

## Creación de Usuarios, Permisos y Roles



FACULTAD de INGENIERÍA de  
**SISTEMAS E INFORMÁTICA**

## Creación de Login

❑ Un **login** es la capacidad de poder utilizar una instancia del **Servidor SQL**, está asociado con un usuario de Windows o con un usuario de **SQL**.



# Creación de Login

```
CREATE LOGIN <login_name> WITH PASSWORD =
'<enterStrongPasswordHere>;
```

```
CREATE LOGIN <login_name> WITH PASSWORD =
'<enterStrongPasswordHere>' MUST_CHANGE,
CHECK_EXPIRATION = ON;
```

```
CREATE LOGIN [MyUser]
WITH PASSWORD = 'MyPassword',
DEFAULT_DATABASE = MyDatabase,
CHECK_POLICY = OFF,
CHECK_EXPIRATION = OFF ;
```

# Creación de un login

## Roles

---

- ◆ Los roles son los conjuntos de permisos.
- ◆ Los roles permiten agrupar los derechos y gestionar más fácilmente los diferentes usuarios y las conexiones.
- ◆ Siempre es preferible asignar los derechos a los roles y posteriormente asignar los roles a los usuarios.
- ◆ Con una estructura como esta, la adición y la modificación de permisos o de usuarios son más sencillas.

## Roles

---

### ◆ **Nivel de servidor**

Para controlar el acceso a los recursos del servidor

### ◆ **Nivel de BD**

Acciones administrativas de la BD

# Roles predefinidos

Rol fijo de base de datos	Descripción
db_owner	Puede realizar todas las actividades de configuración y mantenimiento en la base de datos y también pueden eliminar la base de datos en SQL Server.
db_securityadmin	Puede modificar la pertenencia a un rol y administrar permisos. La adición de principiantes a este rol podría permitir la escalada de privilegios no deseados.
db_accessadmin	Puede agregar o quitar acceso a la base de datos para los inicios de sesión de Windows, los grupos de Windows y los inicios de sesión de SQL Server.
db_backupoperator	Puede hacer un backup de la base de datos.
db_ddladmin	Puede ejecutar cualquier comando de lenguaje de definición de datos (DDL) en una base de datos.
db_datawriter	Puede agregar, eliminar o cambiar datos en todas las tablas de usuario.
db_datareader	Puede leer todos los datos de todas las tablas de usuario.
db_denydatawriter	No puede agregar, modificar o eliminar datos de las tablas de usuario de una base de datos.
db_denydatareader	No puede leer ningún dato en las tablas de usuario de una base de datos.

# Ejemplo de creación de login

```
create login Miguel with password
'Xyz%1234#'
go
use Ciclismo
go
create user Miguel for login Miguel
```

## GRANT y REVOKE

### ◆ Sentencias DCL:

- GRANT
- REVOKE

## GRANT

**GRANT**  
**lista\_privilegios**  
**ON tabla TO**  
**lista\_usuarios [**  
**WITH GRANT**  
**OPTION ]**

◆ **Lista\_privilegios:**  
**privilegios: SELECT,**  
**INSERT, DELETE,**  
**UPDATE**

◆ **Tabla: nombre de la**  
**tabla**

◆ **Lista\_usuarios:**  
**usuarios a los que se**  
**les dara los**  
**privilegios**

## Ejemplos de GRANT

- ◆ Ejemplo de como se asigna todos los privilegios en la tabla Persona a Juan:  
**GRANT ALL ON Persona TO Juan  
WITH GRANT OPTION**
- ◆ Ejemplo de como se da el privilegio de SELECT sobre la tabla Persona a Juan:  
**GRANT SELECT ON Persona TO Juan**

## Ejemplos de GRANT

- ◆ Ejemplo de como se asigna algunos privilegios en la columna Nombre en la tabla Persona a Juan:  
**GRANT SELECT, INSERT, UPDATE  
(Nombre) ON Persona  
TO Juan  
WITH GRANT OPTION**

## REVOKE

**REVOKE [ GRANT  
OPTION FOR ]  
lista\_privilegios  
ON tabla FROM  
lista\_usuarios {  
RESTRICT |  
CASCADE }**

- ◆ **Lista\_privilegios:**  
privilegios: **SELECT,  
INSERT, DELETE,  
UPDATE**
- ◆ **Tabla:** nombre de la  
tabla
- ◆ **Lista\_usuarios:**  
usuarios a los que se  
les quitara los  
privilegios

## Ejemplos de REVOKE

- ◆ **Ejemplo de como se desasigna el  
privilegio de consultar datos de la tabla  
Persona a Juan:**  
**REVOKE SELECT ON Persona FROM  
Juan**

## Ejemplos de GRANT Y REVOKE

```
GRANT SELECT, INSERT on Cliente to Miguel;
```

```
GO
```

```
GRANT ALL on Cliente to Miguel;
```

```
GO
```

```
REVOKE INSERT on Cliente to Miguel
```

## Ejemplos de ROLES

```
ALTER ROLE db_reader add member Miguel;
```

```
GO
```

```
ALTER ROLE db_reader drop member Miguel;
```

```
GO
```

```
ALTER ROLE db_writer add member Miguel;
```



## Ejemplos de DROP

---

```
DROP USER Miguel;  
GO  
DROP LOGIN Miguel
```