

# Usuarios y permisos

El primer paso para configurar la seguridad en el servidor es crear los inicios de sesion con las siguientes instrucciones

## Creacion de inicios de sesion

### ▼ CREATE LOGIN login\_name WITH

login name: Especifica el nombre de inicio de sesion que se va a crear

### ▼ PASSWORD = 'password' [HASHED][MUST\_CHANGED],

password = 'password': Solo se aplica a inicios de sesion de SQL Server

password = hashed\_password: Solo se aplica a la palabra clave HASHED, especifica el valor con hash de la clave para el inicio de sesion que se esta creando

hashed: Solo se aplica a inicios de sesion SQL Server. Especifica que la clave especificada despues del argumento PASSWORD ya tiene aplicado el algoritmo hash.

must\_change: Solo se aplica a inicios de sesion SQL Server. Si se incluye esta opcion, SQL Server pide al usuario la clave nueva la primera vez que se utilice el inicio de sesion nuevo.

### ▼ SID = sid|

SID = sid: Solo se aplica a inicios de sesion SQL Server. Especifica el GUID del inicio de sesion de SQL Server nuevo.

### ▼ DEFAULT\_DATABASE = database|

default\_database = database: Especifica la base de datos predeterminada que debe asignarse al inicio de sesion, si no se incluye esta opcion el valor predeterminado es master.

### ▼ DEFAULT\_LANGUAGE = language|

Especificas el idioma predeterminado que debe asignarse al inicio de sesion. Si no se incluye esta opcion, el idioma predeterminado es el del servidor.

▼ CHECK\_EXPIRATION = {ON | OFF}

Solo se aplica a inicios de sesion SQL Server, especifica si debe aplicarse la directiva de expiracion de claves en este inicio de sesion

▼ CHECK\_POLICY = {ON | OFF}

Sólo se aplica a inicios de sesión de SQL Server. Especifica que se deben aplicar las directivas de contraseñas de Windows en el equipo que ejecuta SQL Server para este inicio de sesión

## Modificacion de inicios de sesion

▼ ALTER LOGIN login\_name

login\_name: Especifica el nombre del inicio de sesion de SQL Server que se esta cambiando

▼ ENABLE | DISABLE | WITH

ENABLE | DISABLE: Habilita o desabilita este inicio de sesion

▼ PASSWORD = 'password' | hashed\_password

PASSWORD = 'password': Solo se aplica a inicios de sesion de SQL Server. Especifica la clave del inicio de sesion que se esta cambiando (distingue entre mayusculas y minusculas)

PASSWORD = hashed\_password: Solo se aplica a la palabra clave HASHED. Especifica el vvalor con algoritmo hash para la clave del inicio de sesion que se crea.

▼ HASHED

HASHED: Solo aplica a inicios de sesion SQL Server. Especifica que la clave especificada despues del argumento PASSWORD ya que tiene aplicado el algoritmo hash.

▼ [OLD\_PASSWORD = 'oldpassword' |

OLD\_PASSWORD = 'oldpassword': Solo aplica a inicios de sesion de SQL Server. La clave actual del inicio de sesion al que se va a asignar una clave nueva.

▼ MUST\_CHANGE | UNLOCK

MUST\_CHANGE: Solo se aplica a inicios de sesion de SQL Server. Si se incluye esta opcion, SQL Server pedira la clave actualizada la primera vez que se utilice el inicio de sesion modificado

**IMPORTANTE: MUST\_CHANGE SOLO SE PUEDE USAR CUANDO  
CHECK\_EXPIRATION=ON**

UNLOCK: Solo se aplica a inicios de SQL Server. Especifica que un inicio de sesion bloqueado debe desbloquearse.

▼ DEFAULT\_DATABASE = database |

Especifica una base de datos predeterminada que debe asignarse al inicio de sesion

▼ LANGUAGE\_DEFAULT = language |

Especifica el idioma predeterminado que debe asignarse al inicio de sesion

▼ NAME = login\_name |

Especifica el nombre del nuevo inicio de sesion al que se esta cambiando el nombre

▼ CHECK\_POLICY = {ON | OFF}

Sólo se aplica a inicios de sesión de SQL Server. Especifica que se deben aplicar las directivas de contraseñas de Windows en el equipo que ejecuta SQL Server para este inicio de sesión.

