


```

                                @fxNacimiento datetime

AS
BEGIN TRY

        BEGIN TRAN

        INSERT INTO CLIENTES

            (nombre, apellido1, apellido2, nifcif, fxnacimiento) VALUES

            (@nombre, @apellido1, @apellido2, @nifCif, @fxNacimiento)

        COMMIT

END TRY
BEGIN CATCH

        ROLLBACK

        PRINT ERROR_MESSAGE()

END CATCH

```

Si queremos que los parámetros de un procedimiento almacenado sean de entrada-salida debemos especificarlo a través de la palabra clave **OUTPUT** , tanto en la definición del procedure como en la ejecución.

El siguiente ejemplo muestra la definición de un procedure con parámetros de salida.

```

CREATE PROCEDURE spu_ObtenerSaldoCuenta @numCuenta varchar(20),
                                @saldo decimal(10,2) output

AS
BEGIN

        SELECT @saldo = SALDO

        FROM CUENTAS

        WHERE NUMCUENTA = @numCuenta

END

```

Y para ejecutar este procedure:

```

DECLARE @saldo decimal(10,2)
EXEC spu_ObtenerSaldoCuenta '200700000001', @saldo output
PRINT @saldo

```

Un procedimiento almacenado puede devolver valores numericos enteros a través de la instrucción RETURN. Normalmente debemos utilizar los valores de retorno para determinar si la ejecución del procedimiento ha sido correcta o no. Si queremos obtener valores se recomienda utilizar parámetros de salida o funciones escalares (se verán mas adelante en este tutorial).

El siguiente ejemplo muestra un procedimiento almacenado que devuelve valores.

```

CREATE PROCEDURE spu_EstaEnNumerosRojos @numCuenta varchar(20)
AS

```

```

BEGIN

    IF (SELECT SALDO FROM CUENTAS

        WHERE NUMCUENTA = @numCuenta) < 0

        BEGIN

            RETURN 1

        END

    ELSE

        RETURN 0

END

```

El siguiente ejemplo muestra como ejecutar el procedure y obtener el valor devuelto.

```

DECLARE @rv int
EXEC @rv = spu_EstaEnNumerosRojos '200700000001'
PRINT @rv

```

Otra característica muy interesante de los procedimientos almacenados en Transact SQL es que pueden devolver uno o varios conjuntos de resultados.

El siguiente ejemplo muestra un procedimiento almacenado que devuelve un conjunto de resultados.

```

CREATE PROCEDURE spu_MovimientosCuenta @numCuenta varchar(20)
AS
BEGIN

    SELECT @numCuenta,

        SALDO_ANTERIOR,

        SALDO_POSTERIOR,

        IMPORTE,

        FXMOVIMIENTO

    FROM MOVIMIENTOS

    INNER JOIN CUENTAS ON MOVIMIENTOS.IDCUENTA = CUENTAS.IDCUENTA

    WHERE NUMCUENTA = @numCuenta

    ORDER BY FXMOVIMIENTO DESC

END

```

La ejecución del procedimiento se realiza normalmente.

```

EXEC spu_MovimientosCuenta '200700000001'

```

El resultado de la ejecucion ...

NUMCUENTA	SALDO_ANTERIOR	SALDO_POSTERIOR	IMPORTE	FXMOVIMIENTO
200700000001	50.99	100.99	50.00	2007-08-25 16:18:36.490
200700000001	0.99	50.99	50.00	2007-08-23 16:20:41.183
200700000001	50.99	0.99	50.00	2007-08-23 16:16:29.840
200700000001	0.99	50.99	50.00	2007-08-23 16:14:05.900
