# Airport Flight Departures application

**Time 3 hours**

Create a **project solution** (named **P3<Your Name><Student Number> e.g. P3McGowanAidan6048201)).** Create a package named **p3**. Add to the solution the **departures.csv. Ensure your name and student number are placed in the Javadoc comments of all the classes you create.**

The application should run (start) from a **DeparturesControl.java** class, conditionally reading in the data from the **departures.csv** file and then performing time-related updates and several menu-driven operations.

## Part 1 – Data mapping, storage and reading from file - 50%

**Scenario**. You have been tasked to design, implement, and unit test part of a Flight Departures application. The system is expected to support several specific **Flight types** such as *Commercial*, *Private* and *Cargo*. You are the first developer of the system and are tasked with designing and implementing the **Commerical Flights**, although you are encouraged to design the system to support the future development of other specific Flight types.

**All Flight types** will have a **Flight Number**, **Airline,** and **Departure Time**. **Commercial Flights** will also include **Destination**, **Airport Code**, **Country**, **Gate**, **Boarding Status,** and **Passenger Number**.

1. Analyse the data in the **departures.csv** and implement an **OOP-based solution** to support the Airport Flight Departures application.

   **Notes**:

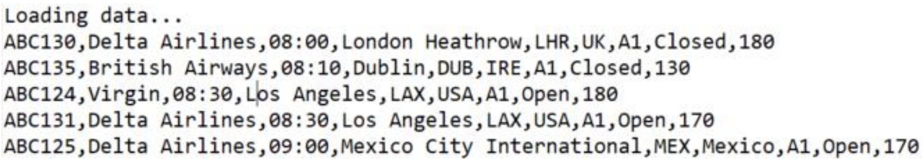
   - It shall be possible to **display all details** (print to console) for a Commercial flight. Your manager has informed you that this is a requirement for the future development of Cargo flights but never for Private flights.
   - It is a requirement to be able to **Cancel** all flight types, however, each flight type will treat the process differently.
   - **Boarding status**: allowable values are **CLOSED** (i.e. the current time is after the scheduled Departure time), **BOARDING** (within one hour inclusive of the scheduled Departure time), or **NOT STARTED** (greater than one hour from scheduled Departure time, **DELAYED**, and **CANCELLED**.

2. **UNIT TEST :** Conduct a simple unit test for the **OOP solution**. (Note there are no other validation or business rules required).

3. **READ CSV DATA** : In the **DeparturesControl.java** class read and store the data in an appropriate JCF container. It is expected that the system will contain data for a very large volume of flights with all records potentially expected to be subject to regular CRUD operations.

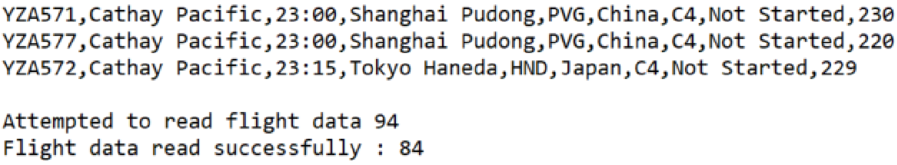
   **Note** :
   **departures.csv** contains some **Military Flights** that should not be included in the application. Military Flights have a Flight Number that starts with **M** or **m**.

   Include the following output when reading the file: (not actual data shown here).



```
Loading data...
ABC130,Delta Airlines,08:00,London Heathrow,LHR,UK,A1,Closed,180
ABC135,British Airways,08:10,Dublin,DUB,IRE,A1,Closed,130
ABC124,Virgin,08:30,Los Angeles,LAX,USA,A1,Open,180
ABC131,Delta Airlines,08:30,Los Angeles,LAX,USA,A1,Open,170
ABC125,Delta Airlines,09:00,Mexico City International,MEX,Mexico,A1,Open,170
```

…



```
YZA571,Cathay Pacific,23:00,Shanghai Pudong,PVG,China,C4,Not Started,230
YZA577,Cathay Pacific,23:00,Shanghai Pudong,PVG,China,C4,Not Started,220
YZA572,Cathay Pacific,23:15,Tokyo Haneda,HND,Japan,C4,Not Started,229

Attempted to read flight data 94
Flight data read successfully : 84
```

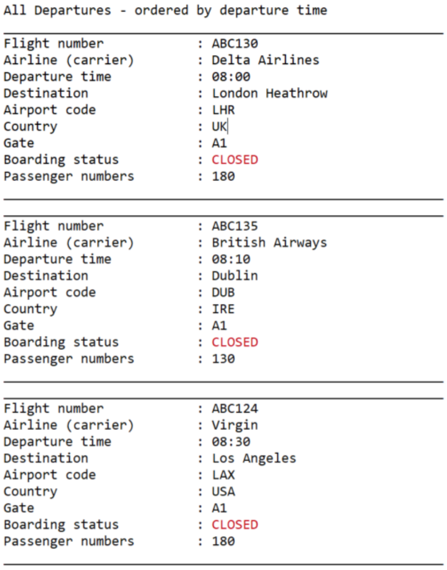
**[CONTINUED ON NEXT PAGE]**

## Part 2 – Functionality – 50%

Having read the data from the CSV file complete the following **menu-driven functions** and **time-related update** as outlined below. An example of the expected format is shown for each function. (Note if you were unable to read and store the data then creating some of your own records to represent the input data would enable you to attempt the following tasks.)

```
1. Display daily schedule departures (ordered by time) - time relative
2. Delay flight - STU904 until 22:15
3. Destination Country analysis
4. Add flight : BAA1234,British Airways,23:30,London,LHR,UK,B12,Not Started,231
5. Write to file (all flights with CLOSED boarding status )
6. Quit
Enter option ...
```

