**UNIDAD 3: Sistema Manejadores de Bases de Datos**

1. [UNIDAD 3: Introducción](https://learn.nextu.com/mod/page/view.php?id=9664&pid=P_WEB_DATABASE)

**Sistemas Manejadores de Bases de Datos**

La identificación de los datos en un contexto y situación determinada, la cual se materializa en la definición de diversas estructuras definidas o flexibles, se consolida en la definición de las bases de datos que son gestionadas por los sistemas manejadores. Una amplia gama de estos sistemas colma el mercado de productos para gestionar bases de datos; sin embargo, en esta unidad explorarás dos de los más utilizados globalmente. Comenzando con MySQL, un sistema manejador de las bases de datos relacionales y terminando en MongoDB, un sistema manejador de base de datos no relacionales orientados a documentos JSON, podrás aplicar los comandos utilizados para manejar estructuras relacionales y no relacionales usando dos de los productos disponibles en el mercado, MySQL y MongoDB.

**Objetivos de aprendizaje**

1. Identificar las funcionalidades de los sistemas manejadores de base de datos con la finalidad de familiarizarse con los productos existentes en el mercado.
2. Usar los comandos básicos para la gestión de bases de datos MySQL con la finalidad de comprender la función de estos comandos en el manejo de las bases de datos relacionales.
3. Utilizar los comandos básicos para la gestión de bases de datos NoSQL orientadas a documentos JSON con la finalidad de comprender la función de estos comandos en el manejo de las bases de datos no relacionales.
4. Lección 1: SMBD
   1. [Funcionalidades de los Sistemas Manejadores de Base de Datos](https://learn.nextu.com/mod/lesson/view.php?id=9665&pid=P_WEB_DATABASE)

Los Sistema Manejadores de DB – facilitan el control de las mismas. Es una colección de software destinado a interactuar con las mismas

Funcionalidades generales:

* Nos permite crear, actualizar y eliminar estructuras
* Nos permiten consultar, actualizar y eliminar datos

Funcionalidades de almacenamiento:

* Almacenar los datos ingresados en estructuras físicas que puedan ser almacenados
* Además herramientas para respaldar los datos y funciones para recuperar los datos.

Funcionalidades de arquitectura:

* Siguen una arquitectura cliente/servidor: varios clientes se pueden conectar a la DB al mismo tiempo, de forma concurrente y desde diferentes localidades

Driver

Configuración en el servidor:

* Dirección IP y puerto
* Número máximo de conexiones simultáneas

Funcionalidad de seguridad:

- un cliente en un DB necesitar poseer un usuario y un password para poder interactuar con el manejador

- definición de usuarios, roles y accesos. Roles generales 🡪 un grupo de usuarios



Funcionalidades de comunicación:

* Comunicación entre diferentes manejadores de DB
* O conexión con otras instancias de la misma: Replicación para poder garantizar disponibilidad de los datos

Funcionalidades de Escalamiento:

* Creación de cluster, para poder mejorar el rendimiento de la DB

Funcionalidades de Monitoreo:



Funcionalidades en respuestas a los errores

En bases de datos relacionales existen las transacciones, cada transacción es un conjunto de operación SQL ejecutadas en una secuencia, si una de ellas falla toda la transacción se debe revertir, lo que se denomina Rollback, el manejo del Rollback lo hace el manejador de base de datos.

Validación de la sintaxis, y ejecución de las sentencias o comandos.

Funcionalidades de consistencia y procesos

Existen índices que se permite que se pueda acceder más rápidamente los datos, el manejo de los índices es responsabilidad del manejador de DB.

