

## Clase 1 y 2 del tema 6

1. Crea un procedimiento (**ejemplo1\_sp.sql**) simple que muestre por pantalla el texto "esto es un ejemplo de procedimiento" (donde la cabecera de la columna sea "mensaje").
2. Crea un procedimiento (**ejemplo2\_sp.sql**) donde se declaren tres variables internas, una de tipo entero con valor de 1, otra de tipo varchar(10) con valor nulo por defecto y otra de tipo decimal con 4 dígitos y dos decimales.  
Dentro del procedimiento, modificar el valor de la variable entera, la variable de tipo texto, realizar alguna operación matemáticas con las variables y mostrar los resultados en una select.
3. Crea un procedimiento (**ejemplo3\_sp.sql**) donde se declaren dos variables internas, una de tipo char y otra de tipo enum.  
Dentro del procedimiento, modificar el valor de la variables y mostrar el resultado.  
Provocar diferentes errores al asignar valores no permitidos a las variables
4. Crea un procedimiento (**ejemplo4\_sp.sql**) donde se declaren varias variables internas, unas de tipo entera y otra de tipo fecha. Poner por defecto la fecha actual.  
Dentro del procedimiento, modificar el valor de la variables y mostrar el resultado.  
Provocar diferentes errores al asignar valores no permitidos a las variables.
5. Asignar cualquier valor a una variable definida por el usuario de distintas formas (mediante una select y mediante la clausula SET)  
Crear un procedimiento (**ejemplo5\_sp.sql**) que modifique dicha variable, muestre el valor de dicha variable tanto dentro del procedimiento, como fuera de él.
6. Crear una tabla llamada usuarios, que almacene varios usuarios y sus passwords.  
Crear un procedimiento (**ejemplo6\_sp.sql**) que reciba dos parámetros de entrada (un usuario y un password) y que inserte un nuevo usuario con esos datos. Comprobar que si modificamos una de las variables de entrada dentro del procedimiento, fuera de él la variable no cambia (los parámetros de entrada son una copia de los parámetros originales).
7. Crear un procedimiento (**ejemplo7\_sp.sql**) con una variable de salida (paso de parámetros por variable) y comprueba que por defecto su valor es Nulo. Modifica su valor y muéstralo tanto dentro del procedimiento como fuera de él.
8. Crea un procedimiento (**ejemplo8\_sp.sql**) con una variable de entrada-salida. El procedimiento debe cambiar el valor de la variable. Realizar una prueba mostrando el valor de la variable antes y después de llamar al procedimiento.
9. Crea un procedimiento (**ejemplo9\_sp.sql**) que compruebe si un usuario existe o no en la base de datos. Mostrar un mensaje en forma de select.
10. Crea un procedimiento (**ejemplo10\_sp.sql**) con dos parámetros de entrada (usuario y contraseña) y uno de salida (mensaje) Hacer uso de IF, ELSE para comprobar si un usuario pasado por parámetros existe en la base de datos con esa contraseña.  
Devolvemos en la variable de salida estos tres posibles valores: usuario no existe, password incorrecto o usuario correcto
11. Crea una nueva tabla datos\_personales con los campos: usuario, ~~edad~~ fecha\_nacimiento, ultimo\_login, num\_accesos, (donde usuario es clave foránea de la tabla creada anteriormente).  
Insertar algunos datos de ejemplos.  
Crea un procedimiento (**ejemplo11\_sp.sql**) Hacer uso de case...when.. para comprobar si un usuario pasado por parámetros es mayor de edad o no.
12. Crea un procedimiento (**ejemplo12\_sp.sql**) que borre si existe la tabla artículos. Si no existe debe crearla e insertar 10 artículos haciendo uso de un bucle WHILE...DO

13. Hacer un procedimiento (**ejemplo13\_sp.sql**) de forma que cuando un usuario se autentique (comprobar que el usuario existe), actualice la tabla datos\_personales (aumente el número de acceso y almacene la fecha actual).  
Nota: Si el usuario existe y es la primera vez que se autentifica, hay que incluirlo en la tabla datos\_personales.  
Mostrar los datos de la tabla datos personales para ese usuario
14. Crear un procedimiento (**ejemplo14\_sp.sql**) que pasándole un número, inserte en la tabla de artículos el artículo con clave el número p, controlar los errores de clave duplicada y error al insertar null como clave.
15. Crear una función (**ejemplo15\_fun.sql**) de manera que reciba una cantidad de euros y devuelva el valor expresado en pesetas. Llamar a dicha función en el interior de una select
16. Crea una función (**ejemplo16\_fun.sql**) de forma que pasándole un usuario nos devuelva su edad actual (debe comprobar su fecha de nacimiento en el caso de existir y calcular su edad actual).
17. Crea una tabla sistema que almacene el número total de usuarios actuales y la fecha de la última actualización (último usuario, modificado o borrado).  
Crear un disparador (**ejemplo17\_trig.sql**) sobre la tabla usuarios de forma que cuando se cree un usuario nuevo, actualice dicha tabla.