1. Display daily schedule departures (ordered by time) - time relative. Example output…



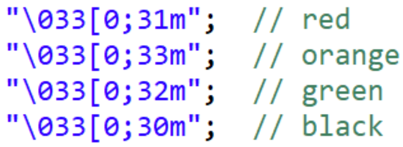
```
All Departures - ordered by departure time

Flight number            : ABC130
Airline (carrier)        : Delta Airlines
Departure time           : 08:00
Destination              : London Heathrow
Airport code             : LHR
Country                  : UK
Gate                     : A1
Boarding status          : CLOSED
Passenger numbers        : 180


Flight number            : ABC135
Airline (carrier)        : British Airways
Departure time           : 08:10
Destination              : Dublin
Airport code             : DUB
Country                  : IRE
Gate                     : A1
Boarding status          : CLOSED
Passenger numbers        : 130


Flight number            : ABC124
Airline (carrier)        : Virgin
Departure time           : 08:30
Destination              : Los Angeles
Airport code             : LAX
Country                  : USA
Gate                     : A1
Boarding status          : CLOSED
Passenger numbers        : 180

```

*etc…*

**Note** : Displaying Boarding Status should have appropriately coloured text, as shown.

```
Boarding status         : CLOSED
Boarding status         : BOARDING
Boarding status         : NOT_STARTED
Boarding status         : DELAYED
Boarding status         : CANCELLED
```

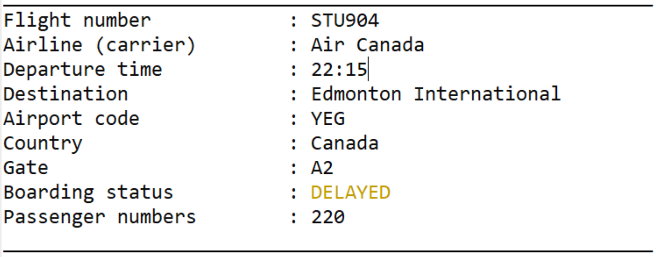
```
// use of Unicode for setting the text colour
System.out.println("\033[0;31m"); // red
System.out.println("Red text");
```



```
"\033[0;31m";  // red
"\033[0;33m";  // orange
"\033[0;32m";  // green
"\033[0;30m";  // black
```

2. **Delay flight - STU904 until 22:15**

   Update this flight and include a confirmation message for the delay, eg **`Flight STU904 delayed until 22:15`**.



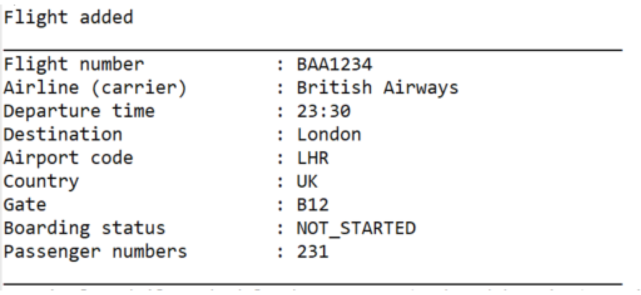
```
Flight number              : STU904
Airline (carrier)          : Air Canada
Departure time             : 22:15
Destination                : Edmonton International
Airport code               : YEG
Country                    : Canada
Gate                       : A2
Boarding status            : DELAYED
Passenger numbers          : 220
```

3. **Country destination analysis**. Show all destination countries with the number of flights planned to that Country. Order by the Country. Example output is shown (not actual answer based on CSV data provided)

```
Australia          :5
Brazil             :1
Canada             :1
China              :23
France             :8
```

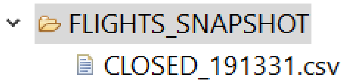
   *Etc …*

4. **Add flight : BAA1234,British Airways,23:30,London,LHR,UK,B12,Not Started,231**. Include a confirmation message e.g.

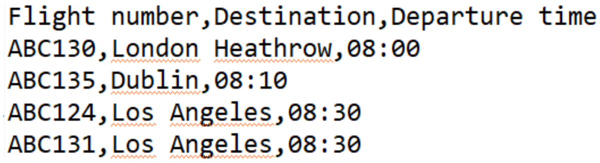


```
Flight added

Flight number              : BAA1234
Airline (carrier)          : British Airways
Departure time             : 23:30
Destination                : London
Airport code               : LHR
Country                    : UK
Gate                       : B12
Boarding status            : NOT_STARTED
Passenger numbers          : 231
```

5. Write to file all flights with Boarding Status **CLOSED**. The file should be written to a directory **FLIGHTS_SNAPSHOT** at the **root of the project**. Note any files that already exist in the directory should be removed before writing the new file. The name of the file should be formatted CLOSED_<HR><MIN><SEC>.csv. For example, a file written at 19:13:31 would be named as shown below.



*The file should be formatted as shown below*



```
Flight number,Destination,Departure time
ABC130,London Heathrow,08:00
ABC135,Dublin,08:10
ABC124,Los Angeles,08:30
ABC131,Los Angeles,08:30
```

6. Quit : Should gracefully end the application.

## Time-related updates

The system should **check the Boarding Status once every minute** updating Boarding Status as appropriate.

When complete compress (zip) the entire ***Eclipse solution*** and upload it to **Assignments** (P3 assessment) on CANVAS.

**Now : check the uploads to ensure you have submitted the correct files (in the correct area).**

**[END]**