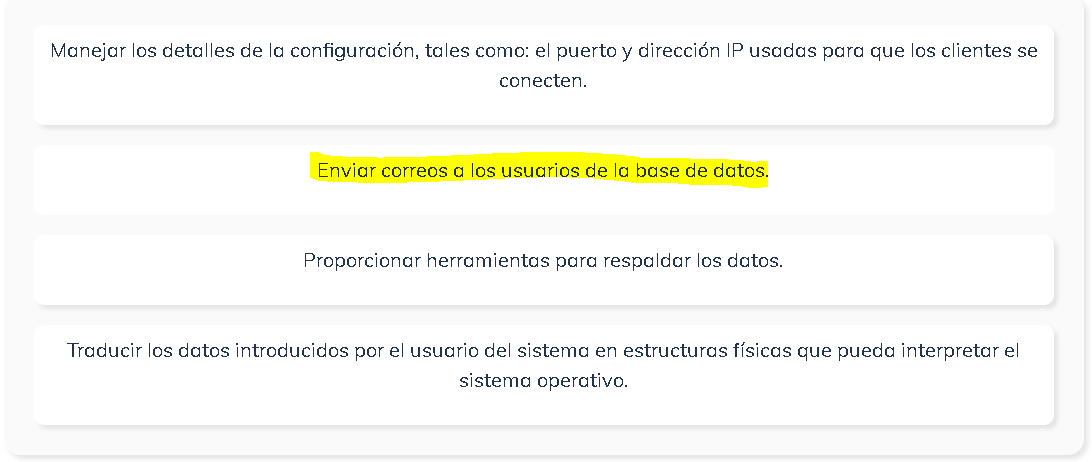
Se puede realizar el registro de un dato en una tabla cuando se realizo el registro en otra tabla. La ejecución de los procesos es responsabilidad del manejador.

Cuando se actualiza un dato y hay dependencia sobre el

* 1. Actividad Interactiva 1

**Prueba tus conocimientos**

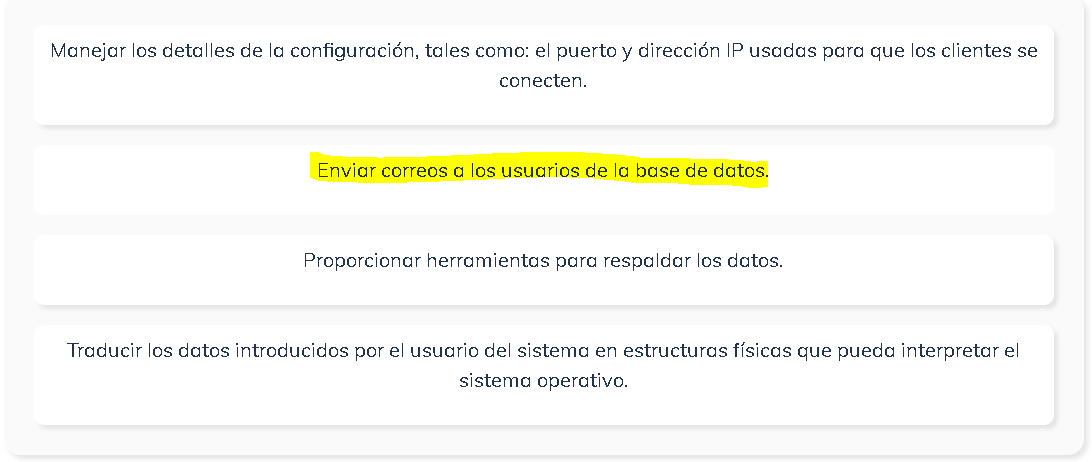
¿Cuál de las siguientes NO es una funcionalidad de un sistema manejador de base de datos?



* 1. Actividad Interactiva 2

**Prueba tus conocimientos**

¿Cuál de las siguientes NO es una funcionalidad de un sistema manejador de base de datos?



* 1. Explorando Manejadores de Base de Datos

Relacionales:

* PostgreSQL
* MySQL

No relacionales:

* MongoDB
* Redis
* Neo4J
  1. Laboratorio 1

**Instalación de MySQL**

En general los pasos para la instalación son:

Descargar el instalador

Instalar el producto

Establecer la contraseña del usuario root

Adicionalmente te proporcionamos algunos enlaces de interés al momento de instalar MySQL en tu computadora:

* [MySQL Downloads](https://dev.mysql.com/downloads/mysql/)
* [MySQL Documentation](https://dev.mysql.com/doc/)
  1. Laboratorio 2

**Instalación de MongoDB**

En general los pasos son los siguientes:

Descargar el instalador

Instalar el producto

Ejecutar el servidor

Adicionalmente te proporcionamos algunos enlaces de interés al momento de instalar MongoDB en tu computadora:

* [MongoDB Downloads](https://www.mongodb.com/download-center#atlas)
* [MongoDB Documentation](https://docs.mongodb.com/)

1. Lección 2: MySQL
   1. [Comandos MySQL: Creando una base de datos](https://learn.nextu.com/mod/lesson/view.php?id=9666&pid=P_WEB_DATABASE)

