

CS410J Project 3: Pretty Printing Your Airline (10 points¹)

In this project you will continue to develop your airline application. An airline's flights will be sorted and you will add the ability to "pretty print" the flights in an airline.

Goal: Learn about Java's facilities for sorting and working with dates.

For this project you will have to add the following functionality to your airline application:

- A `Flight`'s departure and arrival times will now be represented by instances of `java.util.Date`. You should override `AbstractFlight`'s `getArrival` and `getDeparture` methods. Your `getArrivalString` and `getDepartureString` methods should now return strings that are formatted using `DateFormat.SHORT`.
- An `Airline`'s `Flights` should be sorted alphabetically by their code of their source. Flights that depart from same airport should be sorted chronologically by their departure time. Flights that depart from the same airport at the same time are considered to be the same ("equal"). To facilitate sorting, your `Flight` class should implement the `java.lang.Comparable` interface.
- A `PrettyPrinter` class that implements the `AirlineDumper` interface creates a nicely-formatted textual presentation of an airline's flights. I suggest that you make use of `DateFormat` to make the dates look nice. You should also take advantage of the `edu.pdx.cs410J.AirportNames` class. Your pretty printer should include the duration of each flight in minutes.
- Class `Project3` that has all of the functionality of the previous project plus the ability to pretty print an airline. `Project3` should have the following command line interface:

```
usage: java edu.pdx.cs410J.<login-id>.Project3 [options] <args>
args are (in this order):
    name                The name of the airline
    flightNumber         The flight number
    src                  Three-letter code of departure airport
    departTime           Departure date time am/pm
    dest                 Three-letter code of arrival airport
    arriveTime           Arrival date time am/pm
options are (options may appear in any order):
    -pretty file         Pretty print the airline's flights to
                        a text file or standard out (file -)
    -textFile file       Where to read/write the airline info
    -print               Prints a description of the new flight
    -README              Prints a README for this project and exits
```

The three-letter airport codes given on the command line should be case **insensitive**.

The airline should be pretty printed **after** the new flight has been added.

The date and time format on the command line is no longer 24-hour time. An example of the date and time² would be 01/02/2017 9:16 pm.

¹9 for code, 1 for POA

²You might want to use a `java.text.SimpleDateFormat` to parse the date and time.

Error handling: Your program should exit “gracefully” with a user-friendly error message under all reasonable error conditions. Examples of such conditions include

- Something is missing from the command line or there are extraneous command line arguments.
- The format of the day or time is incorrect.
- The three-letter code does not correspond to a known airport.