▼ CHECK\_EXPIRATION = {ON | OFF}

Sólo se aplica a inicios de sesión de SQL Server. Especifica si debe cumplirse la directiva de caducidad de contraseñas en este inicio de sesión.

## USUARIOS DE BASE DE DATOS

Despues de configurar la seguridad de inicio de sesion y establecer la clave, se continua con la configuracion de acceso a las bases de datos, ya que, tener una clave de inicio de sesion a SQL Server no le proporciona acceso a ninguna BD en el servidor

Darle acceso a un inicio de sesion de sesion como usuario de una base de datos:

USE NombreBD

CREATE USER user\_name FOR LOGIN login\_name (Es recomendable tener el mismo nombre de inicio de sesion como usuario de Base de datos)

USE NombreBD

CREATE USER login\_name

Eliminacion de un usuario en una base de datos:

USE NombreBD

DROP USER user\_name

Listar los usuarios de una BD:

USE NombreBD

sp\_HelpUser['NombreUsuarioBD']

## Nombre de usuario GUEST (Invitado)

Este nombre de usuario existe en todas las BD, se usa para que se conecten todos los inicios de sesion del servidor sin tener que darle acceso individual a cada uno de ellos, como su nombre lo indica, es para usuarios invitados. Es necesario activarlo para poder ser utilizado

USE NombreBD

GRANT CONNECT TO GUEST

Para desactivarlo es necesario revocar los permisos al usuario GUEST, solo se desactiva pero no se elimina

USE NombreBD

REVOKE CONNECT FROM GUEST

# Funciones

Las funciones de SQL Server permiten agrupar a usuarios de BD o inicios de sesion para desarrollar ciertas operaciones a nivel BD o nivel servidor. Los miembros de cada funcion tendran derecho a realizar ciertas tareas o permisos sobre ciertos objetos.

- Funciones fijas a nivel servidor
- Funciones fijas a nivel BD
- Funciones creadas por usuarios a nivel BD

## Funciones fijas a nivel servidor

### ▼ SysAdmin

Los miembros de esta funcion pueden hacer TODO en el servidor, aparecen como el DBO de cada BD

### ▼ ServerAdmin

Los miembros pueden asignar opciones de configuracion con el procedimiento SP\_CONFIGURE y puede apagar el servidor con el comando SHUTDOWN. Los operadores deben agregarse a esta funcion

### ▼ SetupAdmin

Los miembros pueden instalar y configurar servidores vinculados y marcar un procedimiento almacenado para su ejecucion durante el arranque.

### ▼ SecurityAdmin

Los miembros pueden crear y controlar claves de inicio de sesion al servidor al igual que permisos para crear BD para poder leer el registro de errores.

### ▼ ProcessAdmin

Los miembros controlan los procesos en ejecucion en el servidor de datos esto suele ser eliminar las consultas descarriadas y proceso de bloqueos.

### ▼ DbCreator

Los miembros pueden crear y alterar bases de datos en el servidor al igual que restaurarlas.

▼ DiskAdmin

Los miembros pueden manejar los archivos y su incremento en el servidor, tambien pueden crear copias de seguridad

▼ BulkAdmin

Los miembros pueden ejecutar inserciones masivas coon le comando bullcopy desde MS-DOS

▼ sp\_AddSrvRoleMember 'Inicio Sesion', 'NombreFuncion'

Asignacion de un inicio de sesion a funciones fijas a nivel servidor

▼ sp\_DropSrvRoleMember 'Inicio Sesion', 'Nombre Funcion'

Eliminacion de un inicio de sesion

▼ sp\_HelprsvRoleMeer 'Inicio Sesion', 'Nombre Funcion'

Muetra los inicios de sesion (hagan cuenta que es como hacer un select de una tabla completa)

▼ sp\_HelprsvRole 'Nombre Funcion'

Muestra las propiedades de una funcion

## Funciones fijas a nivel BD

▼ db\_owner

Los miembros pueden hacer TODO, pero solo dentro de su BD

▼ db\_AccessAdmin

Los miembros pueden agregar y quitar usuarios el acceso a la BD

▼ db\_SecurityAdmin

Los miembros pueden controlar todos los permisos, funciones, membresia de funciones y propietarios de objetos en la BD

▼ db\_ddlAdmin

Los miembros pueden modificar y eliminar todos los objeto de la BD

▼ db\_BackupOperator

Los miembros pueden emitir los comandos CHECKPOINT, BACKUP DATABASE Y BACKUP LOG (copias de seguridad)

