

2. Ejercicios de Procedimientos almacenados. Transact SQL.

- Realizar los siguientes ejercicios en SQL Server con la Base de datos del Ejercicio 2.
- Los ejercicios 1, 2 y 3 están resueltos para que sirvan como guía.
- Controla siempre los errores, trabajar con transacciones.

2.1. Realizar un procedimiento almacenado que cree las tres tablas del ejercicio o ninguna, si las mismas no existen. Si existen, borrarlas y volverlas a crear. Borrar con DROP que es más rápido que DELETE. (Usar transacciones).

```
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
use EMPRESA
go
CREATE PROCEDURE Crear_tablas
AS
begin
BEGIN tran
declare @error int

SET NOCOUNT ON;
if object_id('tcentr2') is not null
drop table temple2
drop table tdepto2
drop table tcentr2
create table tcentr2 (
    numce int primary key,
    nomce varchar(20) not null,
    señas varchar(25) not null
)
set @error = @@error
IF (@ERROR=-1)
    Begin
        rollback tran
        return (@error)
        print 'Error tabla TCentr2 no creada'
    end

create table tdepto2 (
    numde int primary key,
    numce int not null
    constraint fk_tdepto_tcentr_numde foreign key (numce)
references tcentr2 (numce),
    direc int not null,
    tidir char(1) not null,
    presu float not null,
    depde int,
    nomde varchar(30) not null
)
select @error = @@error
IF (@ERROR=-1)
    Begin
        rollback tran
        return (@error)
        print 'Error tabla Tdepto2 no creada'
```

```

        end

create table temple2
(
    numemp int primary key,
    numde int not null
    constraint fk_temple_tdepto_numde foreign key (numde)
references tdepto2 (numde),
    extel smallint not null,
    fecna date not null,
    fecin date not null,
    salar float not null,
    comis float,
    numhi smallint not null,
    nomem varchar(20) not null,
)
select @error = @@error
IF (@ERROR=-1)
    Begin
        rollback tran
        return (@error)
        print 'Error tabla Temple2 no creada'

    end
COMMIT TRAN
PRINT 'Tablas Creadas'

end

```

2.2. Crear un procedimiento almacenado para crear y cargar los datos de cada tabla, y otro donde se llaman a estos tres procedimientos. (Usar transacciones)

```

use empresa
go
CREATE PROCEDURE sp_datos_tcentr2
AS
BEGIN tran

    SET NOCOUNT ON;

        insert into tcentr2 values (10 , 'Sede Central' , 'C.
Alcala, 820, Madrid')

        insert into tcentr2 values (20 , 'Relacion Cliente' , 'C.
Atocha, 405, Madrid')

    declare @error int
    set @error = @@error
    IF (@error=-1)
        Begin
            rollback tran
            return (@error)
            print 'Error en la insercion de datos en tcentr2'

        end

    COMMIT TRAN
    PRINT 'Inserciones realizadas correctamente en Tcntr2'

CREATE PROCEDURE sp_datos_tdepto2

AS
BEGIN tran

