COMP3322 Modern Technologies on World Wide Web

Assignment Two

Total 18 points

Deadline: March 18, 2024 23:59

Overview

You are going to design and develop a Web app that retrieves and displays passenger flight statistics from Hong Kong Airport Open Data. This REST web service provided by HK Airport returns historical flight data in JSON format. The primary function of the app will be to showcase passenger flight statistics for a specific date selected by the user. It is crucial for the Web app to have a user-friendly interface that is compatible with both mobile platforms and desktop computers.

Objectives

1. A learning activity to support ILO 1 and ILO 2.
2. To learn how to make use of Open Data.
3. To practice using JavaScript to (1) create dynamic content, (2) carry out AJAX communication for retrieving Open Data, and (3) selectively display flight information.
4. To practice using CSS styling to design responsive Web application.

Hong Kong Airport Open Data

Our Web App will be built using the data provided by the Airport Authority (HKAA).

Website: https://data.gov.hk/en-data/dataset/aahk-team1-flight-info

You can download the data dictionary about the request parameters and the response data set from here.
https://www.hongkongairport.com/iwov-resources/misc/opendata/Flight_Information_DataSpec_en.pdf

To get the **passenger departure data** for the calendar date **2023-12-20**, use this URL
https://www.hongkongairport.com/flightinfo-rest/rest/flights/past?date=2023-12-20&lang=en&cargo=false&arrival=false

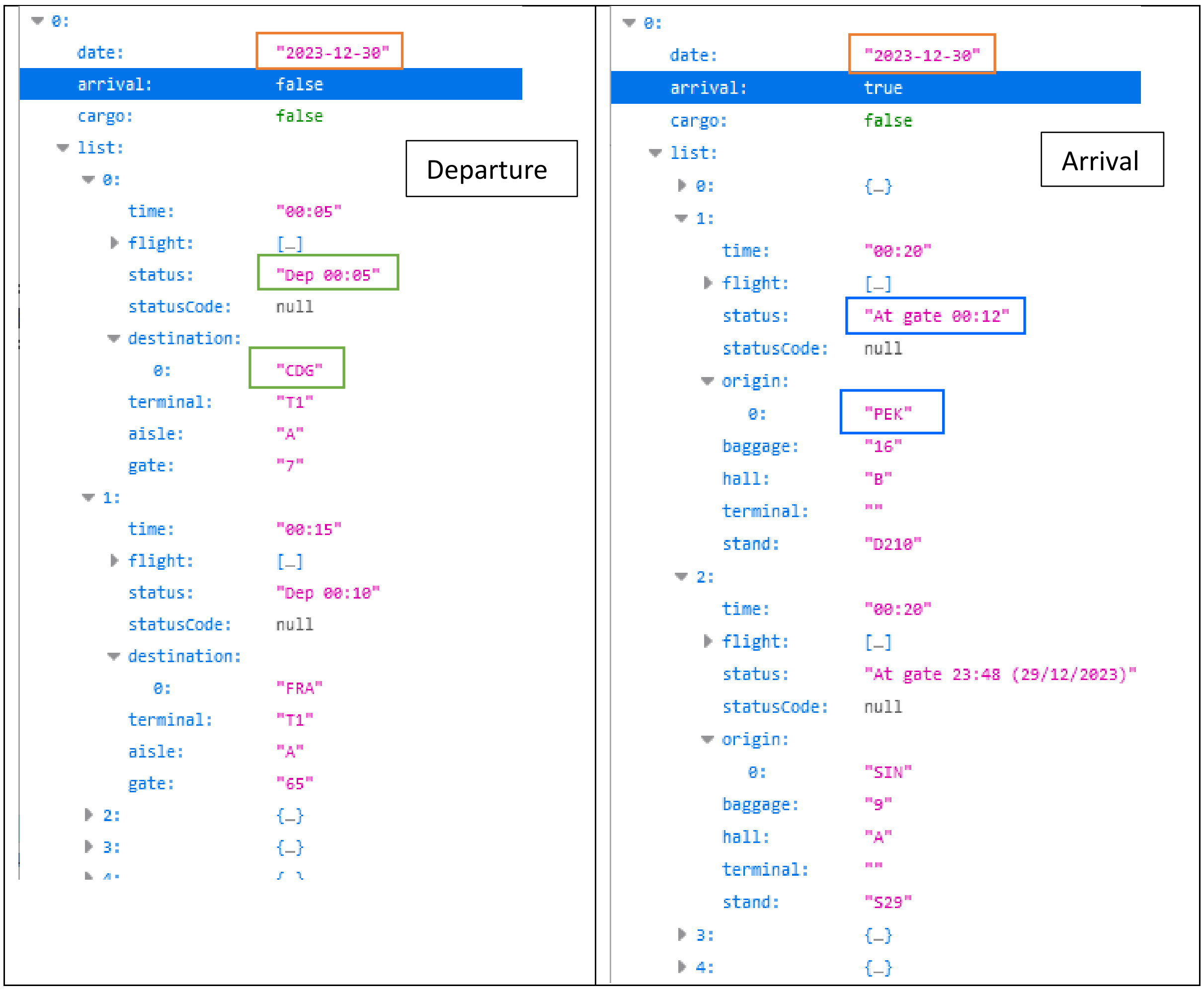
To get the **passenger arrival data** for the calendar date **2023-12-20**, use this URL
https://www.hongkongairport.com/flightinfo-rest/rest/flights/past?date=2023-12-20&lang=en&cargo=false&arrival=true

