

Assignment - Week VII

Domain: SQL

Name: Harsh Bhasin

Student ID: CT_CSI_SQ_3533

Contact No :9098835618

Email ID :Harshbha30@gmail.com

TYPE 0: In this type of SCD no Changes are allowed

Query:

```
CREATE PROCEDURE SP_SCD_TYPE_0_Customer
```

```
    @CustomerID INT,
```

```
    @CustomerName NVARCHAR(100),
```

```
    @Email NVARCHAR(100),
```

```
    @Phone NVARCHAR(20)
```

```
AS
```

```
BEGIN
```

```
    SET NOCOUNT ON;
```

```
    IF EXISTS (SELECT 1 FROM DimCustomer WHERE CustomerID = @CustomerID)
```

```
    BEGIN
```

```
        RAISERROR('SCD Type 0: Changes not allowed for existing customer records.', 16, 1);
```

```
        RETURN;
```

```
    END
```

```
    ELSE
```

```
    BEGIN
```

```
        INSERT INTO DimCustomer (CustomerID, CustomerName, Email, Phone, CreatedDate)
```

```
        VALUES (@CustomerID, @CustomerName, @Email, @Phone, GETDATE());
```

```
        PRINT 'New customer record inserted successfully.';
```

```
    END
```

```
END;
```

```
GO
```

TYPE 1: Overwrites existing data:

```
CREATE PROCEDURE SP_SCD_TYPE_1_Customer
```

```
    @CustomerID INT,
```

```
    @CustomerName NVARCHAR(100),
```

```
    @Email NVARCHAR(100),
```

```
    @Phone NVARCHAR(20)
```

```
AS
```

```
BEGIN
```

```
    SET NOCOUNT ON;
```

```
    IF EXISTS (SELECT 1 FROM DimCustomer WHERE CustomerID = @CustomerID)
```

```
    BEGIN
```

```
        UPDATE DimCustomer
```

```
        SET CustomerName = @CustomerName,
```

```
            Email = @Email,
```

```
            Phone = @Phone,
```

```
            ModifiedDate = GETDATE()
```

```
        WHERE CustomerID = @CustomerID;
```

```
        PRINT 'Customer record updated successfully (SCD Type 1).';
```

```
    END
```

```
    ELSE
```

```
    BEGIN
```

```
        INSERT INTO DimCustomer (CustomerID, CustomerName, Email, Phone, CreatedDate)
```

```
        VALUES (@CustomerID, @CustomerName, @Email, @Phone, GETDATE());
```

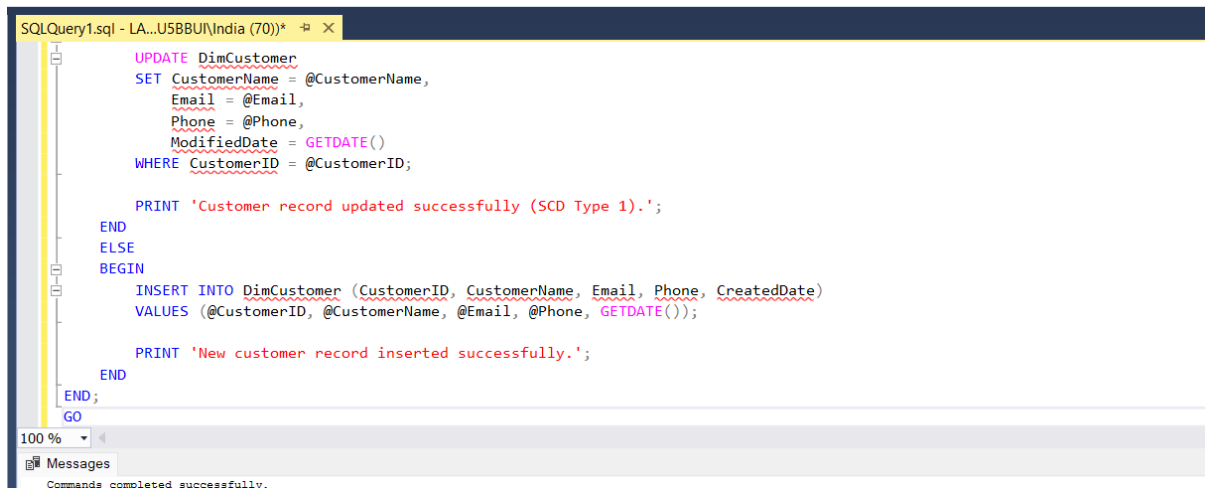
```
        PRINT 'New customer record inserted successfully.';
```

```
    END
```

```
END;
```

```
GO
```

