Indian Railway Reservation system

- Entities and their attributes.

```
-- Table: Station
    StationID INT PRIMARY KEY,
    StationCode VARCHAR(10),
    StationName VARCHAR(100),
    City VARCHAR(100),
    State VARCHAR(100),
    ContactNumber VARCHAR(15),
    NearbyBusStand VARCHAR(100)


-- Table: Train
    TrainID INT PRIMARY KEY,
    TrainName VARCHAR(100),
    TrainType ENUM('Express', 'Superfast', 'Shatabdi', 'Passenger',
'Local'),
    SourceStationID INT,
    DestinationStationID INT,
    TotalCoach INT,
    TotalSeats INT,
    DaysOfOperation ENUM('Mon', 'Tus', 'Wed', 'Thu', 'Fri', 'Sat',
'Sun'),
    StartTime TIME,
    EndTime TIME
    FOREIGN KEY (SourceStationID) REFERENCES Station(StationID),
    FOREIGN KEY (DestinationStationID) REFERENCES Station(StationID)


-- Table: Route
```

```
    RouteID INT PRIMARY KEY,
    TrainID INT,
    StationID INT,
    SequenceNumber INT,
    ArrivalTime TIME,
    DepartureTime TIME,
    HaltTime INT,
    DistanceFromSource DECIMAL,
    PlatformNumber INT,
    IsSource BOOLEAN,
    IsDestination BOOLEAN,
    FOREIGN KEY (TrainID) REFERENCES Train(TrainID),
    FOREIGN KEY (StationID) REFERENCES Station(StationID)

-- Table: Class
    ClassID INT PRIMARY KEY,
    ClassCode VARCHAR(10),
    ClassName VARCHAR(50),
    IsReserved BOOLEAN,
    FarePerKM DECIMAL,
    MaxCapacity INT



-- Table: Coach
    CoachID INT PRIMARY KEY,
    TrainID INT,
    CouchNumber VARCHAR(10),
    CouchCode VARCHAR(10),
    CouchType ENUM('Sleeper', 'AC First', 'AC Second', 'AC Third',
'Chair car', 'General', 'Luggage'),
```

```
    ClassID INT,
    TotalSeats INT,
    IsReserved BOOLEAN,
    FOREIGN KEY (TrainID) REFERENCES Train(TrainID),
    FOREIGN KEY (ClassID) REFERENCES Class(ClassID)

-- Table: Schedule
    ScheduleID INT PRIMARY KEY,
    TrainID INT,
    DaysOfOperation VARCHAR(50),
    FrequencyType ENUM('Daily', 'Weekly', 'BiWeekly', 'Special'),
    DepartureStationID INT,
    ArrivalStationID INT,
    DepartureTime TIME,
    ArrivalTime TIME,
    JourneyDuration VARCHAR(20),
    RouteID INT,
    FOREIGN KEY (TrainID) REFERENCES Train(TrainID),
    FOREIGN KEY (DepartureStationID) REFERENCES Station(StationID),
    FOREIGN KEY (ArrivalStationID) REFERENCES Station(StationID),
    FOREIGN KEY (RouteID) REFERENCES Route(RouteID)


-- Table: Passenger
    PassengerID INT PRIMARY KEY,
    Name VARCHAR(100),
    Age INT,
    Gender ENUM('male', 'female', 'other'),
    Email VARCHAR(100),
    Phone VARCHAR(15),
```

```
    Nationality VARCHAR(50),
    PassengerType ENUM('Adult', 'child', 'senior citizen',
'student'),

-- Table: Booking
    BookingID INT PRIMARY KEY,
    PassengerID INT,
    TrainID INT,
    ScheduleID INT,
    BookingDate DATETIME,
    JourneyDate DATETIME,
    SourceStationID INT,
    DestinationStationID INT,
    ClassID INT,
    CoachID INT,
    SeatID INT,
    BookingStatus ENUM('cnf', 'wl', 'rac', 'cancelled'),
    QuotaType VARCHAR(50),
    IsCancelled BOOLEAN,
    CancelledDate DATETIME,
    RefundAmount DECIMAL,
    FOREIGN KEY (PassengerID) REFERENCES Passenger(PassengerID),
    FOREIGN KEY (TrainID) REFERENCES Train(TrainID),
    FOREIGN KEY (ScheduleID) REFERENCES Schedule(ScheduleID),
    FOREIGN KEY (SourceStationID) REFERENCES Station(StationID),
    FOREIGN KEY (DestinationStationID) REFERENCES
Station(StationID),
    FOREIGN KEY (ClassID) REFERENCES Class(ClassID),
    FOREIGN KEY (CoachID) REFERENCES Coach(CoachID)

-- Table: Seat
```

```
    SeatID INT PRIMARY KEY,
    CoachID INT,
    SeatNumber VARCHAR(10),
    ClassID INT,
    IsAvailable BOOLEAN,
    BookingID INT,
    SeatType ENUM('Lower', 'Middle', 'Upper', 'Side Lower', 'Side
Upper'),
    Position ENUM('Window', 'Middle', 'Aisle'),
    IsReserved BOOLEAN,
    FOREIGN KEY (CoachID) REFERENCES Coach(CoachID),
    FOREIGN KEY (ClassID) REFERENCES Class(ClassID),
    FOREIGN KEY (BookingID) REFERENCES Booking(BookingID)

-- Table: Feedback
    FeedbackID INT PRIMARY KEY,
    BookingID INT,
    Category VARCHAR(50),
    Subject ENUM('Service', 'Train', 'Station', 'App', 'Staff',
'Other'),
    Message TEXT,
    Rating INT,
    SubmittedAt DATETIME,
    FeedbackType ENUM('Complaint', 'Suggestion', 'Praise', 'Query'),
    FOREIGN KEY (BookingID) REFERENCES Booking(BookingID)

-- Table: Payment
       PaymentID INT PRIMARY KEY,
       BookingID INT,
       PaymentDate DATETIME,
       PaymentTime TIME,
```

```
        AmountPaid DECIMAL,
        PaymentMethod VARCHAR(30),
        PaymentGateway VARCHAR(30),
        TransactionID VARCHAR(40),
        PaymentStatus ENUM('Success','Failed','Pending','Cancelled'),
        IsRefundable BOOLEAN,
        RefundStatus ENUM('Not Initiated','In
process','Refunded','Rejected'),
        RefundAmount DECIMAL,
        RefundProcessedAt DATETIME
        FOREIGN KEY (BookingID) REFERENCES Booking(BookingID)

-- Table: SeatAllocation
      AllocationID INT PRIMARY KEY,
      TrainID INT,
      ScheduleID INT,
      JourneyDate DATE,
      CoachID INT,
      SeatID INT,
      SeatNumber VARCHAR(20),
      PassengerID INT,
      BookingID INT,
      ClassID INT,
      BookingStatusID INT,
      AllocationTime DATETIME,
      LastUpdated DATETIME,
      FOREIGN KEY (TrainID) REFERENCES Train(TrainID),
      FOREIGN KEY (ScheduleID) REFERENCES Schedule(ScheduleID),
      FOREIGN KEY (CoachID) REFERENCES Coach(CoachID),
      FOREIGN KEY (SeatID) REFERENCES Seat(SeatID),
```

```
    FOREIGN KEY (BookingID) REFERENCES Booking(BookingID),
    FOREIGN KEY (BookingStatusID) REFERENCES
BookingStatus(StatusID),
    FOREIGN KEY (ClassID) REFERENCES Class(ClassID),
    FOREIGN KEY (PassengerID) REFERENCES Passenger(PassengerID)
```

- Description of the ER – Diagram

The Indian Railway Reservation System is designed to manage passenger bookings, train schedules, seat allocation, and payment processing efficiently. This ER diagram outlines the relationships and attributes across multiple interconnected entities to ensure a functional, scalable, and normalized database structure

---

1. Passenger

Stores information about passengers:

- **Primary Key:** PassengerID
- Includes details like Name, Age, Gender, Phone, Email, Nationality, and DisabilityStatus.