```

```

        SET NOCOUNT ON;
        declare @error int

        insert into tdepto2 ( numde , numce , direc, tidir, presu ,
nomde) values (100 , 10 , 260 , 'P' , 12 , 'direccion general');
        insert into Tdepto2 values (110 , 20, 180 , 'P' , 15 , 100 ,
'dirección comercial');
        insert into Tdepto2 values (111 , 20 , 180 , 'F' , 11 , 110
, 'sector industrial');
        insert into Tdepto2 values (112 , 20 , 270 , 'P' , 9 , 110 ,
'sector servicios');
        insert into Tdepto2 values (120 , 10 , 150 , 'F' , 3 , 100 ,
'organización');
        insert into Tdepto2 values (121 , 10 , 150 , 'P' , 2 , 120 ,
'personal');
        insert into Tdepto2 values (122 , 10 , 350 , 'P' , 6 , 120 ,
'proceso de datos');
        insert into Tdepto2 values (130 , 10 , 310 , 'P' , 2 , 100 ,
'finanzas');
        set @error = @@error
        IF (@ERROR=-1)
        Begin
            rollback tran
            return (@error)
            print 'Error en la insercion de datos en tdepto2'
        end

        COMMIT TRAN

        PRINT 'Inserciones realizadas correctamente en Tdepto2'

CREATE PROCEDURE sp_datos_temple2

AS
BEGIN tran

        SET NOCOUNT ON;
        declare @error int
        insert into temple2
(numemp,numde,extel,fezna,fezn,salar,numhi,nomem) values (110 , 121 ,
350 , '10/11/1929' , '15/02/1950' , 310 , 3 , 'PONS, CESAR');
        insert into temple2 values (120 , 112 , 840 , '09/06/1935' ,
'01/10/1968' , 350 , 110 , 1 , 'LASA, MARIO');
        insert into temple2 values (130 , 112 , 810 , '09/09/1945' ,
'01/02/1969' , 290 , 110 , 2 , 'TEROL, LUCIANO');
        insert into temple2
(numemp,numde,extel,fezna,fezn,salar,numhi,nomem) values (150 , 121 ,
340 , '10/08/1930' , '15/01/1948' , 440 , 0 , 'PEREZ, JULIO');
        insert into temple2 values (160 , 111 , 740 , '09/07/1939' ,
'11/11/1968' , 310 , 110 , 2 , 'AGUIRRE, AUREO');
        insert into temple2 values (180 , 110 , 508 , '18/10/1934' ,
'18/03/1956' , 480 , 50 , 2 , 'PEREZ, MARCOS');
        insert into temple2
(numemp,numde,extel,fezna,fezn,salar,numhi,nomem) values (190 , 121 , 350
, '12/05/1932' , '11/02/1962' , 300 , 4 , 'VEIGA, JULIANA');
        insert into temple2
(numemp,numde,extel,fezna,fezn,salar,numhi,nomem) values (210 , 100 , 200
, '28/09/1940' , '22/01/1959' , 380 , 2 , 'GALVEZ, PILAR');
        insert into temple2 values (240 , 111 , 760 , '26/02/1942' ,
'24/02/1966' , 280 , 100 , 3 , 'SANZ, LAVINIA');
        insert into temple2

```

```

(numemp,numde,extel,fezna,fezn,salar,numhi,nomem) values (250, 100, 250
, '27/10/1946' , '01/03/1967' , 450,0,'ALBA, ADRIANA');
insert into temple2
(numemp,numde,extel,fezna,fezn,salar,numhi,nomem) values (260, 100, 220
, '03/12/1943' , '12/07/1968' , 720,6,'LOPEZ, ANTONIO');
insert into temple2 values (270, 112, 800 , '21/05/1945' ,
'10/09/1966' , 380,80,3,'GARCIA, OCTAVIO');
insert into temple2
(numemp,numde,extel,fezna,fezn,salar,numhi,nomem) values (280, 130, 410
, '11/01/1948' , '08/10/1971' , 290, 5,'FLOR, DOROTEA');
insert into temple2
(numemp,numde,extel,fezna,fezn,salar,numhi,nomem) values (285, 122, 620
, '25/10/1949' , '15/02/1968' , 380, 0,'POLO, OTILIA');
insert into temple2
(numemp,numde,extel,fezna,fezn,salar,numhi,nomem) values (290, 120, 910
, '30/11/1947' , '14/02/1968' , 270, 3,'GIL, GLORIA');
insert into temple2
(numemp,numde,extel,fezna,fezn,salar,numhi,nomem) values (310, 130, 480
, '21/11/1946' , '15/01/1971' , 420, 0,'GARCIA, AUGUSTO');
insert into temple2
(numemp,numde,extel,fezna,fezn,salar,numhi,nomem) values (320, 122, 620
, '25/12/1957' , '05/02/1978' , 405, 2,'SANZ, CORNELIO');
insert into temple2 values (330, 112, 850 , '19/08/1948' ,
'01/03/1972' , 280,90, 0,'DIEZ, AMELIA');
insert into temple2
(numemp,numde,extel,fezna,fezn,salar,numhi,nomem) values (350, 122,
610 , '13/04/1949' , '10/09/1984' , 450, 1,'CAMPS, AURELIO');
insert into temple2 values (360, 111, 750 , '29/10/1958' ,
'10/10/1968' , 250, 100, 2,'LARA, DORINDA');
insert into temple2
(numemp,numde,extel,fezna,fezn,salar,numhi,nomem) values (370, 121, 360
, '22/06/1967' , '20/01/1987' , 190, 1,'RUIZ, FABIOLA');
insert into temple2
(numemp,numde,extel,fezna,fezn,salar,numhi,nomem) values (380, 112, 880
, '22/06/1967' , '20/01/1987' , 190, 1,'MARTIN, MICAELA');
insert into temple2
(numemp,numde,extel,fezna,fezn,salar,numhi,nomem) values (390, 110, 500
, '19/02/1966' , '08/10/1986' , 215, 1,'MORAN, CARMEN');
insert into temple2
(numemp,numde,extel,fezna,fezn,salar,numhi,nomem) values (400, 111, 780
, '18/08/1969' , '01/11/1987' , 185, 0,'LARA, LUCRECIA');
insert into temple2
(numemp,numde,extel,fezna,fezn,salar,numhi,nomem) values (410, 122, 660
, '14/07/1968' , '13/10/1988' , 175, 0,'MUÑOZ, AZUCENA');
insert into temple2
(numemp,numde,extel,fezna,fezn,salar,numhi,nomem) values (420, 130, 450
, '22/10/1966' , '19/11/1988' , 400, 0,'FIERRO, CLAUDIA');
insert into temple2
(numemp,numde,extel,fezna,fezn,salar,numhi,nomem) values (430, 122, 650
, '26/02/1967' , '19/11/1988' , 210, 1,'MORA, VALERIANA');
insert into temple2 values (440, 111, 760 , '26/09/1966' ,
'28/02/1986' , 210,100, 0,'DURAN, LIVIA');
insert into temple2 values (450, 112, 880 , '21/10/1966' ,
'28/02/1986' , 210,100, 0,'PEREZ, SABINA');
insert into temple2 values (480, 111, 760 , '04/04/1965' ,
'28/02/1986' , 210,100, 1,'PINO, DIANA');
insert into temple2 values (490, 112, 880 , '06/06/1964' ,
'01/01/1988' , 280,100, 0,'TORRES, HORACIO');
insert into temple2 values (500, 111, 750 , '08/10/1965' ,
'01/01/1987' , 200,100, 0,'VAZQUEZ, HONORIA');
insert into temple2
(numemp,numde,extel,fezna,fezn,salar,numhi,nomem) values (510, 110, 550
, '04/05/1966' , '01/11/1986' , 200, 1,'CAMPOS, ROMULO');
insert into temple2 values (550, 111, 780 , '10/01/1970' ,
'21/01/1988' , 100,120, 0,'SANTOS, SANCHE');
set @error = @@error

