**1. User Table**

sql

CREATE TABLE User (

UserId INT PRIMARY KEY,

Name VARCHAR(100),

Gender VARCHAR(10),

Age INT,

MobileNo VARCHAR(15),

City VARCHAR(50),

State VARCHAR(50),

Pincode VARCHAR(10),

EmailId VARCHAR(100),

Password VARCHAR(100)

);

**2. Admin Table**

sql

CREATE TABLE Admin (

AdminId INT PRIMARY KEY,

Name VARCHAR(100),

EmailId VARCHAR(100),

Password VARCHAR(100)

);

**3. Passenger Table**

sql

CREATE TABLE Passenger (

PassengerId INT PRIMARY KEY,

Name VARCHAR(100),

Age INT,

Gender VARCHAR(10),

SeatNo INT,

TicketId INT,

UserId INT,

TrainNo INT,

StationNo INT,

FOREIGN KEY (TicketId) REFERENCES Ticket(TicketId),

FOREIGN KEY (UserId) REFERENCES User(UserId),

FOREIGN KEY (TrainNo) REFERENCES Train(TrainNo),

FOREIGN KEY (StationNo) REFERENCES Station(StationNo)

);

**4. Train Table**

sql

CREATE TABLE Train (

TrainNo INT PRIMARY KEY,

ArrivalTime TIME,

DepartureTime TIME,

Destination VARCHAR(100),

Source VARCHAR(100),

Date DATE

);

**5. Station Table**

sql

CREATE TABLE Station (

StationNo INT PRIMARY KEY,

Name VARCHAR(100),

TrainNo INT,

FOREIGN KEY (TrainNo) REFERENCES Train(TrainNo)

);

**6. Ticket Table**

sql

CREATE TABLE Ticket (

TicketId INT PRIMARY KEY,

TrainNo INT,

UserId INT,

NoOfPassengers INT,

FOREIGN KEY (TrainNo) REFERENCES Train(TrainNo),

FOREIGN KEY (UserId) REFERENCES User(UserId)

);

**7. Booking Table**

sql

CREATE TABLE Booking (

BookingId INT PRIMARY KEY,

UserId INT,

TrainNo INT,

TicketId INT,

NoOfPassengers INT,

BookingDate DATE,

TotalFare DECIMAL(10, 2),

FOREIGN KEY (UserId) REFERENCES User(UserId),

FOREIGN KEY (TrainNo) REFERENCES Train(TrainNo),

FOREIGN KEY (TicketId) REFERENCES Ticket(TicketId)

);

**8. Cancellation Table**

sql

CREATE TABLE Cancellation (

CancellationId INT PRIMARY KEY,

BookingId INT,

CancellationDate DATE,

RefundAmount DECIMAL(10, 2),

RefundStatus VARCHAR(50),

FOREIGN KEY (BookingId) REFERENCES Booking(BookingId)

);

**9. Class Table**

sql

CREATE TABLE Class (

ClassId INT PRIMARY KEY,

TrainNo INT,

ClassType VARCHAR(50),

FOREIGN KEY (TrainNo) REFERENCES Train(TrainNo)

);

**10. Payment Table**

sql

CREATE TABLE Payment (

PaymentId INT PRIMARY KEY,

BookingId INT,

UserId INT,

PaymentDate DATE,

PaymentMode VARCHAR(50),

PaymentStatus VARCHAR(50),

AmountPaid DECIMAL(10, 2),

FOREIGN KEY (BookingId) REFERENCES Booking(BookingId),

FOREIGN KEY (UserId) REFERENCES User(UserId)

);

**11. Seat Table**

sql

CREATE TABLE Seat (

SeatId INT PRIMARY KEY,

TrainNo INT,

ClassId INT,

SeatNo INT,

AvailabilityStatus VARCHAR(20),

FOREIGN KEY (TrainNo) REFERENCES Train(TrainNo),

FOREIGN KEY (ClassId) REFERENCES Class(ClassId)

);

**12. Route Table**

sql

CREATE TABLE Route (

RouteId INT PRIMARY KEY,

TrainNo INT,

StationNo INT,

FOREIGN KEY (TrainNo) REFERENCES Train(TrainNo),

FOREIGN KEY (StationNo) REFERENCES Station(StationNo)

);

**13. Schedule Table**

sql

CREATE TABLE Schedule (

ScheduleId INT PRIMARY KEY,

TrainNo INT,

StationNo INT,

ArrivalTime TIME,

DepartureTime TIME,

FOREIGN KEY (TrainNo) REFERENCES Train(TrainNo),

FOREIGN KEY (StationNo) REFERENCES Station(StationNo)

);

**Explanation of Relationships:**

* **User** can have many **Bookings** (One-to-Many).
* **Booking** generates one **Ticket** (One-to-One).
* **Passenger** is associated with each **Ticket** (One-to-Many).
* **Train** operates on a predefined **Route** (One-to-Many).
* **Station** is linked with **Route** and **Schedule** (Many-to-Many).
* **Payment** is linked with **Booking** (One-to-One).
* **Train Classes** and **Seats** are linked to **Train** (One-to-Many).