Output:

A screenshot of the SQL Server Enterprise Manager interface. The top pane shows a T-SQL query in a window titled 'SQLQuery1.sql - LA...U5BBUI\India (70)*'. The query is an UPDATE statement for the DimCustomer table, setting CustomerName, Email, Phone, and ModifiedDate (using GETDATE()) for a specific CustomerID. It includes PRINT statements for success messages and an ELSE clause with an INSERT statement for new records. The bottom pane shows the 'Messages' tab with the text 'Commands completed successfully.'

```
UPDATE DimCustomer
SET CustomerName = @CustomerName,
    Email = @Email,
    Phone = @Phone,
    ModifiedDate = GETDATE()
WHERE CustomerID = @CustomerID;

PRINT 'Customer record updated successfully (SCD Type 1).';
END
ELSE
BEGIN
    INSERT INTO DimCustomer (CustomerID, CustomerName, Email, Phone, CreatedDate)
    VALUES (@CustomerID, @CustomerName, @Email, @Phone, GETDATE());

    PRINT 'New customer record inserted successfully.';
END
END;
GO
```

100 %
Messages
Commands completed successfully.

TYPE 2: Add new record for changes, maintain history

Query:

```
CREATE PROCEDURE SP_SCD_TYPE_2_Customer
```

```
    @CustomerID INT,
```

```
    @CustomerName NVARCHAR(100),
```

```
    @Email NVARCHAR(100),
```

```
    @Phone NVARCHAR(20),
```

```
    @City NVARCHAR(50)
```

```
AS
```

```
BEGIN
```

```
    SET NOCOUNT ON;
```

```
    DECLARE @CurrentSurrogateKey INT;
```

```
    DECLARE @MaxVersion INT;
```

```
    SELECT @CurrentSurrogateKey = CustomerSurrogateKey
```

```
    FROM DimCustomer_SCD2
```

```
    WHERE CustomerID = @CustomerID AND IsCurrentRecord = 1;
```

```
    IF @CurrentSurrogateKey IS NOT NULL
```

```
    BEGIN
```

```
IF EXISTS (
    SELECT 1 FROM DimCustomer_SCD2
    WHERE CustomerSurrogateKey = @CurrentSurrogateKey
    AND CustomerName = @CustomerName
    AND Email = @Email
    AND Phone = @Phone
    AND City = @City
)
BEGIN
    PRINT 'No changes detected. Record not updated.';
    RETURN;
END

UPDATE DimCustomer_SCD2
SET EffectiveEndDate = GETDATE(),
    IsCurrentRecord = 0
WHERE CustomerSurrogateKey = @CurrentSurrogateKey;

SELECT @MaxVersion = ISNULL(MAX(VersionNumber), 0)
FROM DimCustomer_SCD2
WHERE CustomerID = @CustomerID;

INSERT INTO DimCustomer_SCD2 (
    CustomerID, CustomerName, Email, Phone, City,
    EffectiveStartDate, EffectiveEndDate, IsCurrentRecord, VersionNumber
)
VALUES (
    @CustomerID, @CustomerName, @Email, @Phone, @City,
    GETDATE(), '9999-12-31', 1, @MaxVersion + 1
);
```

```

PRINT 'New version of customer record created (SCD Type 2).';

END

ELSE

BEGIN

INSERT INTO DimCustomer_SCD2 (

    CustomerID, CustomerName, Email, Phone, City,

    EffectiveStartDate, EffectiveEndDate, IsCurrentRecord, VersionNumber

)

VALUES (

    @CustomerID, @CustomerName, @Email, @Phone, @City,

    GETDATE(), '9999-12-31', 1, 1

);

PRINT 'New customer record inserted (SCD Type 2).';

END

END;

GO

