*Claudia Nerea Borrachero Mallofret*

1. Realizar un procedimiento que reciba como parámetro de entrada el código de un cliente y nos devuelva como parámetro de salida la cantidad de dinero que se ha gastado dicho cliente. Adicionalmente el procedimiento deberá mostrar por pantalla esa cantidad y “ES UN BUEN CLIENTE” si esa cantidad que se ha gastado es mayor de 100000. Mostrar cómo llamarías a ese procedimiento.

**CREATE** **PROCEDURE** ej1(**IN** codigo **INT** )

**BEGIN**

**DECLARE** cantidad **INT**;

**SELECT** **sum**(p.amount) **AS** amount **INTO** cantidad

**FROM** customers c **INNER** **JOIN** payments p **ON**

c.customerNumber=p.customerNumber

**WHERE** c.customerNumber=codigo;

**IF** cantidad<100000 **THEN**

**SELECT** cantidad;

**END** **IF**;

**IF** cantidad>100000 **THEN**

**SELECT** **CONCAT**(CANTIDAD, 'Es un buen cliente') **AS** MENSAJE;

**END** **IF**;

**END**

**CALL** ej1(112);

1. Realizar un función que dado un código producto nos diga la cantidad de unidades que se han vendido de ese producto.

• Si no se ha vendido ninguna unidad se deberá también mostrar por pantalla “ESE PRODUCTO NO SE VENDIÓ”

• Si se han vendido más de 100 unidades deberá aparecer por pantalla “ES UN PRODUCTO QUE SE VENDE BIEN”.

• Y, adicionalmente, si es el producto que más se vendió deberá mostrarse “ES NUESTRO PRODUCTO ESTRELLA”.

**CREATE** **PROCEDURE** ej2(**IN** codigo **VARCHAR**(10))

**BEGIN**

**DECLARE** cantidad **INT** **DEFAULT** -1;

**DECLARE** mensaje **VARCHAR**(50);

**SELECT** **sum**(od.quantityOrdered) **INTO** CANTIDAD

**FROM** orderdetails od

**WHERE** od.productCode = codigo;

**if** cantidad = -1 **then**

**SET** mensaje='no se vendio';

**ELSEif** cantidad > 100 **then**

**SET** mensaje = 'Es un producto que se vende bien' ;

**ELSEIF** cantidad = (**SELECT** **max**(od.quantityOrdered) **FROM** orderdetails od) **then**

**SET** mensaje = 'Es nuestro producto estrella';

**END** **if**;

**SELECT** mensaje;

**END**

**CALL** ej2('S18\_2248');

**CREATE** **FUNCTION** F3(codigo **VARCHAR**(15))

**RETURNS** **VARCHAR**(100) **DETERMINISTIC**

**BEGIN**

**DECLARE** CANTIDAD **INT** **DEFAULT** -1;

**DECLARE** MENSAJE **VARCHAR**(200);

**IF** **EXISTS** (**SELECT** p.productCode **FROM** products p

**WHERE** p.productCode=codigo) **THEN**

**SELECT** **SUM**(od.quantityOrdered) **INTO** CANTIDAD

**FROM** orderdetails od

**WHERE** od.productCode=codigo;

**IF** CANTIDAD=-1 **THEN**

**SET** MENSAJE='EL PRODUCTO NO SE HA VENDIDO';

**ELSEIF** CANTIDAD>100 **THEN**

**SET** MENSAJE='EL PRODUCTO SE VENDE BIEN';

**ELSEIF** CANTIDAD = (**SELECT** **max**(od.quantityOrdered) **FROM** orderdetails od) **then**

**SET** MENSAJE = 'ES EL PRODUCTO ESTRELLA';

**END** **IF**;

**RETURN** MENSAJE;

**END**

**SELECT** F3('s');

1. Realizar una función que muestre por pantalla los datos del jefe del empleado que se encarga de un cliente determinado que será el parámetro de entrada (puedes hacerlo con más de una función). Devolverá 1 si el cliente existe y 0 en caso contrario.

**CREATE** **FUNCTION** mostrarJefe(codigo **INTEGER**)

**RETURNS** **tinyint**(1) **DETERMINISTIC**

**BEGIN**

**DECLARE** encontrado **tinyint**(1) **DEFAULT** 0;

**DECLARE** JEFE **varchar**(4000) **DEFAULT** **NULL**;

**SELECT** **COUNT**(\*) **INTO** JEFE

**FROM** employees

**WHERE** employeeNumber = (**SELECT** reportsTo **FROM** employees

**WHERE** employeeNumber = codigo);

**IF** JEFE >= 1 **THEN**

**SET** encontrado = 1;

**END** **IF**;

**RETURN** encontrado;

**END**

**SELECT** mostrarJefe(1009);

1. Realizar una que muestre por pantalla los datos de un pedido determinado que será el parámetro de entrada. Deberán aparecer los datos del cliente, los nombres de cada producto comprado, el precio por el que se compró, la cantidad y al final el total del pedido. Devolverá 1 si el cliente existe y 0 en caso contrario.

**CREATE** **PROCEDURE** mostrar\_estadisticas()

**BEGIN**

**DECLARE** resultado **VARCHAR**(4000);

**DECLARE** ventas **DOUBLE**;

**DECLARE** anio **INT**;

**DECLARE** salida **INT** **DEFAULT**=0;

**DECLARE** dia **VARCHAR**(20);

**DECLARE** CURSOR1 **CURSOR** **FOR**

**SELECT** **YEAR**(v.fecha),**ROUND**(**SUM**(v.unidades\_vendidas\*v.precio\_unitario),2)

**FROM** venta v

**GROUP** **BY** **YEAR**(v.fecha);

**DECLARE** CURSOR2 **CURSOR** **FOR**

**SELECT** **DAYNAME**(v.fecha),**ROUND**(**SUM**(v.unidades\_vendidas\*v.precio\_unitario),2)

**FROM** venta v

**GROUP** **BY** **DAYNAME**(v.fecha)

**ORDER** **BY** **ROUND**(**SUM**(v.unidades\_vendidas\*v.precio\_unitario),2) **DESC**;

**DECLARE** **CONTINUE** **HANDLER** **FOR** **NOT** **FOUND** **SET** salida=1;

**SELECT** **ROUND**(**SUM**(v.unidades\_vendidas\*v.precio\_unitario),2)

**FROM** venta v;

**SET** resultado=**CONCAT**('===ESTADISTICAS===\n

-----TOTALES-----\n

VENTAS TOTALES: ',ventas,'\n');

**OPEN** CURSOR1;

**FETCH** CURSOR1 **INTO** anio,ventas;

**WHILE** salida=0 **DO**

**SET** resultado=**CONCAT**(resultado,'En ',anio,': ',ventas,' $\n');

**FETCH** CURSOR1 **INTO** anio,ventas;

**END** **WHILE**;

**CLOSE** CURSOR1;

**SET** resultado=**CONCAT**('===LISTADOS===\n

-----VALOR DE VENTAS POR DIA-----'\n);

**OPEN** CURSOR2;

**FETCH** CURSOR2 **INTO** dia,ventas;

**WHILE** salida=0 **DO**

**SET** resultado=**CONCAT**(resultado,I'. ',dia,': ',ventas,' $\n');

**FETCH** CURSOR2 **INTO** dia,ventas;

**SET** i=i+1;

**END** **WHILE**;

**CLOSE** CURSOR2;

**SELECT** resultado;

**END**

**DROP** **PROCEDURE** mostrar\_estadisticas();