ASSIGNMENT 9

1. Create a stored procedure with input parameter Entity and Entity Id which gives information based on entity passed.
   For example: When entity user is passed, Sp should return all information of user.
   Note: Entities can be User or Customer.

   Ans:

```
CREATE OR ALTER PROCEDURE dbo.[SpEntitySel](

@Entity NVARCHAR(MAX),
@EntityId NVARCHAR(MAX)

)
AS
BEGIN
SET NOCOUNT ON;

DECLARE @SqlQuery NVARCHAR(MAX)
SELECT @SqlQuery = 'SELECT * FROM dbo.[' +@Entity+ '] et WHERE et.'+@Entity+'Id ='
+@EntityId+';'

PRINT (@SqlQuery);
EXEC (@SqlQuery);

END;
```

```
DECLARE @Entity VARCHAR(100) = 'User',
        @EntityId VARCHAR = 1
EXEC dbo.[SpEntitySel] @Entity, @EntityId;
```

2. Write a query to show the address data in following format. (Refer previous assignment tables)

	PersonAddressId			
FirstName	101	102	103	104
Aaron	0	1	0	0
Melisa	1	0	0	0
John	0	0	1	0
Jane	0	0	0	1

Ans:

```
-- Create the temporary table
CREATE TABLE #AddressData
(
   PersonAddressId INT IDENTITY(101, 1),
   FirstName VARCHAR(100),
   [101] INT,
   [102] INT,
   [103] INT,
   [104] INT
);

-- Insert data into the temporary table
INSERT INTO #AddressData (FirstName, [101], [102], [103], [104])
```

```
VALUES
   ('Aaron', 0, 1, 0, 0),
   ('Melisa', 1, 0, 0, 0),
   ('John', 0, 0, 1, 0),
   ('Jane', 0, 0, 0, 1);

select * from #AddressData

SELECT

   FirstName,

   [101], [102], [103], [104]
FROM
(
   SELECT
      FirstName,
      PersonAddressId

   FROM
      #AddressData
) AS src
PIVOT
(
   COUNT(PersonAddressId)
   FOR PersonAddressId IN ([101], [102], [103], [104])
) AS pvt;
```

[MAKING DYNAMIC]

```
CREATE OR ALTER PROCEDURE dbo.[SpPivotSel]

@PivotColumn VARCHAR(100),
@PivotColList VARCHAR(200)

AS
BEGIN
DECLARE @Query NVARCHAR(MAX)
SELECT @Query = N'SELECT

    FirstName,

  ' +@PivotColList+ '
FROM
(
  SELECT
    FirstName,
    ' +@PivotColumn+ '

  FROM
    #AddressData
) AS src
PIVOT
(
  COUNT(PersonAddressId)
```

```
    FOR '+@PivotColumn+' IN ('+@PivotColList+')
) AS pvt'

EXEC (@Query)

END;


Exec SpPivotSel @PivotColumn = 'PersonAddressId', @PivotColList = '[101], [102], [103], [104]'
```

3. Create a dynamic SP to return data based on the entity provided and dataRequest made.

   {

   “entity”:”customer/user/hotel/branch/room”

   entityId:”1,2” / “” / “All” / Null,

   “dataReq":”address,contact”/ “all”/ “”

   }

   In case if entityId is null or All or ‘’, you need to return all data for that entity.

   Ans:

```
USE Swastika_DB;
GO

CREATE OR ALTER PROCEDURE dbo.SpEntitySel
(
   @Json VARCHAR(MAX),
```

```
    @SqlQuery VARCHAR(MAX) OUTPUT,
    @DataReq VARCHAR(100) OUTPUT,
    @Entity VARCHAR(100) OUTPUT,
    @EntityId VARCHAR(100) OUTPUT
)
AS
BEGIN
    SET NOCOUNT ON;

    SELECT @DataReq = oj.dataReq,
           @Entity = oj.entity,
           @EntityId = oj.entityId
    FROM OPENJSON(@Json)
    WITH (
        dataReq VARCHAR(100),
        entity VARCHAR(100),
        entityId VARCHAR(100)
    ) AS oj;

    SELECT @DataReq, @EntityId, @Entity;

    SELECT @SqlQuery = 'SELECT * FROM dbo.' + @Entity + ' et '
                     + CASE
                         WHEN @Entity = 'Branch' THEN ' INNER JOIN dbo.Hotel hh ON hh.HotelId
= et.HotelId'
                         WHEN @Entity = 'Room' THEN ' INNER JOIN dbo.RoomType hh ON
hh.RoomTypeId = et.RoomTypeId INNER JOIN dbo.Branch bb ON et.BranchId = bb.BranchId
INNER JOIN dbo.Hotel ht ON ht.HotelId = bb.HotelId'
                         WHEN @Entity IN ('Customer', 'User') THEN ' INNER JOIN dbo.Person pp
ON et.PersonId = pp.PersonId'
                         ELSE ''
```

```
                END;

    IF (@Entity <> 'Room')
    BEGIN
        IF (@DataReq = 'ALL' OR @DataReq = '' OR @DataReq LIKE '%address%')
        BEGIN
            SELECT @SqlQuery = @SqlQuery + ' INNER JOIN dbo.' + CASE
                                                    WHEN @Entity IN ('Customer', 'User') THEN
'PersonAddress ba ON ba.PersonId = et.PersonId'
                                                    ELSE @Entity
                                                END + 'Address ba ON ba.' + @Entity + 'Id = et.' +
@Entity + 'Id'
                        + ' INNER JOIN dbo.Address ad ON ad.AddressId = ba.AddressId';
        END;

        IF (@DataReq = 'ALL' OR @DataReq = '' OR @DataReq LIKE '%contact%')
        BEGIN
            SELECT @SqlQuery = @SqlQuery + ' INNER JOIN dbo.' + CASE
                                                    WHEN @Entity IN ('Customer', 'User') THEN
'PersonContact ba ON ba.PersonId = et.PersonId'
                                                    ELSE @Entity
                                                END + 'Contact ba ON ba.' + @Entity + 'Id = et.' +
@Entity + 'Id'
                        + ' INNER JOIN dbo.Contact ac ON ac.ContactId = ba.ContactId';
        END;
    END;

    IF (@EntityId IS NULL OR @EntityId = 'ALL' OR @EntityId = ''  OR @EntityId = '1,2')
    BEGIN
        SELECT @SqlQuery = @SqlQuery + ' WHERE et.[' + @Entity + ']Id IN (' + @EntityId
+ ')';
```

```
    END;

    PRINT @SqlQuery;
END;
GO


DECLARE @SqlQuery VARCHAR(MAX);
DECLARE @DataReq VARCHAR(100);
DECLARE @Entity VARCHAR(100);
DECLARE @EntityId VARCHAR(100);

EXEC dbo.SpEntitySel
    @Json = '{
        "dataReq": "contact",
        "entity": "customer",
        "entityId": "1,2"
    }',
    @SqlQuery = @SqlQuery OUTPUT,
    @DataReq = @DataReq OUTPUT,
    @Entity = @Entity OUTPUT,
    @EntityId = @EntityId OUTPUT;
```