```

Ouput:

The screenshot displays the SQL Server Enterprise Manager interface. The top pane shows the execution of a stored procedure named `SP_SCD_TYPE_2_Customer`. The bottom pane shows the output messages, indicating that the commands were completed successfully.

```

SQLQuery1.sql - LA...U5BBUI\India (70))*
CREATE PROCEDURE SP_SCD_TYPE_2_Customer
    @CustomerID INT,
    @CustomerName NVARCHAR(100),
    @Email NVARCHAR(100),
    @Phone NVARCHAR(20),
    @City NVARCHAR(50)
AS
BEGIN
    SET NOCOUNT ON;
    DECLARE @CurrentSurrogateKey INT;
    DECLARE @MaxVersion INT;

    SELECT @CurrentSurrogateKey = CustomerSurrogateKey
    FROM DimCustomer_SCD2
    WHERE CustomerID = @CustomerID AND IsCurrentRecord = 1;

    IF @CurrentSurrogateKey IS NOT NULL
    BEGIN

```

100 %
 Messages
 Commands completed successfully.

TYPE 3: *Store previous values in additional columns*

Query:

```
CREATE PROCEDURE SP_SCD_TYPE_3_Customer
```

```
    @CustomerID INT,
```

```
    @CustomerName NVARCHAR(100),
```

```
    @Email NVARCHAR(100),
```

```
    @Phone NVARCHAR(20),
```

```
    @City NVARCHAR(50)
```

```
AS
```

```
BEGIN
```

```
    SET NOCOUNT ON;
```

```
    IF EXISTS (SELECT 1 FROM DimCustomer_SCD3 WHERE CustomerID = @CustomerID)
```

```
    BEGIN
```

```
        UPDATE DimCustomer_SCD3
```

```
        SET PreviousCity = CurrentCity,
```

```
            PreviousCityDate = CurrentCityDate,
```

```
            CurrentCity = @City,
```

```
            CurrentCityDate = GETDATE(),
```

```
            CustomerName = @CustomerName,
```

```
            Email = @Email,
```

```
            Phone = @Phone,
```

```
            ModifiedDate = GETDATE()
```

```
        WHERE CustomerID = @CustomerID
```

```
        AND (CurrentCity <> @City OR CurrentCity IS NULL);
```

```
    UPDATE DimCustomer_SCD3
```

```
    SET CustomerName = @CustomerName,
```

```
        Email = @Email,
