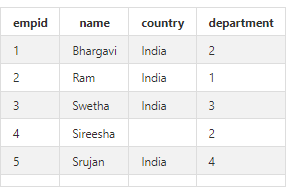
**Slowly Changing Dimension (SCD) Type I**

1. If row is matching with Sink row then update it with the new details for that matching row without maintaining historical/existing data for that row.
2. Insert newly available data/records as new records in the Sink for non-matching rows.

Let’s take below source in csv:



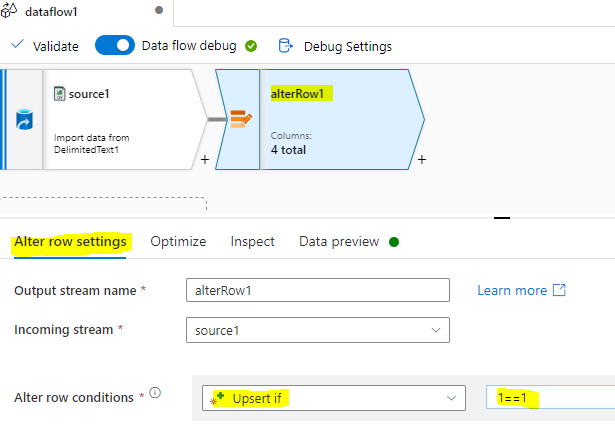
Destination in sql db:



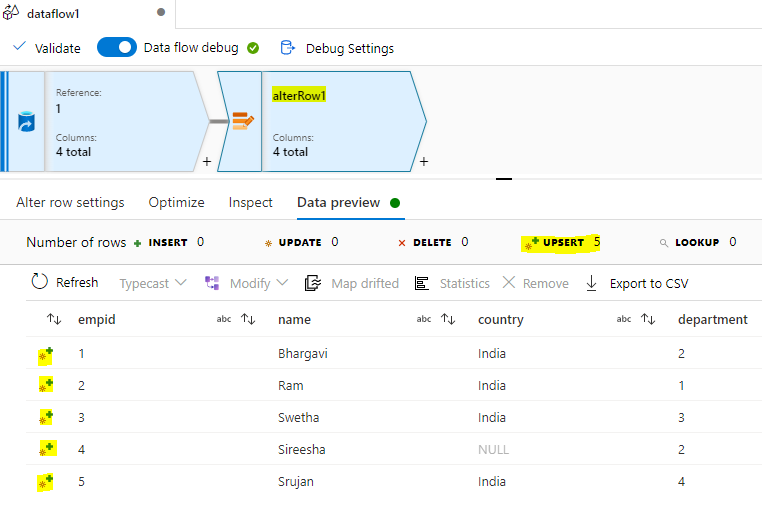
**Upsert**: The term upsert is a portmanteau – a combination of the words “update” and “insert.” In the context of relational databases, an upsert is a database operation that will update an existing row if a specified value already exists in a table, and insert a new row if the specified value doesn't already exist.

**Demo**

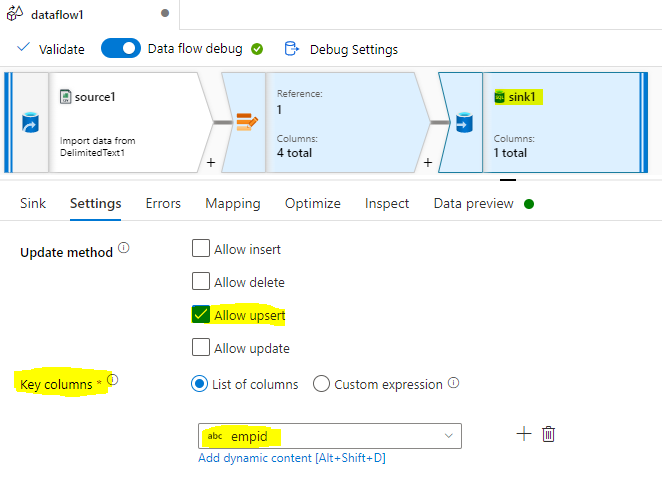
1. Create Data Flow by adding emp.csv and create dataset.
2. Add AlterRow Transformation using UpsertIf with 1==1 expression to update/insert all the rows in the emp.csv file. 1==1 applies update/insert on top of all the rows.



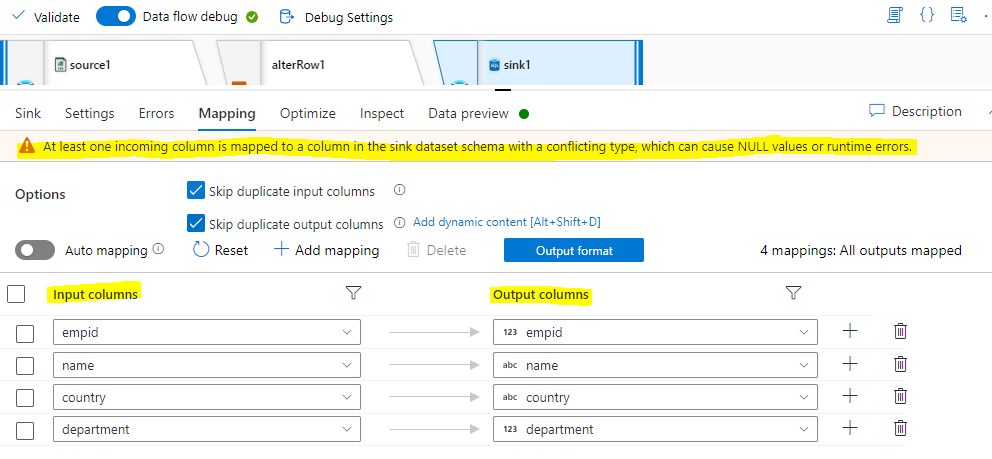
1. Data Preview: Upsert policy is applied on every row which either updates or inserts as per the match.