---

2. Train

Represents train-level information:

- **Primary Key:** TrainID
- Includes TrainName, TrainType, SourceStationID, DestinationStationID, DaysOfOperation, StartTime, and EndTime.

---

3. Station

Details of railway stations:

- **Primary Key:** StationID
- Includes StationName, City, State, StationCode, and indicators like IsJunction.

---

4. Route

Defines routes and station-wise movement of a train:

- **Primary Key:** Composite key including RouteID, StationID
- Tracks ArrivalTime, DepartureTime, DistanceFromSource, and stop information.

---

5. Schedule

Captures train schedules:

- **Primary Key:** ScheduleID
- Contains TrainID, DepartureStationID, ArrivalStationID, DaysOfOperation, FrequencyType, and JourneyDuration.

---

6. Class

Defines class details:

- **Primary Key:** ClassID
- Includes ClassCode, ClassName, FarePerKm, and MaxCapacity.

---

7. Coach

Represents coaches attached to trains:

- **Primary Key:** CoachID
- Includes CoachNumber, CoachCode, CoachType, TotalSeats, and its ClassID.

---

8. Seat

Tracks seat-level data:

- **Primary Key:** SeatID
- Contains SeatNumber, SeatType, Position, and availability (IsAvailable).

---

9. Booking

Captures user bookings:

- **Primary Key:** BookingID
- Foreign Keys: PassengerID, TrainID, ScheduleID, CoachID, ClassID, SeatID

- Includes BookingDate, JourneyDate, SourceStationID, DestinationStationID, QuotaType, IsCancelled, RefundAmount.

---

10. SeatAllocation

Handles allocation of seats post booking:

- **Primary Key:** AllocationID
- Contains info like JourneyDate, CoachID, SeatID, PassengerID, BookingStatusID, AllocationTime.

---

11. Payment

Manages transaction and refund details:

- **Primary Key:** PaymentID
- Includes BookingID, AmountPaid, PaymentMethod, TransactionID, PaymentStatus, RefundStatus, RefundAmount.

---

12. Feedback

Stores passenger feedback post-journey:

- **Primary Key:** FeedbackID
- Foreign Key: BookingID
- Includes Category, Subject, Message, Rating, FeedbackType.

All tables

Seat

```
mysql> select * from seat;
+--------+---------+------------+---------+-------------+-----------+------------+----------+------------+
| SeatID | CoachID | SeatNumber | ClassID | IsAvailable | BookingID | SeatType   | Position | IsReserved |
+--------+---------+------------+---------+-------------+-----------+------------+----------+------------+
|      1 |       1 | S1         |       1 |           1 |      NULL | Lower      | Window   |          0 |
|      2 |       1 | S2         |       1 |           1 |      NULL | Middle     | Middle   |          0 |
|      3 |       1 | S3         |       1 |           1 |      NULL | Upper      | Aisle    |          0 |
|      4 |       1 | S4         |       1 |           1 |      NULL | Side Lower | Window   |          0 |
|      5 |       1 | S5         |       1 |           1 |      NULL | Side Upper | Middle   |          0 |
|      6 |       1 | S6         |       1 |           1 |      NULL | Lower      | Aisle    |          0 |
|      7 |       1 | S7         |       1 |           1 |      NULL | Middle     | Window   |          0 |
|      8 |       1 | S8         |       1 |           1 |      NULL | Upper      | Middle   |          0 |
|      9 |       1 | S9         |       1 |           1 |      NULL | Side Lower | Aisle    |          0 |
|     10 |       1 | S10        |       1 |           1 |      NULL | Side Upper | Window   |          0 |
|     11 |       1 | S11        |       1 |           1 |      NULL | Lower      | Middle   |          0 |
|     12 |       1 | S12        |       1 |           1 |      NULL | Middle     | Aisle    |          0 |
|     13 |       1 | S13        |       1 |           1 |      NULL | Upper      | Window   |          0 |
|     14 |       1 | S14        |       1 |           1 |      NULL | Side Lower | Middle   |          0 |
|     15 |       1 | S15        |       1 |           1 |      NULL | Side Upper | Aisle    |          0 |
|     16 |       1 | S16        |       1 |           1 |      NULL | Lower      | Window   |          0 |
|     17 |       1 | S17        |       1 |           1 |      NULL | Middle     | Middle   |          0 |
|     18 |       1 | S18        |       1 |           1 |      NULL | Upper      | Aisle    |          0 |
|     19 |       1 | S19        |       1 |           1 |      NULL | Side Lower | Window   |          0 |
|     20 |       1 | S20        |       1 |           1 |      NULL | Side Upper | Middle   |          0 |
|     21 |       1 | S21        |       1 |           1 |      NULL | Lower      | Aisle    |          0 |
|     22 |       1 | S22        |       1 |           1 |      NULL | Middle     | Window   |          0 |
|     23 |       1 | S23        |       1 |           1 |      NULL | Upper      | Middle   |          0 |
|     24 |       1 | S24        |       1 |           1 |      NULL | Side Lower | Aisle    |          0 |
|     25 |       1 | S25        |       1 |           1 |      NULL | Side Upper | Window   |          0 |
|     26 |       1 | S26        |       1 |           1 |      NULL | Lower      | Middle   |          0 |
|     27 |       1 | S27        |       1 |           1 |      NULL | Middle     | Aisle    |          0 |
|     28 |       1 | S28        |       1 |           1 |      NULL | Upper      | Window   |          0 |
|     29 |       1 | S29        |       1 |           1 |      NULL | Side Lower | Middle   |          0 |
|     30 |       1 | S30        |       1 |           1 |      NULL | Side Upper | Aisle    |          0 |
|     31 |       1 | S31        |       1 |           1 |      NULL | Lower      | Window   |          0 |
|     32 |       1 | S32        |       1 |           1 |      NULL | Middle     | Middle   |          0 |
|     33 |       1 | S33        |       1 |           1 |      NULL | Upper      | Aisle    |          0 |
|     34 |       1 | S34        |       1 |           1 |      NULL | Side Lower | Window   |          0 |
|     35 |       1 | S35        |       1 |           1 |      NULL | Side Upper | Middle   |          0 |
|     36 |       1 | S36        |       1 |           1 |      NULL | Lower      | Aisle    |          0 |
|     37 |       1 | S37        |       1 |           1 |      NULL | Middle     | Window   |          0 |
```