```

```
        Phone = @Phone,
```

```
        ModifiedDate = GETDATE()
```

```
WHERE CustomerID = @CustomerID

AND (CurrentCity = @City OR (@City IS NULL AND CurrentCity IS NULL));

PRINT 'Customer record updated with previous city history (SCD Type 3).';

END

ELSE

BEGIN

    INSERT INTO DimCustomer_SCD3 (

        CustomerID, CustomerName, Email, Phone,

        CurrentCity, CurrentCityDate, CreatedDate

    )

    VALUES (

        @CustomerID, @CustomerName, @Email, @Phone,

        @City, GETDATE(), GETDATE()

    );

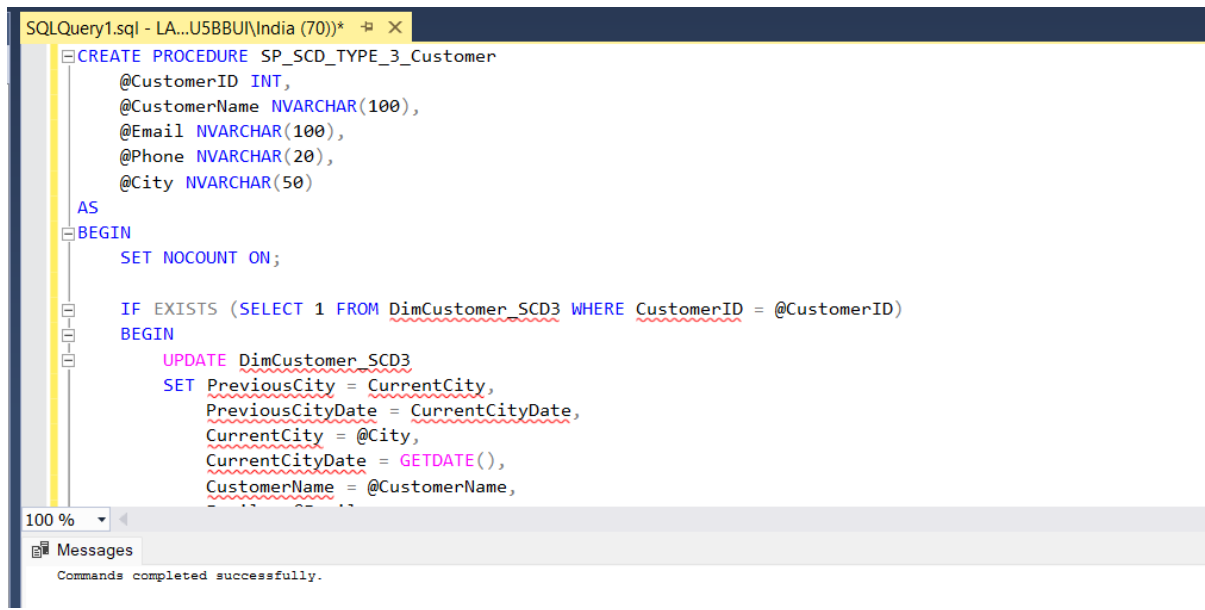
    PRINT 'New customer record inserted (SCD Type 3).';

END

END;

GO
```


Output:



The screenshot shows a SQL Server Enterprise Manager window titled 'SQLQuery1.sql - LA...U5BBUI\India (70))' with a standard toolbar. The main pane displays a T-SQL script for a stored procedure named 'SP_SCD_TYPE_3_Customer'. The script includes parameters for CustomerID, CustomerName, Email, Phone, and City, followed by an AS keyword, a BEGIN statement, and a SET NOCOUNT ON; statement. It then checks for the existence of a record in 'DimCustomer_SCD3' and updates it with new city and date information. The script ends with a 100% zoom level and a Messages pane at the bottom showing 'Commands completed successfully.'

```
SQLQuery1.sql - LA...U5BBUI\India (70)) *
CREATE PROCEDURE SP_SCD_TYPE_3_Customer
    @CustomerID INT,
    @CustomerName NVARCHAR(100),
    @Email NVARCHAR(100),
    @Phone NVARCHAR(20),
    @City NVARCHAR(50)
AS
BEGIN
    SET NOCOUNT ON;

    IF EXISTS (SELECT 1 FROM DimCustomer_SCD3 WHERE CustomerID = @CustomerID)
    BEGIN
        UPDATE DimCustomer_SCD3
        SET PreviousCity = CurrentCity,
            PreviousCityDate = CurrentCityDate,
            CurrentCity = @City,
            CurrentCityDate = GETDATE(),
            CustomerName = @CustomerName,
```

100 %
Messages
Commands completed successfully.

