A3 - Coldest Hour of the Day

Submit Assignment

Due Friday by 5:30pm **Points** 30 **Submitting** a file upload
Available Feb 7 at 5:30pm - Feb 16 at 5:30pm 9 days

In this assignment you will process a CSV file that contains **weather data**. The file is self-documenting - review the first line of the file to determine what the fields are.

The data file is called "weatherdata.csv". You have been taught how to place this folder in the top level folder of your project so that it can be easily opened when running your program.

The first operable field of each line is in "unix epoc time" format. You can (and should) read about this on Wikipedia (https://en.wikipedia.org/wiki/Unix_time.) It is important for you to understand that for a programmer, time is not necessarily what you think it is, and you must be aware that you'll sometimes have to make conversions from one format to another.

The basic assignment is as follows.

Create a project that contains a driver (i.e. main)and a WeatherDataPoint class.

In class WeatherDataPoint:

- define private variables for "date" and "temperature".
- provide accessors/mutators for each of the private variables.
- provide a constructor that takes a String argument
  - the string passed in will be a single line containing CSV data from the file
  - split the data into pieces and assign the appropriate fields to the respective private variables
  - see the note below concerning the "time" field
- provide a second constructor that takes a "time" and "temperature" parameter

Method "main" will be the driver for this program. In main

- create a data structure that will allow you to store an undetermined number of WeatherDataPoint objects.
- open the file and process each line in it, ignoring any invalid lines
- when you read a line, create a WeatherDataPoint object and pass the line to the constructor.
- when the object has been created, store it in your data structure.
- once the file has been completely processed, close it

You should now have a data structure that contains an unknown number of WeatherDataPoints. Each WeatherDataPoint represents a single temperature reading at a given point in time. The source data is structured in such a way that there are multiple data points (typically 5 minutes apart although some data may be missing) in a given hour of a given day.

Write an algorithm in main that scans through the data and looks for all of the data during a given day.

Within that day's entries, extract all of the data for a given hour and average that data.

Create a new WeatherDataPoint object passing in the time (of the hour in question) and the averaged data point.

Store these points in a second data structure.

Finally, scan through this second data structure and for a given day, find the point that contains the coldest hour. Output the date and temperature at which this coldest temperature occurs.

For the purposes of this assignment you will have to convert from epoch time to a simple date format. The following code that will make this conversion. This should go in the first constructor for the WeatherDataPoint.

```
    ---x-- snip ---x---

    //https://stackoverflow.com/questions/535004/unix-epoch-time-to-java-date-object

    long epoc = Long.parseLong(fields[DATE_POS]);

    Date d = new Date(epoc * 1000);

    // https://stackoverflow.com/questions/11046053/how-to-format-date-string-in-java //https://docs.oracle.com/j
avase/6/docs/api/java/text/SimpleDateFormat.html _

    date = new SimpleDateFormat("yyyy/MM/dd kk:mm:ss").format(d);

---x-- snip ---x---
```

I DO NOT ACCEPT LATE WORK

Ensure you follow all of the Code Style requirements.