Train

```
mysql> select * from train;
+---------+-----------------+-----------------+----------------------+-----------+------------+-----------------+-----------+----------+------------+
| TrainID | TrainName       | SourceStationID | DestinationStationID | TotalCoach | TotalSeats | DaysOfOperation | StartTime | EndTime  | TrainType  |
+---------+-----------------+-----------------+----------------------+-----------+------------+-----------------+-----------+----------+------------+
|       1 | Rajdhani Express |               3 |                    1 |          5 |        300 | Mon             | 10:00:00  | 22:00:00 | Express    |
|       2 | Shatabdi Express |               2 |                    6 |          5 |        300 | Mon             | 09:45:00  | 13:45:00 | Shatabdi   |
|       3 | Duronto Superfast |               1 |                    6 |          5 |        300 | Fri             | 14:15:00  | 01:15:00 | Superfast  |
|       4 | Garib Rath Express |               4 |                    7 |          5 |        300 | Mon             | 22:30:00  | 02:30:00 | Express    |
|       5 | Intercity Passenger |               1 |                    2 |          5 |        300 | Thu             | 05:15:00  | 11:15:00 | Passenger  |
+---------+-----------------+-----------------+----------------------+-----------+------------+-----------------+-----------+----------+------------+
5 rows in set (0.00 sec)
```

Station

```
mysql> select * from station;
+-----------+---------------------+---------------+----------------+------------+---------------+---------------+-------------+
| StationID | StationName         | City          | State          | IsJunction | ContactNumber | NearbyBusStand | StationCode |
+-----------+---------------------+---------------+----------------+------------+---------------+---------------+-------------+
|         1 | New Delhi Station   | New Delhi     | Delhi          |       NULL | +916949510404 | +916949510404 | ND          |
|         2 | Howrah Junction     | Kolkata       | West Bengal    |       NULL | +917754600909 | +917754600909 | H           |
|         3 | Chennai Central     | Chennai       | Tamil Nadu     |       NULL | +916125843516 | +916125843516 | CC          |
|         4 | Mumbai CST          | Mumbai        | Maharashtra    |       NULL | +917959015068 | +917959015068 | MC          |
|         5 | Bangalore City      | Bangalore     | Karnataka      |       NULL | +918818257315 | +918818257315 | BC          |
|         6 | Secunderabad Junction | Hyderabad     | Telangana      |       NULL | +916890792958 | +916890792958 | S           |
|         7 | Pune Junction       | Pune          | Maharashtra    |       NULL | +917911697070 | +917911697070 | P           |
|         8 | Lucknow NR          | Lucknow       | Uttar Pradesh  |       NULL | +919102449952 | +919102449952 | LN          |
|         9 | Patna Junction      | Patna         | Bihar          |       NULL | +917278122947 | +917278122947 | P           |
|        10 | Bhopal Junction     | Bhopal        | Madhya Pradesh |       NULL | +918155671763 | +918155671763 | B           |
|        11 | Nagpur Junction     | Nagpur        | Maharashtra    |       NULL | +916240912798 | +916240912798 | N           |
|        12 | Kanpur Central      | Kanpur        | Uttar Pradesh  |       NULL | +919309670932 | +919309670932 | KC          |
|        13 | Ahmedabad Junction  | Ahmedabad     | Gujarat        |       NULL | +919227995378 | +919227995378 | A           |
|        14 | Coimbatore Junction | Coimbatore    | Tamil Nadu     |       NULL | +916952558432 | +916952558432 | C           |
|        15 | Vijayawada Junction | Vijayawada    | Andhra Pradesh |       NULL | +919606481750 | +919606481750 | V           |
|        16 | New Jalpaiguri Station | Siliguri      | West Bengal    |       NULL | +919034384201 | +919034384201 | NJ          |
|        17 | Guwahati Station    | Guwahati      | Assam          |       NULL | +918315290334 | +918315290334 | G           |
|        18 | Jaipur Junction     | Jaipur        | Rajasthan      |       NULL | +919528407799 | +919528407799 | J           |
|        19 | Amritsar Junction   | Amritsar      | Punjab         |       NULL | +917263569873 | +917263569873 | A           |
|        20 | Visakhapatnam Junction | Visakhapatnam | Andhra Pradesh |       NULL | +916548283891 | +916548283891 | V           |
+-----------+---------------------+---------------+----------------+------------+---------------+---------------+-------------+
20 rows in set (0.00 sec)
```

Schedule

```
mysql> select * from schedule;
+------------+---------+---------------------------+---------------+--------------------+--------------------+---------------+-------------+-----------------+---------+
| ScheduleID | TrainID | DaysOfOperation           | FrequencyType | DepartureStationID | ArrivalStationID | DepartureTime | ArrivalTime | JourneyDuration | RouteID |
+------------+---------+---------------------------+---------------+--------------------+--------------------+---------------+-------------+-----------------+---------+
|          1 |       1 | Mon,Tue                   | Special       |                  3 |                  1 | 10:00:00      | 22:00:00    | 12:00:00        |       1 |
|          2 |       2 | Mon,Tue                   | Weekly        |                  2 |                  6 | 09:45:00      | 13:45:00    | 4:00:00         |       2 |
|          3 |       3 | Fri,Sat                   | BiWeekly      |                  1 |                  6 | 14:15:00      | 01:15:00    | 11:00:00        |       3 |
|          4 |       4 | Mon,Sat                   | Special       |                  4 |                  7 | 22:30:00      | 02:30:00    | 4:00:00         |       4 |
|          5 |       5 | Mon,Tue,Wed,Thu,Fri,Sat,Sun | Daily         |                  1 |                  2 | 05:15:00      | 11:15:00    | 6:00:00         |       5 |
+------------+---------+---------------------------+---------------+--------------------+--------------------+---------------+-------------+-----------------+---------+
5 rows in set (0.00 sec)
```