TYPE 4: Separate current and history tables

Query:

```
CREATE PROCEDURE SP_SCD_TYPE_4_Customer

    @CustomerID INT,

    @CustomerName NVARCHAR(100),

    @Email NVARCHAR(100),

    @Phone NVARCHAR(20),

    @City NVARCHAR(50)

AS

BEGIN

    SET NOCOUNT ON;

    BEGIN TRANSACTION;

    BEGIN TRY

        IF EXISTS (SELECT 1 FROM DimCustomer_SCD4_Current WHERE CustomerID = @CustomerID)

        BEGIN

            INSERT INTO DimCustomer_SCD4_History (
```

```
        CustomerID, CustomerName, Email, Phone, City,
        EffectiveStartDate, EffectiveEndDate, CreatedDate
    )
SELECT
    CustomerID, CustomerName, Email, Phone, City,
    CreatedDate, GETDATE(), CreatedDate
FROM DimCustomer_SCD4_Current
WHERE CustomerID = @CustomerID;

UPDATE DimCustomer_SCD4_Current
SET CustomerName = @CustomerName,
    Email = @Email,
    Phone = @Phone,
    City = @City,
    CreatedDate = GETDATE()
WHERE CustomerID = @CustomerID;

PRINT 'Customer record updated and history preserved (SCD Type 4).';
END
ELSE
BEGIN
    INSERT INTO DimCustomer_SCD4_Current (
        CustomerID, CustomerName, Email, Phone, City, CreatedDate
    )
    VALUES (
        @CustomerID, @CustomerName, @Email, @Phone, @City, GETDATE()
    );

    PRINT 'New customer record inserted (SCD Type 4).';
END
```

```
COMMIT TRANSACTION;

END TRY

BEGIN CATCH

    ROLLBACK TRANSACTION;

    DECLARE @ErrorMessage NVARCHAR(4000) = ERROR_MESSAGE();

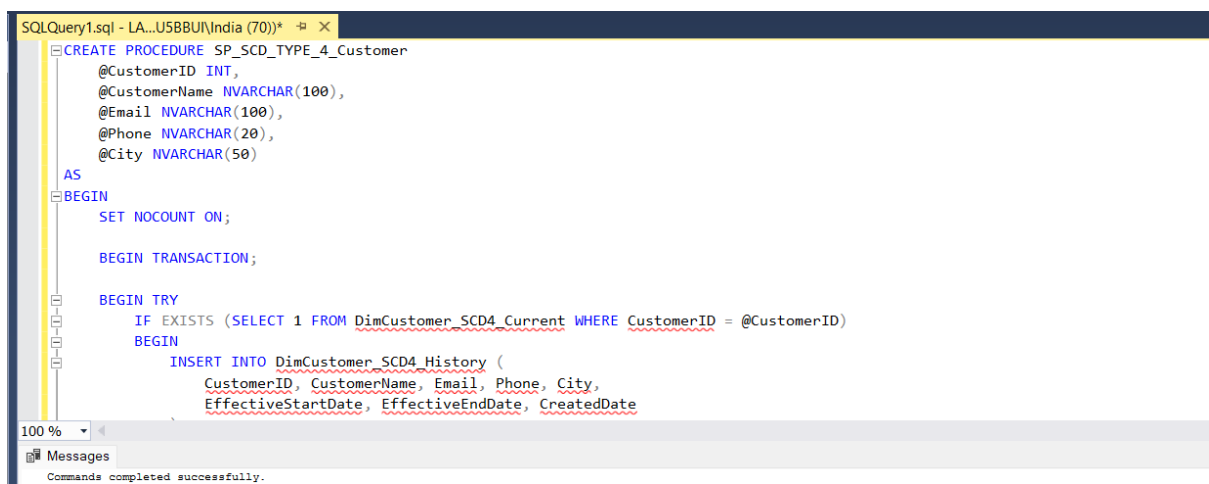
    RAISERROR(@ErrorMessage, 16, 1);

END CATCH

END;

GO
```

Output:



