

Procedimientos con la BD EMPLEADOS.

**1.-** Procedimiento "Cambiar precios": Se le entrega un porcentaje y debe subir los precios de todos los productos de los que existan 10 o menos unidades en dicho porcentaje.

**CREATE PROC CAMBIAR\_PRECIOS**

@porcentaje decimal(4,2)

AS

UPDATE PRODUCTOS

SET PRECIO = PRECIO \* (1 + @porcentaje/100)

WHERE (EXISTENCIAS <= 10)

EXEC CAMBIAR\_PRECIOS

@porcentaje=15

**2.-** Procedimiento "Dar de baja": Se le entrega el nombre de un cliente (EMPRESA) y una fecha y da de baja a dicho cliente si no ha realizado ningún pedido después de dicha fecha.

**CREATE PROC DAR\_DE\_BAJA**

@EMPRESA char(20),

@FECHA datetime

AS

DELETE

FROM CLIENTES

WHERE EMPRESA=@EMPRESA AND NOT EXISTS

(SELECT \*

FROM CLIENTES C INNER JOIN PEDIDOS P

ON C.NUM\_CLIE =P.CLIE

WHERE FECHA\_PEDIDO>@FECHA AND EMPRESA=@EMPRESA)

EXEC DAR\_DE\_BAJA

@EMPRESA = 'FRED LEWIS CORP.',

@FECHA = '18/02/90'

**3.-** Procedimiento "Grabar pedido": Se le entregan los datos necesarios para insertar un pedido. Si el código del cliente, del producto o del representante no se corresponde con ninguno de los almacenados en la BD, deberá emitir el mensaje de error correspondiente y detener el proceso.

**CREATE PROC GRABAR\_PEDIDO**

@NUM\_PEDIDO numeric(18,0),

@CLIE int,

@REP int,

```

        @FAB                char(3),
        @PRODUCTO          char(5),
        @CANT               int,
        @IMPORTE            numeric(18,0)
AS

    IF NOT EXISTS (SELECT *
                    FROM CLIENTES
                    WHERE NUM_CLIE = @CLIE)
        SELECT 'Código del cliente erróneo, fin del proceso'
    ELSE
        IF NOT EXISTS (SELECT *
                        FROM PRODUCTOS
                        WHERE ID_FAB = @FAB AND ID_PRODUCTO = @PRODUCTO)

            SELECT 'Código del producto erróneo, fin del proceso'
        ELSE
            IF NOT EXISTS (SELECT *
                            FROM REPVENTAS
                            WHERE NUM_EMPL = @REP)
                SELECT 'Código del representante erróneo, fin del proceso'
            ELSE
                BEGIN
                    INSERT PEDIDOS
                        VALUES
                        (@NUM_PEDIDO, GETDATE(), @CLIE, @REP, @FAB, @PRODUCTO, @CANT,
                        @IMPORTE)
                END

EXEC GRABAR_PEDIDO
@NUM_PEDIDO = 1111111,
@CLIE = 2101,
@REP      = 101,
@FAB      = 'ACI',
@PRODUCTO = '41001',
@CANT = 10,
@IMPORTE = 30

```

#### Procedimientos con la BD COMPRAVENTA

**4.- Procedimiento: “Valor pedidos”:** Se le entrega el nombre de un cliente (NOMBRECOMPAÑÍA) y dos fechas y el procedimiento debe devolver el valor de los pedidos realizados por el cliente entre dichas fechas, así como el número de ellos.