```

```

        IF (@ERROR=-1)
            Begin
                rollback tran
                return (@error)
                print 'Error en la insercion de datos en temple2'
            end

        COMMIT TRAN

        PRINT 'Inserciones realizadas correctamente en Temple2'

```

Ahora usando los 3 procedimientos a la vez

```

CREATE PROCEDURE sp_backup_de_todas_las_tablas2

AS
BEGIN tran

    SET NOCOUNT ON;
    declare @error int
    exec sp_datos_tcentr2
    set @error = @@error
    IF (@ERROR=-1)
        Begin
            rollback tran
            return (@error)
            print 'Error en la insercion de datos en tcentr2'
        end
    exec sp_datos_tdepto2
    set @error = @@error
    IF (@ERROR=-1)
        Begin
            rollback tran
            return (@error)
            print 'Error en la insercion de datos en tdepto2'
        end
    exec sp_datos_temple2
    set @error = @@error
    IF (@ERROR=-1)
        Begin
            rollback tran
            return (@error)
            print 'Error en la insercion de datos en temple2'
        end

    COMMIT TRAN

    PRINT 'Base de datos cargada y lista para usarse'

```

2.3. Crear un sp que actualice los datos de la tabla empleados, pasándole toda la información del empleado. Si se produce un error no realizar la actualización.

```

CREATE PROCEDURE sp_actualiza_temple2_SIN_NO_REALIZACION
@numemp int,
@numde int,
@extel int,
@fecna date,
@fecin date,
@salar int,
@comis int,

```

```

@numhi int,
@nomem VARCHAR(30)
AS
begin
begin try
begin tran
IF EXISTS (SELECT * FROM temple2
WHERE (numemp=@numemp and NUMDE=@numde AND EXTEL=@extel AND
FECNA=@fecna AND FECIN=@fecin AND SALAR=@salar AND COMIS=@comis AND
NUMHI=@numhi AND NOMEM=@nomem ))
BEGIN
PRINT 'REGISTRO NO MODIFICADO'
END
--END
IF EXISTS (SELECT * FROM temple2
WHERE ((@numde<>NUMDE OR @extel<>EXTEL OR @fecna<>FECNA OR
@FECIN<>FECIN OR @salar<>SALAR OR @comis<>COMIS OR @numhi<>NUMHI OR
@nomem<>NOMEM ) and numemp=@numemp))
--PARA MODIFICAR REGISTROS
BEGIN
UPDATE temple2
SET TEMPLE2.NUMEMP=@numemp, TEMPLE2.NUMDE=@numde,
TEMPLE2.EXTEL=@extel, TEMPLE2.FECNA=@fecna, TEMPLE2.FECIN=@fecin,
TEMPLE2.SALAR=@salar, TEMPLE2.COMIS=@comis, TEMPLE2.NUMHI=@numhi,
TEMPLE2.NOMEM=@nomem
WHERE TEMPLE2.NUMEMP= @numemp
PRINT 'REGISTRO MODIFICADO'
End

commit tran

end try
begin catch
rollback tran
print 'Se ha producido error'
end catch
END