The screenshot displays the SQL Server Enterprise Manager interface. The top pane shows the SQL script for a stored procedure named `SP_SCD_TYPE_4_Customer`. The script includes parameters for `@CustomerID`, `@CustomerName`, `@Email`, `@Phone`, and `@City`. The logic involves a `BEGIN TRY` block with an `IF EXISTS` check on `DimCustomer_SCD4_Current` and an `INSERT` into `DimCustomer_SCD4_History`. The bottom pane shows the execution results, indicating that the commands completed successfully.

```
SQLQuery1.sql - LA...U5BBUI\India (70)*  X
CREATE PROCEDURE SP_SCD_TYPE_4_Customer
    @CustomerID INT,
    @CustomerName NVARCHAR(100),
    @Email NVARCHAR(100),
    @Phone NVARCHAR(20),
    @City NVARCHAR(50)
AS
BEGIN
    SET NOCOUNT ON;

    BEGIN TRANSACTION;

    BEGIN TRY
        IF EXISTS (SELECT 1 FROM DimCustomer_SCD4_Current WHERE CustomerID = @CustomerID)
        BEGIN
            INSERT INTO DimCustomer_SCD4_History (
                CustomerID, CustomerName, Email, Phone, City,
                EffectiveStartDate, EffectiveEndDate, CreatedDate
            )
        END TRY
    END TRY
END
```

100 %
Messages
Commands completed successfully.

TYPE 6: Hybrid approach combining Type 1, 2, and 3

Query:

```
CREATE PROCEDURE SP_SCD_TYPE_6_Customer_CollegeProject
    @CustomerID INT,
    @CustomerName NVARCHAR(100),
    @Email NVARCHAR(100),
    @Phone NVARCHAR(20),
    @City NVARCHAR(50),
    @CustomerSegment NVARCHAR(50)
AS
BEGIN
    SET NOCOUNT ON;
    DECLARE @CurrentSurrogateKey INT;
    DECLARE @MaxVersion INT;

    BEGIN TRANSACTION;

    BEGIN TRY

        SELECT @CurrentSurrogateKey = CustomerSurrogateKey
        FROM DimCustomer_SCD6
        WHERE CustomerID = @CustomerID AND IsCurrentRecord = 1;

        IF @CurrentSurrogateKey IS NOT NULL
        BEGIN

            IF EXISTS (
                SELECT 1 FROM DimCustomer_SCD6
```

```

WHERE CustomerSurrogateKey = @CurrentSurrogateKey

AND (City <> @City OR Email <> @Email)

)

BEGIN

UPDATE DimCustomer_SCD6

SET EffectiveEndDate = GETDATE(),

    IsCurrentRecord = 0

WHERE CustomerSurrogateKey = @CurrentSurrogateKey;


SELECT @MaxVersion = ISNULL(MAX(VersionNumber), 0)

FROM DimCustomer_SCD6

WHERE CustomerID = @CustomerID;


INSERT INTO DimCustomer_SCD6 (

    CustomerID, CustomerName, Email, Phone, City, CustomerSegment,

    EffectiveStartDate, EffectiveEndDate, IsCurrentRecord, VersionNumber,

    CurrentCustomerName, CurrentCustomerSegment

)

VALUES (

    @CustomerID, @CustomerName, @Email, @Phone, @City, @CustomerSegment,

    GETDATE(), '9999-12-31', 1, @MaxVersion + 1,

    @CustomerName, @CustomerSegment

);


PRINT 'created for Type 2 ';

END


UPDATE DimCustomer_SCD6