Mysql -u username -p -h localhost

Mysql>

Crear una nueva DB:

Show databases;

Create dabases registro\_autos;

STATUS para ver el estatus de la DB

USE database\_name;

Exit; para salir del prompt de mysql

* 1. Comandos MySQL: Creando una tabla

CREATE Table autos (id INT AUTO\_INCREMENT, modelo varchar(255), color varchar(255), PRIMERY KEY (id))

Show tables;

Describe autos;

* 1. Laboratorio 1

**Aprende haciendo**

**Taller de videojuegos (MySQL)**

Se te ha encargado la misión de crear una base de datos de videojuegos. En esta fase se necesita crear la tabla juegos donde se registrará el id, nombre del juego, puntos máximos y niveles. Para completar este taller debes utilizar el cliente por línea de comandos o terminal de MySQL (mysql), siguiendo los siguientes pasos:

1. Asegúrate de que tengas instalado el MySQL Server. Ten a la mano el usuario y la contraseña para conectarte.

User: root

Pass:

1. Conéctate usando el comando mysql con sus respectivos parámetros.

# mysql -u root -p

1. Lista las bases de datos disponibles.

MariaDB [(none)]> show databases;

+--------------------+

| Database |

+--------------------+

| agenda\_db |

| information\_schema |

| mysql |

| performance\_schema |

| phpmyadmin |

| test |

| transporte\_db |

+--------------------+

7 rows in set (0.007 sec)

MariaDB [(none)]>

1. Crea una base de datos llamada tienda\_videojuegos.

MariaDB [(none)]> create database tienda\_videojuegos;

Query OK, 1 row affected (0.003 sec)

MariaDB [(none)]>

1. Lista las bases de datos disponibles y verifica que tengas la tuya.

MariaDB [(none)]> show databases;

+--------------------+

| Database |

+--------------------+

| agenda\_db |

| information\_schema |

| mysql |

| performance\_schema |

| phpmyadmin |

| test |

| tienda\_videojuegos |

| transporte\_db |

+--------------------+

8 rows in set (0.001 sec)

MariaDB [(none)]>

1. Cámbiate a esa base de datos.

MariaDB [(none)]> use tienda\_videojuegos;

Database changed

MariaDB [tienda\_videojuegos]>

1. Crea una tabla llamada videojuegos con los campos que se te pidieron en el enunciado (Recuerda colocar el id autonumérico).

MariaDB [tienda\_videojuegos]> Create table juegos (id int auto\_increment, nombre\_del\_juego varchar (255), puntos\_maximo int, niveles int, primary key (id));

Query OK, 0 rows affected (0.019 sec)

MariaDB [tienda\_videojuegos]> show tables;

+------------------------------+

| Tables\_in\_tienda\_videojuegos |

+------------------------------+

| juegos |

+------------------------------+

1 row in set (0.001 sec)

MariaDB [tienda\_videojuegos]> describe juegos;

+------------------+--------------+------+-----+---------+----------------+

| Field | Type | Null | Key | Default | Extra |

+------------------+--------------+------+-----+---------+----------------+

| id | int(11) | NO | PRI | NULL | auto\_increment |

| nombre\_del\_juego | varchar(255) | YES | | NULL | |

| puntos\_maximo | int(11) | YES | | NULL | |

| niveles | int(11) | YES | | NULL | |

+------------------+--------------+------+-----+---------+----------------+

4 rows in set (0.013 sec)

MariaDB [tienda\_videojuegos]>

1. Inserta 3 filas en esa tabla.

MariaDB [tienda\_videojuegos]> Insert into juegos VALUES (null,"Juego #1",3000,10);

Query OK, 1 row affected (0.012 sec)

MariaDB [tienda\_videojuegos]> Insert into juegos VALUES (null,"Juego #2",3600,15);

Query OK, 1 row affected (0.002 sec)

MariaDB [tienda\_videojuegos]> Insert into juegos VALUES (null,"Juego #3",5000,13);

Query OK, 1 row affected (0.002 sec)

MariaDB [tienda\_videojuegos]>

1. Consulta la tabla videojuegos.

MariaDB [tienda\_videojuegos]> Select \* from juegos;

