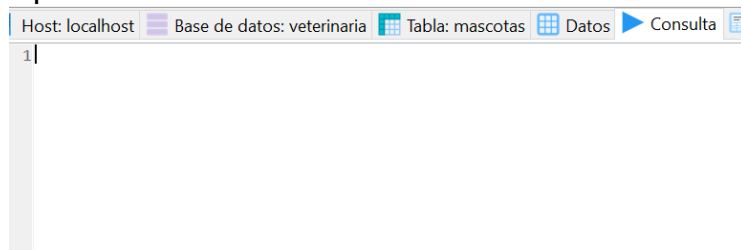
- **Instrucciones ejecutadas:** En esta zona aparecen las instrucciones que se han ejecutado, mostrando si se efectuaron satisfactoriamente.

```

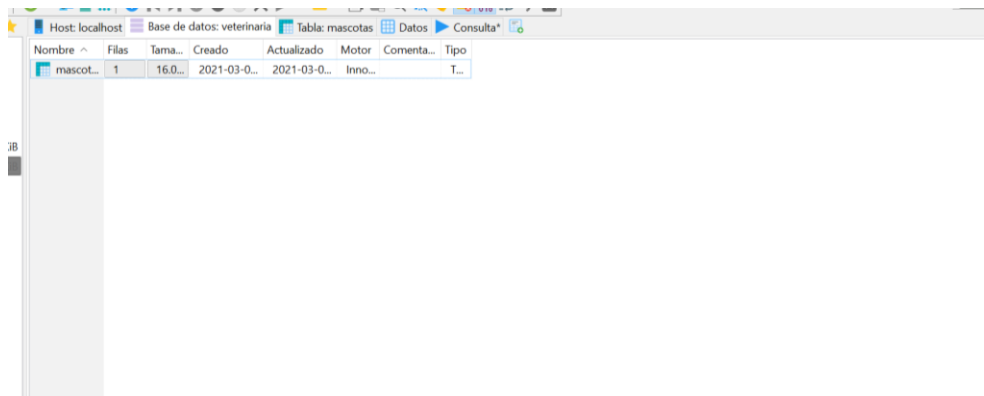
83
84 SELECT * FROM mascotas;
85 /* Filas afectadas: 0 Filas encontradas: 8 Advertencias: 0 Duración p

```

- **Consultas o instrucciones MySQL:** En esta zona se escriben las instrucciones de MySQL que el usuario decida.

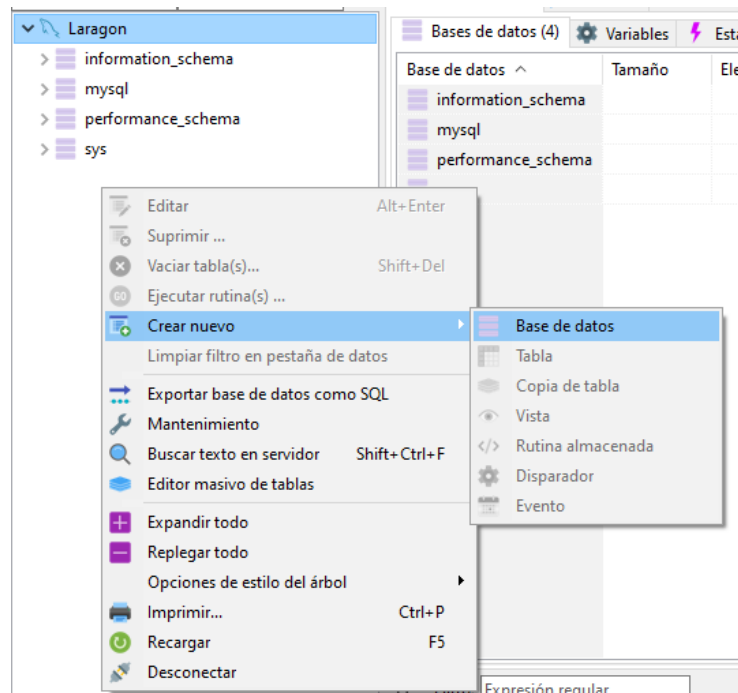


- **Zona de edición de base de datos:** En esta zona aparece de manera visual las posibles modificaciones que se le puede hacer a las bases de datos o a las tablas de datos.