▼ db\_DataReader

Los miembros tienen permisos de seleccion sobre cualquier tabla o vista de la BD (El nombre lo dice solo puede leer los datos pero nada mas)

▼ db\_DataWriter

Los miembros tienen permisos de insercion, actualizacion y eliminacion sobre cualquier tabla o vista (literalmente es el opuesto del anterior)

▼ db\_DenyDataReader

Los miembros no pueden seleccionar datos de ninguna tabla o vista de la BD

▼ db\_DenyDataWriter

Los miembros no pueden modificar ningun dato con las instrucciones insert, update o delete sobre cualquier tabla o vista de la BD

▼ sp\_AddRoleMember 'Funcion', 'Nombre Usuario'

Agregar un usuario de BD a una funcion

▼ sp\_DropRoleMember 'Funcion', 'Nombre Usuario'

Eliminar un usuario de BD o una funcion

▼ sp\_HelpRole 'Funcion'

Muestra las propiedades de la funcion

▼ sp\_HelpRoleMember 'Funcion'

Muestra los usuarios de una funcion

## Permisos a objetos

## Asignacion de permisos

GRANT {all [privileges] | lista\_permisos}

ON {tabla|vista} [(lista\_columnas)]

TO {usuarios|funciones\_usuario\_bd|public}

WITH GRANT OPTION —> Permite que el usuario al que le han concedido los permisos pueda concederlos a otros usuarios

## Quitar permisos

REVOKE {all [privileges] | lista\_permisos}

ON {tabla|vista} [(lista\_columnas)]

FROM {usuario|funciones\_usuario\_bd|public}

## Eliminar permisos

DENY {all [privileges] | lista\_permisos}

ON {tabla|vista} [(lista\_columnas)]

TO {usuarios|funciones\_usuario\_bd|public}

## Conceder permisos a todos los objetos de un esquema

GRANT ListaPermiso

ON SCHEMA :: NombreEsquema

TO usuarios | funciones\_usuario\_bd | public

WITH GRANT OPTION

## Permisos de instruccion

se pueden otorgar o negar los siguientes

- create table
- create proc
- create function



- create view
- create default
- create rule
- backup database
- backup log

## Asignacion de permisos

```
GRANT {all | lista_permisos} TO  
{usuarios | funciones_usuario_bd | public}
```

## Quitar permisos

```
REVOKE {all | lista_permisos} TO  
{usuarios | funciones_usuario_bd | public}
```

## Eliminar permisos explicitamente

```
DENY {all | lista_permisos} TO  
{usuarios | funciones_usuario_bd | public}
```

## Permiso para modificar un esquema

```
GRANT ALTER ON SCHEMA :: NombreEsquema  
TO usuarios | funciones_usuario_bd | public
```

## ▼ Examen 1

**Base de datos Ventas:**

- **Ventas:** folio, fecha, importe, cteid, empid.
- **Cientes:** cteid, nombre, apellidos, domicilio, celular.
- **Empleados:** empid, nombre, apellidos, correo, celular.

**Base de datos Almacenes:**

- **Envíos:** folio, fecha, importe, peso, cteid, camid.
- **Cientes:** cteid, nombre, apellidos, domicilio, celular.
- **Camiones:** camid, nombre, placas, motor, chofer.

Realizar las siguientes políticas de seguridad:

- 1.- Dar de alta al IS **ALMA** para que pueda ejecutar el comando **KILL** para eliminar procesos bloqueados.
- 2.- Dar de alta al IS **ANA** y posteriormente darle permiso para que pueda ejecutar **sp\_addrolemember**, **grant**, **revoke**, **deny** en la bd **ALMACENES**.
- 3.- Dar de alta al IS **LUIS** para que pueda ejecutar los comandos **BACKUP DATABASE** Y **RESTORE DATABASE**, ósea que pueda realizar copias de seguridad y restaurar la base de dato **VENTAS**.
- 4.- Dar de alta al IS **CONRADO** para que pueda crear cualquier objeto (tabla, vista, proc.alm.,función) en la **BD VENTAS**.
- 5.- Dar de alta al IS **PEDRO** para que puedan crear solamente vistas y proc. almacenados en **VENTAS**.
- 6.- Crear la función de usuario (role) llamada **DESARROLLADORES** para que pueda crear proc. almacenados y funciones escalares en las **BD VENTAS** y **ALMACENES**. Agregar al IS **JOSE** en la función **DESARROLLADORES** de ambas bases de datos.
- 7.- Dar de alta al IS **LOPEZ** y posteriormente pueda solamente ejecutar el comando **BACKUP DATABASE** en ambas bases de datos.
- 8.- Permitir que cualquier usuario de la **BD VENTAS** pueda actualizar (**update**) en todas sus tablas.
- 9.- Dar de alta al IS **CARLOS** para que seleccione solamente la segunda columna de todas las tablas de la **BD ALMACENES**.
- 10.- En la **BD VENTAS** cambiar las tablas a un nuevo esquema llamado **RH**.