+----+------------------+---------------+---------+

| id | nombre\_del\_juego | puntos\_maximo | niveles |

+----+------------------+---------------+---------+

| 1 | Juego #1 | 3000 | 10 |

| 2 | Juego #2 | 3600 | 15 |

| 3 | Juego #3 | 5000 | 13 |

+----+------------------+---------------+---------+

3 rows in set (0.001 sec)

MariaDB [tienda\_videojuegos]>

1. Actualiza una fila y consulta la tabla para verificar los cambios.

MariaDB [tienda\_videojuegos]> update juegos SET puntos\_maximo=5300 where id=1;

Query OK, 1 row affected (0.003 sec)

Rows matched: 1 Changed: 1 Warnings: 0

MariaDB [tienda\_videojuegos]> Select \* from juegos;

+----+------------------+---------------+---------+

| id | nombre\_del\_juego | puntos\_maximo | niveles |

+----+------------------+---------------+---------+

| 1 | Juego #1 | 5300 | 10 |

| 2 | Juego #2 | 3600 | 15 |

| 3 | Juego #3 | 5000 | 13 |

+----+------------------+---------------+---------+

3 rows in set (0.002 sec)

MariaDB [tienda\_videojuegos]>

1. Elimina una fila usando el id como condición y consulta la tabla para verificar los cambios.

MariaDB [tienda\_videojuegos]> delete from juegos WHERE id=2;

Query OK, 1 row affected (0.004 sec)

MariaDB [tienda\_videojuegos]> Select \* from juegos;

+----+------------------+---------------+---------+

| id | nombre\_del\_juego | puntos\_maximo | niveles |

+----+------------------+---------------+---------+

| 1 | Juego #1 | 5300 | 10 |

| 3 | Juego #3 | 5000 | 13 |

+----+------------------+---------------+---------+

2 rows in set (0.001 sec)

MariaDB [tienda\_videojuegos]>

1. Sal de la línea de comandos de mysql.

Exit;

Esperamos que este taller haya sido de utilidad para practicar tus habilidades con los comandos de MySQL.

* 1. MySQL Workbench

Permite:

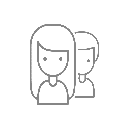


* 1. Laboratorio 2

Aprende haciendo

Promoción para clientes (MySQL Workbench)

Bienvenido a este taller práctico de MySQL. Vamos a utilizar en esta ocasión la herramienta MySQL Workbench. Para ello es necesario que lleves a cabo la siguiente actividad, la cual consiste en guardar los clientes de una tienda a fin de enviarle publicidad por correo electrónico. Se quiere guardar la información básica de los clientes: (nombre, edad y correo electrónico).



A continuación, se encuentran las actividades que debes realizar para que practiques tus conocimientos en este manejador:

1. Asegúrate de que tengas instalado el MySQL Server. Ten a la mano el usuario y la contraseña para conectarte.
2. Conéctate usando la herramienta MySQL Workbench.
3. Crea una base de datos llamada **promociones**.
4. Crea una tabla llamada **clientes**.
5. Inserta 3 filas en la tabla con los campos que se te pidieron en el enunciado.
6. Consulta la tabla **clientes**.
7. Edita alguno de los clientes cambiando su correo electrónico.
8. Elimina un cliente.

Esperamos que este taller haya sido interesante y te haya servido para practicar un poco más tus habilidades con los comandos de MySQL.

* 1. ¿Sabías qué?

**Cápsula de conocimiento**

**Resumen de comandos MySQL**

A lo largo de esta unidad hemos estudiado y utilizado diferentes comandos de MySQL, mediante los cuales creamos, editamos, eliminamos y utilizamos tanto la estructura, como la información dentro de nuestra base de datos y sabemos que es muy importante recordar la estructura de los comandos y los caracteres especiales.

A continuación, te presentamos un breve resumen con los comandos básicos que hemos estudiado hasta este momento:

* 1. Para conectarse:

mysql -u [usuario] -p -h [host]

* 1. Para ver las bases de datos disponibles:

SHOW DATABASES;

* 1. Para crear una base de datos:

CREATE DATABASE [nombre\_de\_base\_datos];

* 1. Para ver el STATUS:

STATUS;

* 1. Para cambiar la base de datos:

USE [nombre\_de\_base\_datos];

* 1. Para ver las tablas disponibles:

SHOW TABLES;

* 1. Para crear una tabla: Se debe consultar el SQL de creación y los tipos, la sintaxis general:

CREATE TABLE [nombre\_tabla] ([campo1] [tipo1], [campo2] [tipo2], ... , PRIMARY KEY(campo));

* 1. Para salir:

exit

Recuerda que existen muchas variantes de los comandos que hemos utilizado, por lo que pueden existir diferentes soluciones e implementaciones para nuestros proyectos. Igualmente existen otros comandos disponibles, algunos de los cuales aprenderemos más adelante en nuestro curso.

