# UNIDAD 12 ACTIVIDAD PRÁCTICA MANEJO DE DATOS

Para realizar esta actividad he optado por dividirla en 5 grandes bloques o pasos. Se va a tratar de analizar y justificar cada elección para tener la actividad lo más aclarada posible.  
Por no alargar este documento y hacer repetitivo el ejercicio, las sentencias realizadas estarán únicamente en el fichero “PedroBerruecoUnd12.sql”, y las explicaciones las pondré aquí.  
  
Paso 1. Crear los Usuarios.  
Paso 2. Crear los Roles.  
Paso 3. Asignar Privilegios a cada Rol.  
Paso 4. Asignar Privilegios o Roles a cada usuario.  
Paso 5. Comprobaciones.

PASO 1  
  
Se han creado 7 usuarios en este paso, uno por cada persona que ha aparecido en el enunciado. La única particularidad de este paso es que todos los usuarios menos Ana han sido creados con **@’localhost’** para que solo puedan conectarse desde el equipo local dónde tenemos la Base de Datos. Ana ha sido creada con **@’%’** por lo que podrá conectarse desde cualquier IP.

# PASO 2

Si bien es verdad que en un problema de la vida real optaría por crear roles para cada tipo de función dada la rotación de personal en los puestos de trabajo, en este ejercicio me voy a ceñir a crear roles solo para los casos en que una función sea realizada por más de una persona, o me indiquen expresamente que el equipo se puede ampliar. De este modo utilizó distintos tipos de privilegios.

Por tanto, he decidido crear dos roles, uno llamado “**inse\_cons**” que usarán Raúl, Pedro y Javier y otro llamado “**actu\_cons**” que usarán Sandra, Jesús y Javier.

En el siguiente paso amplió la información de privilegios que tiene cada rol.

# PASO 3

El rol “**inse\_cons**” se utilizará para **insertar** datos en las tablas (**producto, principal y extra**) y tendrá permisos de **consulta** en la tabla **categoría**. Dando el acceso mínimo imprescindible para permitir las labores que deben realizar Raúl, su futuro compañero, Pedro o Javier.   
Al ser unos permisos transversales a tantas personas he decidido asociarlos a un rol.

EL rol “**actu\_cons**” se utilizará para **actualizar** registros de la tabla **producto** y **consultar** datos de la tabla **principal**. Dando el acceso mínimo imprescindible para permitir las labores que deben realizar Sandra Jesús o Javier.  
Al ser unos permisos transversales a tantas personas he decidido asociarlos a un rol.

# PASO 4

En este caso hemos asignado privilegios directamente a los usuarios Enrique, Pedro y Ana ya que cada uno tiene unas funciones únicas que solo realizan ellos, aunque como he dicho antes en la vida real esta gente se tomaría sus vacaciones y no sería operativo andar asignando y quitando privilegios, es mucho más sencillo crear roles con los permisos predefinidos y activarlos en función de la necesidad.

Cabe indicar que estos permisos se asignan a todas las tablas de la BD de ahí el ‘macking.\*’.

En el caso de Ana hemos incluido ‘@%’ para asegurarnos de que estos permisos son extensibles a cualquier IP desde la que se conecte, ya que ésta es la única usuaria que puede conectarse desde fuera de la red. Para el resto de los usuarios que solo pueden conectarse desde ‘@localhost’ no he considerado necesario ponerlo.

Como Ana tiene privilegios totales y en el enunciado no especifica lo contrario, he decido añadirle la cláusula “WITH GRANT OPTION” para permitir a Ana asignar los permisos que fueran necesarios en el futuro.

# PASO 5

Para el último paso he seleccionado a los usuarios enrique y pedro.  
Se activan los ROLES.   
En el caso de enrique tenía permisos de consulta sobre toda la BD así que puede realizar consultas en cualquier tabla, pero no borrados, inserts o updates.  
  
-- Con el usuario enrique:  
SELECT \* FROM menú;  
![Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente]()

DELETE FROM menu WHERE Nombre = 'Mc Desayuno 1';  
![Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente]()

En el caso del usuario pedro, tiene permisos de borrado en cualquier tabla, y permisos de consulta únicamente en la tabla categoría, debido a que está incluido en el rol “inse\_cons” que tiene permisos de consulta en esa tabla. Lo comprobamos:

-- Con el usuario pedro:

SELECT \* FROM categoria;

![Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente]()

DELETE FROM menu WHERE Nombre = 'Mc Desayuno2';  
  
\*Para poder realizar este DELETE, se necesita asignar permisos de lectura en la tabla al usuario, pues un DELETE con un WHERE conlleva una consulta interna de la tabla.

![Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente]()