home() function:

This function returns the 'search\_result.html' template with no error messages. It serves as the default route when the user navigates to the root path of the application.

flightlist() function:

This function retrieves flight information from the database based on the user's input such as start time, end time, departure airport, and arrival airport. The SQL query is constructed dynamically based on the provided input and retrieves the relevant flights that match the search criteria.

planelist() function:

This function retrieves all airplanes for the logged-in user's airline from the database. The SQL query retrieves all airplane records that belong to the airline associated with the user's session.

addPlane() function:

This function either displays the 'addPlane.html' template with a list of airline names or processes the form submission to insert a new airplane record into the database. The SQL query inserts the new airplane record with the provided plane ID and airline name.

addFlight() function:

This function either displays the 'addFlight.html' template with a list of airport and airplane details or processes the form submission to insert a new flight record into the database. The SQL query inserts the new flight record with the provided flight details.

flightstatus() function:

This function updates the status of a flight in the database based on the provided flight number and new status. The SQL query updates the flight record's status in the database and then redirects the user back to the flight list page.

airportlist() function:

This function retrieves a list of all airports from the database and displays them in the 'airportlist.html' template. The SQL query selects all records from the airport table, and the fetched data is passed to the template for rendering.

addAirport() function:

This function either displays the 'addAirport.html' template or processes the form submission to insert a new airport record into the database. The SQL query inserts the new airport record with the provided name and city.

agentlist() function:

This function retrieves a list of all agents from the database and displays them in the 'workforlist.html' template. The SQL query selects all records from the works\_for table, and the fetched data is passed to the template for rendering.

addagent() function:

This function either displays the 'addWorkfor.html' template or processes the form submission to insert a new agent record into the works\_for table. The SQL query inserts the new agent record with the current session's airline and the provided agent email.

chome() function:

This function redirects the user to the '/viewCus' route, which likely displays a customer-related page. It ensures the connection with the database is maintained by pinging it before performing the redirect.

bhome() function:

This function ensures the connection with the database is maintained by pinging it and then renders the 'Bhome.html' template, likely displaying a booking agent's home page.

ahome() function:

This function pings the database connection, prints the session type, and renders the 'Ahome.html' template, likely displaying an airline staff's home page.

logout() function:

This function pings the database connection and then clears the session's username and type fields. It then redirects the user back to the root route (homepage).

login() function:

This function is responsible for handling user login. Depending on the user type (Customer, Agent, or Airline Staff), it executes different SQL queries to authenticate the user. If the authentication is successful, the function sets up the session with relevant information and redirects the user to the appropriate page based on their type.

register() function:

This function pings the database connection and renders the 'register.html' template, likely displaying a user registration page.

registerAuth() function:

This function authenticates the user registration based on the user type (Customer, Booking Agent, or Airline Staff) and renders the corresponding registration template (Cregister.html, Bregister.html, or Aregister.html) for each user type.

Cregister() function:

This function registers a new customer by inserting their details into the 'customer' table. The SQL queries are used to check if a user with the same email already exists and to insert the new user's information if it doesn't.

Bregister() function:

This function registers a new booking agent by inserting their details into the 'agent' table. The SQL queries are used to check if a user with the same email already exists and to insert the new user's information if it doesn't.

Aregister() function:

This function registers a new airline staff member by inserting their details into the 'airline\_staff' table. The SQL queries are used to check if a user with the same username already exists, verify if the airline exists, and to insert the new user's information if the conditions are met.

search() function:

This function searches for flights based on the user's input for departure city/port, destination city/port, and date. The SQL query retrieves flight information (flight number, departure time, and arrival time) from the 'flight' and 'airport' tables for flights that match the user's search criteria.

cbuy() function:

This function allows a customer to purchase a ticket for a selected flight. The SQL query inserts a new row into the 'ticket' table with the flight number and the customer's email address.

view\_my\_flight() function:

This function allows a customer to view their flight history based on optional filters like start and end times, departure airport, and arrival airport. The SQL query retrieves flight information from the 'flight' and 'ticket' tables for flights that match the user's specified filters.

view\_agent\_flight() function:

This function allows a booking agent to view their customers' flight history based on optional filters like start and end times, departure airport, and arrival airport. The SQL query retrieves flight information from the 'flight' and 'ticket' tables for flights that match the user's specified filters.

getagentchart1() function:

This function generates a bar chart of the top 5 customers who have purchased the most tickets through a booking agent within the past six months. The SQL query retrieves the count of tickets and customer emails from the 'ticket' and 'flight' tables, groups them by customer email, and orders the results by ticket count in descending order, limiting the results to the top 5 customers.

getagentchart2():

This function generates a bar chart that displays the top 5 customers by total flight price within the last year for a specific agent.

The SQL query retrieves the sum of flight prices for each customer and sorts them in descending order, limiting the result to the top 5 customers.

chartAgent1():

This function renders the 'chartAgent1.html' template.

No SQL queries are executed in this function.

aagentTop():

This function retrieves and displays the top 5 agents in terms of revenue, sales count, and monthly sales count within the last year.

Three SQL queries are executed to obtain the required data for each metric (revenue, sales count, and monthly sales count) and then passed to the 'agentTop.html' template.

getchart():

This function generates a bar chart that displays the monthly sales between a specified start time and end time.

The SQL query groups the sales data by month and calculates the sum of prices for each month.

report1():

This function renders the 'chartStafff.html' template.

No SQL queries are executed in this function.

getchartStaff():

This function generates a bar chart that displays the monthly revenue for a specific airline within the last year.

The SQL query groups the sales data by month and calculates the sum of prices for each month.

getpie1():

This function generates a pie chart that displays the revenue distribution between direct sales and agent sales for a specific airline within the last three months.

Two SQL queries are executed to obtain the total revenue for direct sales and agent sales, respectively.

getpie2():

This function generates a pie chart that displays the revenue distribution between direct sales and agent sales for a specific airline within the last year.

Two SQL queries are executed to obtain the total revenue for direct sales and agent sales, respectively.

Viewreports():

This function renders the 'Viewreports.html' template.

No SQL queries are executed in this function.

Viewfrequentcustomers():

This function retrieves and displays the most frequent customer and their flight records for a specific airline within the last year.

Two SQL queries are executed: one to find the most frequent customer and another to fetch their flight records.

airstafflist():

This function retrieves and displays a list of airline staff for a specific airline.

The SQL query selects all records from the airline\_staff table for the specified airline.

addairstaffauth():

This function updates the permission level of a specified airline staff member.

The SQL query updates the permission\_id in the airline\_staff table for the specified username.

abuy():

This function allows an agent to purchase a ticket on behalf of a customer by inserting a new record into the ticket table.

The SQL query inserts the flight number, customer email, and agent email into the ticket table.

showagent():

This function displays an agent's total sales revenue and the number of tickets sold within a specific date range, defaulting to the past month.

The SQL query retrieves the sum of ticket prices and the count of tickets sold for the agent within the specified date range.

ViewTopdestinations():

This function displays the top 3 most popular destination cities for the past three months and the past year.

Two SQL queries are executed: one for the past three months and another for the past year. Both queries count the number of tickets per city and sort the results in descending order, limiting the result to the top 3 cities.

chartCus():

This function displays the total revenue generated from ticket sales within a specified date range, defaulting to the past 180 days.

The SQL query retrieves the sum of ticket prices within the specified date range.

tickets():

This function retrieves and displays a list of tickets for a specific flight number.

The SQL query selects all records from the ticket table where the flight number matches the provided value.