**RETO 1: BASES DE DATOS RELACIONALES ------ TRAINING QA-------**

**NATALIA GARCIA BAHAMÓN**

**Ejercicios propuestos**

**1.** En una conexión local de base de datos, crear la base de datos llamada Sakila:

**a.** Descomprimir el archivo “sakila-db.zip”.

Una captura de pantalla de una computadora

Descripción generada automáticamente

Captura de pantalla de un celular

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente

**b**. Importar el script “sakila-schema.sql” y ejecutar las sentencias.

Una captura de pantalla de una computadora

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

c. Importar el script “sakila-data.sql” y ejecutar las sentencias.

Una captura de pantalla de una computadora

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

**d.** Ejecutar el archivo “sakila.mwb” para análisis posteriores:

Una captura de pantalla de una computadora

Descripción generada automáticamente

**2.** Indique cuáles son las relaciones uno a uno, uno a muchos, muchos a muchos.

* Relación uno a uno: No hay
* Relación n-m:

Categoria- film\_category

* Relación uno a muchos:

Film-Inventory

Inventory-Rental

Film-Language

Address-Store

City-Adress

Country-city

Adress-Customer

Adress-Staff

Staff-Store

Staff-Payment

Customer-Payment

Customer-Rental

Rental-Payment

Store-Inventory

Store-Customer

Store-Staff

**3.** Cuales son las claves primarias y foráneas de las tablas: Actor, Film, Customer, Staff:

* Actor:
* actor\_id🡪Clave primaria
* No tiene claves foráneas
* Film:
* film\_id🡪 Clave primaria
* original\_language\_id 🡪Clave foránea
* language\_id🡪Clave foránea
* Customer:
* customer\_id🡪 Clave primaria
* store\_id🡪Clave foránea
* address\_id🡪Clave foránea
* Staff:
* staff\_id🡪Clave primaria
* address\_id🡪Clave foránea
* store\_id🡪Clave foránea

**4.** Contar y consultar todos los actores que existen en la BD usando sentencias SQL.

SELECT \* FROM actor; -- Buscar todos los actores

Una captura de pantalla de una computadora

Descripción generada automáticamente

SELECT COUNT(\*) AS "Actores existentes" FROM actor; -- Contar a los actores

Una captura de pantalla de una computadora

Descripción generada automáticamente

**5.** Consultar cuáles actores participan en las películas ACADEMY DINOSAUR, BERETS AGENT.

SELECT n.first\_name AS "Nombre", n.last\_name AS "Apellido", film.title AS "Pelicula" FROM actor n JOIN film\_actor fi ON n.actor\_id = fi.actor\_id JOIN film ON film.film\_id = fi.film\_id WHERE title="ACADEMY DINOSAUR" OR title = "BERETS AGENT";

Una captura de pantalla de una computadora

Descripción generada automáticamente

**6.** Consultar en cuáles películas ha participado KARL BERRY y cuál es la categoría de dichas películas.

Una captura de pantalla de una computadora

Descripción generada automáticamente

**7.** Consultar el costo promedio de (replacement\_cost) para las películas cuya categoría sea Drama.

SELECT c.name AS "Categoria",(sum(film.replacement\_cost)/count(film.replacement\_cost)) AS "Costo Promedio" FROM film JOIN film\_category fc ON fc.film\_id = film.film\_id JOIN category c ON fc.category\_id = c.category\_id WHERE c.name = "Drama";

Una captura de pantalla de una computadora

Descripción generada automáticamente

**8.** Consultar el total de ventas de cada store por categoría de película.

SELECT \* FROM payment;

SELECT \* FROM sales\_by\_store;

SELECT c.name "Categoria",co.country AS "País de la tienda",ct.city AS "Ciudad de la tienda",(count(p.amount)) AS "Total ventas"

FROM payment p

JOIN rental r ON p.rental\_id = r.rental\_id

JOIN inventory i ON r.inventory\_id = i.inventory\_id

JOIN film f ON i.film\_id = f.film\_id

JOIN film\_category fc ON f.film\_id = fc.film\_id

JOIN category c ON fc.category\_id = c.category\_id

JOIN store s ON i.store\_id = s.store\_id

JOIN address a ON s.address\_id = a.address\_id

JOIN city ct ON a.city\_id = ct.city\_id

JOIN country co ON ct.country\_id = co.country\_id

group by c.name, s.store\_id;

Una captura de pantalla de una computadora

Descripción generada automáticamente

9. Contar todos los clientes existentes por País.

SELECT count(name) AS "Cantidad de Cliente", country AS "País" FROM customer\_list group by country;

Una captura de pantalla de una computadora

Descripción generada automáticamente

10. Crear una tabla temporal que contenga las películas que cada cliente ha alquilado.

DROP VIEW IF EXISTS peliculas\_cliente;

CREATE VIEW peliculas\_cliente AS

SELECT cu.first\_name AS "Nombre",cu.last\_name AS "Apellido", f.title AS "Pelicula"

FROM customer cu

JOIN store s ON cu.store\_id = s.store\_id

JOIN inventory i ON s.store\_id =i.store\_id

JOIN film f ON i.film\_id = f.film\_id

order by cu.last\_name;

SELECT \* FROM peliculas\_cliente;

Una captura de pantalla de una computadora

Descripción generada automáticamente

**11.** Consulte el tiempo máximo en días que cada cliente ha tenido una película.

SELECT cu.first\_name AS "Nombre",cu.last\_name AS "Apellido", f.title AS "Pelicula", datediff(r.return\_date,r.rental\_date) AS "Tiempo máximo de días"

FROM customer cu

JOIN rental r ON cu.customer\_id = r.customer\_id

JOIN inventory i ON r.inventory\_id =i.inventory\_id

JOIN film f ON i.film\_id = f.film\_id;

Una captura de pantalla de una computadora

Descripción generada automáticamente

12. Realice el siguiente ejercicio de acuerdo a la siguiente definición de negocio y tener presente:

**a.** Diseñar el model entidad relación.

Diagrama, Esquemático

Descripción generada automáticamente

**b.** Crear el modelo lógico debidamente normalizado.

Interfaz de usuario gráfica, Diagrama, Aplicación, Tabla, Excel

Descripción generada automáticamente