Procedimientos que

6.1 Nos muestre el producto (Id y nombre) más vendido de un cliente en un año determinado (Por defecto 2012)

#6.1 Nos muestre el producto (Id y nombre) más vendido de un cliente en un año determinado (Por defecto 2012)

USE neGOcios2011;

DELIMITER $$

DROP PROCEDURE IF EXISTS SP\_PRODUCTO\_CLIENTE $$

CREATE PROCEDURE SP\_PRODUCTO\_CLIENTE(IN IDCLIENTE VARCHAR(5),AÑO INT )

BEGIN

SELECT pro.idproducto ,pro.nomproducto

FROM productos pro

JOIN pedidosdeta pedd ON pro.idproducto = pedd.idproducto

JOIN pedidoscabe pedc ON pedd.idpedido = pedc.idpedido

WHERE pedc.idcliente = IDCLIENTE AND

YEAR(FECHAPEDIDO) = AÑO

group by pro.idproducto ,pro.nomproducto

ORDER BY sum(PEDD.PRECIOUNIDAD\*CANTIDAD) DESC LIMIT 1 ;

END $$

##LLAMADA PROCEDIMIENTO

USE neGOcios2011;

CALL SP\_PRODUCTO\_CLIENTE('NCBCK',2012);

6.2 Reciba un cargo, Devuelva el id del empleado y el salario del que más gana de un cargo determinado.

#6.2 Reciba un cargo, Devuelva el id del empleado y el salario del

#que más gana de un cargo determinado.

USE neGOcios2011;

DELIMITER $$

DROP PROCEDURE IF EXISTS SP\_CARGO\_SALARIO $$

CREATE PROCEDURE SP\_CARGO\_SALARIO(IN IDCARGO INT )

BEGIN

SELECT empleados.idempleado ,MAX(empleados.Salario)

FROM empleados

WHERE empleados.idcargo IN(SELECT cars.idcargo FROM cars

WHERE cars.idcargo= IDCARGO )

group by empleados.idempleado

ORDER BY MAX(empleados.Salario) DESC LIMIT 1 ;

END $$

##LLAMADA PROCEDIMIENTO

USE neGOcios2011;

CALL SP\_CARGO\_SALARIO(16);

##OTRO PROCEDIMIENTO:

delimiter $$

DROP PROCEDURE IF EXISTS seistres $$

CREATE PROCEDURE seistres (IN PORC INT)

BEGIN

select idempleado, salario

from empleados empl

where empl.idcargo = IDCARG

order by salario desc limit 1;

END $$

delimiter ;

call seistres();

6.3 Suba en un determinado porcentaje el precio del producto a una categoría determinada.

#6.3 Suba en un determinado porcentaje el precio del producto a una categoría determinada.

USE neGOcios2011;

DELIMITER $$

DROP PROCEDURE IF EXISTS SP\_SUBIR\_PRECIO\_CATEGORIA $$

CREATE PROCEDURE SP\_SUBIR\_PRECIO\_CATEGORIA(IN PORCENTAJE INT, NOMBRE\_CATEGORIA VARCHAR(40))

BEGIN

UPDATE productos SET productos.preciounidad = (productos.preciounidad\*(PORCENTAJE/100))

WHERE productos.idcategoria IN(SELECT categorias.idcategoria FROM categorias WHERE

categorias.nomcategoria LIKE NOMBRE\_CATEGORIA);

END $$

##LLAMADA PROCEDIMIENTO

USE neGOcios2011;

CALL SP\_SUBIR\_PRECIO\_CATEGORIA(10, 'Verduras');

6.4 Reciba un Id Empleado. Nos devuelva el número de pedidos y la suma de las cantidades de los pedidos suyos . Todo en un mes y año determinado , por defecto 2012 y 03.(dos select). MIRAR

#6.4 Reciba un Id Empleado. Nos devuelva el número de pedidos y la suma de las cantidades

# de los pedidos suyos . Todo en un mes y año determinado , por defecto 2012 y 03.(dos select) 2012 2013 02

USE neGOcios2011;

DELIMITER $$

DROP PROCEDURE IF EXISTS NUMEROS\_SUMA\_CANTIDAD\_PEDIDOS\_EMPLEADO $$

CREATE PROCEDURE NUMEROS\_SUMA\_CANTIDAD\_PEDIDOS\_EMPLEADO(ID\_EMPLEADO INT, AÑO INT, MES INT, OUT SALIDA1 INT,

OUT SALIDA2 INT)

BEGIN

#primera consulta

SELECT COUNT(pedidoscabecera.idpedido) INTO SALIDA1

FROM pedidoscabecera, pedidosdetalle WHERE pedidoscabecera.idempleado = ID\_EMPLEADO

AND YEAR(pedidoscabecera.fechapedido) LIKE AÑO

AND MONTH(pedidoscabecera.fechapedido) LIKE MES;

#SEGUNDA CONSULTA

SELECT SUM(pedidosdetalle.cantidad) INTO SALIDA2

FROM pedidoscabecera, pedidosdetalle WHERE pedidoscabecera.idempleado = ID\_EMPLEADO