Please note that the date string must be in the format YYYY-MM-DD.

The datasets of the departure and arrival flights have common fields and a few unique fields. Here are the example JSON data returned by the API for **2023-12-30**.

Each departure flight schedule consists of a list of flight information, which in turn, contains a time field, the airline info, the status of the flight, the destination IATA code, and other info. Similarly, the arrival flight schedule consists of a list of flight information and each flight contains a time field, the airline info, the status of the flight, the origin IATA code, and other info.

Please read the data dictionary file to learn the meaning of individual fields and perform a few GET requests **using Firefox** to examine the returned JSON datasets.

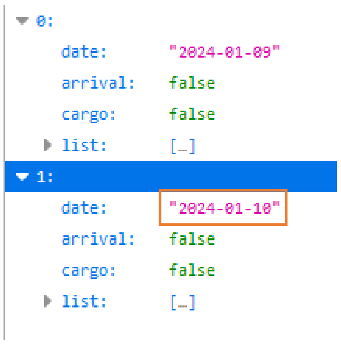


```
▾ 0:
    date:            "2023-12-30"
    arrival:         false
    cargo:           false
  ▾ list:
    ▾ 0:
        time:          "00:05"
      ▸ flight:        […]
        status:        "Dep 00:05"
        statusCode:    null
      ▾ destination:
          0:           "CDG"
        terminal:      "T1"
        aisle:         "A"
        gate:          "7"
    ▾ 1:
        time:          "00:15"
      ▸ flight:        […]
        status:        "Dep 00:10"
        statusCode:    null
      ▾ destination:
          0:           "FRA"
        terminal:      "T1"
        aisle:         "A"
        gate:          "65"
    ▸ 2:               {…}
    ▸ 3:               {…}
```

```
▾ 0:
    date:            "2023-12-30"
    arrival:         true
    cargo:           false
  ▾ list:
    ▸ 0:             {…}
    ▾ 1:
        time:          "00:20"
      ▸ flight:        […]
        status:        "At gate 00:12"
        statusCode:    null
      ▾ origin:
          0:           "PEK"
        baggage:       "16"
        hall:          "B"
        terminal:      ""
        stand:         "D210"
    ▾ 2:
        time:          "00:20"
      ▸ flight:        […]
        status:        "At gate 23:48 (29/12/2023)"
        statusCode:    null
      ▾ origin:
          0:           "SIN"
        baggage:       "9"
        hall:          "A"
        terminal:      ""
        stand:         "S29"
    ▸ 3:             {…}
    ▸ 4:             {…}
```

For this assignment, we **only use** the following information for building the app.