Route

```
mysql> select * from route;
+---------+---------+-----------+----------------+-------------+---------------+----------+--------------------+----------------+----------+---------------+
| RouteID | TrainID | StationID | SequenceNumber | ArrivalTime | DepartureTime | HaltTime | DistanceFromSource | PlatformNumber | IsSource | IsDestination |
+---------+---------+-----------+----------------+-------------+---------------+----------+--------------------+----------------+----------+---------------+
|       1 |       1 |         3 |              1 | 06:00:00    | 06:00:00      |        0 |                  0 |              3 |        1 |             0 |
|       2 |       1 |         4 |              2 | 07:30:00    | 07:40:00      |       10 |                 86 |              1 |        0 |             0 |
|       3 |       1 |         6 |              3 | 09:00:00    | 09:06:00      |        6 |                141 |              4 |        0 |             0 |
|       4 |       1 |        10 |              4 | 10:30:00    | 10:35:00      |        5 |                190 |              2 |        0 |             0 |
|       5 |       1 |        12 |              5 | 12:00:00    | 12:10:00      |       10 |                238 |              2 |        0 |             0 |
|       6 |       1 |        14 |              6 | 13:30:00    | 13:30:00      |        0 |                260 |              1 |        0 |             1 |
|       7 |       2 |         1 |              1 | 06:00:00    | 06:00:00      |        0 |                  0 |              3 |        1 |             0 |
|       8 |       2 |         8 |              2 | 07:30:00    | 07:33:00      |        3 |                 55 |              5 |        0 |             0 |
|       9 |       2 |        12 |              3 | 09:00:00    | 09:08:00      |        8 |                 97 |              1 |        0 |             0 |
|      10 |       2 |        14 |              4 | 10:30:00    | 10:35:00      |        5 |                175 |              4 |        0 |             0 |
|      11 |       2 |        18 |              5 | 12:00:00    | 12:00:00      |        0 |                226 |              4 |        0 |             1 |
|      12 |       3 |         1 |              1 | 06:00:00    | 06:00:00      |        0 |                  0 |              1 |        1 |             0 |
|      13 |       3 |         2 |              2 | 07:30:00    | 07:35:00      |        5 |                 77 |              1 |        0 |             0 |
|      14 |       3 |         8 |              3 | 09:00:00    | 09:07:00      |        7 |                130 |              5 |        0 |             0 |
|      15 |       3 |        10 |              4 | 10:30:00    | 10:35:00      |        5 |                164 |              5 |        0 |             0 |
|      16 |       3 |        11 |              5 | 12:00:00    | 12:08:00      |        8 |                248 |              3 |        0 |             0 |
|      17 |       3 |        12 |              6 | 13:30:00    | 13:30:00      |        0 |                283 |              4 |        0 |             1 |
|      18 |       4 |         2 |              1 | 06:00:00    | 06:00:00      |        0 |                  0 |              1 |        1 |             0 |
|      19 |       4 |         3 |              2 | 07:30:00    | 07:33:00      |        3 |                 52 |              3 |        0 |             0 |
|      20 |       4 |        12 |              3 | 09:00:00    | 09:09:00      |        9 |                 75 |              3 |        0 |             0 |
|      21 |       4 |        13 |              4 | 10:30:00    | 10:39:00      |        9 |                156 |              4 |        0 |             0 |
|      22 |       4 |        14 |              5 | 12:00:00    | 12:00:00      |        0 |                199 |              1 |        0 |             1 |
|      23 |       5 |         5 |              1 | 06:00:00    | 06:00:00      |        0 |                  0 |              3 |        1 |             0 |
|      24 |       5 |         6 |              2 | 07:30:00    | 07:34:00      |        4 |                 62 |              2 |        0 |             0 |
|      25 |       5 |         8 |              3 | 09:00:00    | 09:03:00      |        3 |                106 |              5 |        0 |             0 |
|      26 |       5 |        14 |              4 | 10:30:00    | 10:40:00      |       10 |                160 |              3 |        0 |             0 |
|      27 |       5 |        20 |              5 | 12:00:00    | 12:00:00      |        0 |                253 |              4 |        0 |             1 |
+---------+---------+-----------+----------------+-------------+---------------+----------+--------------------+----------------+----------+---------------+
27 rows in set (0.00 sec)
```

Payment

```
mysql> select * from payment;
+-----------+-----------+-------------+-------------+------------+---------------+----------------+---------------+---------------+---------------+--------------+---------------------+
| PaymentID | BookingID | PaymentDate | PaymentTime | AmountPaid | PaymentMethod | PaymentGateway | TransactionID | PaymentStatus | IsRefundable | RefundStatus | RefundAmount | RefundProcessedAt |
+-----------+-----------+-------------+-------------+------------+---------------+----------------+---------------+---------------+---------------+--------------+---------------------+
|         1 |         1 | 2025-04-14  | 12:06:01    |       1345 | Credit Card   | VISA           | TXN9874521    | Success       |             1 | Not Initiated |            0 | NULL                |
|         2 |        51 | 2025-04-14  | 12:30:59    |       1340 | UPI           | GooglePay      | TRI1023521    | Success       |             0 | Not Initiated |            0 | NULL                |
|         3 |        52 | 2025-04-15  | 22:46:06    |       1500 | UPI           | PayTM          | TYU876278     | Success       |             1 | Refunded      |         1500 | 2025-04-16 01:22:04 |
+-----------+-----------+-------------+-------------+------------+---------------+----------------+---------------+---------------+---------------+--------------+---------------------+
3 rows in set (0.00 sec)
```

Passenger

```
mysql> select * from passenger;
+-------------+---------------+------+--------+---------------------------+--------------+-------------+---------------+-----------------+
| PassengerID | Name          | Age  | Gender | Email                     | Phone        | Nationality | PassengerType | DisabilityStatus |
+-------------+---------------+------+--------+---------------------------+--------------+-------------+---------------+-----------------+
|           1 | Raj Mehta     |   64 | Male   | raj.mehta01@example.in    | 919812759037 | Indian      | Senior Citizen | NO              |
|           2 | Sneha Das     |   17 | Female | sneha.das02@example.in    | 917933239303 | Indian      | Student       | YES             |
|           3 | Mohit Verma   |   90 | Male   | mohit.verma03@example.in  | 918784996514 | Indian      | Senior Citizen | NO              |
|           4 | Nitin Mehta   |   86 | Male   | nitin.mehta04@example.in  | 916578309734 | Indian      | Senior Citizen | NO              |
|           5 | Vikram Das    |   16 | Male   | vikram.das05@example.in   | 916931846426 | Indian      | Student       | NO              |
|           6 | Ravi Joshi    |   10 | Male   | ravi.joshi06@example.in   | 917193674272 | Indian      | Child         | NO              |
|           7 | Vikram Joshi  |   11 | Male   | vikram.joshi07@example.in | 919885884568 | Indian      | Child         | NO              |
|           8 | Suresh Kumar  |   72 | Male   | suresh.kumar08@example.in | 913741268827 | Indian      | Senior Citizen | NO              |
|           9 | Arjun Sharma  |   17 | Male   | arjun.sharma09@example.in | 918225366892 | Indian      | Student       | NO              |
|          10 | Sneha Gupta   |   22 | Female | sneha.gupta10@example.in  | 916224157080 | Indian      | Adult         | NO              |
|          11 | Kavita Joshi  |   77 | Female | kavita.joshi11@example.in | 914873788718 | Indian      | Senior Citizen | YES             |
|          12 | Suresh Sharma |    9 | Male   | suresh.sharma12@example.in | 913046266055 | Indian      | Child         | NO              |
|          13 | Kavita Sharma |   84 | Female | kavita.sharma13@example.in | 919809295306 | Indian      | Senior Citizen | NO              |
|          14 | Pooja Joshi   |   32 | Female | pooja.joshi14@example.in  | 919506558200 | Indian      | Adult         | NO              |
|          15 | Ravi Patel    |    7 | Male   | ravi.patel15@example.in   | 916369228330 | Indian      | Child         | NO              |
|          16 | Amit Das      |   76 | Male   | amit.das16@example.in     | 919226472163 | Indian      | Senior Citizen | NO              |
|          17 | Arjun Reddy   |   67 | Male   | arjun.reddy17@example.in  | 917674108476 | Indian      | Senior Citizen | YES             |
|          18 | Mohit Gupta   |   24 | Male   | mohit.gupta18@example.in  | 916919052014 | Indian      | Adult         | NO              |
|          19 | Arjun Reddy   |   73 | Male   | arjun.reddy19@example.in  | 914203987218 | Indian      | Senior Citizen | NO              |
|          20 | Sneha Verma   |   87 | Female | sneha.verma20@example.in  | 913749304790 | Indian      | Senior Citizen | NO              |
|          21 | Karan Patel   |   71 | Male   | karan.patel21@example.in  | 919486376525 | Indian      | Senior Citizen | NO              |
|          22 | Raj Mehta     |   34 | Male   | raj.mehta22@example.in    | 919427793377 | Indian      | Adult         | NO              |
|          23 | Sneha Gupta   |   61 | Female | sneha.gupta23@example.in  | 916982397561 | Indian      | Senior Citizen | NO              |
|          24 | Ravi Sharma   |   84 | Male   | ravi.sharma24@example.in  | 917107102417 | Indian      | Senior Citizen | NO              |
|          25 | Ritu Mehta    |   24 | Female | ritu.mehta25@example.in   | 915490178642 | Indian      | Adult         | NO              |
|          26 | Sneha Das     |   13 | Female | sneha.das26@example.in    | 912906310417 | Indian      | Student       | YES             |
|          27 | Rahul Patel   |   56 | Male   | rahul.patel27@example.in  | 916088492045 | Indian      | Adult         | NO              |
|          28 | Nisha Mehta   |   72 | Female | nisha.mehta28@example.in  | 914557078647 | Indian      | Senior Citizen | NO              |
|          29 | Neha Patel    |   80 | Female | neha.patel29@example.in   | 917596356891 | Indian      | Senior Citizen | NO              |
|          30 | Sunita Gupta  |   65 | Female | sunita.gupta30@example.in | 917076879725 | Indian      | Senior Citizen | NO              |
|          31 | Arjun Gupta   |   89 | Male   | arjun.gupta31@example.in  | 917200248281 | Indian      | Senior Citizen | NO              |
|          32 | Amit Patel    |   10 | Male   | amit.patel32@example.in   | 918355445606 | Indian      | Child         | NO              |
|          33 | Ritu Sharma   |   48 | Female | ritu.sharma33@example.in  | 914001194216 | Indian      | Adult         | NO              |
|          34 | Meena Singh   |   72 | Female | meena.singh34@example.in  | 914600703348 | Indian      | Senior Citizen | NO              |
|          35 | Nisha Das     |   18 | Female | nisha.das35@example.in    | 912872124906 | Indian      | Student       | NO              |
```