```

2.5. Mostrar los datos de los empleados cuyos nombres de empleados empiecen por una letra dada.

```

create proc SP_LISTA_INICIAL_NOMBRE
@LETRA VARCHAR=NULL
as begin
if @LETRA is null
print 'No ha introducido ninguna inicial de nombre'
else
if (select COUNT(*) from temple2 where nomem like '%,
'+@LETRA+'%')=0
print 'No hay ningún nombre que empiece por esa letra'
else
select * from temple2 where nomem like '%, '+@LETRA+'%'
end

```

2.4. Repetir el anterior pero comprobando después de la modificación del usuario, que si se queda lo mismo que está en la tabla, no se produzca dicha modificación (usar triggers y

transacciones).

2.5. Mostrar los datos de los empleados cuyos nombres de empleados empiecen por una letra dada.

```
--2.5.  Mostrar los datos de los empleados cuyos nombres de empleados empiecen por una letra dada

if OBJECT_ID('nombreemp') is not null
begin
    drop procedure nombreemp
end
go

create procedure nombreemp
@nomemp char(1)
as
begin
    (SELECT SUBSTRING(NOMEM,CHARINDEX(',',NOMEM)+1,LEN(NOMEM)) FROM TEMPLE WHERE NOMEM LIKE @nomemp+'%')
end

EXEC nombreemp 'A'
```

100 % <

Resultados Mensajes

	(Sin nombre de colum...
1	AUREO
2	ADRIANA

2.6. Repetir el anterior, pero con el apellido.

```
--2.6.  Repetir el anterior, pero con el apellido.

if OBJECT_ID('apellidoemp') is not null
begin
    drop procedure apellidoemp
end
go

create procedure apellidoemp
@apelemp char(1)
as
begin
    (SELECT SUBSTRING(NOMEM,1,CHARINDEX(',',NOMEM)-1) FROM TEMPLE WHERE NOMEM LIKE @apelemp+'%')
end

EXEC apellidoemp 'C'
```

100 % <

Resultados Mensajes

	(Sin nombre de colum...
1	CAMPS
2	CAMPOS

2.7. Mostrar los datos de los empleados donde nomce se adapten a un patrón de búsqueda.

```
--2.7. Mostrar los datos de los empleados donde nomce se adapten a un patrón de búsqueda.

if OBJECT_ID('datosemp') is not null
begin
    drop procedure datosemp
end
create Procedure datosemp
@datos varchar(40)
as
begin
select temple2.numemp, temple2.numde, temple2.nomem, temple2.fecna
from temple2 inner join tdepto2 on
temple2.numde=tdepto2.numde inner join tcentr2 on
tdepto2.numce=tcentr2.numce where tcentr2.nomce like '%'+@datos+'%'
end

Exec datosemp 'sede'
```

100 %

Resultados Mensajes

	numemp	numde	nomem	fecna
1	110	121	Pons, Cesar	1929-11-10
2	150	121	PEREZ, JULIO	1930-08-10
3	190	121	VEIGA, JULIANA	1932-05-12
4	210	100	GALVEZ, PILAR	1940-09-28
5	250	100	ALBA, ADRIANA	1946-10-27
6	260	100	LOPEZ, ANTONIO	1943-12-03
7	280	130	ELOR, DOROTEA	1948-01-11

2.8. Mostrar el número de empleado, el nombre y la fecha de nacimiento, mostrando el mes de nacimiento con letras.

```
--2.8. Mostrar el número de empleado, el nombre y la fecha de nacimiento,
--mostrando el mes de nacimiento con letras

if OBJECT_ID('fechletra') is not null
begin
    drop procedure fechletra
end
create procedure fechletra
@numemp int
as
begin
SELECT numemp, nomem, CONCAT(datepart(day, fecna), ' de ', datename(month, fecna), ' de ', datepart(year, fecna)) as fecha
FROM Temple2 where numemp=@numemp
end

EXEC fechletra 110
```