- The returned JSON dataset from the HKAA server may contain sets of flight data of different dates. The app only extracts the set of data with the date field matching the user-selected date. For example, the program received the following JSON data for the date Jan 10, 2024.



```
▾ 0:
    date:      "2024-01-09"
    arrival:   false
    cargo:     false
  ▸ list:      […]
▾ 1:
    date:      "2024-01-10"
    arrival:   false
    cargo:     false
  ▸ list:      […]
```

  In this case, the program **only uses** the list of flights for the date 2024-01-10.
- For each departure flight, the program **only uses** the status field and the destination[0] field. For each arrival flight, the program **only uses** the status field and the origin[0] field.

Please note that each departure flight may go to more than one destination airport, and it is also possible that an arrival flight may have more than one origin airport. However, the app only extracts the first entry and ignores the others.

The destination and the origin airports are encoded by the standard IATA code. To provide useful information to the users, we have prepared another JSON file – **iata.json**, which contains more descriptive information about an airport. For examples,

IATA code: HKG
{
  "iata_code": "HKG",
  "name": "Hong Kong International Airport",
  "continent": "AS",
  "iso_country": "HK",
  "iso_region": "HK-U-A",
  "municipality": "Hong Kong"
}
	IATA code: MEL
	{
	  "iata_code": "MEL",
	  "name": "Melbourne International Airport",
	  "continent": "OC",
	  "iso_country": "AU",
	  "iso_region": "AU-VIC",
	  "municipality": "Melbourne"
	}

IATA code: KIX
{
  "iata_code": "KIX",
  "name": "Kansai International Airport",
  "continent": "AS",
  "iso_country": "JP",
  "iso_region": "JP-27",
  "municipality": "Osaka"
}
	IATA code: WUH
	{
	  "iata_code": "WUH",
	  "name": "Wuhan Tianhe International Airport",
	  "continent": "AS",
	  "iso_country": "CN",
	  "iso_region": "CN-42",
	  "municipality": "Wuhan"
	}

Unfortunately, we **cannot directly fetch** HK Airport flight data from the HKA server using JavaScript in our web app. It is because the hongkongairport.com server does not set up the CORS setting. Cross-origin resource sharing (CORS) is a mechanism that allows restricted resources on a web page to be requested from another domain outside the domain from which the first resource was served. To overcome this limitation, we prepared a **flight.php** program, which allows our web server to act as a client to request flight data from the Hong Kong Airport server and relay the data to our web app. The flight.php program accepts the same set of query strings as the Hong Kong airport server.

