C212/A592 Spring 17 Lab 10

Intro to Software Systems

Instructions:

- Review the requirements given below and Complete your work. Please submit all files through Canvas (including all input and output files).
- The grading scheme is provided on Canvas

Lab10: Input / Output File

Part1: CSV format

The CSV (or *comma-separated values*) format is commonly used for tabular data. Each table row is a line, with columns separated by commas. Items may be enclosed in quotation marks, and they contain commas and quotation marks. Quotation marks inside quoted fields are doubled. Here is a line with 4 fields:

1729, San Francisco, "Hello, World", "He asked: ""Quo vadis?"""

Implement a class CVSReader that reads a CVS file and provide methods
int numberOfRows()
int numberOfFields()
String fields(int row, int column)

Also write a driver to test you class and provide some summary / statistics of data such as Minimum, maximum or average of few columns.

Note: A sample CVS file is provided on canvas

Part2: HTML Converter (using File I/O)

The purpose of this assignment is to create a program that works like a simple html file converter. Your program will read a text file and will produce an html file that can be viewed using a web browser, like Mozilla Firefox or Internet Explorer.

Write code that converts an input file, *myfile*, into a simple Hyper Text Markup Language (HTML) file, *myfile.html*. In order to do this, you will read the input file, *myfile*, from the beginning to the end and will create the output file, *myfile.html,* by following the instructions given below.

- Every time you run your code, the first thing your program will do is to write the following 5 lines at the beginning of the output file before you start reading the input file. This is a very minimum header for your html file. You will use a function called *write_header* to do this part of the program.

```
<html>
<title>
This is my Java html converter
</title>
<body>
```

- From this point on, you will read the contents of the input file, *myfile*, and will copy it to the output file, *myfile.html*. However, as the input file is read, you will write a "<br>" (without the double-quotes) into the output file each time a newline is reached. The <br> in html means the same thing as the "\n" in Java. Continue reading and copying the rest of the file. Repeat this (as needed) until you are done reading the entire file.

  For example, suppose your input file has the following two lines:
  Line 1 on the input file: I have a black cat.
  Line 2 on the input file: I have a white dog.

  As you have noticed, at the end of the first line there is an invisible "\n". So, once you reached that, your program will write <br> into the output file. Thus, these two lines in the output file will look like this:

  I have a black cat.
  I have a white dog.

- After reading the input file and copying its entire contents into the output file, you will write the following two lines at the end of the output file. You will use a function called *write_footer* to write these lines.

  ```
  </body>
  </html>
  ```

  The output that you will generate after you run the program will be an html file. If you have done everything correctly, you can open it using a web browser like Mozilla Firefox or Internet Explorer.

Example:

A sample input file:

Hi my name is David.
I wrote this program that uses the file I/O to create html files from text files.

Sample output:
```
<html>
<title>
```

This is my java html converter, YourName.
</title>
<body>

Hi my name is David.
I wrote this program that uses the file I/O to create html files from text files.

</body>
</html>