100 %

Resultados Mensajes

	numemp	nomem	fecha
1	110	Pons, Cesar	10 de Noviembre de 1929

2.9. Repetir el ejercicio anterior, añadiendo el nombre del departamento donde trabaja y el nombre del centro y la dirección.

```
--2.9. Repetir el ejercicio anterior, añadiendo el nombre del
-- departamento donde trabaja y el nombre del centro y la dirección.
if OBJECT_ID('fechletra2') is not null
begin
drop procedure fechletra2
end
create procedure fechletra2
@numemp int
as
begin
SELECT temple2.numemp, temple2.numde, temple2.nomem, CONCAT(datepart(day, temple2.fecna), ' de ',
datename(month, temple2.fecna), ' de ', datepart(year, temple2.fecna)) 'Fecha de nac.', tdepto2.numce,
tcentr2.nomce FROM Temple2 inner join Tdepto2 on Temple2.numde=Tdepto2.numde inner join Tcentr2 on
Tdepto2.numce=Tcentr2.numce where Temple2.numemp = @numemp
end

EXEC fechletra2 110
```

	numemp	numde	nomem	Fecha de nac.	num...	nomce
1	110	121	Pons, Cesar	10 de Noviembre de 1929	10	Sede Central

2.10. Obtener los nombres de los empleados que tengan más edad que la que se le pasa como parámetro.

```
--2.10. Obtener los nombres de los empleados que tengan más edad que la que se le pasa como parámetro.
if OBJECT_ID('edademple') is not null
begin
drop procedure edademple
end
create procedure edademple
@edad int
as
begin
SELECT nomem, datediff(year, fecna, getdate()) as edad from temple2 where
@edad < (SELECT datediff(year, fecna, getdate()))
end

EXEC edademple 60
```

	nomem	edad
1	Pons, Cesar	88
2	LASA, MARIO	82
3	TEROL, LUCIANO	72
4	PEREZ, JULIO	87
5	AGUIRRE, AUREO	78
6	PEREZ, MARCOS	83
7	VEIGA, JULIANA	85

2.11. Obtener los nombres de los empleados que lleven en la empresa menos años que los dado como parámetro

```
--2.11. Obtener los nombres de los empleados que lleven en la empresa menos años que los dado como parámetro
if OBJECT_ID ('fechempez') is not null
begin
    drop procedure fechempez
end
create procedure fechempez
@añs int
as
begin
SELECT nomem, datediff(year,fecin,getdate())'Años en la empresa' from temple2 where
@años > (SELECT datediff(year,fecin,getdate()))
end

EXEC fechempez 30
```

100 %

Resultados Mensajes

	nomem	Años en la empr...
1	MUÑOZ, AZUCENA	29
2	FIERRO, CLAUDIA	29
3	MORA, VALERIANA	29
4	TORRES, HORACIO	29
5	SANTOS, SANCHO	29

2.12. Mostrar el número de empleado, nombre, edad y fecha de ingreso de aquellos empleados que tengan más edad que la que se pasa como parámetro y lleven en la empresa más de los años pasados también como parámetro.

```
--2.12. Mostrar el número de empleado, nombre, edad y fecha de ingreso de aquellos empleados
--que tengan más edad que la que se pasa como parámetro y lleven en la empresa más de los años
--pasados también como parámetro.
if OBJECT_ID ('fechtot') is not null
begin
    drop procedure fechtot
end
create procedure fechtot
@edad int,
@años int
as
begin
SELECT numemp,nomem, datediff(year,fecna,getdate())'Edad', datediff(year,fecin,getdate())'Años en la empresa'
from Temple2 where @edad < (SELECT datediff(year,fecna,getdate())) and @años < (SELECT datediff(year,fecin,getdate()))
end

EXEC fechtot 60,30
```

100 %

Resultados Mensajes

	numemp	nomem	Ed...	Años en la empr...
1	110	Pons, Cesar	88	67
2	120	LASA, MARIO	82	49
3	130	TEROL, LUCIANO	72	48
4	150	PEREZ, JULIO	87	69
5	160	AGUIRRE, AUREO	78	49
6	180	PEREZ MARCOS	83	61