1. Lección 3: MongoDB
   1. [Comandos básicos de mongo](https://learn.nextu.com/mod/lesson/view.php?id=9667&pid=P_WEB_DATABASE)

Mongo

C:\Users\jrodrigue253>"C:\Program Files\MongoDB\Server\4.4\bin\mongo.exe"

> show databases;

admin 0.000GB

agenda 0.000GB

c7 0.000GB

config 0.000GB

estudiantes 0.000GB

local 0.000GB

nodeDriver 0.000GB

nuevaDB 0.000GB

users 0.000GB

>

> use registro\_autos;

switched to db registro\_autos

>

> db.createCollection("autos");

{ "ok" : 1 }

>

> db.getCollectionNames();

[ "autos" ]

>

> db.autos.insert({\_id:1,modelo:"escarabajo",color:"verde"})

WriteResult({ "nInserted" : 1 })

>

> db.autos.find();

{ "\_id" : 1, "modelo" : "escarabajo", "color" : "verde" }

>

> db.autos.insert({modelo:"escarabajo",color:"rojo"})

WriteResult({ "nInserted" : 1 })

> db.autos.find();

{ "\_id" : 1, "modelo" : "escarabajo", "color" : "verde" }

{ "\_id" : ObjectId("5ffe2851d9098c0c866b0b70"), "modelo" : "escarabajo", "color" : "rojo" }

>

> db.autos.insert({modelo:"escarabajo",color:"rojo",marca:"WV"})

WriteResult({ "nInserted" : 1 })

> db.autos.find();

{ "\_id" : 1, "modelo" : "escarabajo", "color" : "verde" }

{ "\_id" : ObjectId("5ffe2851d9098c0c866b0b70"), "modelo" : "escarabajo", "color" : "rojo" }

{ "\_id" : ObjectId("5ffe2897d9098c0c866b0b71"), "modelo" : "escarabajo", "color" : "rojo", "marca" : "WV" }

>

>exit

* 1. Laboratorio 1

**Taller de videojuegos (MongoDB)**

Se te ha encargado la misión de crear una base de datos de videojuegos, en este caso la haremos utilizando documentos JSON en MongoDB. En esta fase se necesita crear la colección juegos donde se registrará el id, nombre del juego, puntos máximos y niveles. Para completar este taller debes utilizar el cliente por la línea de comandos o terminal de MongoDB (mongo), siguiendo los siguientes pasos:

1. Asegúrate de que tengas instalado el MongoDB Server. Ten a la mano el usuario y la contraseña para conectarte.

"C:\Program Files\MongoDB\Server\4.4\bin\mongo.exe"

1. Conéctate usando el comando **mongo** con sus respectivos parámetros.

"C:\Program Files\MongoDB\Server\4.4\bin\mongo.exe"

1. Crea una base de datos llamada **tienda\_videojuegos**.

> use tienda\_videojuegos;

switched to db tienda\_videojuegos

>

1. Lista las bases de datos disponibles y verifica que tengas la tuya.
2. Cámbiate a esa base de datos.
3. Consulta la colección **videojuegos**.

> db.videojuegos.find();

>

1. Inserta 3 documentos en esa tabla con los campos que se te pidieron en el enunciado.

> db.videojuegos.insert({\_id:1,nombre\_del\_juego:"Juego #1",puntos\_maximos:1500,niveles:15})

WriteResult({ "nInserted" : 1 })

> db.videojuegos.insert({\_id:2,nombre\_del\_juego:"Juego #2",puntos\_maximos:2500,niveles:25})

WriteResult({ "nInserted" : 1 })

> db.videojuegos.insert({\_id:3,nombre\_del\_juego:"Juego #3",puntos\_maximos:3500,niveles:35})

WriteResult({ "nInserted" : 1 })

1. Consulta la tabla **videojuegos**.

> db.videojuegos.find();

{ "\_id" : 1, "nombre\_del\_juego" : "Juego #1", "puntos\_maximos" : 1500, "niveles" : 15 }