flight.php?**date**=2024-1-15&**lang**=en&**cargo**=false&**arrival**=true

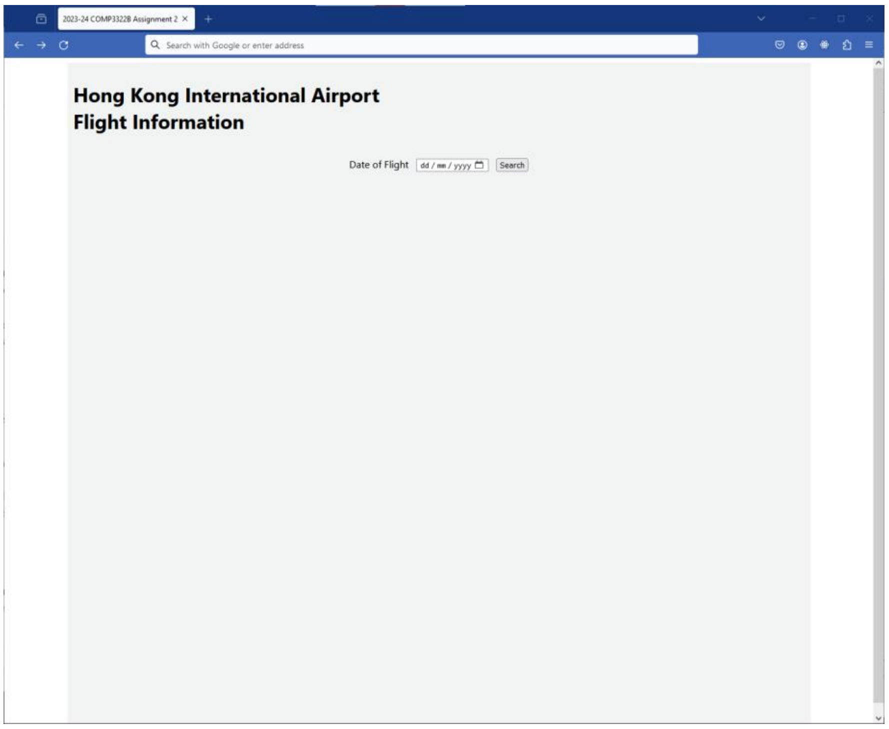
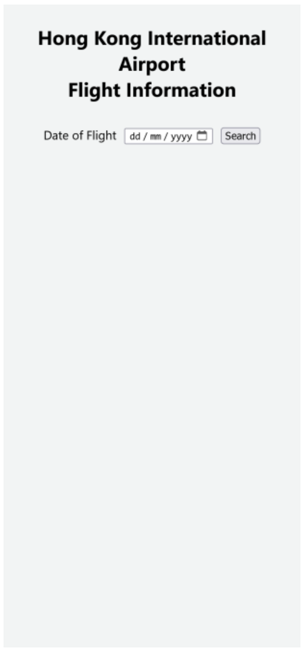
Please place the flight.php program and the iata.json file under the public_html folder of your web server container.

```
public_html
|
├── flight.php
├── iata.json
├── styles.css
├── main.js
└── index.html
```

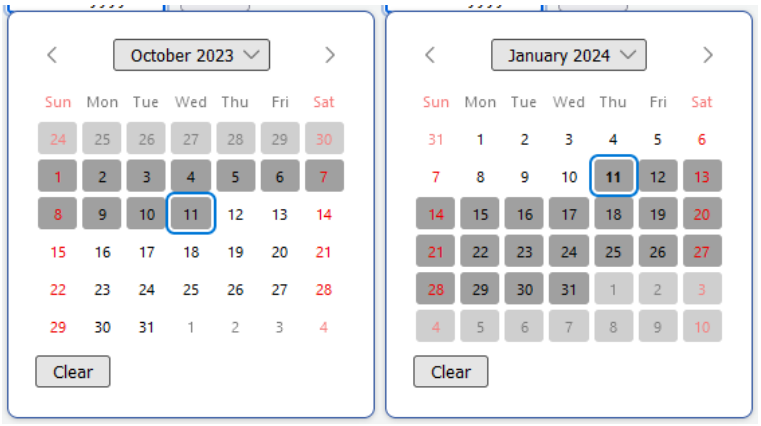
For this assignment, you are going to develop the index.html, main.js, and styles.css files.

Requirements

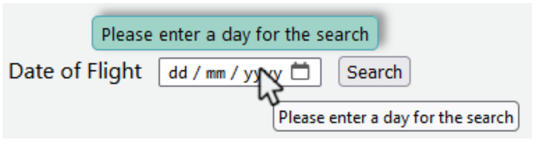
- Use AJAX (**XHR or fetch()**) technique to retrieve the Airport data and the iata.json data.

- You **should implement** appropriate CSS settings for rendering the Web app on a mobile device (with a viewport width between 350px and 500px) and a desktop browser (≥1000px).
    - For the desktop view, create a container with a width of **1200px** measured by the border-box and align the container in the middle of the window. Add the title for the app and an input form for the user to select/enter the date for the search. Align the input form in the middle of the container.
      e.g.,
      
      2023-24 COMP3322B Assignment 2
Search with Google or enter address
Hong Kong International Airport
Flight Information
Date of Flight
dd / mm / yyyy
Search
    - For the mobile view, set the container to use the full width of the device, and align the title and the input form in the middle.
      e.g.,
      

- The app only accepts a date within this range **Today-91 to Yesterday** (inclusive). For example, consider today is 2024-1-11, then the valid range is [2023-10-12 – 2024-01-10]. (Note: it is the HKAA API limitation which sets a limit on the range of the dates we can request for the data.)
    - Set the form validation attributes to guide the user in selecting an appropriate date.



