

Celebal Assignment week-7

Let's assume a base dimension table named StudentDim

```
CREATE TABLE StudentDim (  
    StudentID INT PRIMARY KEY,  
    Name VARCHAR(100),  
    Subject VARCHAR(100),  
    StartDate DATETIME,  
    EndDate DATETIME,  
    IsCurrent BIT,  
    PreviousSubject VARCHAR(100) -- for SCD Type 3  
);
```

Also assume a staging table StudentStaging with the latest records

```
CREATE TABLE StudentStaging (  
    StudentID INT,  
    Name VARCHAR(100),  
    Subject VARCHAR(100)
```

);

SCD Type 0 – Retain Original

```
CREATE PROCEDURE SCD_Type_0
```

```
AS
```

```
BEGIN
```

```
    INSERT INTO StudentDim (StudentID, Name, Subject, StartDate,  
EndDate, IsCurrent)
```

```
    SELECT s.StudentID, s.Name, s.Subject, GETDATE(), NULL, 1
```

```
    FROM StudentStaging s
```

```
    WHERE NOT EXISTS (
```

```
        SELECT 1 FROM StudentDim d WHERE d.StudentID = s.StudentID
```

```
    );
```

```
END
```

SCD Type 1 – Overwrite

```
CREATE PROCEDURE SCD_Type_1
```

```
AS
```

```
BEGIN
```

```
    MERGE StudentDim AS target
```

```
    USING StudentStaging AS source
```

```
    ON target.StudentID = source.StudentID
```

```
    WHEN MATCHED AND (
```

```
        target.Name <> source.Name OR
        target.Subject <> source.Subject
    )
    THEN

        UPDATE SET

            target.Name = source.Name,

            target.Subject = source.Subject

    WHEN NOT MATCHED THEN

        INSERT (StudentID, Name, Subject, StartDate, EndDate,
        IsCurrent)

            VALUES (source.StudentID, source.Name, source.Subject,
        GETDATE(), NULL, 1);

    END
```

SCD Type 2 – Add Row with History

```
CREATE PROCEDURE SCD_Type_2
AS
BEGIN

    DECLARE @Today DATETIME = GETDATE();

    -- End-date the old record

    UPDATE StudentDim
```

```

SET EndDate = @Today, IsCurrent = 0

WHERE EXISTS (

    SELECT 1 FROM StudentStaging s

    WHERE StudentDim.StudentID = s.StudentID

    AND (StudentDim.Subject <> s.Subject)

    AND StudentDim.IsCurrent = 1

);

-- Insert new current record

INSERT INTO StudentDim (StudentID, Name, Subject, StartDate,
EndDate, IsCurrent)

SELECT s.StudentID, s.Name, s.Subject, @Today, NULL, 1

FROM StudentStaging s

LEFT JOIN StudentDim d ON s.StudentID = d.StudentID AND
d.IsCurrent = 1

WHERE d.Subject <> s.Subject OR d.StudentID IS NULL;

END

```

SCD Type 3 – Track Changes in Separate Column

```

CREATE PROCEDURE SCD_Type_3

AS

BEGIN

    UPDATE StudentDim

```

```

SET PreviousSubject = Subject,
    Subject = s.Subject
FROM StudentStaging s
WHERE StudentDim.StudentID = s.StudentID
AND StudentDim.Subject <> s.Subject;

INSERT INTO StudentDim (StudentID, Name, Subject, StartDate,
EndDate, IsCurrent)
SELECT s.StudentID, s.Name, s.Subject, GETDATE(), NULL, 1
FROM StudentStaging s
WHERE NOT EXISTS (
    SELECT 1 FROM StudentDim d WHERE d.StudentID = s.StudentID
);
END

```

SCD Type 4 – Historical Table

-- Historical table

```

CREATE TABLE StudentDimHistory (
    StudentID INT,
    Name VARCHAR(100),
    Subject VARCHAR(100),
    ChangeDate DATETIME
);

```

```
CREATE PROCEDURE SCD_Type_4
AS
BEGIN
    -- Archive old data

    INSERT INTO StudentDimHistory (StudentID, Name, Subject,
ChangeDate)

    SELECT d.StudentID, d.Name, d.Subject, GETDATE()
    FROM StudentDim d
    JOIN StudentStaging s ON d.StudentID = s.StudentID
    WHERE d.Subject <> s.Subject;

    -- Update current dimension

    UPDATE StudentDim
    SET Name = s.Name,
        Subject = s.Subject
    FROM StudentStaging s
    WHERE StudentDim.StudentID = s.StudentID;

    -- Insert new records

    INSERT INTO StudentDim (StudentID, Name, Subject, StartDate,
EndDate, IsCurrent)

    SELECT s.StudentID, s.Name, s.Subject, GETDATE(), NULL, 1
```

```
FROM StudentStaging s
WHERE NOT EXISTS (
    SELECT 1 FROM StudentDim d WHERE d.StudentID = s.StudentID
);
END
```

SCD Type 6 – Hybrid (Types 1 + 2 + 3)

```
CREATE PROCEDURE SCD_Type_6
AS
BEGIN
    DECLARE @Today DATETIME = GETDATE();

    -- End-date old record and insert new one with previous value
    INSERT INTO StudentDim (StudentID, Name, Subject,
PreviousSubject, StartDate, EndDate, IsCurrent)
    SELECT
        s.StudentID,
        s.Name,
        s.Subject,
        d.Subject AS PreviousSubject,
        @Today,
        NULL,
```

1

FROM StudentStaging s

JOIN StudentDim d ON s.StudentID = d.StudentID

WHERE d.IsCurrent = 1 AND d.Subject <> s.Subject;

-- Expire old record

UPDATE StudentDim

SET EndDate = @Today, IsCurrent = 0

WHERE EXISTS (

SELECT 1 FROM StudentStaging s

WHERE StudentDim.StudentID = s.StudentID

AND StudentDim.Subject <> s.Subject

AND StudentDim.IsCurrent = 1

);

-- Insert new students

INSERT INTO StudentDim (StudentID, Name, Subject,
PreviousSubject, StartDate, EndDate, IsCurrent)

SELECT s.StudentID, s.Name, s.Subject, NULL, @Today, NULL, 1

FROM StudentStaging s

WHERE NOT EXISTS (

SELECT 1 FROM StudentDim d WHERE d.StudentID = s.StudentID

);

END