{ "\_id" : 2, "nombre\_del\_juego" : "Juego #2", "puntos\_maximos" : 2500, "niveles" : 25 }

{ "\_id" : 3, "nombre\_del\_juego" : "Juego #3", "puntos\_maximos" : 3500, "niveles" : 35 }

>

1. Sal de la línea de comandos de **mongo**.

> exit;

###### Esperamos que este taller haya sido de utilidad para practicar tus habilidades con los comandos de MongoDB.

* 1. Mongo gui client

Mongo Compass Community GUI

* 1. Laboratorio 2

**Aprende haciendo**

**Promoción para clientes**

Bienvenido a este taller práctico de MongoDB. Vamos a utilizar en esta ocasión la herramienta MongoDB Compass. Para ello es necesario que lleves a cabo la siguiente actividad, la cual consiste en guardar los clientes de una tienda a fin de enviarle publicidad por correo electrónico. Se quiere guardar la información básica de los clientes: (nombre, edad y correo electrónico).

A continuación, se encuentran las actividades que debes realizar para que practiques tus conocimientos en este manejador:

1. Asegúrate de que tengas instalado el MongoDB Server. Ten a la mano el usuario y la contraseña para conectarte.
2. Conéctate usando la herramienta **MongoDB Compass**.
3. Crea una base de datos llamada **promociones**.
4. Crea una colección llamada **clientes**.
5. Inserta 3 documentos en esa colección con los campos que se te pidieron en el enunciado.
6. Consulta la colección **clientes**.
7. Edita alguno de los clientes cambiando su correo electrónico.
8. Agrega un atributo al último cliente que guarde la información de interés del mismo. Por ejemplo (fútbol, comida, electrónicos).
9. Agrega el género a otro cliente.
10. Elimina un documento.

Esperamos que este taller haya sido interesante y te haya servido para practicar un poco más tus habilidades con los comandos de MongoDB.

* 1. ¿Sabías qué?

**Resumen de comandos MongoDB**

A lo largo de esta unidad hemos estudiado y utilizado diferentes comandos de MongoDB, mediante los cuales creamos, editamos, eliminamos y utilizamos tanto la estructura, como la información dentro de nuestra base de datos y sabemos que es muy importante recordar la estructura de los comandos y los caracteres especiales.  
  
A continuación te presentamos un breve resumen con los comandos básicos que hemos estudiado hasta este momento:

* 1. Para conectarse:

mongo

* 1. Para ver las bases de datos disponibles:

show databases

* 1. Para crear una base de datos:

use [nombre\_base\_de\_datos]

* 1. Para cambiar la base de datos actual:

use [nombre\_base\_de\_datos]

* 1. Para crear una colección:

db.createCollection("[nombre\_coleccion]")

* 1. Para ver las colecciones disponibles:

db.getCollectionNames()

* 1. Para insertar en una colección:

db.[nombre\_coleccion].insert([json])

* 1. Para ver los documentos en una colección:

db.[nombre\_coleccion].find()

* 1. Para salir:

exit

###### Recuerda que existen muchas variantes de los comandos que hemos utilizado, por lo que pueden existir diferentes soluciones e implementaciones para nuestros proyectos. Igualmente existen otros comandos disponibles, algunos de los cuales aprenderemos más adelante en nuestro curso.

1. UNIDAD 3: Prueba

|  |  |
| --- | --- |
| **Comenzado en** | Tuesday, 12 de January de 2021, 17:20 |
| **Estado** | Terminados |
| **Finalizado en** | Tuesday, 12 de January de 2021, 17:27 |
| **Tiempo empleado** | 7 mins 31 segundos |
| **Puntos** | 10/10 |
| **Calificación** | **100** de un total de 100 |

Top of Form

### Pregunta 1

Correcta

Puntúa 1 sobre 1

Señalar con bandera la pregunta

#### **Texto de la pregunta**

MySQL Workbench es \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.

Seleccione una:

a. Una herramienta soportada por una compañía independiente a MySQL para facilitar la gestión de bases de datos MySQL a través de un ambiente gráfico.

b. na herramienta que se descarga del sitio oficial de MySQL para facilitar la gestión de bases de datos MySQL a través de un ambiente gráfico.

c. Un Sistema Manejador de Bases de Datos relacionales.

d. Una herramienta que actúa como el conector entre el lenguaje de programación que estamos usando para desarrollar nuestra aplicación y la base de datos MySQL.