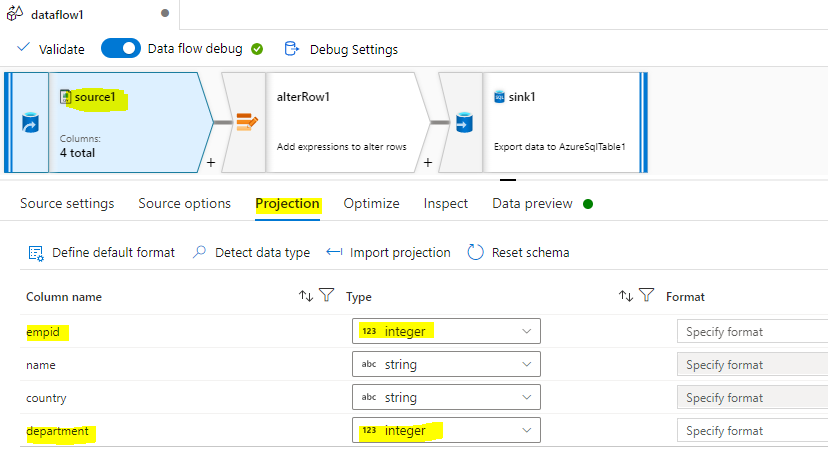
1. Add Sink -> SQL DB dataset and in Settings-> uncheck AllowInsert and check the box for upsert which handles insert+update. Select the empid as key column to check the match condition on.



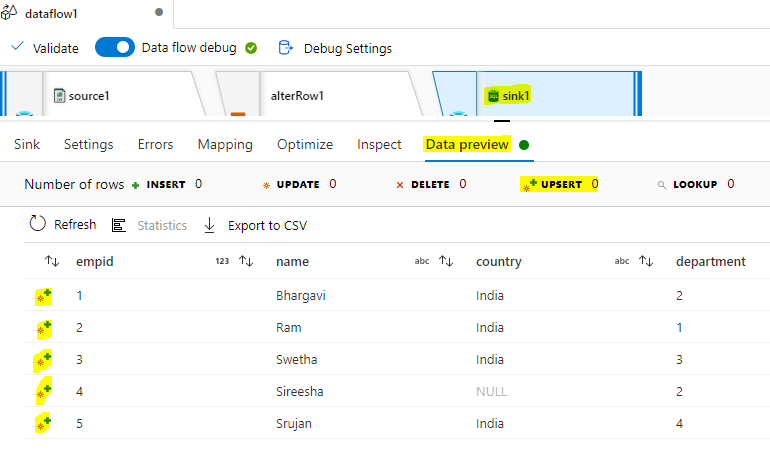
1. Set the Mappings as below:



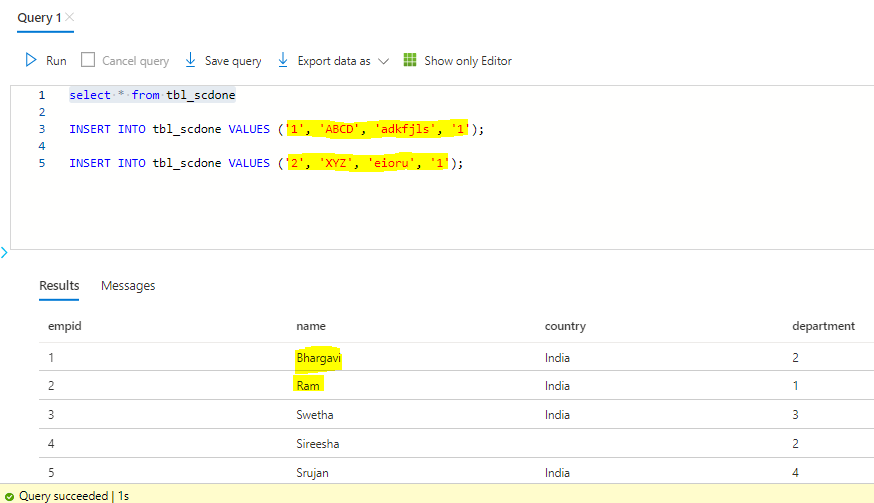
1. There is a warning of conflicting type. Change the data type under Projection of **Source** to the matching columns with Sink SQL DB:



1. Data Preview:



1. Publish this Data Flow and Create Pipeline Activity with Data Flow then Debug. Check the SQL DB for the upsertion of records:



Hence Emp id 1 and 2 updated and 3,4,5 inserted in the Sink SQL DB.