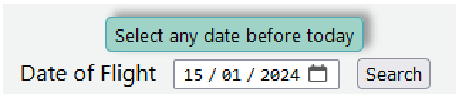
- Perform the following checking on the date entered/selected by the user via the input form.
    - If the user did not select or enter a date and clicked the "Search" button, show a message above the input form to alert the user.


Please enter a day for the search

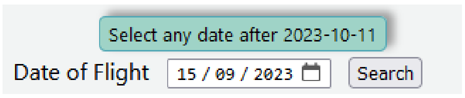
Date of Flight dd / mm / yyyy Search

Please enter a day for the search

      Please note that this alert message is generated by your program; and may not be the same as the browser's built-in validation message. You can use the Constraint Validation API to set the browser's message too, but it is optional.
    - If the user entered a date that is not valid and clicked the "Search" button, show a message above the input form to alert the user. For example, consider today is 2024-1-11


Select any date before today

Date of Flight 15 / 01 / 2024 Search


Select any date after 2023-10-11

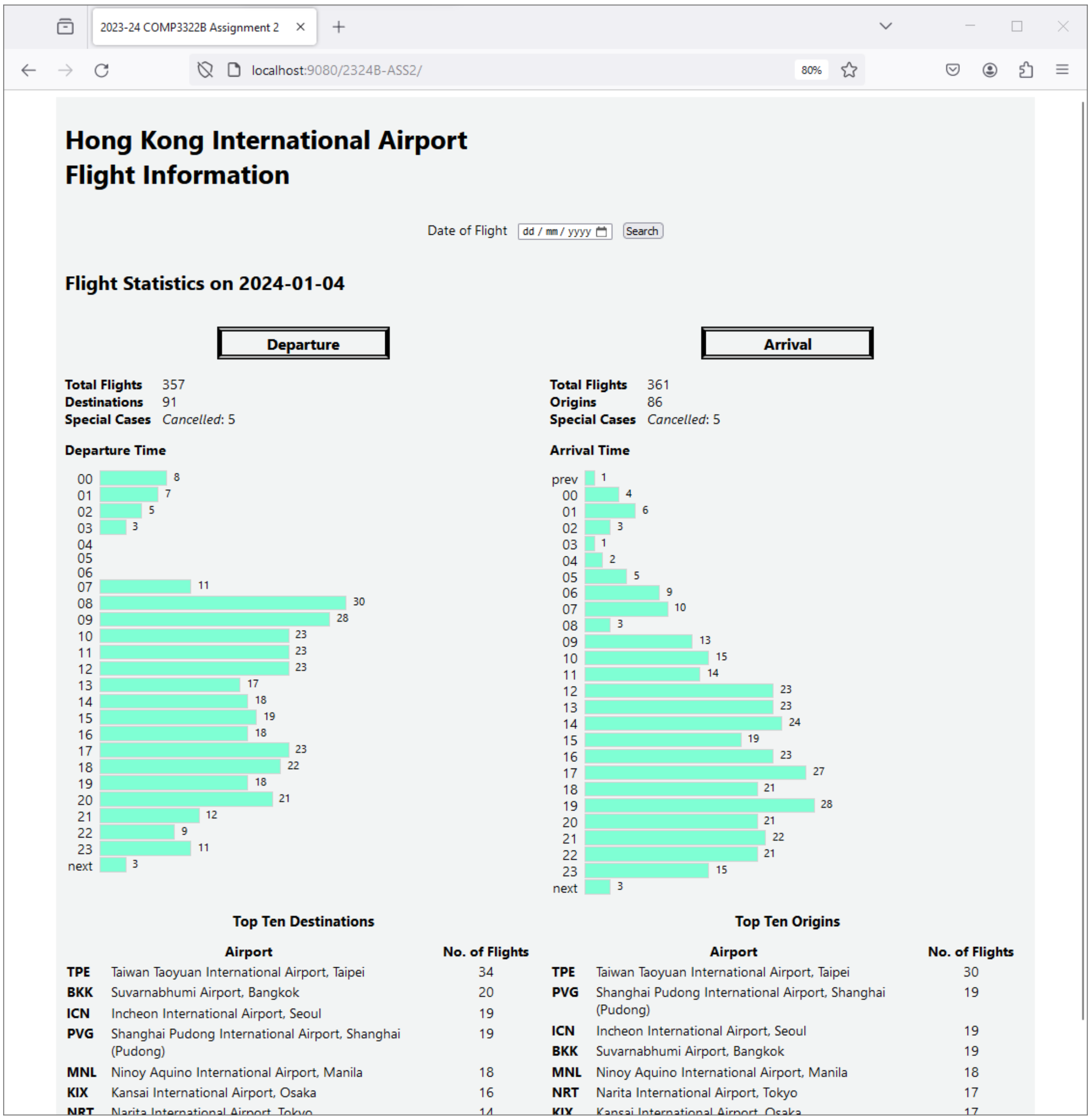
Date of Flight 15 / 09 / 2023 Search

    - When the user has entered/selected a valid date, clear any alert message (and reset the browser validation message), clear the date input field, and perform the AJAX requests for the departure and arrival data.
- After receiving the departure and arrival datasets, the app presents the following information to the user.
    - A header indicates the **date** of the datasets.
    - Total numbers of departure flights and arrival flights in the datasets.
    - Total unique destinations and origins (IATA code) in the departure and arrival datasets.