#### **Retroalimentación**

Muy  bien, esta herramienta es proporcionada por el sitio de MySQL

La respuesta correcta es: na herramienta que se descarga del sitio oficial de MySQL para facilitar la gestión de bases de datos MySQL a través de un ambiente gráfico.

### Pregunta 2

Correcta

Puntúa 1 sobre 1

Señalar con bandera la pregunta

#### **Texto de la pregunta**

¿Cuál de los siguientes es un producto de un Sistema Manejador de Bases de Datos No Relacionales Orientadas a Grafos?

Seleccione una:

a. Neo4j.

b. MySQL.

c. Postgres.

d. MongoDB.

#### **Retroalimentación**

Muy bien, Neo4J es de este tipo.

La respuesta correcta es: Neo4j.

### Pregunta 3

Correcta

Puntúa 1 sobre 1

Señalar con bandera la pregunta

#### **Texto de la pregunta**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ es el comando en MongoDB usado para ver las colecciones disponibles.

Seleccione una:

a. db.getCollectionNames()

b. show databases

c. db.createCollection (“[nombre\_coleccion]”)

d. db.[nombre\_coleccion].find()

#### **Retroalimentación**

Este es el comando correcto.

La respuesta correcta es: db.getCollectionNames()

### Pregunta 4

Correcta

Puntúa 1 sobre 1

Señalar con bandera la pregunta

#### **Texto de la pregunta**

¿Cuál es la funcionalidad de la tecla tabular (tab) cuando estamos en el cliente MongoDB?.

Seleccione una:

a. Mostrar todos los comandos que contienen las palabras que tipeamos antes presionar el Tab.

b. Salir del cliente MongoDB.

c. Insertar un documento predeterminado en la colección que estamos usando.

d. Borrar el último documento insertado en la colección que estamos usando.

#### **Retroalimentación**

Esta funcionalidad es muy útil

La respuesta correcta es: Mostrar todos los comandos que contienen las palabras que tipeamos antes presionar el Tab.

### Pregunta 5

Correcta

Puntúa 1 sobre 1

Señalar con bandera la pregunta

#### **Texto de la pregunta**

¿Cuál de los siguientes elementos de un sitio oficial de un producto de un Sistema Manejador de Bases de Datos es el más importante que visites cuando deseas a comenzar a usar la herramientas?.

Seleccione una:

a. Acerca (About).

b. Documentación (Documentation).

c. Tu cuenta (Your Account).

d. Soporte (Support).

#### **Retroalimentación**

La documentación permite obtener ayuda acerca del manejador.

La respuesta correcta es: Documentación (Documentation).

### Pregunta 6

Correcta

Puntúa 1 sobre 1

Señalar con bandera la pregunta

#### **Texto de la pregunta**

Supongamos que te encuentras en el terminal de comandos de tu sistema operativo , ¿cuál de las siguientes es la secuencia de comandos MongoDB correcta para conectarse al servidor de la base de datos usando parámetros predeterminados, crear una base de datos llamada programas y una colección llamada cursos e insertar un curso cuyo id es el 1, su nombre es “Base de Datos”, y su objetivo general es “Mi objetivo general”, consultar los documentos en la  colección  cursos y finalmente salir del cliente mongo? Asuma que ya está en el directorio donde se instaló MongoDB y el servidor está ejecutándose en su máquina.

Seleccione una:

a. > mongo; use programas; db.createCollection (“cursos”); db.cursos.insert ({\_id: 1, titulo:  “Base de Datos”, objetivo\_general: “Mi objetivo general”}); db.cursos.find(); exit.

b. > mongo; use programas; db.createCollection (“cursos”); db.cursos.insert (\_id: 1, titulo:  “Base de Datos”, objetivo\_general: “Mi objetivo general”); exit.

c. > mongo; use programas; db.createCollection (“cursos”); db.cursos.insert ({titulo:  “Base de Datos”, objetivo\_general: “Mi objetivo general”}); db.cursos.find(); exit.

d. > mongo; use programas; db.createCollection (“cursos”, {\_id: 1, titulo:  “Base de Datos”, objetivo\_general: “Mi objetivo general”}); db.cursos.find(); exit.

