

## **Flight Explorer Code Structure Document**

### **Functions :**

- `connect_to_mysql`: Handles database connection.
- `show_available_flights`: Fetches and displays available flights based on user input.
- `check_seats_availability`: Checks if the desired number of seats are available.
- `update_seats_after_payment`: Updates seat availability after payment is made.
- `get_flight_details`: Fetches details of a specific flight.

### **Function Descriptions :**

- `connect_to_mysql`
  - Purpose: Establishes a connection to the MySQL database.
  - Parameters: None
  - Returns: Connection object if successful, otherwise None.
- `show_available_flights`
  - Purpose: Displays available flights based on user input.
  - Parameters: `connection`, `user_month`, `user_departure`, `user_destination`
  - Returns: True if flights are available, otherwise False.
  - Example Usage: `flights = show_available_flights(connection, user_month, user_departure, user_destination)`
- `check_seats_availability`
  - Purpose: Checks if the desired number of seats are available for a given flight.
  - Parameters: `connection`, `flight_id`, `no_passengers`
  - Returns: True if seats are available, otherwise False.
- `update_seats_after_payment`
  - Purpose: Updates the seat availability after payment is made.
  - Parameters: `connection`, `flight_id`, `no_passengers`
  - Returns: None
- `get_flight_details`
  - Purpose: Fetches details of a specific flight.
  - Parameters: `connection`, `flight_id`
  - Returns: Tuple with flight details (`airline`, `departure_date`, `departure_time`, `arrival_time`).

## **Main Program Workflow :**

- The program establishes a connection to the MySQL database using the `connect_to_mysql` function.
- The program displays a menu with options such as "Book a Flight", "Exit", etc.
- User Selection: The user selects the "Book a Flight" option.
- Departure and Destination: The user is prompted to enter the departure city and the destination country.
- destination City: Based on the destination country, the user selects the specific destination city.
- departure Month: The user selects the desired departure month.
- retrieve Flights: The program uses the `show_available_flights` function to query the database and retrieve a list of flights matching the user's criteria.
- Display Flights: The program displays the available flights along with details such as flight ID, airline, departure time, and arrival time.
- Flight Selection: The user selects a specific flight from the list of available options.
- Enter Passenger Details: The user enters the number of passengers and their details.
- Check Seat Availability: The program checks if the desired number of seats are available on the selected flight using the `check_seats_availability` function.
- Process Payment: Once seat availability is confirmed, the program proceeds to the payment step.
- Update Seats: After successful payment, the program updates the seat availability in the database using the `update_seats_after_payment` function.
- Confirm Booking: The program confirms the booking and provides a summary of the booking details to the user.
- Retrieve Flight Details: The program fetches and displays detailed information about the booked flight using the `get_flight_details` function.
- Booking Summary: The user is presented with a booking summary, including flight details, passenger details, and a confirmation message.
- Return to Main Menu: The user can choose to return to the main menu to perform another operation.
- Exit Program: Alternatively, the user can choose to exit the program

## **Data Structures :**

- `flight_ids_list`: List to store flight id
- `passenger_list`: Dictionary to store passenger details

### **flight\_details Table :**

- flight\_id : unique id for each flight
- departure\_time : date of departure
- arrival\_time : date of arrival
- airline : name of airline
- Seats : number of available seats

### **available\_flights table:**

- flight\_id : unique flight id
- Departure\_city : departure city
- destination\_city : destination city
- Departure\_month : month of departure
- departure\_date : date of departure

### **Error handling:**

- Database connection error :
  - The try block attempts to establish a connection.
  - If an error occurs, the except block catches the Error exception and prints an error message.
  - The function returns None if the connection fails, allowing the program to handle the failed connection gracefully.
- SQL Query Errors:
  - The try block attempts to execute the given SQL query.
  - If an error occurs, the except block catches the Error exception and prints an error message.
- User Input Validation:
  - User input is validated throughout the program.
  - The program checks if the user input is a digit and if it is a valid option.
  - If the input is invalid, an error message is printed, and the loop continues to prompt for valid input.

