Brief description of what each file does

- `static/` web page style design
  - `airplane_bg.jpg`
  - `Agent-icon.jpg`
  - `Agent-icon.svg`
  - `Airplane_bg2.jpg`
  - `Bootstrap.bundle.min.js`
  - `Bootstrap.bundle.min.js.map`
  - `Bootstrap.min.css`
  - `Bootstrap.min.map`
  - `Bootstrap.rtl.min.css`
  - `Bootstrap.rtl.min.css.map`
  - `Carousel.css`
  - `Carousel.rtl.css`
  - `Customer-icon.jpg`
  - `Destination1.jpg`
  - `Destination1.png`
  - `Destination2.jpg`
  - `Destination2.png`
  - `Echarts.min.js`
  - `Flight-search-icon.jpg`
  - `Form-validation.css`
  - `Form-validation.js`
  - `Index-airline.jpg`
  - `Login-flight.jpg`
  - `Login.css`
  - `Sea-bg.jpg`
  - `Staff-icon.jpg`
  - `Starter-template.css`
- `templates/` all htmls files
  - `addAirplane.html` page for staff to add an airplane
  - `addAirplaneConfirm.html` page confirming adding airplane was successful
  - `addAirport.html` page for staff to add an airport
  - `agent.html` main page for booking agent
  - `agentUpcomingFlights.html` search page for booking agent
  - `changeFlight.html` page for staff to change status of flights
  - `commission.html` page to show commision in a specify time range

- `ComparisonRevenue.html` page to show the revenue comparison for booking agent
- `createFlight.html` page for staff to create a new flight
- `customer.html` main page for customer
- `CustomerError.html` error page for customer
- `error.html` general error page which returns back to the main page (index.html)
- `index.html` main page of the app
- `login.html` login page of the app
- `purchaseAgent.html` page for booking agent to search and purchase tickets
- `purchaseCustomer.html` page for customer to search and purchase tickets
- `registerAgent.html` register page for booking agent
- `registerCustomer.html` register page for customer
- `registerStaff.html` register page for airline staff
- `search.html` page for users to search without logging in
- `Searchall.html` page to search for all flights without logging in
- `searchFlights.html` results page when users search without logging in
- `searchStaff.html` page for staff to search for flights
- `searchStaffResults.html` results for staff searching for flights (limited to their airline)
- `searchstatus.html` page for user to search status of flights without logging in
- `staff.html` main page for staff
- `viewAgents.html` page for staff to search info about agents
- `viewAgentsCommission.html` results page for staff after searching for agents based on commission
- `viewAgentsSales.html` results page for staff after searching for agents based on ticket sales
- `viewCustomers.html` search for customers on a flight, and search for the most frequent customer
- `viewCustomerSpending.html` page for tracking for customers's spending
- `viewCustomersResults.html` results for searching for customers on a flight
- `viewReports.html` Show graph for tickets sold in last year and search for stats based on range
- `viewReportsDate.html` Show tickets sold in date range specified
- `viewReportsPast.html` Show tickets sold in last year/month
- `viewTopCustomer.html` show top customers for agent
- `viewUpcomingFlights.html` show upcoming flights for customer

- `agent.py` functions for agent's home page to: 1) get ticket counts, commission 2) search through flights
- `appdef.py` defines the configurations of the app
- `customer.py` functions for customers to: 1) view all flights 2) search through flights
- `init.py` initialize the app
- `login.py` login authentication
- `publicinfo.py` search flights without logging in
- `purchase.py` functions for purchasing tickets by customers and booking agents
- `register.py` register functions for customer, agent, and staff
- `staff.py` functions for staff home page and all of their use cases

- `sql/` all sql files + texts
    - `Insert.sql` modify the inserted flight information
    - `Create_tables.sql` create all tables for the databases