- ▼ 1-Dar de alta al IS **ALMA** para que pueda ejecutar el comando **KILL** para eliminar procesos bloqueados

```
CREATE LOGIN ALMA WITH PASSWORD = '123'  
GO  
SP_AddSrvRoleMember ALMA, ProccessAdmin
```

- ▼ 2-Dar de alta al IS **ANA** y posteriormente darle permiso para que pueda ejecutar **sp\_addrolemember**, **grant**, **revoke**, **deny** en la bd **ALMACENES**

```
CREATE LOGIN ANA WITH PASSWORD='123'  
USE ALMACENES  
CREATE USER ANA FOR LOGIN ANA  
GO  
SP_ADDROLEMEMBER DB_OWNER, ANA
```

- ▼ 3-Dar de alta al IS **LUIS** para que pueda ejecutar los comandos **BACKUP DATABASE** y **RESTORE DATABASE**, ósea que pueda realizar copias de seguridad y restaurar la base de datos **VENTAS**.

```

CREATE LOGIN LUIS WITH PASSWORD='123'
USE VENTAS
CREATE USER LUIS FOR LOGIN LUIS
GO
SP_ADDROLEMEMBER DB_BACKUPOPERATOR, LUIS

```

- ▼ 4-Dar de alta al IS CONRADO para que pueda crear cualquier objeto (tabla, vista, proc.almacenado, funcion) en la BD VENTAS

```

CREATE LOGIN CONRADO WITH PASSWORD = '123'
USE VENTAS
CREATE USER CONRADO FOR LOGIN CONRADO
GO
SP_ADDROLEMEMBER DB_DDLADMIN, CONRADO

```

- ▼ 5.-Dar de alta al IS PEDRO para que puedan crear solamente vistas y proc.almacenados en la BD VENTAS

```

CREATE LOGIN PEDRO WITH PASSWORD = '123'
USE VENTAS
CREATE USER PEDRO FOR LOGIN PEDRO
GRANT CREATE PROC, CREATE VIEW TO PEDRO

```

- ▼ 6-Crear la funcion de usuario(role) llamada DESARROLLADORES para que pueda crear proc. almacenados y funciones escalares en las BD VENTAS y ALMACENES. Agregar al IS JOSE en la funcion DESARROLLADORES de ambas bases de datos

```

CREATE LOGIN JOSE WITH PASSWORD = '123'
GO
USE VENTAS
GO
SP_ADDROLE DESARROLLADORES
GRANT CREATE PROC, CREATE FUNCTION TO DESARROLLADORES
GO
CREATE USER JOSE FOR LOGIN JOSE
GO
SP_ADDROLEMEMBER DESARROLLADORES, JOSE
GO
USE ALMACENES
GO

```

```

SP_ADDROLE DESARROLLADORES
GRANT CREATE PROC, CREATE FUNCTION TO DESARROLLADORES
GO
CREATE USER JOSE FOR LOGIN JOSE
GO
SP_ADDROLEMEMBER DESARROLLADORES, JOSE

```

- ▼ 7.-Dar de alta al IS LOPEZ y posteriormente solo pueda ejecutar el comando BACKUP DATABASE en ambas bases de datos

```

CREATE LOGIN LOPEZ WITH PASSWORD='123'
USE VENTAS
CREATE USER LOPEZ FOR LOGIN LOPEZ
GRANT BACKUP DATABASE TO LOPEZ
GO
USE ALMACENES
CREATE USER LOPEZ FOR LOGIN LOPEZ
GRANT BACKUP DATABASE TO LOPEZ

