

## B.D. COMPRAVENTA

- 1.- **Procedimiento Ciudades.** Se le entrega el apellido de un empleado y debe visualizar el Nombre de las compañías clientes que tienen su sede en la misma ciudad que el empleado. Si no hay ninguno deberá emitir el mensaje correspondiente.

```
CREATE PROCEDURE Ciudades
@Apellidos nvarchar(20)
AS
IF EXISTS (SELECT *
            FROM CLIENTES C INNER JOIN EMPLEADOS E
            ON C.Ciudad=E.Ciudad
            WHERE Apellidos=@Apellidos)

    SELECT NombreCompañía
    FROM CLIENTES C INNER JOIN EMPLEADOS E
    ON C.Ciudad=E.Ciudad
    WHERE Apellidos=@Apellidos

ELSE
    SELECT 'No hay ningún cliente'

EXEC Ciudades
'Suyama'

EXEC Ciudades
'Peacock'
```

- 2.- **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. Sin/Con parámetros de salida.

```
CREATE VIEW ValorVentasEmpleados
AS
SELECT Apellidos, SUM(ValorConCargo) AS ValorVentas
FROM EMPLEADOS E INNER JOIN PEDIDOS P
    INNER JOIN ValorPedidosConCargo VP
    ON VP.IdPedido=P.IdPedido
    ON E.IdEmpleado=P.IdEmpleado
GROUP BY Apellidos

CREATE PROC EJ5
@ValorVentas float OUTPUT,
@Apellidos nvarchar(20)
```

```
AS
SELECT @ValorVentas=ValorVentas
      FROM ValorVentasEmpleados
      WHERE Apellidos=@Apellidos
```

```
DECLARE @ValorVentas nvarchar(20)
EXEC EJ5
@ValorVentas      OUTPUT, 'Suyama'
SELECT @ValorVentas
```

**3.- 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 100 NombreCategoría, NombreProducto, PrecioUnidad
      FROM CATEGORÍAS C INNER JOIN PRODUCTOS P
            ON C.IdCategoría=P.IdCategoría
      ORDER BY NombreCategoría, PrecioUnidad DESC
```

```
CREATE PROC ProductoMásCaro
@NombreProducto nvarchar(40) OUTPUT,
@NombreCategoría nvarchar(15)
AS
IF NOT EXISTS (SELECT *
              FROM CATEGORÍAS
              WHERE NombreCategoría=@NombreCategoría)
      SELECT 'Categoría no válida'
ELSE
      SELECT TOP 1 @NombreProducto=NombreProducto
            FROM Categorías_Productos_Precios
            WHERE NombreCategoría=@NombreCategoría

DECLARE @NombreProducto nvarchar(40)
EXEC ProductoMásCaro
@NombreProducto OUTPUT,
'bebidas'
      SELECT 'Producto más caro. ', @NombreProducto
```

**4.- 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 25 NombreCategoría, YEAR(FechaPedido) AÑO,
SUM(DDP.PrecioUnidad*Cantidad*(1-Descuento)) ValorVentas
FROM CATEGORÍAS C INNER JOIN
PRODUCTOS PR INNER JOIN
[Detalles de Pedidos] DDP INNER JOIN PEDIDOS P
ON DDP.IdPedido=P.IdPedido
ON PR.IdProducto=DDP.IdProducto
ON C.IdCategoría=PR.IdCategoría
GROUP BY NombreCategoría, YEAR(FechaPedido)
ORDER BY NombreCategoría,Año
```

```
CREATE PROC Ventas_Anuales_Por_Categoría
@Año int
AS
SELECT NombreCategoría,ValorVentas
FROM Categorías_Años_Ventas
WHERE Año=@Año
```

```
EXEC Ventas_Anuales_Por_Categoría
1998
```

