Database Schema v2

1. User Table

```
CREATE TABLE User (
    user_id INT PRIMARY KEY,
    company VARCHAR(255),
    name VARCHAR(255),
    gender ENUM('Male', 'Female', 'Other'),
    age INT,
    -- HotelBookingReviews TEXT,  -- Consider a separate revi
    -- FlightBookingReviews TEXT, -- Consider a separate revi
    -- CarRentalReviews TEXT,    -- Consider a separate review
    -- Rating INT,                -- Consider a separate revie
    travelCode INT,  -- Consider moving this to a separate Tr
    FOREIGN KEY (travelCode) REFERENCES Trips(travelCode) --
);
```

2. Trips Table (Recommended)

It's generally better practice to separate trip information into its own table, as one user might have multiple trips.

```
CREATE TABLE Trips (
    travelCode INT PRIMARY KEY,  -- Or use a UUID for more un
    user_id INT,
    FOREIGN KEY (user_id) REFERENCES User(user_id),
    -- Add other trip-related details here if needed (e.g., t
);
```

3. Flight Table

```
CREATE TABLE Flight (
    travelCode INT,  -- Foreign key to Trips
    User_ID INT,     -- Foreign key to User (consider removin
    Departure VARCHAR(255),
    Arrival VARCHAR(255),
    flightType ENUM('Domestic', 'International'),
    Flight_price DECIMAL(10,2),
    Flight_duration DECIMAL(5,2),
    Flight_Distance DECIMAL(10,2),
    Flight_agency VARCHAR(255),
    Departure_date DATETIME,
    PRIMARY KEY (travelCode, User_ID), -- Composite key for t
    FOREIGN KEY (travelCode) REFERENCES Trips(travelCode),
    FOREIGN KEY (User_ID) REFERENCES User(user_id)
);
```

4. Hotel Table

```
CREATE TABLE Hotel (
    travelCode INT,  -- Foreign key to Trips
    User_ID INT,     -- Foreign key to User (consider removin
    Hotel_Name VARCHAR(255),
    Arrival_place VARCHAR(255),
    Hotel_stay INT,
    Hotel_per_day_price DECIMAL(10,2),
    Check_in DATETIME,
    Hotel_TotalPrice DECIMAL(10,2),
    PRIMARY KEY (travelCode, User_ID), -- Composite key for t
    FOREIGN KEY (travelCode) REFERENCES Trips(travelCode),
    FOREIGN KEY (User_ID) REFERENCES User(user_id)
);
```

5. Car Rental Table

```
CREATE TABLE CarRental (
    travelCode INT,  -- Foreign key to Trips
    User_ID INT,     -- Foreign key to User (consider removin
    Check_in DATETIME,
    pickupLocation VARCHAR(255),
    dropoffLocation VARCHAR(255),
    carType ENUM('Sedan', 'SUV', 'Hatchback', 'Luxury'),
    rentalAgency VARCHAR(255),
    rentalDuration INT,
    Car_total_distance INT,
    fuelPolicy ENUM('Full-to-Full', 'Prepaid'),
    Car_bookingStatus ENUM('Confirmed', 'Cancelled', 'Pending
    total_rent_price DECIMAL(10,2),
    PRIMARY KEY (travelCode, User_ID), -- Composite key for t
    FOREIGN KEY (travelCode) REFERENCES Trips(travelCode),
    FOREIGN KEY (User_ID) REFERENCES User(user_id)
);
```

6. Passenger Table

```
CREATE TABLE Passenger (
    User_ID INT,     -- Foreign key to User
    company VARCHAR(255),
    Name VARCHAR(255),
    gender_x ENUM('Male', 'Female', 'Other'),
    PRIMARY KEY (User_ID, Name), -- Composite key so a user c
    FOREIGN KEY (User_ID) REFERENCES User(user_id)

);
```

7. Customer Call Table

```
CREATE TABLE CustomerCall (
    User_ID INT,     -- Foreign key to User
    Arrival_date DATETIME,
    issueType VARCHAR(255),
    resolutionStatus ENUM('Resolved', 'Pending', 'Escalated')
    supervisorID INT,  -- Consider a foreign key to a Supervi
    Call_Date DATETIME,
    PRIMARY KEY (User_ID, Call_Date), -- Composite key so a u
    FOREIGN KEY (User_ID) REFERENCES User(user_id)
);
```

8. Guest Table

```
CREATE TABLE Guest (
    Guest_ID INT PRIMARY KEY AUTO_INCREMENT,
    travelCode INT,  -- Foreign key to Trips
    Guest_name VARCHAR(255),
    Guest_Gender ENUM('Male', 'Female', 'Other'),
    Age INT,
    Guest_PhoneNo VARCHAR(20),
    Guest_email VARCHAR(255),
    idProof VARCHAR(255),
    FOREIGN KEY (travelCode) REFERENCES Trips(travelCode)
);
```

9. Review Table

```
CREATE TABLE Review (
    review_id INT PRIMARY KEY AUTO_INCREMENT,  -- Add a prima
    travelCode INT,  -- Foreign key to Trips
    User_ID INT,     -- Foreign key to User
    Car_rented VARCHAR(255), -- Consider making this a boolea
    review_car TEXT,
    review_hotel TEXT,
    review_flights TEXT,
    flight_rating INT,
    hotel_rating INT,
    car_rating INT,
    overall_rating DECIMAL(3,2),
    FOREIGN KEY (travelCode) REFERENCES Trips(travelCode),
    FOREIGN KEY (User_ID) REFERENCES User(user_id)
);
```

Key points:

- **`user_id` and `travelCode`:** Neither of these columns has `AUTO_INCREMENT`. You *must* provide unique values for these when inserting data. This is the most critical change.
- **Foreign Keys:** The foreign key relationships are still in place. This means that the `user_id` and `travelCode` values you insert into other tables *must* correspond to existing values in the `User` and `Trips` tables, respectively.
- **Data Insertion:** You will need to modify your data insertion logic to provide the `user_id` and `travelCode` values explicitly. **Make absolutely sure these values are unique to avoid errors.**