

DDL:

```
CREATE TABLE Airline (  
    airline_id INT PRIMARY KEY,  
    airline_code VARCHAR(10) UNIQUE NOT NULL,  
    airline_name VARCHAR(100) NOT NULL  
);
```

```
CREATE TABLE Airport (  
    airport_id INT PRIMARY KEY,  
    airport_name VARCHAR(100) NOT NULL,  
    city VARCHAR(100) NOT NULL,  
    state VARCHAR(100) NOT NULL  
);
```

```
CREATE TABLE Flight (  
    flight_id INT PRIMARY KEY,  
    flight_name VARCHAR(100) NOT NULL,  
    seat_available INT CHECK (seat_available >= 0),  
    distance INT CHECK (distance > 0),  
    departure_time TIMESTAMP NOT NULL,  
    arrival_time TIMESTAMP NOT NULL,  
    source INT,  
    destination INT,
```

```
    airline_id INT,  
  
    FOREIGN KEY (source) REFERENCES Airport(airport_id) ON DELETE SET  
NULL,  
  
    FOREIGN KEY (destination) REFERENCES Airport(airport_id) ON DELETE  
SET NULL,  
  
    FOREIGN KEY (airline_id) REFERENCES Airline(airline_id) ON DELETE  
CASCADE,  
  
    CHECK (source <> destination),  
  
    CHECK (arrival_time > departure_time)  
);
```

```
CREATE TABLE Class (  
  
    class_id INT PRIMARY KEY,  
  
    class_type VARCHAR(50) NOT NULL,  
  
    price DECIMAL(10, 2) CHECK (price >= 0) NOT NULL  
);
```

```
CREATE TABLE Ticket (  
  
    ticket_id INT PRIMARY KEY,  
  
    flight_id INT NOT NULL,  
  
    date_time TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
  
    seat_number VARCHAR(10) NOT NULL,  
  
    class_id INT NOT NULL,
```

FOREIGN KEY (flight_id) REFERENCES Flight(flight_id) ON DELETE CASCADE,

FOREIGN KEY (class_id) REFERENCES Class(class_id) ON DELETE CASCADE

);

CREATE TABLE "User" (

user_id INT PRIMARY KEY,

first_name VARCHAR(100) NOT NULL,

last_name VARCHAR(100) NOT NULL,

gender VARCHAR(10) CHECK (gender IN ('Male', 'Female', 'Other')),

date_of_birth DATE NOT NULL CHECK (date_of_birth <= CURRENT_DATE),

phone_number VARCHAR(15) UNIQUE,

city VARCHAR(100) NOT NULL,

state VARCHAR(100) NOT NULL

);

CREATE TABLE Booking (

booking_id INT PRIMARY KEY,

ticket_id INT NOT NULL,

user_id INT NOT NULL,

FOREIGN KEY (ticket_id) REFERENCES Ticket(ticket_id) ON DELETE CASCADE,

```
FOREIGN KEY (user_id) REFERENCES "User"(user_id) ON DELETE  
CASCADE,  
  
UNIQUE (ticket_id, user_id) -- Each ticket can only be booked by one user  
);
```

```
CREATE TABLE Payment (  
  
    payment_id INT PRIMARY KEY,  
  
    booking_id INT NOT NULL,  
  
    payment_mode VARCHAR(50) CHECK (payment_mode IN ('Credit Card',  
'Debit Card', 'Cash', 'Online Payment')) NOT NULL,  
  
    amount DECIMAL(10, 2) CHECK (amount >= 0) NOT NULL,  
  
    FOREIGN KEY (booking_id) REFERENCES Booking(booking_id) ON  
DELETE CASCADE  
);
```

❖ Giải thích cách hoạt động và chức năng của từng quan hệ trong database:

1. Airlines

Bảng Airline lưu trữ thông tin về các hãng hàng không khác nhau.

- **airline_id**: Một định danh duy nhất cho hãng hàng không (Khóa chính).
- **airline_code**: Một mã ngắn đại diện cho hãng hàng không, thường là 2-3 ký tự, ví dụ như "AA" cho American Airlines. Được đặt ràng buộc UNIQUE và NOT NULL để đảm bảo mỗi mã hãng hàng không là duy nhất và không rỗng.
- **airline_name**: Tên đầy đủ của hãng hàng không, ví dụ như "American Airlines". Được đặt ràng buộc NOT NULL để đảm bảo không rỗng.

2. Airport

Bảng Airport chứa thông tin về các sân bay khác nhau.

- **airport_id**: Một định danh duy nhất cho sân bay (Khóa chính).
- **airport_name**: Tên của sân bay, ví dụ như "Hartsfield-Jackson Atlanta International Airport". Được đặt ràng buộc NOT NULL để đảm bảo không rỗng.
- **city**: Thành phố nơi sân bay tọa lạc, ví dụ như "Atlanta". Được đặt ràng buộc NOT NULL để đảm bảo không rỗng.
- **state**: Bang hoặc khu vực nơi sân bay tọa lạc, ví dụ như "GA" cho Georgia. Được đặt ràng buộc NOT NULL để đảm bảo không rỗng.

3. Flights

Bảng Flight chứa chi tiết về các chuyến bay.