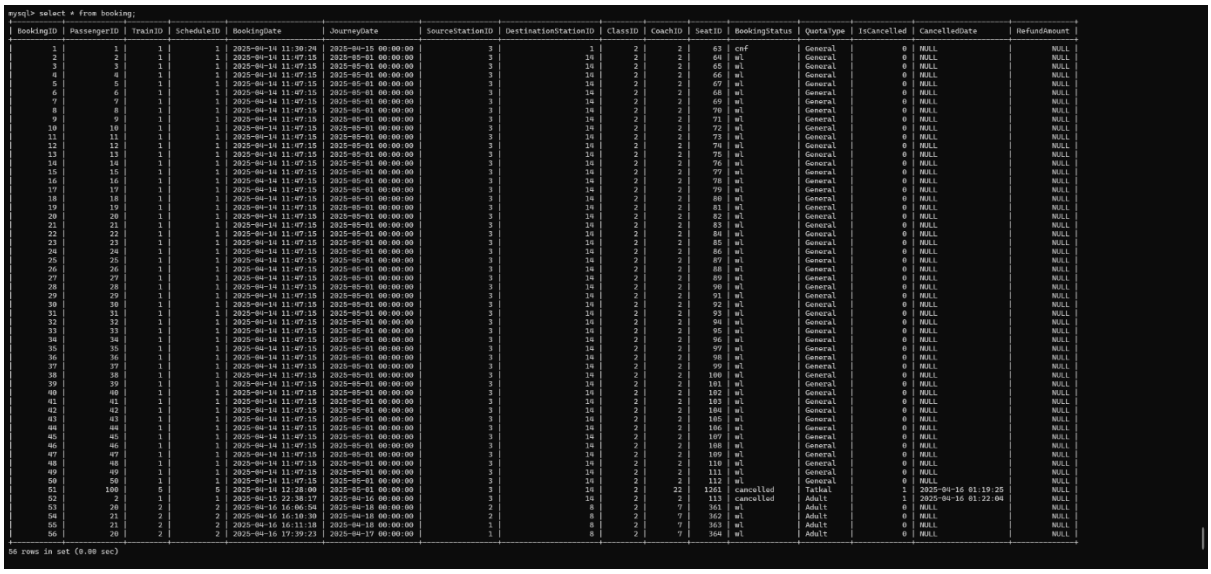
Coach

```
mysql> select * from coach;
+---------+---------+--------------+------------+------------+---------+------------+------------+
| CoachID | TrainID | CoachNumber | CoachCode | CoachType | ClassID | TotalSeats | IsReserved |
+---------+---------+--------------+------------+------------+---------+------------+------------+
|       1 |       1 | SL1          | SL         | Sleeper    |       1 |         60 |          1 |
|       2 |       1 | 1A2          | 1A         | AC First   |       2 |         60 |          1 |
|       3 |       1 | 2A3          | 2A         | AC Second  |       3 |         60 |          1 |
|       4 |       1 | CC4          | CC         | Chair car  |       5 |         60 |          0 |
|       5 |       1 | 3A4          | 3A         | AC Third   |       4 |         60 |          0 |
|       6 |       2 | SL1          | SL         | Sleeper    |       1 |         60 |          1 |
|       7 |       2 | 1A2          | 1A         | AC First   |       2 |         60 |          1 |
|       8 |       2 | 2A3          | 2A         | AC Second  |       3 |         60 |          1 |
|       9 |       2 | GN4          | GN         | General    |       6 |         60 |          0 |
|      10 |       2 | LG5          | LG         | Luggage    |       7 |         60 |          0 |
|      11 |       3 | SL1          | SL         | Sleeper    |       1 |         60 |          1 |
|      12 |       3 | 1A2          | 1A         | AC First   |       2 |         60 |          1 |
|      13 |       3 | 2A3          | 2A         | AC Second  |       3 |         60 |          1 |
|      14 |       3 | GN4          | GN         | General    |       6 |         60 |          0 |
|      15 |       3 | LG5          | LG         | Luggage    |       7 |         60 |          0 |
|      16 |       4 | SL1          | SL         | Sleeper    |       1 |         60 |          1 |
|      17 |       4 | 1A2          | 1A         | AC First   |       2 |         60 |          1 |
|      18 |       4 | 2A3          | 2A         | AC Second  |       3 |         60 |          1 |
|      19 |       4 | GN4          | GN         | General    |       6 |         60 |          0 |
|      20 |       4 | LG5          | LG         | Luggage    |       7 |         60 |          0 |
|      21 |       5 | SL1          | SL         | Sleeper    |       1 |         60 |          1 |
|      22 |       5 | 1A2          | 1A         | AC First   |       2 |         60 |          1 |
|      23 |       5 | 2A3          | 2A         | AC Second  |       3 |         60 |          1 |
|      24 |       5 | GN4          | GN         | General    |       6 |         60 |          0 |
|      25 |       5 | LG5          | LG         | Luggage    |       7 |         60 |          0 |
+---------+---------+--------------+------------+------------+---------+------------+------------+
25 rows in set (0.00 sec)
```