AND YEAR(pedidoscabecera.fechapedido) LIKE AÑO

AND MONTH(pedidoscabecera.fechapedido) LIKE MES;

END $$

##LLAMADA PROCEDIMIENTO

USE neGOcios2011;

CALL NUMEROS\_SUMA\_CANTIDAD\_PEDIDOS\_EMPLEADO(1,2012,02,@SALIDA1 ,@SALIDA2 );

select @SALIDA1 AS NUMERO\_PEDIDOS, @SALIDA2 AS SUMA\_CANTIDADES;

6.5 Mostrar en cuantos países distintos se vende un determinado producto en un año determinado. MIRAR

USE negocios2011

DELIMITER $$

DROP PROCEDURE IF EXISTS agenda.Insertar $$

CREATE PROCEDURE USP\_PRODVENTAPAISES( IN PROD INT, IN AÑO INT, OUT NPAIS INT)

BEGIN

SELECT COUNT(DISTINCT C.idpais) INTO NPAIS

FROM clientes c

JOIN pedidoscabecera vp ON c.idcliente = vp.idcliente

JOIN pedidosdetalle pd ON vp.idpedido = pd.idpedido

WHERE pd.idproducto = PROD

AND YEAR(vp.fechapedido) = AÑO;

END $$

DELIMITER ;

CALL USP\_PRODVENTAPAISES(15,2012,@NPAIS );

select @NPAIS AS PAISES;

6.6 Mostrar el nombre de las tres categorías de las que más se ha vendido en un año determinado.

#6.6 Mostrar el nombre de las tres categorías de las que más se ha vendido en un año determinado.

USE neGOcios2011;

DELIMITER $$

DROP PROCEDURE IF EXISTS TRES\_CATEGORIAS\_MAS\_VENDIDAS $$

CREATE PROCEDURE TRES\_CATEGORIAS\_MAS\_VENDIDAS(IN AÑO INT, OUT SALIDA1 VARCHAR(40))

BEGIN

select distinct categorias.nomcategoria INTO SALIDA1

from categorias WHERE EXISTS (SELECT productos.idcategoria

FROM productos WHERE EXISTS ( SELECT pedidosdetalle.idproducto

FROM pedidosdetalle WHERE EXISTS (SELECT pedidoscabecera.idpedido

FROM pedidoscabecera WHERE year(pedidoscabecera.fechapedido) LIKE AÑO

)# FIN PEDIDOS CABECERA

GROUP BY pedidosdetalle.idproducto

order by MAX(pedidosdetalle.cantidad) desc limit 3

)## FIN PEDIDOS DETALLES

)#FIN PRODUCTOS

ORDER BY categorias.nomcategoria DESC LIMIT 3;

END $$

DELIMITER ;

CALL TRES\_CATEGORIAS\_MAS\_VENDIDAS(2012,@SALIDA1 );

select @SALIDA1 AS CATEGORIAS;

6.7 Realizar el alta de un pedido y una línea y actualizar stock con control de errores (try) y de transacciones. (Está en el Manual pag 112)

#6.7 Realizar el alta de un pedido y una línea y actualizar stock con control de errores (try) y

# de transacciones. (Está en el Manual pag 112)

# Inserta una cabecera de pedido y una linea de pedido (es pedidoDetalles)

USE neGOcios2011;

DELIMITER $$

DROP PROCEDURE IF EXISTS USP\_AGREGAPEDIDO $$

CREATE PROCEDURE USP\_AGREGAPEDIDO(IN IDPED INT,IN IDCLI VARCHAR(5) ,IN IDEMP INT ,IN FECPED DATE

,IN IDPROD INT, PRE DECIMAL,IN CANT INT ,OUT SALIDA varchar(100) )

BEGIN

START TRANSACTION;

BEGIN TRY;

#INSERTAMOS EN PEDIDO CABECERA

INSERT INTO pedidoscabecera (idpedido, idcliente,idempleado, fechapedido, fechaentrega, fechaenvio,

enviopedido, destinatario, dirdestinatario)

VALUES(IDPED,IDCLI,IDEMP,FECPED, NULL,NULL, '','Isabel','C/ Marqués de Murrieta, nº 43');

#INSERTAMOS EN pedidosdetalle - EL IDPEDIDO SE INCREMENTARA AUTOMATICAMENTE

INSERT INTO pedidosdetalle (idpedido, idproducto,preciounidad, cantidad, descuento)

VALUES(IDPED,IDPROD,PRE,CANT, 1.48);

END TRY;

COMMIT TRANSACTION;

END $$

DELIMITER ;

CALL USP\_AGREGAPEDIDO(101, 'NCBCK',1,'2013-02-01',89,6.59,85,@SALIDA1 );

select @SALIDA1 AS CATEGORIAS;

6.8 Modificar el anterior para que vuelva hacia atrás si el resultado de las existencias del producto da menor que 0.