- 2.13. Se va a eliminar un centro, pasándole el código del centro. Borrar todos los empleados del centro, todos los departamentos y por último el centro. Si hay algún error en algunas de las operaciones de borrado, el contenido de las tablas debe quedar como estaba al principio (es decir, que no se realiza la operación).

```
--2.13. Se va a eliminar un centro, pasándole el código del centro. Borrar todos los empleados del centro,  
-- todos los departamentos y por último el centro.  
if OBJECT_ID ('borcentro') is not null  
begin  
    drop procedure borcentro  
end  
create procedure borcentro  
@codcentro int  
as  
begin  
    delete from Temple2 where Temple2.numde in  
    (SELECT Tdepto2.numde from Tdepto2 where Tdepto2.numce = @codcentro)  
    delete from Tdepto2 where Tdepto2.numce = @codcentro  
    delete from Tcentr2 where Tcentr2.numce = @codcentro  
end  
Exec borcentro 10
```

100 % <

Mensajes

(16 filas afectadas)

(5 filas afectadas)

(1 filas afectadas)

- 2.14. Borrar un empleado pasándole el número de empleado.

```
--2.14. Borrar un empleado pasándole el número de empleado.  
if OBJECT_ID ('boremp') is not null  
begin  
    drop procedure boremp  
end  
create procedure boremp  
@emple int  
as  
begin  
    delete from Temple2 where numemp = @emple  
end  
exec boremp 120
```

100 % <

Mensajes

(1 filas afectadas)

2.15. Insertar un registro en la tabla tdepto.

```
--2.15. Insertar un registro en la tabla tdepto.  
if OBJECT_ID ('insertdepto') is not null  
begin  
    drop procedure insertdepto  
end  
create procedure insertdepto  
@numde int, @numce int,  
@direc int, @tidir char(1),  
@presu int, @depde int,  
@nomde varchar(50)  
as  
begin  
insert into Tdepto2(numde,numce,direc,tidir,presu,depde,nomde)  
values (@numde,@numce,@direc,@tidir,@presu,@depde,@nomde)  
end  
  
Exec insertdepto 115,20,180,'F',12,100,'dpto. informatico'  
select * from tdepto2
```

100 % <

Resultados Mensajes

	numde	num...	dir...	tidir	pre...	dep...	nomde
1	110	20	180	P	15	100	dirección comercial
2	111	20	180	F	11	110	sector industrial
3	112	20	270	P	9	110	sector servicios
4	115	20	180	F	12	100	dpto. informatico

- 2.16. Insertar un registro en una tabla, pedir dni, antes de escribir comprobar la letra, llamando a la función creada anteriormente. Si es correcta, escribir registro, sino no escribir el registro y mostrar un mensaje de error.

```
create function dbo.dnicorrecto(@dni varchar(9))
returns varchar(1)
as
begin
declare @letra varchar(1)
declare @numeros varchar(8)
SET @numeros = SUBSTRING(@dni,1,8)
SET @letra = SUBSTRING('TRWAGMYFPDXBNJZSQVHLCKE', @numeros % 23 + 1,1)
if (@letra=SUBSTRING(@dni,9,1))
set @letra='S'
else
set @letra='N'
return @letra
end
print dbo.dnicorrecto('42213469')
```

SQLQuery1.sql - DE...AVJ4D5\danic (51))*

```
create procedure insertemp
@NUMEMP INT,
@NUMDE INT,
@EXTel SMALLINT,
@FECNA DATE,
@FECIN DATE,
@SALAR FLOAT,
@COMIS FLOAT,
@NUMHI SMALLINT,
@NOMEM VARCHAR(20),
@DNI VARCHAR(9)
as
begin
begin tran
declare @error int
if (dbo.FN_COMPROBAR_DNI(@DNI)=1)
begin
print '¡El DNI es correcto!'
insert into temple2 (numemp, numde, extel, fecna, fecin, salar, comis, numhi, nomem)
values (@numemp, @numde, @extel, @fecna, @fecin, @salar, @comis, @numhi, @nomem)
print 'Se ha insertado el empleado'
end
else
begin
print 'El DNI introducido no es correcto'
print 'No se ha guardado el usuario en la base de datos'
end
set @error = @@error
if (@error=-1)
begin
rollback tran
return (@error)
print 'Ha sucedido un error y se han deshecho los cambios'
end
commit tran
```

EXEC dbo.insertemp 150, 121, 340, '1984-10-28', '2006-09-16', 550, 300, 1, 'CABRERA, DANIEL', '42213469M'

- 2.17. Insertar un registro en una tabla de usuarios y cuentas corrientes. Comprobar antes de escribir que el dc que indica el usuario se corresponde con el real. En caso contrario, no escribir el registro y mostrar un error.