**The Script in the file explains the ETL process as discussed in the class:**

Here the data is taken from the source file which is a CSV file and through the ETL process which can be defined as the process of extracting, transforming and loading in a three-phase process where data is extracted, transformed and loaded into an output data container.

This script explains the fact that the data from the source is taken into a staging table where the staging table can be considered as a warehouse where we can use the data and remove that after the ascertained task is accomplished at the end of the ETL process.

Here our aim is to take the data from the source table > staging table > peripheral linkup table > direct linkup with peripheral linkup table > other direct linkup table > take the IDs from all link up tables of fact table into staging table > import the data from the staging table into the Fact table at the end.

Overview:

The Script executes the SQL statements to populate dimension tables and aims to updating the staging table with respective keys from the populated tables. Finally, it populates the fact table with the transformed data.

Explanation:

1. Loading and Transforming State Data

-- Get States data from Staging Table and Load into DimState

INSERT INTO Crime.DimState([State])

SELECT

DISTINCT [State]

FROM [Crime].[Source\_Data\_Staging]

ORDER BY [State] ASC

This code explains the idea of taking data of states from the staging table and loading with the insert statements in the DimState by ordering according to the states in ascending order. As the state data is a peripheral link up table to the fact through city table data

Here the requirement of extraction is not needed, as it takes the pre-extracted data. Now it comes to the transformation phase, with the selection of distinct states, it ensures that there are no duplicate states that would be entered into `DimState`. Later, it executes to populate the `DimState` dimension table with distinct state names.

2.

-- UPDATE StateID in Staging table

UPDATE Crime.Source\_Data\_Staging

SET Crime.Source\_Data\_Staging.StateID = Crime.DimState.StateID

FROM Crime.Source\_Data\_Staging

INNER JOIN Crime.DimState ON

Crime.Source\_Data\_Staging.[State] = Crime.DimState.[State]

Here the script aims to link the `State` between `DimState` and `Source\_Data\_Staging` to get consistent and establishing appropriate relations by using `StateID`.Later it aims to update the staging table with the appropriate `StateID` from `DimState`.

3. Loading and Transforming City Data

Here the script aims to select the distinct city and state relations in order to reduce any incorrect matching in DimCity. Later it populates DimCIty with distinct City and associates with StateID.

4. Update Staging Table with CityID

Here the Script aims in establishing link between `City` and `StateID` columns in both `DimCity` and `Source\_Data\_Staging`. Later, it focuses on updating the staging table with equivalent `CityID` from `DimCity`.

5. Loading and Transforming Year Data

Here the script aims in selecting only distinct year to prevent any duplicate data and loads them into DimYear

6. Update Staging Table with YearID and Additional Year Data Transformation

Here the script aims to updating the Chinese year data through multiple UPDATE statements for and followed by updating them in the staging table. Later, it takes the additional data from the Chinese new year data into the DimYear with equivalent year. It also focuses on ensuring the YearID is being updated in the staging table subsequently.

7. Loading the Fact Table

Here the entire transformed data from the staging table is inserted into the `FactCrime` fact table, in a way to complete the ETL process.