

TEMA 6: EJERCICIOS PROCEDURES

Procedimientos sin sentencias SQL

1. Escribe un procedimiento (HolaMundo) que no tenga ningún parámetro de entrada ni de salida y que muestre el texto ¡Hello from top of the world! Himalayan Yeti!!!!.
2. Escribe un procedimiento (negPosCero) que reciba un número real de entrada y muestre un mensaje indicando si el número es positivo, negativo o cero.
3. Modifique el procedimiento diseñado en el ejercicio anterior para que tenga un parámetro de entrada, con el valor un número real y un parámetro de salida con una cadena de caracteres indicando si el número es positivo, negativo o cero. Se llamará negPosCero2
4. Escribe un procedimiento (nota) que reciba un número real de entrada, que representa el valor de la nota de un alumno, y muestre un mensaje indicando qué nota ha obtenido teniendo en cuenta las siguientes condiciones:
 - [0,5) = Insuficiente
 - [5,6) = Aprobado
 - [6, 7) = Bien
 - [7, 9) = Notable
 - [9, 10] = Sobresaliente
 - En cualquier otro caso la nota no será válida.
5. Modifique el procedimiento diseñado en el ejercicio anterior para que tenga un parámetro de entrada, con el valor de la nota en formato numérico y un parámetro de salida, con una cadena de texto indicando la nota correspondiente. Llamalo nota2
6. Resuelva el procedimiento diseñado en el ejercicio anterior haciendo uso de la estructura de control CASE o IF-ELSE si lo has hecho con CASE. Llamalo nota3
7. Escribe un procedimiento que reciba como parámetro de entrada un valor numérico que represente un día de la semana y que devuelva una cadena de caracteres con el nombre del día de la semana correspondiente. Por ejemplo, para el valor de entrada 1 debería devolver la cadena lunes. (DiaDeLaSemana)

Procedimientos con sentencias SQL

A partir de la BD Jardinería:

1. Escribe un procedimiento que reciba el nombre de un país como parámetro de entrada y realice una consulta sobre la tabla cliente para obtener todos los clientes que existen en la tabla de ese país. ClientePais
2. Escribe un procedimiento que reciba como parámetro de entrada una forma de pago, que será una cadena de caracteres (Ejemplo: PayPal, Transferencia, etc). Y devuelva como salida el pago de máximo valor realizado para esa forma de pago. Deberá hacer uso de la tabla pago de la base de datos jardinería. PagoMáximoPorTipo
3. Escribe un procedimiento que reciba como parámetro de entrada una forma de pago, que será una cadena de caracteres (Ejemplo: PayPal, Transferencia, etc). Y devuelva como salida los siguientes valores teniendo en cuenta la forma de pago seleccionada como parámetro de entrada:
 - el pago de máximo valor,
 - el pago de mínimo valor,
 - el valor medio de los pagos realizados,
 - la suma de todos los pagos,
 - el número de pagos realizados para esa forma de pago.

Deberá hacer uso de la tabla pago de la base de datos jardinería.

Llamalo CalculoTipoPago

4. Crea una base de datos llamada **procedimientos** que contenga una tabla llamada **secuencia**, que contenga una única columna llamada **numero** de tipo INT UNSIGNED.
5. Crea un procedimiento (secuenciaWhile) que, mediante un bucle WHILE:
 - Reciba un parámetro de entrada llamado valorInicial de tipo INT UNSIGNED.
 - Almacene en la tabla **secuencia** toda la secuencia de números desde el **valorInicial** hasta el 1.
 - El procedimiento deberá eliminar el contenido de la tabla antes de insertar los nuevos valores.
6. Resuelve el ejercicio anterior pero utilizando un bucle REPEAT (secuenciaREPEAT)
7. Vuelve a resolver el ejercicio pero ahora con un bucle LOOP (secuenciaLOOP)
8. Dentro de la base de datos de procedimientos, crea dos tablas **par** y **impar**. Ambas deben contener una única columna llamada **numero** de tipo INT UNSIGNED.

9. Crea un procedimiento (paresImpares) que reciba un parámetro de entrada llamado **tope** de tipo INT UNSIGNED y deberá almacenar en la tabla **par** aquellos números pares que existan entre el número 1 y el tope. De igual manera, insertará los impares en **impar**.
 - El procedimiento deberá eliminar el contenido de la tabla antes de insertar los nuevos valores.
10. Resuelve el ejercicio anterior pero utilizando un bucle REPEAT (secuenciaRepeat).
11. Vuelve a resolver el ejercicio pero ahora con un bucle LOOP (211secuenciaLoop).