    - Count each special case in each dataset. For the departure dataset, any flights with the status not showing "Dep hh:mm" or "Dep hh:mm (dd/MM/yyyy)" are considered special cases. For the arrival dataset, any flights with the status not showing "At gate hh:mm" or "At gate hh:mm (dd/MM/yyyy)" are considered special cases. For each special case, we show the frequency of that special case, e.g. Cancelled, Delayed, …
    - The departure and arrival histograms that show the number of flights (excluding special cases) in each hour (from 00 to 23).

      In the departure dataset, it may include flights that departed not on the date of the report but departed on the next calendar date. We use the label '**next**' to represent these cases.

The arrival dataset may include flights that arrived not on the date of the report but arrived earlier on the previous calendar date. We use the label '**prev**' to represent these cases. It may also include flights that arrived on the next calendar date. We use the label '**next**' to represent these cases.

   - Display the top ten destination and origin airports and sort them in **descending** order of the number of flights. Show the IATA codes, the airport names, and their counts.



Hong Kong International Airport
Flight Information

Date of Flight dd / mm / yyyy Search

Flight Statistics on 2024-01-04

Departure

Total Flights	357
Destinations	91
Special Cases	*Cancelled*: 5

Departure Time

00	8
01	7
02	5
03	3
04	
05	
06	
07	11
08	30
09	28
10	23
11	23
12	23
13	17
14	18
15	19
16	18
17	23
18	22
19	18
20	21
21	12
22	9
23	11
next	3

Top Ten Destinations

	Airport	No. of Flights
TPE	Taiwan Taoyuan International Airport, Taipei	34
BKK	Suvarnabhumi Airport, Bangkok	20
ICN	Incheon International Airport, Seoul	19
PVG	Shanghai Pudong International Airport, Shanghai (Pudong)	19
MNL	Ninoy Aquino International Airport, Manila	18
KIX	Kansai International Airport, Osaka	16
NRT	Narita International Airport, Tokyo	14