```

```

SET CurrentCustomerName = @CustomerName,

    CurrentCustomerSegment = @CustomerSegment,

    CustomerName = CASE WHEN IsCurrentRecord = 1 THEN @CustomerName ELSE
CustomerName END,

    Phone = CASE WHEN IsCurrentRecord = 1 THEN @Phone ELSE Phone END,

    ModifiedDate = GETDATE()

WHERE CustomerID = @CustomerID;


PRINT 'Type 1 attributes';

END

ELSE

BEGIN

    INSERT INTO DimCustomer_SCD6 (

        CustomerID, CustomerName, Email, Phone, City, CustomerSegment,

        EffectiveStartDate, EffectiveEndDate, IsCurrentRecord, VersionNumber,

        CurrentCustomerName, CurrentCustomerSegment, CreatedDate

    )

    VALUES (

        @CustomerID, @CustomerName, @Email, @Phone, @City, @CustomerSegment,

        GETDATE(), '9999-12-31', 1, 1,

        @CustomerName, @CustomerSegment, GETDATE()

    );


    PRINT 'New record ';

END


COMMIT TRANSACTION;

END TRY

BEGIN CATCH

    ROLLBACK TRANSACTION;

    DECLARE @ErrorMessage NVARCHAR(4000) = ERROR_MESSAGE();

```

```
        RAISERROR(@ErrorMessage, 16, 1);

    END CATCH

END;

GO
```

```
IF NOT EXISTS (SELECT * FROM sys.objects WHERE object_id = OBJECT_ID(N'DimCustomer') AND
type in (N'U'))
```

```
BEGIN
```

```
    CREATE TABLE DimCustomer (
        CustomerSurrogateKey INT IDENTITY(1,1) PRIMARY KEY,
        CustomerID INT NOT NULL,
        CustomerName NVARCHAR(100),
        Email NVARCHAR(100),
        Phone NVARCHAR(20),
        CreatedDate DATETIME2,
        ModifiedDate DATETIME2
    );
    PRINT 'DimCustomer table';
```

```
END;
```

```
GO
```

```
IF NOT EXISTS (SELECT * FROM sys.objects WHERE object_id = OBJECT_ID(N'DimCustomer_SCD2')
AND type in (N'U'))
```

```
BEGIN
```

```
    CREATE TABLE DimCustomer_SCD2 (
        CustomerSurrogateKey INT IDENTITY(1,1) PRIMARY KEY,
        CustomerID INT NOT NULL,
        CustomerName NVARCHAR(100),
        Email NVARCHAR(100),
        Phone NVARCHAR(20),
        City NVARCHAR(50),
        EffectiveStartDate DATETIME2,
```

```

        EffectiveEndDate DATETIME2,

        IsCurrentRecord BIT DEFAULT 1,

        VersionNumber INT,

        CreatedDate DATETIME2 DEFAULT GETDATE()

    );

    PRINT 'Type 2 table is ready';

END;

GO


IF NOT EXISTS (SELECT * FROM sys.objects WHERE object_id = OBJECT_ID(N'DimCustomer_SCD3')
AND type in (N'U'))

BEGIN

    CREATE TABLE DimCustomer_SCD3 (

        CustomerSurrogateKey INT IDENTITY(1,1) PRIMARY KEY,

        CustomerID INT NOT NULL,

        CustomerName NVARCHAR(100),

        Email NVARCHAR(100),

        Phone NVARCHAR(20),

        CurrentCity NVARCHAR(50),

        CurrentCityDate DATETIME2,

        PreviousCity NVARCHAR(50),

        PreviousCityDate DATETIME2,

        CreatedDate DATETIME2,

        ModifiedDate DATETIME2

    );

    PRINT 'Type 3 table created';

END;

GO


IF NOT EXISTS (SELECT * FROM sys.objects WHERE object_id =
OBJECT_ID(N'DimCustomer_SCD4_Current') AND type in (N'U'))

BEGIN

```



```

CREATE TABLE DimCustomer_SCD4_Current (
    CustomerSurrogateKey INT IDENTITY(1,1) PRIMARY KEY,
    CustomerID INT NOT NULL,
    CustomerName NVARCHAR(100),
    Email NVARCHAR(100),
    Phone NVARCHAR(20),
    City NVARCHAR(50),
    CreatedDate DATETIME2
);
PRINT 'Type 4 table';
END;
GO