```

- ▼ 8-Permitir que cualquier usuario de la BD VENTAS pueda actualizar (update) en todas sus tablas

```

USE VENTAS
GRANT UPDATE TO PUBLIC

```

- ▼ 9-Dar de alta al IS CARLOS para que seleccione solamente la segunda columna de todas las tablas de la BD almacenes

```

CREATE LOGIN CARLOS WITH PASSWORD = '123'
CREATE USER CARLOS FOR LOGIN CARLOS
GRANT SELECT ON VENTAS(FECHA) TO CARLOS
GRANT SELECT ON CLIENTES(NOMBRE) TO CARLOS
GRANT SELECT ON EMPLEADOS(NOMBRE) TO CARLOS

```

- ▼ 10-En la BD VENTAS cambiar las tablas a un nuevo esquema llamado RH

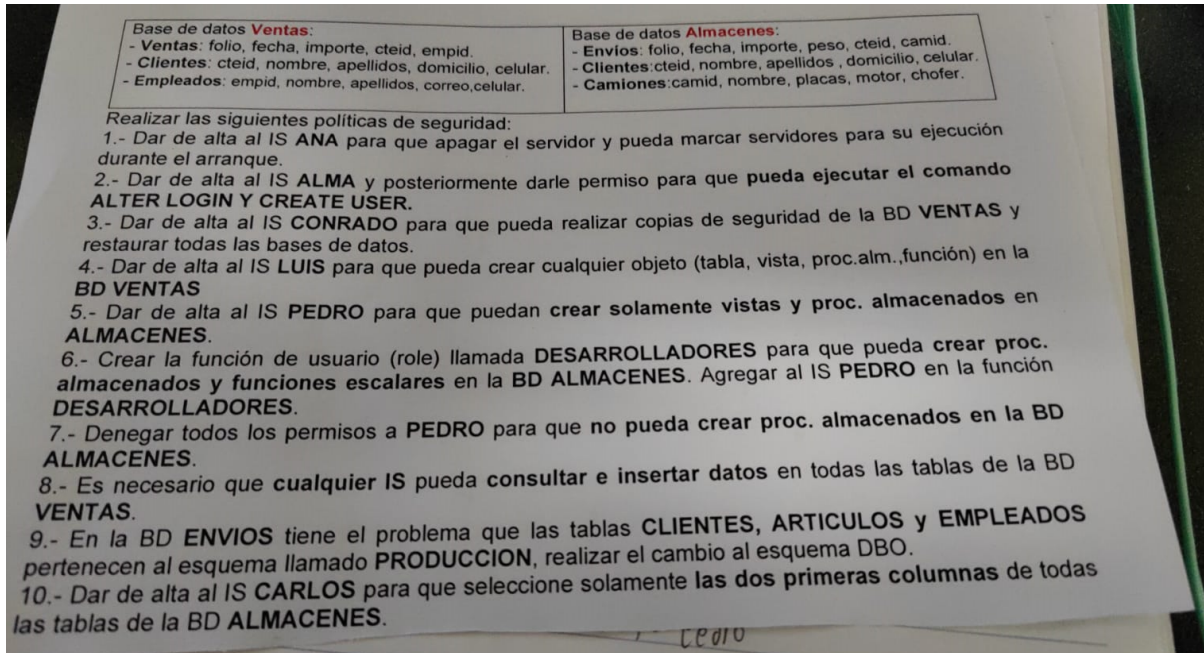
```

USE VENTAS
CREATE SCHEMA RH AUTHORIZATION DBO
ALTER SCHEMA RH TRANSFER DBO.VENTAS

```

```
ALTER SCHEMA RH TRANSFER DBO.CLIENTES
ALTER SCHEMA RH TRANSFER DBO.EMPLEADOS
```

## ▼ Examen 2



▼ 1-Dar de alta al IS ANA para que pueda apagar el servidor y pueda marcar servidores para su ejecución durante el arranque

```
CREATE LOGIN ANA WITH PASSWORD='123'
GO
SP_ADDSRVROLEMEMBER ANA, SERVERADMIN
```

▼ 2-Dar de alta al IS ALMA y posteriormente darle permiso para que pueda ejecutar el comando ALTER LOGIN Y CREATE USER

```
CREATE LOGIN ALMA WITH PASSWORD='123'
GO
SP_ADDSRVROLEMEMBER ALMA, SECURITYADMIN
```