Class

```
mysql> select * from class;
+---------+-----------+------------+------------+------------+-------------+
| ClassID | ClassCode | ClassName  | IsReserved | FarePerKm  | MaxCapacity |
+---------+-----------+------------+------------+------------+-------------+
|       1 | SL        | Sleeper    |          1 |       0.50 |          72 |
|       2 | 1A        | AC First   |          1 |       3.00 |          24 |
|       3 | 2A        | AC Second  |          1 |       2.00 |          48 |
|       4 | 3A        | AC Third   |          1 |       1.50 |          64 |
|       5 | CC        | Chair car  |          1 |       1.00 |          78 |
|       6 | GN        | General    |          0 |       0.20 |          90 |
|       7 | LG        | Luggage    |          0 |       0.10 |          20 |
+---------+-----------+------------+------------+------------+-------------+
7 rows in set (0.00 sec)
```

Booking



The normalization process used here is aimed at reducing redundancy and ensuring data integrity. Let's go through the **normalization levels (1NF to 3NF)** and how they are applied here:

1NF (First Normal Form)

Criteria:

- Atomic (indivisible) values.
- No repeating groups or arrays.

Application:

All tables follow 1NF. For example:

- Passenger(Name, Age, Gender, Email, ...) – atomic attributes.
- Seat(SeatNumber, SeatType, Position, ...) – atomic values.

All tables use atomic values with no repeating groups.

2NF (Second Normal Form)

Criteria:

- Should be in 1NF.
- No partial dependency (i.e., non-prime attributes must depend on the **whole** primary key, not part of a composite key).

Application:

All primary keys are **single-column (surrogate keys)** like PassengerID, BookingID, TrainID, etc.

Since there are **no composite keys**, **no partial dependencies** exist. So all tables are in 2NF.

3NF (Third Normal Form)

Criteria:

- Should be in 2NF.
- No **transitive dependency** (non-prime attributes should not depend on other non-prime attributes).

Application:

Each table separates concepts cleanly:

- Station keeps only station-related details.
- Train links only via foreign keys to Station.
- Coach separates class and type information from the train.
- Seat links to Coach, Class, and optionally a Booking.
- Payment and Feedback relate only to a Booking.

No transitive dependencies observed — every non-key attribute depends only on the primary key of the table

---

Interesting Queries:::

-----> waiting list count in a train <-----

```
delimiter $$
create procedure wlist(
    in p_trainID int
)
begin
select Train.TrainName as Train, count(Booking.TrainID) as WaitingList
from Booking
join Train on Train.TrainID = Booking.TrainID
where BookingStatus = 'wl' and Booking.TrainID = p_trainID
group by Train.TrainName;
end$$
delimiter ;
```

------> Total amount with railways <-------

```
delimiter $$
create procedure totalAmount()
begin
declare amountPaid_var int default 0;
declare refundAmount_var int default 0;
select ifnull(sum(AmountPaid), 0) into amountPaid_var from Payment
where PaymentStatus = 'Success';
select ifnull(sum(refundAmount), 0) into refundAmount_var from Payment
where RefundStatus = 'Refunded';
select amountPaid_var - refundAmount_var as TotalAmount;
end$$
```

delimiter ;

-------> PNR tracking <-----

```
delimiter $$
create procedure track(
    in pnr int
)
begin
select
b.BookingID,
b.PassengerID,
(select TrainName from Train where b.TrainID = Train.TrainID) as Train,
b.BookingDate,
b.JourneyDate,
(select StationName from station where stationID =  SourceStationID) as
SourceStation,
(select StationName from station where stationID =  DestinationStationID)
as DestinationStation,
concat(seatNumber, ' ', SeatType) as seat,
b.bookingstatus as status
from booking b
join seat s on s.seatID = b.seatID
where b.BookingID = pnr;
end$$
delimiter ;
```

------>cancel ticket <-----

```
DELIMITER $$

CREATE PROCEDURE cancel(IN p_BookingID INT)
BEGIN
    UPDATE Booking
```

```
    SET
    BookingStatus = 'cancelled',
        IsCancelled = 1,
        CancelledDate = NOW()
    WHERE BookingID = p_BookingID;
END $$

DELIMITER ;
```

------->trigger for all associated tasks after cancel ticket <----

```
DELIMITER $$

CREATE TRIGGER after_booking_cancel
AFTER UPDATE ON Booking
FOR EACH ROW
BEGIN
    DECLARE v_SeatID INT;
    DECLARE v_IsRefundable BOOLEAN;
    DECLARE v_AmountPaid DECIMAL(10,2);

    -- Only run if BookingStatus changed to 'cancelled'
    IF NEW.BookingStatus = 'cancelled' AND OLD.BookingStatus !=
'cancelled' THEN

        -- Get seat ID
        SET v_SeatID = NEW.SeatID;

        -- Make seat available
        UPDATE Seat
        SET IsAvailable = 1
        WHERE SeatID = v_SeatID;
```

```
        -- Check if refundable & amount paid
        SELECT IsRefundable, AmountPaid
        INTO v_IsRefundable, v_AmountPaid
        FROM Payment
        WHERE BookingID = NEW.BookingID;

        -- Refund if applicable
        IF v_IsRefundable = TRUE THEN
            UPDATE Payment
            SET
                RefundStatus = 'Refunded',
                RefundAmount = v_AmountPaid,
                RefundProcessedAt = NOW()
            WHERE BookingID = NEW.BookingID;
        END IF;

        -- Update SeatAllocation
        UPDATE SeatAllocation
        SET
            BookingStatusID = 0,
            LastUpdated = NOW()
        WHERE BookingID = NEW.BookingID;
    END IF;
END $$

DELIMITER ;
```

----->schedule lookup for a particular train <----

```
delimiter $$
create procedure slookup(
    in p_trainID int
)
begin
select daysofoperation from schedule where trainid = p_trainID;
select routeID, (select stationName from station s where s.stationID =
r.stationID), ArrivalTime, DepartureTime, HaltTime from route r
join train on train.trainID = r.trainID
where r.trainID = p_trainID;
end$$
delimiter ;
```

----->list all passengers travelling in a specific train on specific date<----

```
delimiter $$
create procedure passengerlist(
    in p_trainid int,
    in p_date date
)
begin
select p.Name, p.age, p.gender from passenger p
join booking b on b.passengerid = p.passengerid
where b.JourneyDate = p_date and b.trainid = p_trainid;
end$$
delimiter ;
```

---->bussiest route<----

```
create view SDbooking

select TrainID, concat(SourceStationID,DestinationStationID) as combined
from Booking;


select sum(combined) from SDbooking

groupby combined

limit 1;
```

------>revenue generated for a specific period<-----

```
delimiter $$

create procedure revgen(

    in stDate date,

    in endDate date

)

begin

declare amountPaid_var int default 0;

declare refundAmount_var int default 0;

select ifnull(sum(AmountPaid), 0) into amountPaid_var from Payment

where PaymentStatus = 'Success' and paymentDate between stDate and
endDate;


select ifnull(sum(refundAmount), 0) into refundAmount_var from Payment

where RefundStatus = 'Refunded' and RefundProcessedAt between stDate and
endDate;

select amountPaid_var - refundAmount_var as TotalAmount;

end$$

delimiter ;
```