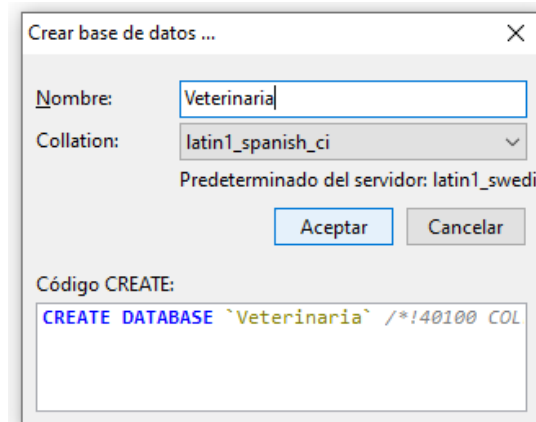


Crear una base de datos en Laragon

Para crear una base de datos nueva en la conexión actual, una forma básica de realizarlo es presionar con el botón derecho en la zona de base de datos, en la conexión y seleccionar en el menú contextual la instrucción “Crear nuevo” y posteriormente seleccionar “Base de datos”:

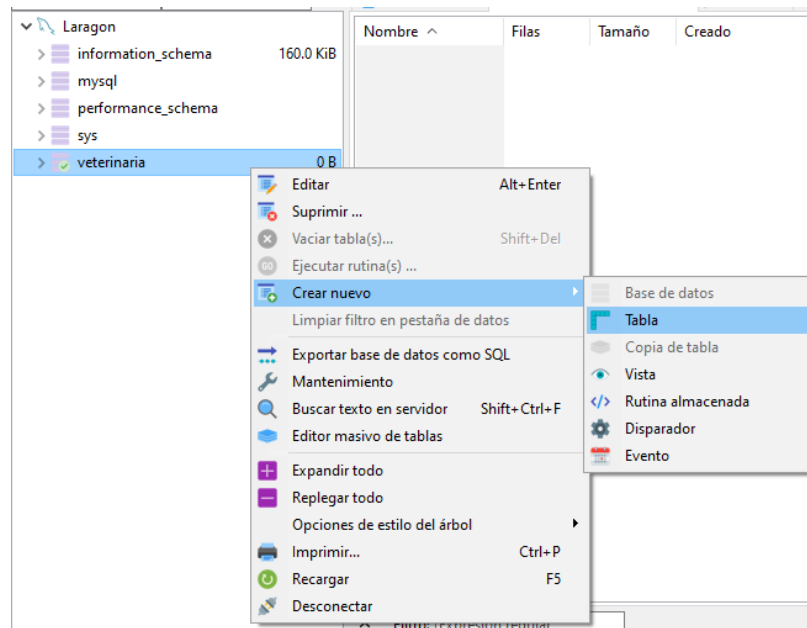


Posteriormente; aparecerá un cuadro de diálogo , en donde se escribirá el nombre de la base de datos, por lo que procederemos a aceptar la creación de base de datos.

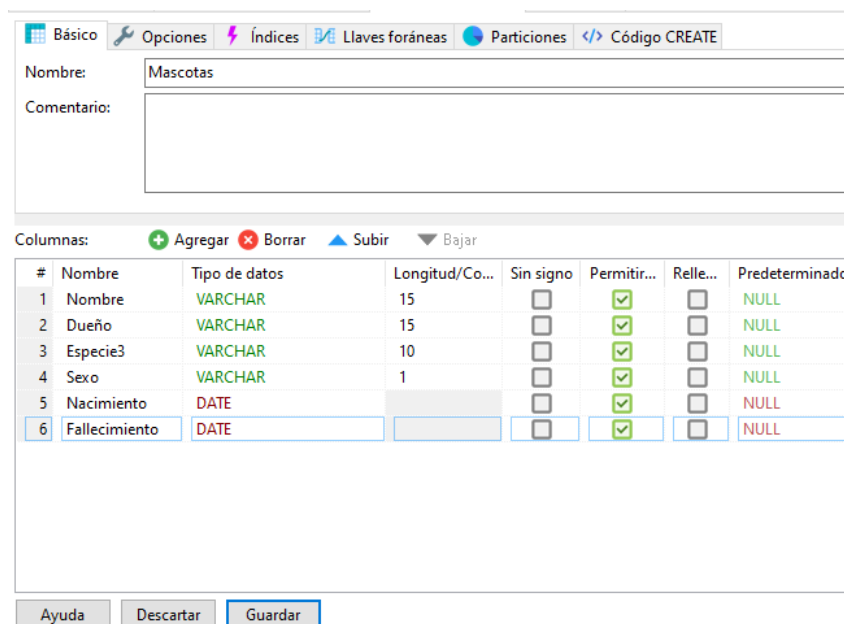


Crear tablas con Laragon

Una vez que se ha creado la base de datos, presiona con el botón derecho del ratón, en la base de datos que quieras crear una tabla, posteriormente en el menú contextual selecciona "Crear nuevo" y posteriormente selecciona "Tabla".