**5.- Procedimiento “Cargos Demoras”.** Recibe el nombre de una compañía de envíos y devuelve, mediante parámetros de salida, la media de los cargos cobrados por ésta así como su demora media (demora: días transcurridos entre la fecha de envío de un pedido y su fecha de entrega).

```
CREATE PROC CargosDemoras
@MediaCargos numeric(10,4) OUTPUT,
@MediaDemoras numeric(10,4) OUTPUT,
@Nombre nvarchar(40)
AS
DECLARE @IdCompañía int
SELECT @IdCompañía=IdCompañíaEnvíos
FROM [Compañías De Envíos]
WHERE NombreCompañía=@Nombre
SELECT @MediaCargos=AVG(Cargo),
@MediaDemoras=AVG(datediff(dd,FechaEnvío,FechaEntrega))
FROM Pedidos
WHERE FormaEnvío=@IdCompañía

DECLARE @MediaCargosS numeric(10,4), @MediaDemorasS numeric(10,4)
EXEC CargosDemoras
@MediaCargosS OUTPUT,
```

```
@MediaDemorasS OUTPUT,
'Speedy Express'
SELECT 'Media Cargos: ',@MediaCargosS,'Media Demoras: ',@MediaDemorasS
```

**6.- Procedimiento Producto Estrella.** Recibe un mes y un año y devuelve, mediante parámetro de salida, el nombre del producto “estrella”, es decir, aquél producto del que se pidieron más unidades en dicho mes de dicho año.

```
CREATE PROC ProductoEstrella
@NombreProducto nvarchar(40) OUTPUT,
@Mes int,
@Año int
AS
DECLARE @IdProducto int
SELECT TOP 1 @Idproducto=IdProducto
FROM Pedidos P INNER JOIN [Detalles de Pedidos] DDP
ON P.IdPedido=DDP.IdPedido
WHERE YEAR(FechaPedido)=@Año AND MONTH(FechaPedido)=@Mes
GROUP BY IdProducto
ORDER BY SUM(Cantidad) DESC
SELECT @NombreProducto=NombreProducto
FROM Productos
WHERE IdProducto=@IdProducto

DECLARE @NombreProductoS nvarchar(40)
EXEC ProductoEstrella
@NombreProductoS OUTPUT,7,1997
SELECT 'Producto estrella: ',@NombreProductoS
```

**7.- Procedimiento Jefes.** Se le entrega el apellido de un empleado y debe visualizar el nombre y apellido de su jefe, así como el nombre de la región a la que el empleado está asignado. Utilizar parámetros de salida.

```
CREATE PROCEDURE JefesRegiones
@NombreJefe nvarchar(10) OUTPUT,
@Apellidosjefe nvarchar(20) OUTPUT,
@Región nchar(50) OUTPUT,
@Apellidos nvarchar(20)
AS
DECLARE @Jefe int,@IdEmpleado int
SELECT @Idempleado=IdEmpleado
FROM EMPLEADOS
WHERE Apellidos=@Apellidos
SELECT @jefe=Jefe
FROM EMPLEADOS
WHERE IdEmpleado=@IdEmpleado
SELECT @NombreJefe=Nombre,@ApellidosJefe=Apellidos
```

```

FROM EMPLEADOS
WHERE IdEmpleado=@jefe
SELECT @Región=R.Descripción
FROM EMPLEADOSTERRITORIOS ET INNER JOIN TERRITORIOS T
INNER JOIN REGIONES R
ON T.IdRegión=R.IdRegión
ON ET.IdTerritorio=T.IdTerritorio

WHERE IdEmpleado=@IdEmpleado

```

```

DECLARE @NombreJefe      nvarchar(10),
@Apellidosjefe    nvarchar(20) ,
@Región          nchar(50)
EXEC JefesRegiones
@NombreJefe      OUTPUT,
@Apellidosjefe    OUTPUT,
@Región          OUTPUT,
'Peacock'

SELECT @Nombrejefe, @ApellidosJefe, @Región

```