# 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 y trabajar de forma ordenada el código.  
3. Una captura de pantalla del error que aparece al intentar un DELETE (DENY DELETE). 4. Agregar comentarios indicando como se fue realizando el ejercicio.

**Ejercicio 1: Aplicación con CRUD parcial**

Se requiere la creación de un login y usuario de aplicación denominados **ex1\_app\_gym**, así como de un rol **ex1\_db\_app** en SQL Server.  
El usuario debe configurarse con la opción **CHECK\_EXPIRATION = OFF**, ya que corresponde a una cuenta de aplicación.  
El rol asociado debe contar con permisos **SELECT, INSERT y UPDATE** sobre las tablas dbo.Cliente y dbo.Membresia.  
Adicionalmente, se debe **negar explícitamente el permiso DELETE** en la tabla dbo.Cliente.

**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 tratar de ejecutar el DELETE.

**Ejercicio 2: Capturista de Membresías sin modificaciones**

Se solicita crear un login y usuario de aplicación llamados **ex2\_data\_entry** junto con un rol denominado **ex2\_db\_entry**.  
Al ser un usuario de aplicación, la opción **CHECK\_EXPIRATION = OFF** debe establecerse.  
El rol debe disponer de permisos de **SELECT sobre la tabla dbo.Cliente**, así como **SELECT e INSERT sobre la tabla dbo.Membresia**.  
Para garantizar la integridad de la información, se deben **denegar los permisos UPDATE y DELETE** en la tabla 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 tratar de ejecutar el DELETE.

**Ejercicio 3: Backoffice de Membresías con actualización controlada**

Se requiere implementar un login y usuario de backoffice denominados **ex3\_backoffice** y un rol **ex3\_db\_backoffice**.

En este caso, el usuario debe crearse con la opción **CHECK\_EXPIRATION = ON**, dado que las cuentas de backoffice deben someterse a vencimiento de contraseñas.  
El rol tendrá permisos de **SELECT sobre dbo.Cliente** y de **SELECT, INSERT y UPDATE sobre dbo.Membresia**. Para mantener control sobre la información, se debe **denegar el permiso 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 tratar de ejecutar 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.