Victoria University of Wellington
School of Engineering and Computer Science

SWEN221: Software Development
Lab Handout 5

The primary purpose of this lab is to get used to working with exceptions. By the end of the lab, the tutor should have given you a grade based on your performance. **Be sure that you have been given a grade before you leave, since attendance at labs is mandatory**.

NOTE: it is also recommended that you submit your final lab code via the online submission system, which can be found on the SWEN221 homepage. This is useful in case your lab grade is lost, or you believe you have been given the incorrect grade.

To get started, download the file `lab5.jar` from course website. This contains the following files:

```
swen221/lab5/Identifier.java
swen221/lab5/RowFileReader.java
swen221/lab5/Main.java
swen221/lab5/RowFile.java
swen221/tests/Lab5Tests.java
swen221/util/WebServer.java
```

For this lab, do not concern yourself with understanding the class `WebServer` — it is not necessary.

1 Reading from a file

The file “input.dat” contains some comma separated data — columns are separated by commas, rows by newlines. The first row is a title row. The data is a bigger version of this:

```
Name,dept,dd,yu
Nick,d,45,32
Dave,a,99,2
```

For simplicity, we can assume that, in the file “input.dat”, there should be no whitespace other than newlines. Your first task is to read in this file, create a corresponding instance of `RowFile` and display its contents to the console. A class `Main` has been provided which attempts to do this; however, you will need to complete the method `RowFileReader.read()` in order for it to work. Furthermore, since `RowFile` is an interface, you cannot directly create instances of it. Instead, you should create a class (e.g. `RowFileImpl`) which implements `RowFile` and stores the necessary information. For this part, only the methods `RowFile.getIdentifiers()`, `RowFile.getRow()` and `RowFile.addRow()` need to be implemented.

Hint: most of the code for reading a file has been provided for you, and uses the `Scanner` class which you may remember from COMP102. To complete this section, you should find the method `Integer.parseInt()` helpful. You might also want to look at Formatter for doing the output.

2 Manipulating the Data

The objective now is to complete your implementation class (i.e. `RowFileImpl`) to support all features indicated in the Javadoc, including exceptions. You should make all the tests in `Lab5Tests.java` run without errors.

To implement the method `RowFile.toHtmlTable()` you will need to need to generate a summary of the data stored in the `RowFile`. More specifically, you should sort the data alphabetically, first by 'dept', then by name (**hint:** you can use the Java libraries). You should calculate totals and averages for rows and columns.

3 Generating HTML Output

Now, modify the method `RowFile.toHtmlTable()` to output the ordered data (including header row, totals, and averages) as an HTML table. You should be able to read the output in a web browser. You can make the table as pretty as you like, using CSS or whatever. Example output for the above table is:

```
<html>
 <body>
  <table border="1">
   <tr>
    <td>Name</td>
    <td>dept</td>
    <td>dd</td>
    <td>yu</td>
    <td>total</td>
    <td>average</td>
   </tr><tr>
    <td>Dave</td>
    <td>a</td>
    <td>99</td>
    <td>2</td>
    <td>101</td>
    <td>50.5</td>
   </tr><tr>
    <td>Nick</td>
    <td>d</td>
    <td>45</td>
    <td>32</td>
    <td>77</td>
    <td>38.5</td>
   </tr><tr>
    <td>total</td>
    <td>&nbsp;</td>
    <td>144</td>
    <td>34</td>
    <td>&nbsp;</td>
    <td>&nbsp;</td>
   </tr><tr>
    <td>average</td>
    <td>&nbsp;</td>
    <td>72</td>
    <td>17</td>
    <td>&nbsp;</td>
    <td>&nbsp;</td>
   </tr>
  </table>
 </body>
</html>
```

A very simple web server is provided that uses the `RowFile.toHtmlTable()` function to read an input file and transmit its summary to a web-browser for display. To run the web server, you can use the following command:

```
java lab5.Main
```

At this stage, a web server will be running on port 8080 by default. You can see the server by pointing your web browser to the url `http://localhost:8080/input.dat`. The server will serve up summary data for files in the current directory (by default).

Marking Guide

Each lab is worth just under 1% of your overall mark for SWEN221. The lab should be marked during the lab sessions, according to the following grade scale:

- **0**: Student didn't attend lab.
- **E**: Student did not really participate in the lab.
- **D**: Student's participation was *poor.* For example, he/she made some attempt to work on the lab, but did not complete any activities.
- **C**: Student's participation was *satisfactory.* That is, he/she completed at least one activity (e.g. reading from file).
- **B**: Student's participation was *good.* That is, he/she has completed activities 2 and 3.
- **A**: Student's participation was *excellent.* That is, he/she completed all activities.