Individual 5.1

version de las tablas examples: MySQL dump 10.13 Distrib 8.0.19, for osx10.14 (x86\_64)

--

-- Host: 127.0.0.1 Database: world

-- ------------------------------------------------------

-- Server version 8.0.19-debug

Defina los siguientes conceptos:

- Base de datos:Una base de datos es un conjunto organizado de datos relacionados que se almacenan y se pueden acceder electrónicamente.

- Tabla:Una tabla es una estructura dentro de una base de datos que almacena los datos en filas y columnas.

- Registro:Un registro es una fila específica dentro de una tabla que contiene los datos relacionados entre sí.

- Columna:Una columna es una estructura vertical dentro de una tabla que almacena un tipo específico de datos.

- Llave primaria:Una llave primaria es un campo o conjunto de campos en una tabla que identifica de manera única cada registro en esa tabla.

- Llave foránea: Una llave foránea es un campo o conjunto de campos en una tabla que establece una relación con la llave primaria de otra tabla.

Respondan: ¿Cuál es la relación entre una llave primaria y una llave foránea?

La relación entre una llave primaria y una llave foránea es que la llave primaria de una tabla se utiliza como referencia en otra tabla a través de la llave foránea. Esto permite establecer una relación entre las dos tablas y garantiza la integridad referencial de los datos. La llave foránea enlaza los registros de una tabla con los registros correspondientes en otra tabla mediante la llave primaria.

Para comenzar este proceso, en este primer ejercicio se solicitará instalar MySQL en un entorno local.

Asumiendo que ya está instalado el aplicativo, se pide hacer lo siguiente:

Para crear un usuario llamado "explorador" en SQL, debes utilizar la siguiente sintaxis:

mysql> CREATE USER 'explorador'@'localhost' IDENTIFIED BY 'password';

Luego, para asignarle permisos de lectura y escritura en todas las tablas de la base de datos "world", puedes utilizar el siguiente comando:

GRANT SELECT, INSERT, UPDATE, DELETE ON world.\* TO 'explorador'@'localhost';

Después de esto, puedes ingresar a la base de datos con el usuario "explorador" utilizando el siguiente comando:

mysql -u explorador -p

Una vez dentro de la base de datos, puedes verificar todas las tablas visibles utilizando el siguiente comando:

SHOW TABLES;

Para obtener el nombre de cada tabla y la cantidad de registros en un archivo de texto, puedes utilizar el siguiente script en un archivo .sql:

SELECT TABLE\_NAME, TABLE\_ROWS

FROM INFORMATION\_SCHEMA.TABLES

WHERE TABLE\_SCHEMA = 'world'

INTO OUTFILE '/ruta/del/archivo.txt';

Recuerda reemplazar "/ruta/del/archivo.txt" con la ubicación y nombre de archivo deseado.

El documento de texto generado contendrá el listado de tablas y la cantidad de registros en cada caso.

3.- Usando como datos de referencia la base de datos “world” presente en MySQL Workbench, contesta

las siguientes preguntas:

● ¿Cuántas ciudades existen en la base de datos?

SELECT COUNT(\*) FROM world.city;

4079

● Si se ordenan los países de acuerdo con su nombre ¿cuál sería el primero y cuál el último?

Primero:'AFG', '**Afghanistan**', 'Asia', 'Southern and Central Asia', '652090.00', '1919', '22720000', '45.9', '5976.00', NULL, 'Afganistan/Afqanestan', 'Islamic Emirate', 'Mohammad Omar', '1', 'AF'

Ultimo:'ZWE', '**Zimbabwe**', 'Africa', 'Eastern Africa', '390757.00', '1980', '11669000', '37.8', '5951.00', '8670.00', 'Zimbabwe', 'Republic', 'Robert G. Mugabe', '4068', 'ZW'

● Tomando en cuenta todos los países de Sudamérica ¿cuántas ciudades existen entre todos ellos?

SELECT COUNT(\*) AS total\_ciudades FROM country JOIN city ON country.Code = city.CountryCode WHERE country.Region = 'South America';

470

● Considerando todos los idiomas declarados en la base, ¿cuántos idiomas corresponden a países de Sudamérica?

