

Data Modeling

SQL:

1) Flat Main Table (Railways)

- Transaction_ID
- Date_of_Purchase
- Time_of_Purchase
- Departure_Station
- Arrival_Destination
- Date_of_Journey
- Departure_Time,
- Arrival_Time
- Actual_Arrival_Time
- Journey_Status
- Reason_for_Delay, Refund_Request
- Purchase_Type
- Payment_Method
- Railcard
- Ticket_Class
- Ticket_Type,
- Price

2) Cleaning

We performed multiple cleaning steps:

- **Checked for duplicates:** verified Transaction_ID uniqueness.
- **Checked for NULLs:** especially in Transaction_ID, Actual_Arrival_Time, and Reason_for_Delay.
- **Decided NULL handling:** left NULLs in columns like Reason_for_Delay to preserve meaning (NULL = No Delay).
- **Standardized data types:**
 - Converted date/time columns → DATE, TIME.
- **Added constraints:** prepared for primary key (Transaction_ID).
- **Price Validation:** price data checked that it's above zero and valid

3) Dimensions and Facts

a) TicketInfo (Fact-like)

- Transaction_ID (PK)
 - Date_of_Purchase
 - Time_of_Purchase
 - Purchase_Type
 - Payment_Method
 - Railcard
 - Ticket_Class
 - Ticket_Type
 - Price
-

b) RouteInfo (Dimension)

- Route_ID (PK, surrogate with IDENTITY)
 - Departure_Station
 - Arrival_Destination
 - Added UNIQUE constraint that each pair exists only once (64 Routes).
-

c) Journey (Dimension)

- Journey_ID (PK, surrogate with IDENTITY)
 - Transaction_ID (FK from TicketInfo)
 - Departure_Station
 - Arrival_Destination
 - Date_of_Journey
 - Departure_Time
 - Arrival_Time
 - Actual_Arrival_Time
 - Journey_Status
-

d) Delay (Dimension Table)

- Delay_ID (PK, surrogate with IDENTITY)
 - Transaction_ID (FK from TicketInfo)
 - Journey_ID (FK from Journey)
 - Route_ID (FK from RouteInfo)
 - Journey_Status
 - Reason_for_Delay
 - Refund_Request
-

d) Date (Dimension Table)

- DateKey (PK)
 - FullDate
 - Day
 - Month
 - Year
-

Python:

We used python to ensure the validation of the cleaning done by the SQL

A) Duplicate Check:

Ensured no repeated Transaction IDs.

Guarantee: each transaction is unique.

B) Null Check:

Checked critical fields (ID, Dates, Times, Price, Status).

Guarantee: no missing data in essential columns.

C) Primary Key Check:

Verified Transaction ID is unique and not null.

Guarantee: dataset has a valid primary key for joins/analysis.

D) Price Check:

Confirmed Price is present and non-negative.

Guarantee: financial values are reliable.

