

REPUBLICA DOMINICANA
INSTITUTO TECNOLOGICO DE LAS AMERICAS



INTRODUCCION A LA BASE DE DATOS

SECCION:

2025-C-1-1746-3090-TCD-005

PRACTICA No. 5

DATA CONTROL LANGUAGE

Rosy Elania Arvelo Pérez

20240861

Rolando Oscar Garcia Guzman

Práctica: Uso de DCL en SQL

Objetivo de la Práctica

Desarrollar habilidades para gestionar permisos y control de acceso en bases de datos utilizando los comandos DCL (GRANT y REVOKE).

Descripción del Escenario

Eres el administrador de una base de datos para una empresa de comercio electrónico. La base de datos contiene información sensible sobre clientes, productos y ventas. Es necesario configurar permisos específicos para diferentes roles de usuarios:

- **Administrador:** Tiene acceso total a la base de datos.
- **Analista de Datos:** Solo puede consultar la información de ventas y clientes.
- **Equipo de Ventas:** Puede agregar nuevas ventas, pero no modificar datos existentes.

Instrucciones

1. Creación de la Base de Datos y Tablas

Crea una base de datos llamada comercio y las siguientes tablas con datos de ejemplo:

```
“sql
CREATE DATABASE comercio;
USE comercio;
```

```
CREATE TABLE clientes (
id INT PRIMARY KEY,
nombre VARCHAR(50),
correo VARCHAR(50),
telefono VARCHAR(15) );
```

```
CREATE TABLE ventas (
id INT PRIMARY KEY,
cliente_id INT,
producto VARCHAR(50),
monto DECIMAL(10, 2),
fecha DATE );”
```

2. Insertar Datos de Ejemplo

Llena las tablas con datos ficticios:

```
“sql
Introducción a Base de Datos
INSERT INTO clientes VALUES (1, 'Juan Perez', 'juan@example.com', '8091234567');
INSERT INTO ventas VALUES (1, 1, 'Laptop', 1200.50, '2024-01-01');”
```

3. Crear Usuarios

Crea usuarios ficticios para cada rol:

```
“sql  
CREATE USER 'admin'@'localhost' IDENTIFIED BY 'admin123';  
CREATE USER 'analista'@'localhost' IDENTIFIED BY 'analista123';  
CREATE USER 'ventas'@'localhost' IDENTIFIED BY 'ventas123'; “
```

4. Asignar Permisos

Utiliza el comando GRANT para asignar permisos:

- **Administrador:** Acceso completo.

```
“sql / GRANT ALL PRIVILEGES ON comercio.* TO 'admin'@'localhost' WITH GRANT  
OPTION;”
```

- **Analista de Datos:** Solo consulta las tablas clientes y ventas.

```
“sql / GRANT SELECT ON comercio.clientes TO 'analista'@'localhost';  
GRANT SELECT ON comercio.ventas TO 'analista'@'localhost';”
```

- **Equipo de Ventas:** Permiso para insertar datos en ventas.

```
"sql / GRANT INSERT ON comercio.ventas TO 'ventas'@'localhost';”
```

5. Revocar Permisos

Imagina que decides revocar el permiso de inserción para el usuario ventas. Usa el commando REVOKE:

```
“sql / REVOKE INSERT ON comercio.ventas FROM 'ventas'@'localhost';“
```

Tareas

1. Verifica los permisos asignados utilizando los comandos:

```
“sql / SHOW GRANTS FOR 'usuario'@'localhost';“
```

2. Intenta realizar operaciones no autorizadas con cada usuario y documenta los errores obtenidos.

3. Configura un nuevo usuario llamado gestor con permisos de consulta y actualización en la tabla clientes.

Desarrollo:

Creación de la Base de Datos y Tablas

```
CREATE DATABASE comercio;
```

```
USE comercio;
```

```
CREATE TABLE clientes (  
    id INT PRIMARY KEY,  
    nombre VARCHAR(50),  
    correo VARCHAR(50),  
    telefono VARCHAR(15)  
);
```

```
CREATE TABLE ventas (  
    id INT PRIMARY KEY,  
    cliente_id INT,  
    producto VARCHAR(50),  
    monto DECIMAL(10, 2),  
    fecha DATE,  
    FOREIGN KEY (cliente_id) REFERENCES clientes(id)  
);
```

Datos de Ejemplo

```
INSERT INTO clientes VALUES (1, 'Juan Perez', 'juan@example.com', '8091234567');
```

```
INSERT INTO clientes VALUES (2, 'Maria Gomez', 'maria@example.com', '8299876543');
```

```
INSERT INTO ventas VALUES (1, 1, 'Laptop', 1200.50, '2024-01-01');
```

```
INSERT INTO ventas VALUES (2, 2, 'Celular', 850.00, '2024-02-15');
```

Crear Usuarios y Asignar Permisos

Administrador (Acceso completo)

```
CREATE USER 'admin'@'localhost' IDENTIFIED BY 'admin123';
```

```
GRANT ALL PRIVILEGES ON comercio.* TO 'admin'@'localhost' WITH GRANT  
OPTION;
```

Analista de Datos (Solo lectura en clientes y ventas)

```
CREATE USER 'analista'@'localhost' IDENTIFIED BY 'analista123';
```

```
GRANT SELECT ON comercio.clientes TO 'analista'@'localhost';
```

```
GRANT SELECT ON comercio.ventas TO 'analista'@'localhost';
```

Equipo de Ventas (Solo inserción en ventas)

```
CREATE USER 'ventas'@'localhost' IDENTIFIED BY 'ventas123';
```

```
GRANT INSERT ON comercio.ventas TO 'ventas'@'localhost';
```

Revocar Permisos

```
REVOKE INSERT ON comercio.ventas FROM 'ventas'@'localhost';
```

Verificar Permisos

```
SHOW GRANTS FOR 'analista'@'localhost';
```

```
SHOW GRANTS FOR 'ventas'@'localhost';
```

```
SHOW GRANTS FOR 'admin'@'localhost';
```

Prueba de Operaciones no Autorizadas

```
INSERT INTO ventas VALUES (3, 1, 'Tablet', 500.00, '2024-03-10');
```

ERROR 1142 (42000): INSERT command denied to user 'analista'@'localhost' for table 'ventas'

Nuevo Usuario "Gestor" con Permisos de Consulta y Actualización en Clientes

```
CREATE USER 'gestor'@'localhost' IDENTIFIED BY 'gestor123';
```

```
GRANT SELECT, UPDATE ON comercio.clientes TO 'gestor'@'localhost';
```

Evaluación

1. ¿Qué sucede si intentas que el usuario analista inserte un registro en la tabla ventas?

El usuario analista no tiene permisos de inserción en ventas, por lo que mostrará un error.

2. Describe el proceso para otorgar permisos de delegación a un usuario.

Para permitir que un usuario pueda otorgar permisos a otros, se usa WITH GRANT OPTION.

3. Explica por qué es importante revocar permisos cuando ya no son necesarios.

Es importante para:

- Proteger datos sensibles.
- Evitar accesos no autorizados.
- Cumplir con normativas de seguridad.
- Reducir el riesgo de errores humanos o ataques malintencionados.

4. Diseña una estrategia de permisos para un escenario donde múltiples analistas puedan acceder a datos sensibles, pero solo un supervisor pueda modificarlos.

- **Analistas:** Solo lectura en tablas sensibles.
- **Supervisor:** Puede modificar datos, pero no cambiar la estructura de las tablas.