▼ 3-Dar de alta al IS CONRADO para que pueda realizar copias de seguridad de la BD VENTAS y restaurar todas las bases de datos

```

CREATE LOGIN CONRADO WITH PASSWORD='123'
GO
USE VENTAS
CREATE USER CONRADO FOR LOGIN CONRADO
GO
SP_ADDROLEMEMBER DB_BACKUPOPERATOR, CONRADO

```

- ▼ 4-Dar de alta al IS LUIS para que pueda crear cualquier objeto(tabla, vista, proc.alm, funcion) en la BD VENTAS

```

CREATE LOGIN LUIS WITH PASSWORD='123'
GO
USE VENTAS
CREATE USER LUIS FOR LOGIN LUIS
GO
SP_ADDROLEMEMBER DB_DDLADMIN, LUIS

```

- ▼ 5-Dar de alta al IS PEDRO para que puedan crear solamente vistas y proc. almacenados en ALMACENES

```

CREATE LOGIN PEDRO WITH PASSWORD='123'
USE ALMACENES
CREATE USER PEDRO FOR LOGIN PEDRO
GRANT CREATE VIEW, CREATE PROC TO PEDRO

```

- ▼ 6-Crear la funcion de usuario(role) llamada DESARROLLADORES para que pueda crear proc. almacenados y funciones escalares en las BD ALMACENES. Agregar al IS PEDRO en la funcion DESARROLLADORES

```

USE ALMACENES
SP_ADDROLE DESARROLLADORES
GRANT CREATE PROC, CREATE FUNCTION TO DESARROLLADORES
SP_ADDROLEMEMBER DESARROLLADORES, PEDRO

```

- ▼ 7-Denegar todos los permisos a PEDRO para que no pueda crear proc. almacenados en la BD ALMACENES

```
USE ALMACENES
DENY CREATE PROC TO PEDRO
```

- ▼ 8- Es necesario que cualquier IS pueda consultar e insertar datos en todas las tablas de la BD VENTAS

```
USE VENTAS
GRANT SELECT, INSERT TO PUBLIC
```

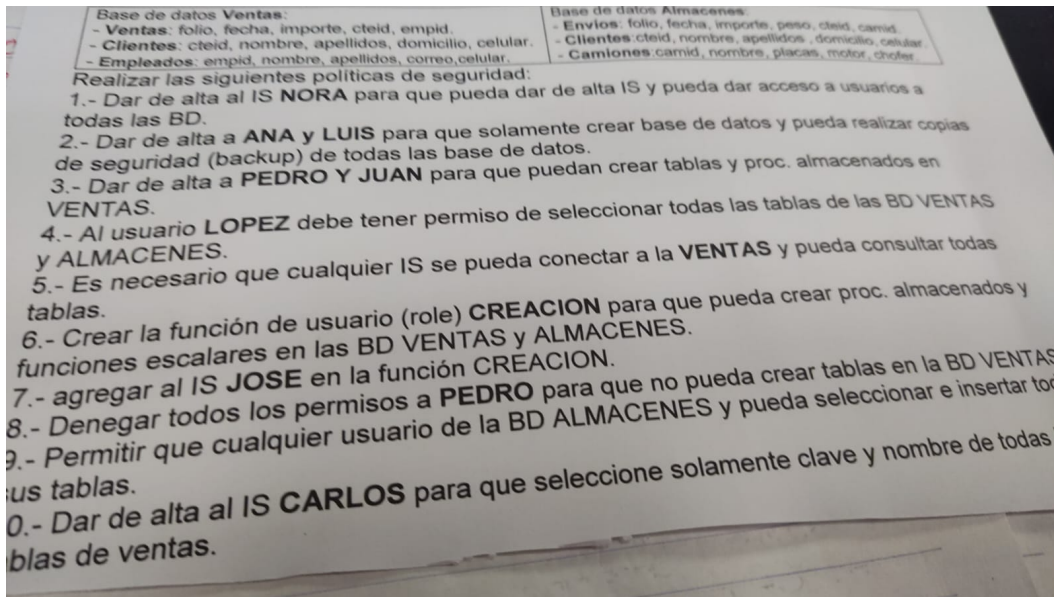
- ▼ 9-En la BD ENVIOS tiene el problema que las tablas CLIENTES, ARTICULOS y EMPLEADOS pertenecen al esquema llamado PRODUCCION, realizar el cambio al esquema DBO

```
USE ENVIOS
ALTER SCHEMA DBO TRANSFER PRODUCCION.CLIENTES
ALTER SCHEMA DBO TRANSFER PRODUCCION.ARTICULOS
ALTER SCHEMA DBO TRANSFER PRODUCCION.EMPLEADOS
```

