# SSIS – Conditional Split Transformation

## 1. Introduction

Conditional Split Transformation in SSIS is used to route data rows to different outputs based on the evaluation of conditions applied to the data. It works similarly to the CASE or IF-ELSE statements in programming languages — sending rows to different destinations depending on the content of certain columns.

## 2. Purpose & Use Cases

Purpose: To filter and direct data into multiple paths within a Data Flow.

Common Use Cases:

- Separating valid and invalid records.

- Routing based on categories (e.g., gender, region, product type).

- Splitting data for different processing logic.

Example Scenario: We have a CSV file with columns: ID, FirstName, LastName, Gender, CompanyName. We want to route Male records to males\_data table and Female records to females\_data table.

## 3. Step-by-Step Implementation

### Step 1 – Create Source Data

Place Testdata\_1.csv in C:\files. File contains records with Gender values: Male / Female.

### Step 2 – Create a New SSIS Package

1. Open SQL Server Data Tools (BIDS/Visual Studio 2014 or later).

2. Drag Data Flow Task from Toolbox → Control Flow.

3. Double-click the Data Flow Task to enter Data Flow design surface.

### Step 3 – Configure Flat File Source

1. Drag Flat File Source to the Data Flow.

2. Double-click to configure.

3. Click New to create a Flat File Connection Manager:

- File Name: Testdata\_1.csv

- Preview data to verify columns and rows.

4. Click OK.

### Step 4 – Add Conditional Split Transformation

1. Drag Conditional Split from the Toolbox to the Data Flow.

2. Connect Flat File Source → Conditional Split.

3. Double-click the Conditional Split to configure:

- Condition 1: Gender == "Male", Output Name: Male

- Condition 2: Gender == "Female", Output Name: Female

4. Click OK.

### Step 5 – Create OLE DB Destinations

For Male Records:

1. Drag OLE DB Destination into the Data Flow.

2. Connect Conditional Split → OLE DB Destination (choose Male output).

3. Configure:

- Create a new OLE DB Connection: Server: Developer, Database: Testing

- Data Access Mode: Table or View - Fast Load

- Create table: males\_data

CREATE TABLE males\_data (  
 ID INT,  
 FirstName VARCHAR(50),  
 LastName VARCHAR(50),  
 Gender VARCHAR(10),  
 CompanyName VARCHAR(100)  
);

For Female Records:

1. Drag another OLE DB Destination.

2. Connect Conditional Split → OLE DB Destination (choose Female output).

3. Use same connection, create table: females\_data

CREATE TABLE females\_data (  
 ID INT,  
 FirstName VARCHAR(50),  
 LastName VARCHAR(50),  
 Gender VARCHAR(10),  
 CompanyName VARCHAR(100)  
);

### Step 6 – Execute the Package

Run the package:

- Example result: males\_data: 12 records, females\_data: 18 records.

## 4. Real-World Applications

- Data Quality Checks: Separate valid/invalid data.

- ETL Processing: Route records to different processing pipelines.

- Data Warehousing: Load different fact tables based on category.

- Business Rules: Apply different transformations per output path.

## 5. Key Points

- Similar to CASE logic in SQL.

- Outputs are mutually exclusive — a row will match the first condition that is true.

- Always include a default output for rows that don’t match any condition.

## 6. Diagram

Flat File Source  
 |  
 v  
Conditional Split ---> Male Output ---> OLE DB Destination (males\_data)  
 ---> Female Output ---> OLE DB Destination (females\_data)