# LABORATORIO 2 ADMINISTRACION DE BASES DE DATOS

# EJERCICIOS PROPUESTOS

# PREREQUISITO

- La base de datos DB\_Gimnasio y sus tablas deben estar creadas previamente en el schema dbo.  
- Se debe iniciar sesión en SQL Server como usuario con permisos administrativos (sa).  
- Cada grupo de laboratorio solo debe resolver UNA de las claves propuestas.

# ENTREGABLES

1. Un archivo .sql con la creación ordenada de logins, usuarios, roles y pruebas.  
2. Documentar cada paso con comentarios en el código.  
3. Una captura de pantalla del error que aparece al intentar un DELETE (DENY DELETE).

# Ejercicio 1: Aplicación con CRUD parcial

Objetivo:  
- Crear un login/usuario de aplicación llamados ex1\_app\_gym y un rol ex1\_db\_app.  
- Los usuarios de aplicación deben tener CHECK\_EXPIRATION = OFF.  
- Al rol se le conceden permisos SELECT, INSERT y UPDATE sobre las tablas dbo.Cliente y dbo.Membresia.  
- Se debe negar la eliminación (DELETE) en dbo.Cliente con DENY DELETE.

Pruebas a realizar:  
- Ejecutar un INSERT en dbo.Cliente (debe funcionar).  
- Ejecutar un UPDATE en dbo.Cliente (debe funcionar).  
- Intentar ejecutar un DELETE en dbo.Cliente (debe fallar).  
- Tomar una captura de pantalla del error al realizar el DELETE.

# Ejercicio 2: Capturista de Membresías sin modificaciones

Objetivo:  
- Crear un login/usuario de aplicación llamados ex2\_data\_entry y un rol ex2\_db\_entry.  
- Los usuarios de aplicación deben tener CHECK\_EXPIRATION = OFF.  
- Al rol se le conceden:  
 \* SELECT sobre dbo.Cliente (para consultar clientes existentes).  
 \* SELECT e INSERT sobre dbo.Membresia (para registrar nuevas membresías).  
- Se debe negar la posibilidad de UPDATE y DELETE en dbo.Membresia.

Pruebas a realizar:  
- Ejecutar un SELECT en dbo.Cliente (debe funcionar).  
- Ejecutar un INSERT en dbo.Membresia (debe funcionar).  
- Intentar ejecutar un UPDATE en dbo.Membresia (debe fallar).  
- Intentar ejecutar un DELETE en dbo.Membresia (debe fallar).  
- Tomar una captura de pantalla del error al realizar el DELETE.

# Ejercicio 3: Backoffice de Membresías con actualización controlada

Objetivo:  
- Crear un login/usuario de backoffice llamados ex3\_backoffice y un rol ex3\_db\_backoffice.  
- Los usuarios de backoffice deben tener CHECK\_EXPIRATION = ON (contraseña con vencimiento).  
- Al rol se le conceden:  
 \* SELECT sobre dbo.Cliente (para consultar clientes).  
 \* SELECT, INSERT y UPDATE sobre dbo.Membresia (para gestionar membresías).  
- Se debe negar la posibilidad de DELETE en dbo.Membresia.

Pruebas a realizar:  
- Ejecutar un SELECT en dbo.Cliente (debe funcionar).  
- Ejecutar un INSERT en dbo.Membresia (debe funcionar).  
- Ejecutar un UPDATE en dbo.Membresia (debe funcionar).  
- Intentar ejecutar un DELETE en dbo.Membresia (debe fallar).  
- Tomar una captura de pantalla del error al realizar el DELETE.

# NOTA FINAL

Recuerde usar EXECUTE AS USER para realizar las pruebas con cada usuario creado, y regresar al contexto original con REVERT después de cada prueba.