WITHOUT PARAMETER

create procedure stpr\_AllEmpRec

as

begin

select \* from EmployeeDetails;

end

exec stpr\_AllEmpRec

WITH PARAMETER

create procedure stpr\_SpecificEmpRec

@eid int

as

begin

select \* from EmployeeDetails where Id = @eid

end

exec stpr\_SpecificEmpRec 333

WITH OUTPUT PARAMETER

create procedure stpr\_SpecificEmpRec1

@eid int,

@ENAME VARCHAR(50) OUTPUT,

@SAL FLOAT OUTPUT

as

begin

select @ENAME=EmpName,@SAL = Salary

from EmployeeDetails where Id = @eid

end

BEGIN

DECLARE @ENAME VARCHAR(50)

DECLARE @SAL FLOAT

EXEC stpr\_SpecificEmpRec1 999,@ENAME OUTPUT,@SAL OUTPUT

PRINT 'NAME = ' + @ENAME + ' HAS SAL = ' + CAST(@SAL AS VARCHAR(50))

END

PROCEDURE ON INSERT

CREATE procedure stpr\_InsertEmpRec

@ID INT,

@ENAME VARCHAR(50),

@SAL FLOAT,

@MGRID INT,

@DEPTID INT

as

begin

INSERT INTO EmployeeDetails(Id,EmpName,Salary,MgrId,DeptId)

VALUES(@ID,@ENAME,@SAL,@MGRID,@DEPTID)

end

EXEC stpr\_InsertEmpRec @ID=777,@ENAME='NAGESH',@SAL=NULL,@MGRID=222,@DEPTID=22

EXEC stpr\_AllEmpRec

PROCEDURE ON UPDATE

CREATE procedure stpr\_UPDATEEmpRec

@ID INT,

@SAL FLOAT,

@MGRID INT

as

begin

UPDATE EmployeeDetails

SET Salary = @SAL, MgrId=@MGRID

WHERE ID = @ID

end

EXEC stpr\_UPDATEEmpRec @ID=111,@SAL=12000,@MGRID=222

EXEC stpr\_AllEmpRec

PROCEDURE ON DELETE

CREATE procedure stpr\_DELETEEmpRec

@ID INT

as

begin

DELETE FROM EmployeeDetails WHERE ID = @ID

end

EXEC stpr\_DELETEEmpRec 444

EXEC stpr\_AllEmpRec

HIGHEST SALARY BETWEEN 2 IDS WITH IF

CREATE PROCEDURE STPR\_CMP

@ID1 INT,

@ID2 INT

AS

BEGIN

DECLARE @SAL1 FLOAT

DECLARE @SAL2 FLOAT

DECLARE @NAME1 VARCHAR(50)

DECLARE @NAME2 VARCHAR(50)

SELECT @SAL1 = SALARY,@NAME1=EmpName FROM EmployeeDetails WHERE ID = @ID1

SELECT @SAL2 = SALARY,@NAME2=EmpName FROM EmployeeDetails WHERE ID = @ID2

IF @SAL1>@SAL2

PRINT @NAME1 +'  HIGHER SAL = ' + CAST(@SAL1 AS VARCHAR(50))

ELSE

PRINT @NAME2 +'   HIGHER SAL = ' + CAST(@SAL2 AS VARCHAR(50))

END

EXEC STPR\_CMP 222,666

WHILE LOOP INTRO

BEGIN

DECLARE @CNT INT

SET @CNT=1

WHILE @CNT<=10

BEGIN

PRINT @CNT

SET @CNT = @CNT+1

END

END

CURSOR

CREATE PROCEDURE MYCURSORFUN

AS

BEGIN

DECLARE @ID INT;

DECLARE @ENAME VARCHAR(50);

--DECLARE A CURSOR AND THE QUERY ON WHICH THE CURSOR IS RUNNING

DECLARE EMPCUR CURSOR FOR

SELECT ID,EmpName FROM EmployeeDetails WHERE SALARY>8000;

OPEN EMPCUR;

FETCH NEXT FROM EMPCUR INTO @ID,@ENAME;

--@@FETCH\_STATUS = 0 MEANS THE STAUS WAS SUCCESFUL IN FETCHING THE RECORD

-- -1 => FAIL -2 => MISSING FETCHED ROW -9=>CURSOR ISNT PERFORMING OPERATION

WHILE @@FETCH\_STATUS=0

BEGIN

PRINT CAST(@ID AS VARCHAR(50)) + ' = ' + @ENAME

FETCH NEXT FROM EMPCUR INTO @ID,@ENAME;

END

CLOSE EMPCUR;

DEALLOCATE EMPCUR;

END

EXEC MYCURSORFUN