

Procedimientos Almacenados para la tabla Producto

PROCEDIMIENTO ALMACENADO UTILIZADO PARA INSERTAR UN
PRODUCTO A LA BASE DE DATOS

```
*/
CREATE PROC dbo.spI_Producto
    @codigoProducto  int = Null OUTPUT,
    @nombre  varchar(100) = Null,
    @precio  decimal(18, 2) = Null
AS
insert into Producto
(
    nombre,
    precio
)
VALUES(
@nombre,
@precio
)
--Obteniendo el codigo autogenerado de producto
SET @codigoProducto = @@IDENTITY;
GO
/*
```

PROCEDIMIENTO ALMACENADO UTILIZADO PARA ACTUALIZAR UN
PROCDUCTO A LA BASE DE DATOS

```
*/
CREATE PROC dbo.spU_Producto
    @codigoProducto  int = Null,
    @nombre  varchar(100) = Null,
    @precio  decimal(18, 2) = Null
AS

UPDATE Producto
SET
    nombre = @nombre,
    precio = @precio
WHERE
    codigoProducto = @codigoProducto
GO
/*
```

PROCEDIMIENTO ALMACENADO UTILIZADO PARA OBTENER TODOS
LOS PRODUCTOS DE LA BASE DE DATOS

```
*/
```

```

CREATE PROC dbo.spF_Producto_All
AS
SELECT
    p.codigoProducto,
    p.nombre,
    p.precio
FROM
    Producto p
ORDER BY
    P.nombre

```

Procedimientos Almacenados para la tabla Venta

```

/*
-----
PROCEDIMIENTO ALMACENADO UTILIZADO PARA INSERTAR UNA
VENTA A LA BASE DE DATOS
-----
*/
CREATE PROC dbo.spI_Venta
    @codigoVenta  int = Null OUTPUT,
    @cliente  varchar(100) = Null
AS

insert into Venta
(
    cliente,
    fecha
)
VALUES(
@cliente,
GETDATE()
)
--Obteniendo el codigo autogenerado de la venta
SET @codigoVenta = @@IDENTITY
GO

/*
-----
PROCEDIMIENTO ALMACENADO UTILIZADO PARA OBTENER EL
REPORTE DE LA VENTA DE LA BASE DE DATOS
-----
*/
CREATE PROCEDURE dbo.spF_Venta_One
    @codigoVenta int
AS
SELECT

```

```

v.codigoVenta AS CodigoVenta,
v.cliente AS Cliente,
v.fecha AS Fecha,
d.codigoProducto AS CodigoProducto,
p.nombre AS Nombre,
p.precio AS Precio,
d.cantidad AS Cantidad,
d.descuento AS Descuento,
p.precio*d.cantidad AS Parcial,
((p.precio*d.cantidad)-d.descuento) AS SubTotal,
(
SELECT
SUM((dT.cantidad * pT.precio)-dT.descuento) AS TotalPagar
FROM
DetalleVenta AS dT INNER JOIN
Producto AS pT ON dT.codigoProducto = pT.codigoProducto
WHERE
dT.codigoVenta=v.codigoVenta
) AS TotalPagar
FROM
Venta AS v INNER JOIN
DetalleVenta AS d ON v.codigoVenta = d.codigoVenta INNER JOIN
Producto AS p ON d.codigoProducto = p.codigoProducto
WHERE
v.codigoVenta=@codigoVenta
ORDER BY
Nombre

```

GO

/*

PROCEDIMIENTO ALMACENADO UTILIZADO PARA OBTENER TODAS
LAS VENTAS DE LA BASE DE DATOS

*/

```

CREATE PROCEDURE dbo.spF_Venta_All
AS
SELECT

```

```

v.codigoVenta AS CodigoVenta,
v.cliente AS Cliente,
v.fecha AS Fecha,
d.codigoProducto AS CodigoProducto,
p.nombre AS Nombre,
p.precio AS Precio,
d.cantidad AS Cantidad,
d.descuento AS Descuento,
p.precio*d.cantidad AS Parcial,

```

```

        ((p.precio*d.cantidad)-d.descuento) AS SubTotal,
    (
SELECT
        SUM((dT.cantidad * pT.precio)-dT.descuento) AS TotalPagar
FROM
        DetalleVenta AS dT INNER JOIN
        Producto AS pT ON dT.codigoProducto = pT.codigoProducto
WHERE
        dT.codigoVenta=v.codigoVenta
    ) AS TotalPagar
FROM
        Venta AS v INNER JOIN
        DetalleVenta AS d ON v.codigoVenta = d.codigoVenta INNER JOIN
        Producto AS p ON d.codigoProducto = p.codigoProducto
ORDER BY
        CodigoVenta, Nombre

```

Procedimientos Almacenados para la tabla DetalleVenta

```

/*
PROCEDIMIENTO ALMACENADO UTILIZADO PARA INSERTAR UN
DETALLE DE VENTA A LA BASE DE DATOS
-----
*/
CREATE PROC dbo.spI_DetalleVenta
    @codigoVenta    int = Null,
    @codigoProducto int = Null,
    @cantidad       decimal(18, 2) = Null,
    @descuento      decimal(18, 2) = Null
AS
insert into DetalleVenta
(
    codigoVenta,
    codigoProducto,
    cantidad,
    descuento
)
VALUES(
    @codigoVenta,
    @codigoProducto,
    @cantidad,
    @descuento
)

```