

Actividades tema 3

Ejercicios sobre la base de datos creaBD_pedidos.sql:

Esquema relacional de la base de datos pedidos:

PEDIDO (refPed, fecped)
ARTICULO (codart, desart, pvpart)
LINEAPEDIDO (refped, codart, cantart)
Claves ajenas: refped → PEDIDO(refped)
codart → ARTICULO(desart)

CURSORES: base de datos PEDIDO.

1. Crea un procedimiento llamado `verPedido`, que muestre sólo una fila de la tabla pedido, debe de mostrar su referencia y fecha mediante el empleo de un cursor.

Ejemplo de salida:

```
+-----+
| Datos Pedidos                |
+-----+
| Referencia: P0001 Fecha: 2014-02-16 |
+-----+
```

2. Modifica `verPedido`, para que muestre cada una de las filas de la tabla pedido. Además, se debe de mostrar un mensaje dependiendo de la cantidad en años que tenga el pedido, siendo éste “vigente” si tienes menos de 2 años, “antiguo” si tiene 3 ó 4 años y “muy antiguo” si tiene 5 o más años.
3. Realiza un procedimiento llamado `verArticuloPedido`, en que solicite por parámetro el código de un producto y muestre cada uno de los pedidos que tiene asociado, así como el número de unidades solicitadas del artículo.
Si el pedido tiene un número de unidades de 5 o menos, se indicará que es un pedido pequeño, si tiene entre 6 y 10 unidades, será un pedido mediano y si tiene 15 o más unidades, será un pedido grande.
4. Crea un procedimiento `insertarPedido` que se encargue de añadir un nuevo pedido a la tabla PEDIDO, este procedimiento recibirá como parámetros la referencia del pedido que se desea añadir y su fecha. Debe de mostrar un mensaje si se ha realizado correctamente.

Excepciones:

5. Modifica el procedimiento `insertarPedido` para que, en el caso que se produzca la excepción al insertar un pedido repetido (clave duplicada), no finalice anormalmente mostrando la descripción por defecto del error por pantalla, y en su lugar, termine

normalmente y guarde en una variable de salida `estado`, un mensaje con el tipo de error “Clave duplicada”. Se debe de utilizar un manejador o handler.

6. Crea un procedimiento llamado `insertar` que se encargue de llamar a `insertarPedido` y nos informe si la inserción se ha podido llevar a cabo o no. Nos debe de mostrar un mensaje de error confeccionado por nosotros para el tipo de error que nos devuelve, por ejemplo “ERROR. Ya hay un pedido con la referencia XXXX”.
7. Crea un procedimiento que reciba como parámetro el código de un artículo y nos muestre su descripción en caso de existir. En caso de que no exista ningún artículo con dicho código, se mostrará un mensaje como el siguiente: “No existe ningún artículo con el código XXXXX”.

Ejercicios sobre la base de datos `creaBD_empresa.sql`:

Esquema relacional de la base de datos empresa:

DEPART (dept_no, dnombre, loc)

EMPLE (emp_no, apellido, oficio, dir, fecha_alt, salario, comision, dept_no)

Clave ajenas: dir → EMPL (emp_no)

dept_no → DEPART (dept_no)

8. Crea un procedimiento llamado `cambiarJefe` que cambie el director o jefe de un empleado en la tabla `emple` de la base de datos Empresa.
Este procedimiento recibirá el apellido del empleado cuyo jefe se desea modificar y el apellido de su nuevo empleado jefe.
En este procedimiento se pueden dar dos situaciones de error o excepciones (código 1329):
 - Que no exista el empleado cuyo jefe se desea modificar.
 - Que no exista el nuevo empleado jefe que se le quiere asignar.

Ejemplos de salida:

```
+-----+
| Mensaje |
+-----+
| El nuevo jefe de AAAA es BBBB |
+-----+
```

```
+-----+
| Error jefe |
+-----+
| No existe ningún empleado jefe apellidado CCCCC |
+-----+
```

```
+-----+
| Error empleado |
+-----+
| No existe ningún empleado apellidado CCCCC |
+-----+
```