#### **Retroalimentación**

Muy bien esta es la sintaxis correcta y secuencia de comandos

La respuesta correcta es: > mongo; use programas; db.createCollection (“cursos”); db.cursos.insert ({\_id: 1, titulo:  “Base de Datos”, objetivo\_general: “Mi objetivo general”}); db.cursos.find(); exit.

### Pregunta 7

Correcta

Puntúa 1 sobre 1

Señalar con bandera la pregunta

#### **Texto de la pregunta**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_es el encargado de traducir los datos introducidos por el usuario del sistema en estructuras físicas que pueda interpretar el sistema operativo.

Seleccione una:

a. El Sistema Manejador de Base de Datos.

b. El driver del sistema de base de datos.

c. Un campo.

d. Un atributo.

#### **Retroalimentación**

Una de las responsabilidades de un SMBD es guarda la base de datos en el sistema operativo.

La respuesta correcta es: El Sistema Manejador de Base de Datos.

### Pregunta 8

Correcta

Puntúa 1 sobre 1

Señalar con bandera la pregunta

#### **Texto de la pregunta**

MongoDB Compass Community es \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.

Seleccione una:

a. Una herramienta que se descarga del sitio oficial de MySQL para facilitar la gestión de bases de datos MySQL a través de un ambiente gráfico.

b. Una herramienta que proporciona un ambiente gráfico que facilita la interacción con las bases de datos MongoDB.

c. Un Sistema Manejador de Bases de Datos MongoDB.

d. Una herramienta que actúa como el conector entre el lenguaje de programación que estamos usando para desarrollar nuestra aplicación y la base de datos MongoDB.

#### **Retroalimentación**

Es una herramient muy util para trabajar con bases de datos MongoDB.

La respuesta correcta es: Una herramienta que proporciona un ambiente gráfico que facilita la interacción con las bases de datos MongoDB.

### Pregunta 9

Correcta

Puntúa 1 sobre 1

Señalar con bandera la pregunta

#### **Texto de la pregunta**

Supongamos que te encuentras en el terminal de comandos de tu sistema operativo: ¿Cuál de las siguientes es la secuencia de comandos MySQL correcta para conectarse al servidor de la base de datos, crear una base de datos llamada programas y una tabla llamada cursos que contiene los campos: id, título y objetivo\_general, consultar la tabla recién creada y finalmente salir de MySQL? Asuma que ya está en el directorio donde se instaló MySQL y el servidor está ejecutándose en su máquina.

Seleccione una:

a. > mysql -u user -p -h localhost; CREATE DATABASE programas; USE programas; CREATE TABLE cursos (id INT AUTO\_INCREMENT, titulo VARCHAR (255),  objetivo\_general VARCHAR (255), primary key (id)); SHOW TABLES; exit.

b. > mysql -u -p -h localhost; CREATE DATABASE programas; USE programas; INSERT INTO cursos (NULL, “base de datos”, “mi objetivo”); SHOW TABLES; exit.

c. > CREATE DATABASE programas; USE programas; CREATE TABLE cursos (id INT AUTO\_INCREMENT, titulo VARCHAR (255),  objetivo\_general VARCHAR (255), primary key (id)); SHOW TABLES; exit.

d. > mysql -u -p -h localhost; CREATE DATABASE programas; USE programas; CREATE TABLE cursos (id INT AUTO\_INCREMENT, titulo VARCHAR (255),  objetivo\_general VARCHAR (255), primary key (id)); exit.

#### **Retroalimentación**

Esta es la secuencia correcta

La respuesta correcta es: > mysql -u user -p -h localhost; CREATE DATABASE programas; USE programas; CREATE TABLE cursos (id INT AUTO\_INCREMENT, titulo VARCHAR (255),  objetivo\_general VARCHAR (255), primary key (id)); SHOW TABLES; exit.

### Pregunta 10

Correcta

Puntúa 1 sobre 1

Señalar con bandera la pregunta

#### **Texto de la pregunta**

Supongamos que deseamos crear una nueva base de datos con el nombre de tienda\_mascotas ¿Cuál es el comando MySQL a usar?.

Seleccione una:

a. mysql -u root -p -h localhost

b. CREATE DATABASE tienda\_mascotas;

c. SHOW DATABASES

d. exit

#### **Retroalimentación**

El comando CREATE DATABASE es usado para este fin.

La respuesta correcta es: CREATE DATABASE tienda\_mascotas;

Bottom of Form