En el área de edición de la tabla de datos, se debe nombrar la tabla y agregar los campos necesarios, de acuerdo con el tipo de dato y si es necesario modificar sus propiedades. Para guardar la tabla, es necesario presionar el botón “Guardar”.



Si observas en la zona de instrucciones ejecutadas, observarás que se ejecutó la instrucción de MySQL para crear la tabla.

```

30 SHOW VARIABLES LIKE 'collation_database';
31 CREATE TABLE `Mascotas` ( `Nombre` VARCHAR(15) NULL DEFAULT NULL, `Dueño` VARCHAR(15) NULL DEFAULT NULL, `Especie3` VARCHAR(10) NULL DEFAULT NULL);
32 SELECT `DEFAULT_COLLATION_NAME` FROM `information_schema`.`SCHEMATA` WHERE `SCHEMA_NAME`='veterinaria';
33 SHOW TABLE STATUS FROM `veterinaria`;
34 SHOW FUNCTION STATUS WHERE `Db`='veterinaria';
35 SHOW PROCEDURE STATUS WHERE `Db`='veterinaria';
36 SHOW TRIGGERS FROM `veterinaria`;
37 SELECT *, EVENT_SCHEMA AS `Db`, EVENT_NAME AS `Name` FROM `information_schema`.`EVENTS` WHERE `EVENT_SCHEMA`='veterinaria';
38 /* El modo de tabla "Mascotas" no se encontró en el árbol. */
39 /* Entrando a la sesión "Laragon" */
40 SHOW CREATE TABLE `veterinaria`.`Mascotas`;

```

Insertar, borrar o actualizar registros con uso de Laragon

Para agregar registros en las tablas de la base de datos, debe estar seleccionada la tabla en la que desea insertar información y posteriormente seleccionar la ficha de “Datos”, por último, presionar con el botón derecho en la tabla de datos y en el menú contextual seleccionar la instrucción “Insertar fila”, esta última acción se debe repetir cada vez que se desea agregar un registro nuevo.

2. Selecciona la persiana Datos

1. Selecciona la tabla

3. Clic derecho en la tabla de datos

4. Selecciona Insertar fila

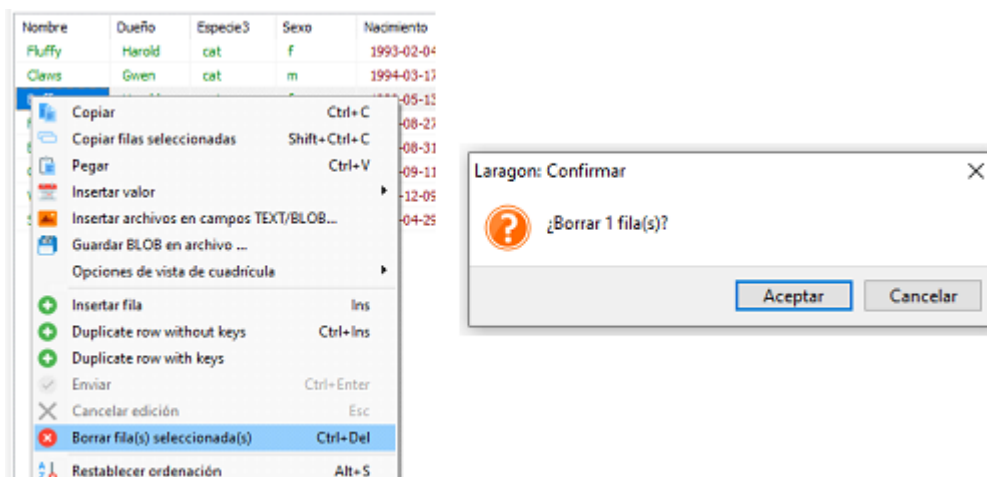
Una vez que se ha agregado una fila nueva, presiona “doble clic” en cada campo que se encuentra vacío, esto se repite hasta que se complete la base de datos, cada campo fue diseñado con un tipo de campo y sus propiedades, por ello debe contener información que respete esas propiedades.

