Índice

1. SEGURIDAD A NIVEL DE TABLA

Las tablas pertenecen al usuario que las crea, y él será quién conceda y quite los privilegios sobre sus tablas a otros usuarios.

Los privilegios que podrá conceder y quitar el propietario de las tablas son los siguientes:

* ALTER	permite modificar la definición de la tabla
* DELETE	permite borrar filas de la tabla
* INDEX	permite crear índices sobre la tabla
* INSERT	permite añadir filas a la tabla
* **SELECT**	permite consultar filas de la tabla
* UPDATE	permite actualizar cualquier columna dentro de cualquier fila
* UPDATE (col1,...,coln)	permite actualizar las columnas listadas
* ALL	todos los privilegios anteriores

El propietario de una tabla puede conceder privilegios sobre la misma utilizando el siguiente comando:

```
SQL> GRANT < lista_de_privilegios > ON < tabla >
                TO < nombre_de_usuario > [ WITH GRANT OPTION ];
```

Si se quiere conceder los privilegios a todos los usuarios de la base de datos se deberá utilizar el nombre de usuario PUBLIC.

```
SQL>GRANT ALL ON autores TO PUBLIC;
```

Todos los usuarios de la base de datos tendrán todo tipo de privilegios sobre la tabla autores después de que su propietario haya ejecutado la sentencia anterior.

Para referirse en sus sentencias a la tabla anterior deberán utilizar el nombre

***nombre_de_propietario*.nombre_de_tabla**

Si NEIL fuese el propietario de la tabla cualquier otro usuario para referirse a ella debe utilizar el nombre NEIL.autores

```
SQL>SELECT * FROM    NEIL.autores;
```

Si el propietario de una tabla quiere que los privilegios que concede a un usuario, éste a su vez los pueda conceder a otros usuarios, deberá utilizar la cláusula **"WITH GRANT OPTION".**

SQL>GRANT SELECT ON autores TO JOHN WITH GRANT OPTION;

El usuario JOHN podrá ahora consultar la tabla autores y además podrá conceder el privilegio de consulta a otros usuarios.

Para quitar los privilegios que un usuario concedió sobre una tabla se utilizará el siguiente comando:

```
SQL>REVOKE <lista_de_privilegios> ON <nombre_tabla>
FROM <nombre_usuario>;
```

Cuando se quita un privilegio a un usuario que a su vez ha concedido este privilegio a otros, automáticamente ORACLE también se lo quitará a los otros.

El nombre de usuario al que se le quitan los privilegios también podrá ser el usuario PUBLIC.

2. SEGURIDAD A NIVEL DE BASE DE DATOS EN GENERAL (ORACLE)

Los usuarios ORACLE, además de tener acceso (identificador y contraseña) al sistema operativo, deberán tener un identificador y una contraseña para poder conectarse con ORACLE. El administrador dispone para conceder este privilegio del siguiente comando SQL:

```
GRANT { CONNECT | RESOURCE | DBA } TO < nombre_de_usuario >
        [ IDENTIFIED BY < contraseña > ];
```

* CONNECT

- posibilidad de acceso a la base de datos
- acceder y manipular tablas de otros usuarios si tiene permiso para ello
- crear vistas y sinónimos
- no puede crear ni tablas ni agrupamientos ni índices

* RESOURCE posibilidad de crear tablas

- crear tablas, agrupamientos e índices
- conceder y quitar privilegios sobre los objetos anteriores
- utilizar el comando AUDIT para controlar el acceso a los objetos propios

* DBA privilegios de administrador como:

- crear tablas y vistas (RESOURCE implícito)
- crear y dar de baja usuarios
- crear sinónimos públicos

Ejemplo:

```
SQL>GRANT CONNECT,  RESOURCE TO NEIL, JUAN
        IDENTIFIED BY AMSTRONG, DE_LA_CIERVA;
```

La cláusula IDENTIFIED BY es obligatoria si se le está concediendo a un usuario privilegio de acceso (CONNECT), sin embargo no será necesario utilizarla si estamos concediendo nuevos privilegios a un usuario.

```
SQL>GRANT DBA TO NEIL;
```

Un usuario podrá cambiarse la contraseña en cualquier momento ejecutando el siguiente comando:

```
SQL>GRANT CONNECT TO  < nombre_de_usuario > IDENTIFIED BY
                    <  nueva_contraseña  >;
```

El identificador ORACLE es independiente del identificador del sistema operativo.
ORACLE comprueba automáticamente la validez del identificador y la contraseña cuando un usuario se conecta con ORACLE.
Sólo un usuario con privilegio DBA puede conceder privilegios a otros usuarios.

Para quitar privilegios a los usuarios el administrador dispone del siguiente comando SQL:

SQL> **REVOKE** { CONNECT | RESOURCE | DBA } **FROM** < *nombre_de_usuario* >;

como por ejemplo:

```
SQL>REVOKE DBA FROM NEIL;
```

Si a un usuario se le retira el privilegio de conectarse a la base de datos, dicho usuario no podrá acceder a los datos hasta que se le vuelva a conceder el privilegio de conexión. Sus tablas no serán borradas y los usuarios que tengan privilegios sobre ellas podrán seguir utilizándolas. Cuando se le vuelvan a conceder a este usuario privilegios de conexión "recuperará" sus tablas.

3. EJERCICIOS PROPUESTOS

1º. Asigne el permiso para modificar la definicion de la tabla TIENDAS al usuario DAI2TYY.

2º. Asigne el permiso para modificar filas de la columna ORDER_DATE de la tabla SALES_ORDER al usuario DAI2TYY.

3º. Asigne el permiso para modificar filas de todas las columnas de la tablaITEM al usuario DAI2TYY.

4º. Elimina el permiso de modificar filas de todas las columnas de la tabla ITEM al usuario DAI2TYY.

5º. Crea al usuario Maria con pasword maria.

6º. Asignale al usuario maria la posibilidad de conectarse a la base de datos.

7º. Asignarle al usuario maria la posibilidad de crear tablas.

8º. Quitar el derecho de conexión a la base de datos al usuario maria.

9º. Borrar al usuario maria.