SELECT COUNT(DISTINCT Language) AS total\_idiomas FROM countrylanguage WHERE CountryCode IN (SELECT Code FROM country WHERE Region = 'South America');

21

5.- Dentro de MySQL Workbench existe una base de pruebas llamada “sakila”, la cual almacena datos de

arriendo de películas, junto con datos de las películas y actores.

Usando estos datos como referencia, conteste las siguientes preguntas:

● ¿Cuántas películas están registradas?

SELECT COUNT(\*) AS total\_peliculas FROM sakila.film;

1000

Esta consulta cuenta el número total de películas registradas en la tabla "film" de la base de datos "sakila".

● ¿Cuántos clientes existen en la base de datos?

SELECT COUNT(\*) AS total\_clientes FROM sakila.customer;

599

Esta consulta cuenta el número total de clientes registrados en la tabla "customer" de la base de datos "sakila".

● ¿Cuántos títulos existen en inventario?

SELECT COUNT(\*) AS total\_titulos FROM sakila.inventory;

4581

Esta consulta cuenta el número total de títulos de películas en inventario en la tabla "inventory" de la base de datos "sakila".

● ¿Cuál es la película más arrendada?

SELECT film.title, COUNT(rental.rental\_id) AS total\_arriendos FROM sakila.film JOIN sakila.inventory ON film.film\_id = inventory.film\_id JOIN sakila.rental ON inventory.inventory\_id = rental.inventory\_id GROUP BY film.title ORDER BY total\_arriendos DESC LIMIT 1;

'BUCKET BROTHERHOOD', '34'

Esta consulta realiza una unión entre las tablas "film", "inventory" y "rental" para obtener el número de arriendos por película. Luego, se agrupa por el título de la película y se ordena de forma descendente por el número total de arriendos. Finalmente, se limita el resultado a la película con el mayor número de arriendos.

El resultado de esta consulta te dará el título de la película más arrendada en la base de datos "sakila".

Esta consulta cuenta el número de arriendos por película en la tabla "rental" de la base de datos "sakila" y luego ordena los resultados de forma descendente. La película más arrendada será la que tenga el mayor número de arriendos.

6.- Luego de utilizar MySQL Workbench, indaguen en el uso de MySQL Shell. ¿Qué diferencias existen

entre utilizar una u otra plataforma?

MySQL Workbench y MySQL Shell son dos herramientas diferentes para interactuar con la base de datos MySQL. A continuación, te mencionaré algunas diferencias entre ambas:

1. Interfaz gráfica vs. línea de comandos: MySQL Workbench es una herramienta con una interfaz gráfica que proporciona una forma visual de administrar y trabajar con la base de datos. Por otro lado, MySQL Shell es una interfaz de línea de comandos que permite interactuar con la base de datos a través de comandos y scripts.

2. Funcionalidades: MySQL Workbench ofrece una amplia gama de funcionalidades, como el diseño de bases de datos, la creación y ejecución de consultas SQL, la administración de usuarios y permisos, la visualización de esquemas y la generación de informes. MySQL Shell, por otro lado, se centra más en la administración y la automatización de tareas, así como en la ejecución de consultas y scripts.

3. Lenguajes de programación: MySQL Workbench es compatible con varios lenguajes de programación, como SQL, Python y JavaScript, lo que permite la integración con diferentes tecnologías y frameworks. MySQL Shell, por su parte, se basa en JavaScript y Python, lo que brinda flexibilidad para interactuar con la base de datos utilizando estos lenguajes.

4. Soporte para protocolos: MySQL Workbench utiliza principalmente el protocolo MySQL para la comunicación con el servidor de base de datos. MySQL Shell, por otro lado, admite tanto el protocolo MySQL como el protocolo X Protocol, que permite la comunicación a través de un canal seguro y proporciona funciones adicionales, como la ejecución de consultas en paralelo.

En resumen, MySQL Workbench es una herramienta más orientada a la administración y visualización de la base de datos, mientras que MySQL Shell es más flexible y se centra en la automatización y ejecución de tareas a través de la línea de comandos. La elección entre una u otra plataforma dependerá de tus necesidades y preferencias personales.