------>Ticket<----

```
DELIMITER $$

CREATE PROCEDURE BookTicket(
    IN p_PassengerID INT,
    IN p_TrainID INT,
    IN p_ScheduleID INT,
    IN p_JourneyDate DATE,
    IN p_SourceStationID INT,
    IN p_DestinationStationID INT,
    IN p_ClassID INT,
    IN p_QuotaType VARCHAR(50)
)
BEGIN
    DECLARE v_CoachID INT;
    DECLARE v_SeatID INT;
    DECLARE v_SeatNumber VARCHAR(10);
    DECLARE v_BookingID INT;
    DECLARE v_SourceDistance INT;
    DECLARE v_DestDistance INT;
    DECLARE v_Distance INT;
    DECLARE v_FarePerKm DECIMAL(5,2);
    DECLARE v_BaseFare DECIMAL(10,2);
    DECLARE v_TotalAmount DECIMAL(10,2);
    DECLARE v_ConvenienceFee INT DEFAULT 50;
    DECLARE v_PlatformFee INT DEFAULT 50;

    -- Find available seat
    SELECT s.SeatID, s.CoachID, s.SeatNumber
    INTO v_SeatID, v_CoachID, v_SeatNumber
    FROM Seat s
    JOIN Coach c ON c.CoachID = s.CoachID
```

```
    WHERE s.IsAvailable = TRUE
      AND s.ClassID = p_ClassID
      AND c.TrainID = p_TrainID
    LIMIT 1;

    -- Mark seat as unavailable
    UPDATE Seat SET IsAvailable = FALSE WHERE SeatID = v_SeatID;

    -- Generate BookingID
    SELECT IFNULL(MAX(BookingID), 0) + 1 INTO v_BookingID FROM Booking;

    -- Insert into Booking
    INSERT INTO Booking (
        BookingID, PassengerID, TrainID, ScheduleID,
        BookingDate, JourneyDate, SourceStationID,
        DestinationStationID, ClassID, CoachID, SeatID,
        BookingStatus, QuotaType, IsCancelled,
        CancelledDate, RefundAmount
    )
    VALUES (
        v_BookingID, p_PassengerID, p_TrainID, p_ScheduleID,
        NOW(), p_JourneyDate, p_SourceStationID,
        p_DestinationStationID, p_ClassID, v_CoachID, v_SeatID,
        'wl', p_QuotaType, FALSE,
        NULL, NULL
    );
```

-- Get distance

```
SELECT ifnull(DistanceFromSource, 0) INTO v_SourceDistance
```

```
    FROM Route
    WHERE StationID = p_SourceStationID AND TrainID = p_TrainID;

    SELECT ifnull(DistanceFromSource,0) INTO v_DestDistance
    FROM Route
    WHERE StationID = p_DestinationStationID AND TrainID = p_TrainID;

    SET v_Distance = ABS(v_DestDistance - v_SourceDistance);

    -- Get fare rate
    SELECT FarePerKm INTO v_FarePerKm FROM Class WHERE ClassID =
p_ClassID;

    -- Calculate fares
    SET v_BaseFare = v_Distance * v_FarePerKm;
    SET v_TotalAmount = v_BaseFare + v_ConvenienceFee + v_PlatformFee;

    -- Show itemized bill
    SELECT
        v_BookingID AS 'Booking ID',
        p_PassengerID AS 'Passenger ID',
        p_TrainID AS 'Train ID',
        p_ClassID AS 'Class ID',
        v_Distance AS 'Distance (km)',
        v_BaseFare AS 'Base Fare',
        v_ConvenienceFee AS 'Convenience Fee',
        v_PlatformFee AS 'Platform Fee',
        v_TotalAmount AS 'Total Fare';
END$$

DELIMITER ;
```

----->available seats train,date,class<-----

```
delimiter $$
create procedure availseats(
    in p_trainid int,
    in p_date date,
    in p_class int
)
begin
declare totalseats int;
declare bookedseats int;
select ifnull(count(*), 0) into totalseats from seat s
join coach c on c.CoachID = s.CoachID
where c.classID = p_class;

select ifnull(count(*),0) into bookedseats from booking
where booking.trainid = p_trainid and booking.JourneyDate = p_date;

select (select trainName from train where train.trainid = p_trainID) as
Train,p_date as Date, (totalseats - bookedseats) as SeatsAvailable;
end$$
delimiter ;
```

----->list of wailist passengers in a train<----

```
delimiter $$
create procedure listwl(
    in p_trainid int,
    in p_date date
)
begin
select p.name, p.age, p.gender from passenger p
```

```
join booking b on b.passengerid = p.passengerid
where trainid = p_trainid and JourneyDate = p_date and BookingStatus =
'wl';
end$$
delimiter ;
```

---> Procedure to BookTicket <---

```
DELIMITER $$

CREATE DEFINER=root@localhost PROCEDURE BookTicket(
    IN p_PassengerID INT,
    IN p_TrainID INT,
    IN p_ScheduleID INT,
    IN p_JourneyDate DATE,
    IN p_SourceStationID INT,
    IN p_DestinationStationID INT,
    IN p_ClassID INT,
    IN p_QuotaType VARCHAR(50)
)
BEGIN
    DECLARE v_CoachID INT;
    DECLARE v_SeatID INT;
    DECLARE v_SeatNumber VARCHAR(10);
    DECLARE v_BookingID INT;

    -- 1. Find an available seat
    SELECT s.SeatID, s.CoachID, s.SeatNumber
    INTO v_SeatID, v_CoachID, v_SeatNumber
    FROM Seat s
    JOIN Coach c ON c.CoachID = s.CoachID
    WHERE s.IsAvailable = TRUE
      AND s.ClassID = p_ClassID
      AND c.TrainID = p_TrainID
```

```
    LIMIT 1;

    -- 2. Mark the seat as unavailable
    UPDATE Seat
    SET IsAvailable = FALSE
    WHERE SeatID = v_SeatID;

    -- 3. Generate new Booking ID
    SELECT IFNULL(MAX(BookingID), 0) + 1
    INTO v_BookingID
    FROM Booking;

    -- 4. Insert the booking
    INSERT INTO Booking (
        BookingID, PassengerID, TrainID, ScheduleID,
        BookingDate, JourneyDate, SourceStationID,
        DestinationStationID, ClassID, CoachID, SeatID,
        BookingStatus, QuotaType, IsCancelled,
        CancelledDate, RefundAmount
    )
    VALUES (
        v_BookingID, p_PassengerID, p_TrainID, p_ScheduleID,
        NOW(), p_JourneyDate, p_SourceStationID,
        p_DestinationStationID, p_ClassID, v_CoachID, v_SeatID,
        'wl', p_QuotaType, FALSE,
        NULL, NULL
    );
END$$

DELIMITER ;
```