Arrival

Total Flights	361
Origins	86
Special Cases	*Cancelled*: 5

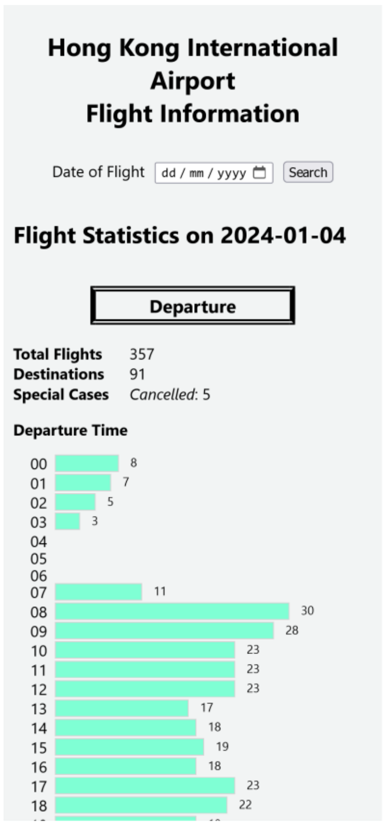
Arrival Time

prev	1
00	4
01	6
02	3
03	1
04	2
05	5
06	9
07	10
08	3
09	13
10	15
11	14
12	23
13	23
14	24
15	19
16	23
17	27
18	21
19	28
20	21
21	22
22	21
23	15
next	3

Top Ten Origins

	Airport	No. of Flights
TPE	Taiwan Taoyuan International Airport, Taipei	30
PVG	Shanghai Pudong International Airport, Shanghai (Pudong)	19
ICN	Incheon International Airport, Seoul	19
BKK	Suvarnabhumi Airport, Bangkok	19
MNL	Ninoy Aquino International Airport, Manila	18
NRT	Narita International Airport, Tokyo	17
KIX	Kansai International Airport, Osaka	17

- For the desktop view, display the departure and arrival statistics **side-by-side**. For the mobile view, display the departure statistics first followed by the arrival statistics.



Hong Kong International Airport
Flight Information

Date of Flight dd / mm / yyyy Search

Flight Statistics on 2024-01-04

Departure

Total Flights	357
Destinations	91
Special Cases	*Cancelled:* 5

Departure Time

- When the user clicks on the input field, the app should **clear** the displayed statistics.
- The base document of our Web app is the **index.html** file. You can add any HTML tags to the <body> part of the file. To display the flight statistics, you use JavaScript to dynamically create all HTML elements and their contents during runtime and use CSS and JavaScript to set the styling and layout.
- You must place all JavaScript code in an external file named **main.js**, and all styling rules in an external file named **style.css**.

Resources

Here are the required resources for building the Web app. You can download a copy via the course's Moodle page.

- iata.json – this JSON file contains more descriptive information about the airports.
- flight.php – the PHP file that acts as a client to request flight data from the HKAA API server.

Please place the flight.php program and the iata.json file under the public_html folder of your web server container.

Testing platform

We shall place all your submitted files in the LAMP container set and use Chrome and Firefox to test the programs.

Submission

Please finish this assignment before **March 18, Monday 23:59**. Submit the following files:

1. index.html
2. styles.css
3. main.js

Grading Policy

Points	Criteria
6.0	CSS styling ▪ Desktop layout (2.5) ▪ Mobile layout (1.0) ▪ Input field (0.5) ▪ Histogram (2.0)
3.5	Input validation ▪ Input checking and show appropriate alert messages (3.0) ▪ Remove any alert message and clear the input field after initiating the AJAX request (0.5)
8.5	Display the flight statistics ▪ Correct flight statistics (2.0) ▪ Correct histogram data (3.0) ▪ Correct top ten lists (3.0) ▪ Clear the statistics when the user clicks on the input field (0.5)
-1.0	Not using index.html as the Web app main page
-5.0	Use external JavaScript/CSS libraries

Plagiarism

Plagiarism is a very serious academic offence. Students should understand what constitutes plagiarism, the consequences of committing an offence of plagiarism, and how to avoid it. ***Please note that we may request you to explain to us how your program is functioning as well as we may also make use of software tools to detect software plagiarism.***