CREATE TABLE EMPLEADO

(

IDEmpleado INT  PRIMARY KEY,

Nombre VARCHAR(50),

Sueldo Decimal (18,0)

)

GO

--creamos un procedimiento almacenado

CREATE PROC GUARDAR\_EMPLEADO(

      @IdEmpleado        integer,

      @Nombre         varchar(50),

      @Sueldo      Decimal(18,0),

-- variable de salida, para conocer el mensaje de la transacción

      @MESSAGE          varchar(500) output

      )

AS

-- Asignamos un nombre a la transaccion

      BEGIN TRAN TRAN\_GUARDAR\_EMPLEADO

            BEGIN TRY

                  BEGIN

                        INSERT INTO EMPLEADO(IdEmpleado,

                                           Nombre,

                                           Sueldo

                                           )

                                  VALUES(@IdEmpleado,

                                           @Nombre,

                                           @Sueldo                                                 )

                        set @MESSAGE= 'Registro guardado'

                  END

            END TRY

            BEGIN CATCH

                  -- si se produciera algun error desacera los cambios

                  ROLLBACK TRAN TRAN\_GUARDAR\_EMPLEADO

                  -- asignara a la variable el mensaje de error

                  SET @MESSAGE      =     ERROR\_MESSAGE()

            RETURN

            END CATCH

--

COMMIT TRAN TRAN\_GUARDAR\_EMPLEADO

GO

--declaro la variable que me permita

--obtener el mensaje devuelto por la transacción

DECLARE @MESSAGE  VARCHAR(500)

EXEC GUARDAR\_EMPLEADO 1, 'XXX', 'YYYY', @MESSAGE OUTPUT

SELECT @MESSAGE

-- Primer mensaje al ejecutarlo

Registro guardado

Segundo mensaje al ejecutarlo por segunda vez, con el mismo

IDEmpleado =1  de la EMPLEADO

aquí podemos ver el mensaje de retorno, nos indica que hay una infracción

en la clave primaria la cual no puede ser duplicada\*/

Infracción de la restricción PRIMARY KEY'PK\_\_EMPLEADO\_\_8F57AAEA08EA5793'. No se puede insertar una clave duplicada en el objeto 'dbo.EMPLEADO'.

**Nota:**

* [BEGIN TRANSACTION](http://msdn.microsoft.com/es-es/library/ms188929.aspx) : Marca el punto de inicio de una transacción local explícita.
* [ROLLBACK](http://msdn.microsoft.com/es-pe/library/ms181299.aspx)  : Revierte una transacción explícita o implícita hasta el inicio de la transacción o hasta un punto de retorno dentro de la transacción. Puede usar ROLLBACK TRANSACTION para borrar todas las modificaciones de datos realizadas desde el inicio de la transacción o hasta un punto de retorno.
* [COMMIT](http://technet.microsoft.com/es-pe/library/ms190295.aspx) : Marca el final de una transacción correcta, implícita o explícita.
* [TRY ... CATCH](http://msdn.microsoft.com/en-us/library/ms175976.aspx) : Implementa el control de errores para Transact-SQL que es similar al manejo de excepciones en Microsoft Visual Basic