Nombre	Dueño	Especie3	Sexo	Nacimiento	Fallecimiento
Fluffy	Harold	cat	f	1993-02-04	(NULL)
Claws	Gwen	cat	m	1994-03-17	(NULL)
Buffy	Harold	dog	f	1989-05-13	(NULL)
Fang	Benny	dog	m	1990-08-27	(NULL)
Bowser	Diane	dog	m	1979-08-31	1979-08-31
Chirpy	Gwen	bird	f	1998-09-11	(NULL)
Whistler	Gwen	bird	(NULL)	1997-12-09	(NULL)
Slim	Benny	snake	m	1996-04-29	(NULL)

Para editar un dato de algún registro en específico, se presiona “doble clic” en el dato a modificar, realizar los cambios y presionar con el botón en otra área de la pantalla y automáticamente se actualizará el dato.

Nombre	Dueño	Especie3	Sexo
Fluffy	Harold	cat	f
Claws	Gwen	cat	m
Buffy	Harold	cat	f
Fang	Benny	dog	m
Bowser	Diane	dog	m

Para eliminar un registro, se presiona “clic derecho” en el registro que se desea eliminar y en el menú contextual se selecciona la instrucción “Borrar” fila(s) seleccionada(s), aparecerá una confirmación de eliminación por lo que se selecciona si está de acuerdo o no en eliminar el registro.



Referencias

- Catherine M. Ricardo. (2009). Base de datos. México: McGraw-Hill.
- MySQL. (2020). Manual de referencia del MySQL 5.5. 20 de enero del 2021, de MySQL Sitio web: <http://download.nust.na/pub6/mysql/index.html>



¡Ponlo en práctica!

Retomando la actividad del tema anterior, el cliente desea que se incluyan información referente a los libros que se encuentran vigentes en la biblioteca, a su vez, estos libros deben estar referenciados con los diferentes autores más comunes.

Esta información se encuentra contenida de manera tradicional en una tabla sin relacionar como se muestra a continuación:

Id_libro	Título	Id_Autor	Nombre	País
1	El Guardian en el centeno	1	JD Salinger	Estados Unidos
2	Nueve historias	1	JD Salinger	Estados Unidos
3	Franny y Zooey	1	JD Salinger	Estados Unidos
4	El gran Gatsby	2	F. Scott. Fitzgerald	Estados Unidos
5	Tierna id la noche	2	F. Scott. Fitzgerald	Estados Unidos
6	Orgullo y prejuicio	3	Jane Austen	Reino Unido
7	Profesional ASP.NET 4.5 en C # y VB	4	Scott Hanselman	Estados Unidos
7	Profesional ASP.NET 4.5 en C # y VB	5	Jason N. Gaylord	Estados Unidos
7	Profesional ASP.NET 4.5 en C # y VB	6	Pranav Rastogi	India
7	Profesional ASP.NET 4.5 en C # y VB	7	Todd miranda	Estados Unidos
7	Profesional ASP.NET 4.5 en C # y VB	8	Christian Wenz	Estados Unidos

De acuerdo con la estructura de la tabla **Autores**, arma la instrucción MySQL para agregar nuevos registros, en el que se debe agregar el **Nombre** y el **País** de cada registro:

Recorta o transcribe cada una de las siguientes partes de la instrucción y arma la instrucción final.



Coloca en el siguiente recuadro el orden que debería tener la sentencia de insertar varios registros en una sola instrucción con la ayuda de los recortes anteriores:

Comprueba tu sintaxis por medio de Laragon.

Ahora realiza la instrucción de MySQL para agregar los registros de la tabla **Libros**, recuerda que solamente es de agregar los **Títulos** de los libros.

Por medio de las siguientes palabras y símbolos, completa los espacios de la siguiente instrucción de MySQL para insertar registros a la tabla **Libros**:

- Libro
- (7,
- Título
-)
- ,
- Id_Libro
- Gatsby),

- (

```

INSERT INTO _____ ( _____ , _____ )
VALUES
_____ 1 , 'The Catcher in the Rye' _____
(2, 'Nine Stories'),
(3, 'Franny and Zooey'),
(4, 'The Great _____
(5, 'Tender id the Night'),
(6, 'Pride and Prejudice'),
_____ 'Professional ASP.NET 4.5 in C# and VB')
;

```

Para concluir la relación de las tablas, se deben completar por medio de la tabla **LibrosAutor**, al completar los datos de los campos **Id_Libro** y **Id_Autor**, toma en cuenta la tabla inicial de los datos registrados y determina qué libro fue escrito por uno o varios autores.