- ▼ 10-Dar de alta al IS CARLOS para que seleccione solamente las dos primeras columnas de todas las tablas de la BD ALMACENES

```
CREATE LOGIN CARLOS WITH PASSWORD='123'
USE ALMACENES
CREATE USER CARLOS FOR LOGIN CARLOS
GRANT SELECT ON ENVIOS(FOLIO, FECHA) TO CARLOS
GRANT SELECT ON CLIENTES(CTEID, NOMBRE) TO CARLOS
GRANT SELECT ON CAMIONES(CAMID, NOMBRE) TO CARLOS
```

## ▼ Examen 3



- ▼ 1-Dar de alta al IS NORA para que pueda dar de alta IS y que pueda dar acceso a usuarios a todas las BD

```
CREATE LOGIN NORA WITH PASSWORD='123'
GO
SP_ADDSRVROLEMEMBER NORA, SECURITYADMIN
```

- ▼ 2-Dar de alta a ANA y LUIS para que solamente crear base de datos y pueda realizar copias de seguridad(backup) de todas las base de datos

```
CREATE LOGIN ANA WITH PASSWORD='123'
GO
SP_ADDSRVROLEMEMBER ANA, DBCREATOR
GO
CREATE LOGIN LUIS WITH PASSWORD='123'
GO
SP_ADDSRVROLEMEMBER LUIS, DBCREATOR
```

- ▼ 3-Dar de alta a PEDRO y JUAN para que puedan crear tablas y proc. almacenados en VENTAS

```
CREATE LOGIN PEDRO WITH PASSWORD='123'
CREATE LOGIN JUAN WITH PASSWORD='123'
USE VENTAS
```



```
CREATE USER PEDRO FOR LOGIN PEDRO
CREATE USER JUAN FOR LOGIN JUAN
GRANT CREATE TABLE, CREATE PROC TO PEDRO
GRANT CREATE TABLE, CREATE PROC TO JUAN
```

▼ 4-Al usuario LOPEZ debe tener permiso de seleccionar todas las tablas de las BD VENTAS y ALMACENES

```
USE VENTAS
GRANT SELECT TO LOPEZ
GO
USE ALMACENES
GRANT SELECT TO LOPEZ
```

▼ 5-Es necesario que cualquier IS se pueda conectar a la VENTAS y pueda consultar todas tablas

```
USE VENTAS
GRANT SELECT TO PUBLIC
```

▼ 6-Crear la funcion de usuario(role) CREACION para que pueda crear proc. almacenados y funciones escalares en las BD VENTAS y ALMACENES

```
USE VENTAS
GO
SP_ADDROLE CREACION
GRANT CREATE PROC, CREATE FUNCTION TO CREACION
GO
USE ALMACENES
GO
SP_ADDROLE CREACION
GRANT CREATE PROC, CREATE FUNCTION TO CREACION
```

▼ 7-Agregar al IS JOSE en la funcion CREACION

```
USE VENTAS
SP_ADDROLEMEMBER CREACION, JOSE
```

```
USE ALMACENES
SP_ADDROLEMEMBER CREACION, JOSE
```

▼ 8-Denegar todos los permisos a PEDRO para que no pueda crear tablas en la BD VENTAS

```
USE VENTAS
DENY CREATE TABLE TO PEDRO
```

▼ 9-Permitir que cualquier usuario de la BD ALMACENES y pueda seleccionar e insertar todo sus tablas

```
USE ALMACENES
GRANT SELECT, INSERT TO PUBLIC
```

▼ 10-Dar de alta al IS CARLOS para que seleccione solamente la clave y nombre de todas las tablas de ventas

te la rifaste we

```
CREATE LOGIN CARLOS WITH PASSWORD = '123'
USE VENTAS
CREATE USER CARLOS FOR LOGIN CARLOS
GRANT SELECT ON VENTAS(FOLIO, FECHA)
GRANT SELECT ON CLIENTES(CTEID, NOMBRE)
GRANT SELECT ON EMPLEADOS(EMPID, NOMBRE)
```