- **flight_id**: Một định danh duy nhất cho chuyến bay (Khóa chính).
- **flight_name**: Tên hoặc số hiệu chuyến bay, ví dụ như "DL 104". Được đặt ràng buộc NOT NULL để đảm bảo không rỗng.
- **seat_available**: Số lượng ghế còn trống trên chuyến bay. Ràng buộc CHECK để đảm bảo số ghế có sẵn không âm.
- **distance**: Khoảng cách mà chuyến bay sẽ đi, thường được tính bằng kilômét hoặc dặm. Ràng buộc CHECK để đảm bảo khoảng cách là một số dương.
- **departure_time**: Thời gian khởi hành dự kiến của chuyến bay. Được đặt ràng buộc NOT NULL để đảm bảo không rỗng.
- **arrival_time**: Thời gian đến dự kiến của chuyến bay. Được đặt ràng buộc NOT NULL để đảm bảo không rỗng.
- **source**: Tham chiếu đến airport_id từ bảng Airport chỉ định sân bay khởi hành. Tham chiếu này có hành động ON DELETE SET NULL để khi sân bay bị xóa, giá trị này sẽ được đặt là NULL.

- **destination:** Tham chiếu đến airport_id từ bảng Airport chỉ định sân bay đến. Tham chiếu này có hành động ON DELETE SET NULL để khi sân bay bị xóa, giá trị này sẽ được đặt là NULL.
- **airline_id:** Tham chiếu đến airline_id từ bảng Airline chỉ định hãng hàng không. Tham chiếu này có hành động ON DELETE CASCADE để khi hãng hàng không bị xóa, các chuyến bay liên quan cũng sẽ bị xóa.
- Ràng buộc CHECK để đảm bảo sân bay khởi hành và đến không giống nhau.
- Ràng buộc CHECK để đảm bảo thời gian đến lớn hơn thời gian khởi hành.

4. Class

Bảng Class lưu trữ thông tin về các hạng vé máy bay khác nhau.

- **class_id:** Một định danh duy nhất cho hạng vé (Khóa chính).
- **class_type:** Loại hạng vé, ví dụ như "Economy", "Business". Được đặt ràng buộc NOT NULL để đảm bảo không rỗng.
- **price:** Giá của hạng vé đó. Ràng buộc CHECK để đảm bảo giá vé không âm và không rỗng.

5. Ticket

Bảng Ticket lưu trữ thông tin về vé máy bay.

- **ticket_id:** Một định danh duy nhất cho vé (Khóa chính).
- **flight_id:** Tham chiếu đến flight_id từ bảng Flight chỉ định chuyến bay. Tham chiếu này có hành động ON DELETE CASCADE để khi chuyến bay bị xóa, vé liên quan cũng sẽ bị xóa.
- **date_time:** Thời gian phát hành vé. Được đặt giá trị mặc định là thời gian hiện tại.
- **seat_number:** Số ghế trên chuyến bay. Được đặt ràng buộc NOT NULL để đảm bảo không rỗng.

- **class_id**: Tham chiếu đến class_id từ bảng Class chỉ định hạng vé. Tham chiếu này có hành động ON DELETE CASCADE để khi hạng vé bị xóa, vé liên quan cũng sẽ bị xóa.

6. User

Bảng User lưu trữ thông tin về hành khách.

- **user_id**: Một định danh duy nhất cho hành khách (Khóa chính).
- **first_name**: Tên của hành khách. Được đặt ràng buộc NOT NULL để đảm bảo không rỗng.
- **last_name**: Họ của hành khách. Được đặt ràng buộc NOT NULL để đảm bảo không rỗng.
- **gender**: Giới tính của hành khách. Ràng buộc CHECK để chỉ cho phép các giá trị 'Male', 'Female', và 'Other'.
- **date_of_birth**: Ngày sinh của hành khách. Ràng buộc CHECK để đảm bảo ngày sinh không vượt quá ngày hiện tại.
- **phone_number**: Số điện thoại của hành khách. Ràng buộc UNIQUE để đảm bảo số điện thoại là duy nhất.
- **city**: Thành phố nơi hành khách sinh sống. Được đặt ràng buộc NOT NULL để đảm bảo không rỗng.
- **state**: Bang hoặc khu vực nơi hành khách sinh sống. Được đặt ràng buộc NOT NULL để đảm bảo không rỗng.

7. Booking

Bảng Booking lưu trữ thông tin về các đặt chỗ.

- **booking_id**: Một định danh duy nhất cho đặt chỗ (Khóa chính).
- **ticket_id**: Tham chiếu đến ticket_id từ bảng Ticket chỉ định vé. Tham chiếu này có hành động ON DELETE CASCADE để khi vé bị xóa, đặt chỗ liên quan cũng sẽ bị xóa.

- **user_id**: Tham chiếu đến user_id từ bảng User chỉ định hành khách. Tham chiếu này có hành động ON DELETE CASCADE để khi hành khách bị xóa, đặt chỗ liên quan cũng sẽ bị xóa.
- Ràng buộc UNIQUE để đảm bảo mỗi vé chỉ có thể được đặt bởi một người dùng duy nhất.

8. Payment

Bảng Payment lưu trữ thông tin về các thanh toán.

- **payment_id**: Một định danh duy nhất cho thanh toán (Khóa chính).
- **booking_id**: Tham chiếu đến booking_id từ bảng Booking chỉ định đặt chỗ. Tham chiếu này có hành động ON DELETE CASCADE để khi đặt chỗ bị xóa, thanh toán liên quan cũng sẽ bị xóa.
- **payment_mode**: Phương thức thanh toán, ví dụ như "Credit Card", "Debit Card", "Cash", và "Online Payment". Ràng buộc CHECK để chỉ cho phép các giá trị này.
- **amount**: Số tiền thanh toán. Ràng buộc CHECK để đảm bảo số tiền thanh toán không âm.