Escribe aquí la instrucción de MySQL

Comprueba tu sintaxis por medio de **Laragon**.

Actividad 3 Actualizar o eliminar

Una vez que ya se ha presentado la información al cliente de cómo está estructurada los datos en la base de datos, el cliente se ha percatado de los siguientes problemas:

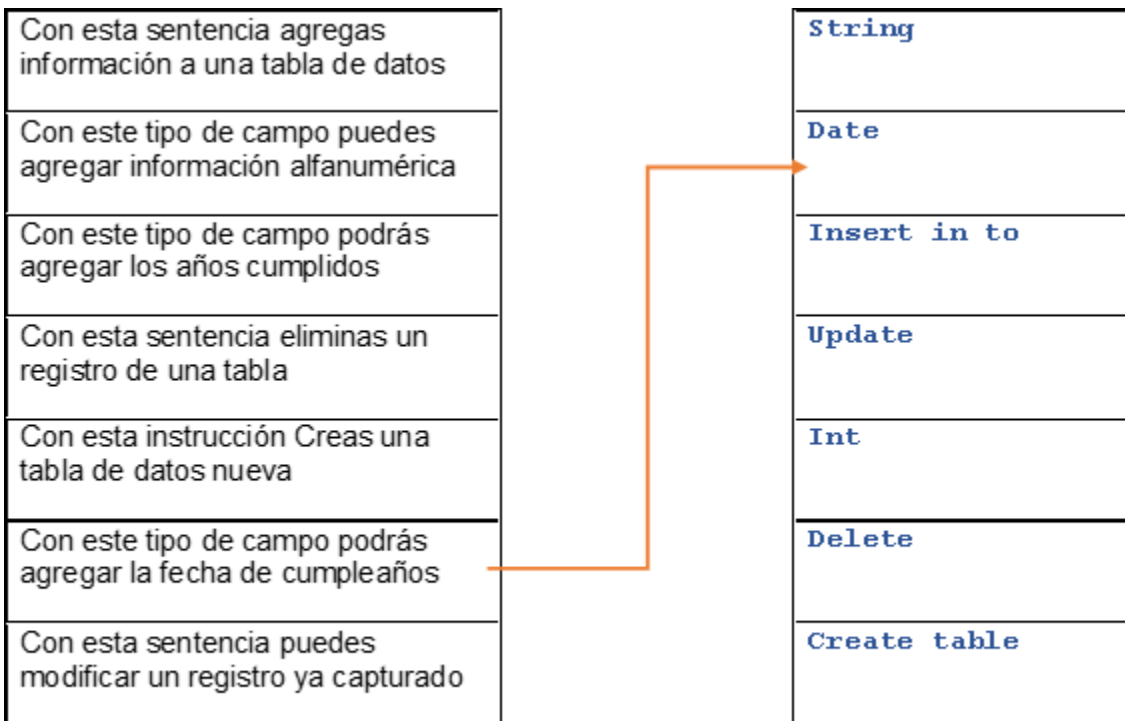
1. El título del registro número 6 en realidad es Sentido y Sensibilidad, escribe a continuación la instrucción para actualizar el registro 6, posteriormente comprueba tu sintaxis en **Laragon**.

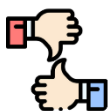
- El registro de "Christian Wenz" en la tabla de autores, en realidad este autor ya no existe en el catálogo de autores de la biblioteca, escribe a continuación la instrucción para eliminar el registro "Christian Wenz", posteriormente comprueba tu sintaxis en **Laragon**.



Actividad de aprendizaje

Une con una línea las sentencias con su descripción de uso.





Autoevaluación

Indicadores	Lo puedo hacer	Tengo dudas	Necesito trabajar más
Comprendiste la sintaxis INSERT INTO y su aplicación para agregar información nueva en una tabla de datos			
Aplicaste la sintaxis correcta para agregar información en el caso de la biblioteca			
Comprendiste la sintaxis UPDATE y su aplicación para modificar o actualizar información en una tabla de datos			
Aplicaste la sintaxis correcta para modificar información en el caso de la biblioteca			
Comprendiste la sintaxis DELETE y su aplicación para eliminar información en una tabla de datos			
Aplicaste la sintaxis correcta para eliminar información en el caso de la biblioteca			