```

-- SCD Type 4 history table for old data, totally correct!

```

IF NOT EXISTS (SELECT * FROM sys.objects WHERE object_id =
OBJECT_ID(N'DimCustomer_SCD4_History') AND type in (N'U'))
BEGIN
    CREATE TABLE DimCustomer_SCD4_History (
        HistoryID INT IDENTITY(1,1) PRIMARY KEY,
        CustomerID INT NOT NULL,
        CustomerName NVARCHAR(100),
        Email NVARCHAR(100),
        Phone NVARCHAR(20),
        City NVARCHAR(50),
        EffectiveStartDate DATETIME2,
        EffectiveEndDate DATETIME2,
        CreatedDate DATETIME2
    );
    PRINT ' Type 4 history table is created';
END;
GO

```

```
IF NOT EXISTS (SELECT * FROM sys.objects WHERE object_id = OBJECT_ID(N'DimCustomer_SCD6')
AND type in (N'U'))
```

```
BEGIN
```

```
CREATE TABLE DimCustomer_SCD6 (
    CustomerSurrogateKey INT IDENTITY(1,1) PRIMARY KEY,
    CustomerID INT NOT NULL,
    CustomerName NVARCHAR(100),
    Email NVARCHAR(100),
    Phone NVARCHAR(20),
    City NVARCHAR(50),
    CustomerSegment NVARCHAR(50),
    EffectiveStartDate DATETIME2,
    EffectiveEndDate DATETIME2,
    IsCurrentRecord BIT DEFAULT 1,
    VersionNumber INT,
    CurrentCustomerName NVARCHAR(100),
    CurrentCustomerSegment NVARCHAR(50),
    CreatedDate DATETIME2 DEFAULT GETDATE(),
    ModifiedDate DATETIME2
);
```

```
PRINT 'Type 6 table ready';
```

```
END;
```

```
GO
```

```
EXEC SP_SCD_TYPE_0_Customer 1, 'Rajesh Kumar', 'rajesh@email.com', '9876543210';
```

```
EXEC SP_SCD_TYPE_1_Customer 1, 'Rajesh Kumar', 'rajesh.kumar@email.com', '9876543210';
```

```
EXEC SP_SCD_TYPE_2_Customer 1, 'Rajesh Kumar', 'rajesh@email.com', '9876543210', 'Mumbai';
```

```
EXEC SP_SCD_TYPE_2_Customer 1, 'Rajesh Kumar', 'rajesh@email.com', '9876543210', 'Delhi';
```

```
EXEC SP_SCD_TYPE_3_Customer 1, 'Rajesh Kumar', 'rajesh@email.com', '9876543210', 'Bangalore';
```

```
EXEC SP_SCD_TYPE_4_Customer 1, 'Rajesh Kumar', 'rajesh@email.com', '9876543210', 'Chennai';
```

EXEC SP_SCD_TYPE_6_Customer 1, 'Rajesh Kumar', 'rajesh@email.com', '9876543210', 'Hyderabad',
'Premium';

EXEC SP_SCD_TYPE_6_Customer 2, 'Priya Sharma', 'priya.sharma@email.com', '9123456789', 'Pune',
'Standard';

EXEC SP_SCD_TYPE_6_Customer 3, 'Arjun Patel', 'arjun.patel@email.com', '9234567890',
'Ahmedabad', 'Premium';

EXEC SP_SCD_TYPE_6_Customer 4, 'Sneha Reddy', 'sneha.reddy@email.com', '9345678901',
'Visakhapatnam', 'Gold';

EXEC SP_SCD_TYPE_6_Customer 5, 'Vikram Singh', 'vikram.singh@email.com', '9456789012', 'Jaipur',
'Standard';

PRINT 'test cases are executed successfully';