- a) Sin utilizar parámetros de salida.
- b) Utilizándolos.

```

CREATE PROC VALOR_PEDIDOS_SIN
    @NOMBRECOMPAÑÍA nvarchar(40),

```

```

        @FECHAINICIO    datetime,
        @FECHAFIN        datetime
AS
    DECLARE @IDCLIENTE nvarchar(5)
    SELECT @IDCLIENTE =IDCLIENTE
        FROM CLIENTES
        WHERE NOMBRECOMPAÑÍA = @NOMBRECOMPAÑÍA
    SELECT COUNT(*) AS NUMPEDIDOS, SUM(VALORPEDIDO) AS
VALORPEDIDOS
        FROM PEDIDOS P INNER JOIN VALORPEDIDOS VP
            ON P.IDPEDIDO = VP.IDPEDIDO
        WHERE (IDCLIENTE = @IDCLIENTE) AND FECHAPEDIDO
BETWEEN @FECHAINICIO AND @FECHAFIN
    GROUP BY IDCLIENTE

```

```

EXEC VALOR_PEDIDOS
@NOMBRECOMPAÑÍA='Bottom-Dollar Markets',
@FECHAINICIO = '23-04-98',
@FECHAFIN = '24-04-98'

```

#### **CREATE PROC VALOR\_PEDIDOS\_CON**

```

        @NUMPEDIDOS    int                OUTPUT,
        @VALORPEDIDOS    numeric(10,2)    OUTPUT,
        @NOMBRECOMPAÑÍA nvarchar(40),
        @FECHAINICIO    datetime,
        @FECHAFIN        datetime
AS
    DECLARE @IDCLIENTE nvarchar(5)
    SELECT @IDCLIENTE =IDCLIENTE
        FROM CLIENTES
        WHERE NOMBRECOMPAÑÍA = @NOMBRECOMPAÑÍA
    SELECT @NUMPEDIDOS = COUNT(*), @VALORPEDIDOS =
SUM(VALORPEDIDO)
        FROM PEDIDOS P INNER JOIN VALORPEDIDOS VP
            ON P.IDPEDIDO = VP.IDPEDIDO
        WHERE (IDCLIENTE = @IDCLIENTE) AND FECHAPEDIDO
BETWEEN @FECHAINICIO AND @FECHAFIN
    GROUP BY IDCLIENTE

```

```

DECLARE @NUMPEDIDOS int
DECLARE @VALORPEDIDOS numeric(10,2)
EXEC VALOR_PEDIDOS
@NUMPEDIDOS OUTPUT,
@VALORPEDIDOS OUTPUT,
@NOMBRECOMPAÑÍA='Bottom-Dollar Markets',
@FECHAINICIO = '23-04-98',
@FECHAFIN = '24-04-98'
SELECT 'Número de pedidos: ',@NUMPEDIDOS, 'Valor: ', @VALORPEDIDOS

```

**5.-** Procedimiento “Valor ventas”: Se le entrega el apellido de un empleado y debe visualizar el valor de sus ventas. Dicho valor deberá obtenerse de una vista creada previamente. La vista estará formada por dos campos: El apellido de cada vendedor y el valor de sus ventas. Utiliza parámetros de salida.

#### **CREATE VIEW VENDEDOR\_VENTAS**

AS

```
SELECT APELLIDOS,SUM(VALORPEDIDO) AS VALORVENTAS
FROM (SELECT APELLIDOS, IDPEDIDO
      FROM EMPLEADOS E INNER JOIN PEDIDOS P
      ON E.IDEMPLEADO = P.IDEMPLEADO) A
      INNER JOIN
      VALORPEDIDOS VP
      ON A.IDPEDIDO = VP.IDPEDIDO
GROUP BY APELLIDOS
```

#### **CREATE PROC VALOR\_VENTAS**

```
@VALORVENTAS numeric(10,2)      OUTPUT,
@APELLIDOS nvarchar(20)
```

AS

```
SELECT @VALORVENTAS = VALORVENTAS
FROM VENDEDOR_VENTAS
WHERE APELLIDOS =@APELLIDOS
```

```
DECLARE @VALORVENTAS numeric(10,2)
EXEC VALOR_VENTAS
@VALORVENTAS OUTPUT,
@APELLIDOS = 'Peacock'
SELECT 'Valor de las ventas: ',@VALORVENTAS
```

**6.-** Procedimiento “Producto más caro”: Se le entrega el nombre de una categoría y debe devolver el nombre del producto más caro de la categoría. Si el nombre de la categoría no existe en la BD deberá emitir un mensaje de error. El procedimiento debe utilizar una vista creada previamente. Dicha vista constará de tres columnas: nombre de categoría, nombre de producto y precio unitario. Deberá estar ordenada en ascendente por categoría y, a igualdad de categoría, en descendente por precio unitario. Utiliza parámetros de salida.