----->Procedure for making payment <---

```
DELIMITER $$


CREATE DEFINER=root@localhost PROCEDURE MakePayment(
    IN p_PaymentID INT,
    IN p_BookingID INT,
    IN p_PaymentDate DATETIME,
    IN p_PaymentTime TIME,
    IN p_AmountPaid DECIMAL(10,2),
    IN p_PaymentMethod VARCHAR(30),
    IN p_PaymentGateway VARCHAR(30),
    IN p_TransactionID VARCHAR(40),
    IN p_PaymentStatus ENUM('Success','Failed','Pending','Cancelled'),
    IN p_IsRefundable BOOLEAN,
    IN p_RefundStatus ENUM('Not Initiated','In process','Refunded','Rejected'),
    IN p_RefundAmount DECIMAL(10,2),
    IN p_RefundProcessedAt DATETIME
)
BEGIN
    -- Insert the payment record
    INSERT INTO Payment (
        PaymentID, BookingID, PaymentDate, PaymentTime,
        AmountPaid, PaymentMethod, PaymentGateway, TransactionID,
        PaymentStatus, IsRefundable, RefundStatus, RefundAmount, RefundProcessedAt
    )
    VALUES (
        p_PaymentID, p_BookingID, p_PaymentDate, p_PaymentTime,
        p_AmountPaid, p_PaymentMethod, p_PaymentGateway, p_TransactionID,
        p_PaymentStatus, p_IsRefundable, p_RefundStatus, p_RefundAmount,
p_RefundProcessedAt
    );


    -- If payment was successful, confirm the booking
    IF p_PaymentStatus = 'Success' THEN
```

```
        UPDATE Booking
        SET BookingStatus = 'cnf'
        WHERE BookingID = p_BookingID;
    END IF;

    -- If payment was cancelled or failed and is refundable, cancel the booking
and store refund info
    IF (p_PaymentStatus = 'Cancelled' OR p_PaymentStatus = 'Failed') AND
p_IsRefundable = TRUE THEN
        UPDATE Booking
        SET
            IsCancelled = TRUE,
            CancelledDate = NOW(),
            RefundAmount = p_RefundAmount
        WHERE BookingID = p_BookingID;
    END IF;
END$$

DELIMITER ;
```

--->Trigger for inserting seat after booking in seat allocation <---

```
DELIMITER $$

CREATE TRIGGER after_booking_insert
AFTER INSERT ON Booking
FOR EACH ROW
BEGIN
    DECLARE v_AllocationID INT;

    -- 1. Generate next AllocationID
    SELECT IFNULL(MAX(AllocationID), 0) + 1
    INTO v_AllocationID
    FROM SeatAllocation;
```

```
    -- 2. Insert into SeatAllocation using data from NEW Booking row
    INSERT INTO SeatAllocation (
        AllocationID, TrainID, ScheduleID, JourneyDate,
        CoachID, SeatID, SeatNumber, PassengerID,
        BookingID, ClassID, BookingStatusID,
        AllocationTime, LastUpdated
    )
    SELECT
        v_AllocationID, t.TrainID, t.ScheduleID, t.JourneyDate,
        t.CoachID, t.SeatID, s.SeatNumber, t.PassengerID,
        t.BookingID, t.ClassID, 1,
        NOW(), NOW()
    FROM (
        SELECT
            NEW.TrainID AS TrainID,
            NEW.ScheduleID AS ScheduleID,
            NEW.JourneyDate AS JourneyDate,
            NEW.CoachID AS CoachID,
            NEW.SeatID AS SeatID,
            NEW.PassengerID AS PassengerID,
            NEW.BookingID AS BookingID,
            NEW.ClassID AS ClassID
    ) t
    JOIN Seat s ON s.SeatID = t.SeatID;
END$$

DELIMITER ;
```

Red diamond - Foreign Key
Blue diamond - Table key
Yellow marker - Primary key
-||- - Table from which foreign key is referenced
claw - Referencing table.

booking
- BookingID INT
- PassengerID INT
- TrainID INT
- ScheduleID INT
- BookingDate DATETIME
- JourneyDate DATETIME
- SourceStationID INT
- DestinationStationID INT
- ClassID INT
- CoachID INT
- SeatID INT
- BookingStatus ENUM(...)
- QuotaType VARCHAR(50)
- IsCancelled TINYINT(1)
- CancelledDate DATETIME
- RefundAmount DECIMAL(10,0)

class
- ClassID INT
- ClassCode VARCHAR(30)
- ClassName VARCHAR(30)
- IsReserved TINYINT(1)
- FarePerKm DECIMAL(10,...
- MaxCapacity INT

train
- TrainID INT
- TrainName VARCHAR(2...
- SourceStationID INT
- DestinationStationID INT
- TotalCoach INT
- TotalSeats INT
- DaysOfOperation SET(...)
- StartTime TIME
- EndTime TIME
- TrainType ENUM(...)

seatallocation
- AllocationID INT
- TrainID INT
- ScheduleID INT
- JourneyDate DATE
- CoachID INT
- SeatID INT
- SeatNumber VARCHAR(1...
- PassengerID INT
- BookingID INT
- ClassID INT
- BookingStatusID INT
- AllocationTime DATETIME
- LastUpdated DATETIME

feedback
- FeedbackID INT
- BookingID INT
- Category VARCHAR(50)
- Subject ENUM(...)
- Message TEXT
- Rating INT
- SubmittedAt DATETIME
- FeedbackType ENUM(....

schedule
- ScheduleID INT
- TrainID INT
- DaysOfOperation VARCHAR(5...
- FrequencyType ENUM(...)
- DepartureStationID INT
- ArrivalStationID INT
- DepartureTime TIME
- ArrivalTime TIME
- JourneyDuration VARCHAR(20)
- RouteID INT

passenger
- PassengerID INT
- Name VARCHAR(30)
- Age INT
- Gender ENUM(...)
- Email VARCHAR(60)
- Phone VARCHAR(20)
- Nationality VARCHAR(15)
- PassengerType ENUM(...)
- DisabilityStatus ENUM('YES', 'N...

payment
- PaymentID INT
- BookingID INT
- PaymentDate DATETIME
- PaymentTime TIME
- AmountPaid DECIMAL(10,0)
- PaymentMethod VARCHAR(30)
- PaymentGateway VARCHAR(3...
- TransactionID VARCHAR(40)
- PaymentStatus ENUM(...)
- IsRefundable TINYINT(1)
- RefundStatus ENUM(...)
- RefundAmount DECIMAL(10,0)
- RefundProcessedAt DATETIME

station
- StationID INT
- StationName VARCHAR(30)
- City VARCHAR(20)
- State VARCHAR(20)
- IsJunction TINYINT(1)
- ContactNumber VARCHAR(20)
- NearbyBusStand VARCHAR(20)
- StationCode VARCHAR(20)

seat
- SeatID INT
- CoachID INT
- SeatNumber VARCHAR(1...
- ClassID INT
- IsAvailable TINYINT(1)
- BookingID INT
- SeatType ENUM(...)
- Position ENUM(...)
- IsReserved TINYINT(1)

route
- RouteID INT
- TrainID INT
- StationID INT
- SequenceNumber INT
- ArrivalTime TIME
- DepartureTime TIME
- HaltTime INT
- DistanceFromSource DECIMAL(10,...
- PlatformNumber INT
- IsSource TINYINT(1)
- IsDestination TINYINT(1)

coach
- CoachID INT
- TrainID INT
- CoachNumber VARCHAR(2...
- CoachCode VARCHAR(20)
- CoachType ENUM(...)
- ClassID INT
- TotalSeats INT
- IsReserved TINYINT(1)