#### **CREATE VIEW CATEGORÍAS\_PRODUCTOS\_PRECIOS**

AS

```
SELECT TOP 77 NOMBRECATEGORÍA, NOMBREPRODUCTO,
PRECIOUNIDAD
FROM CATEGORÍAS C INNER JOIN PRODUCTOS P
ON C.IDCATEGORÍA = P.IDCATEGORÍA
ORDER BY NOMBRECATEGORÍA,PRECIOUNIDAD DESC
```

**CREATE PROC PRODUCTO\_MÁS\_CARO**

```
@NOMBREPRODUCTO nvarchar(40) OUTPUT,  
@NOMBRECATEGORÍA nvarchar(15)  
AS  
    SELECT @NOMBREPRODUCTO =  
        (SELECT TOP 1 NOMBREPRODUCTO  
         FROM CATEGORÍAS_PRODUCTOS_PRECIOS  
         WHERE NOMBRECATEGORÍA=@NOMBRECATEGORÍA)  
  
DECLARE @NOMBREPRODUCTO nvarchar(40)  
EXEC PRODUCTO_MÁS_CARO  
@NOMBREPRODUCTO OUTPUT,  
@NOMBRECATEGORÍA = 'Condimentos'  
SELECT 'Producto más caro: ', @NOMBREPRODUCTO
```

**7.-** Procedimiento “Ventas anuales”. Se le entrega un año y debe mostrar los nombres de las categorías junto con el valor de sus ventas en dicho año. El procedimiento debe utilizar una vista creada anteriormente. Dicha vista constará de tres campos. Nombre de categoría, año y valor de las ventas de productos de la categoría en el año. Deberá estar ordenada por categoría y año.

**CREATE VIEW CATEGORÍAS\_AÑOS\_VENTAS**

```
AS  
    SELECT TOP 24 NOMBRECATEGORÍA, AÑO, SUM(VAORVENTAS)  
AS VENTASCATEGORÍA  
    FROM (SELECT NOMBRECATEGORÍA,IDPRODUCTO  
            FROM CATEGORÍAS C INNER JOIN PRODUCTOS P  
            ON C.IDCATEGORÍA  
            =P.IDCATEGORÍA) A  
        INNER JOIN  
        (SELECT YEAR(FECHAPEDIDO)AS  
AÑO,IDPRODUCTO,SUM(CANTIDAD*PRECIOUNIDAD) AS VAORVENTAS  
        FROM PEDIDOS P INNER JOIN [DETALLES DE  
PEDIDOS] DDP  
        ON P.IDPEDIDO  
        =DDP.IDPEDIDO  
        GROUP BY YEAR(FECHAPEDIDO),IDPRODUCTO)  
B  
        ON A.IDPRODUCTO = B.IDPRODUCTO  
        GROUP BY AÑO,NOMBRECATEGORÍA  
        ORDER BY NOMBRECATEGORÍA,AÑO
```

**CREATE PROC VENTAS\_POR\_CATEGORÍA**

```
@AÑO int  
AS  
    SELECT NOMBRECATEGORÍA,VENTASCATEGORÍA  
        FROM CATEGORÍAS_AÑOS_VENTAS  
        WHERE AÑO=@AÑO
```

EXEC VENTAS\_POR\_CATEGORÍA  
@AÑO=1996