Learning Activity: Packages

Back

Educational Objectives

- Working with packages.
- Creating jar files.

Instructions

We will create a simple application that prints out a table of Celsius and Fahrenheit values.

Download the starting point code:

- Converter.java
- Double.java
- TempTable.java

Notice that all three of these classes are in named packages.

Converter.java contains one method: c2f.
Double.java contains three methods: format1, format3, and format.
TempTable.java contains one method: main.

There are JavaDoc comments for the three classes and for all the methods in these classes. Most of the methods have "dummy implementations" so that they will compile as downloaded. The application method has a brief skeleton of the code.

Very important note:
To get credit for this lab, it is necessary that the submitted source code maintain these class names:

- csc143.packages.temp.Converter
- csc143.packages.string.Double
- csc143.packages.TempTable

Remember that package names are associated with a folder structure. The implied folder structure must be present in the .jar file that is submitted.

Caution: There are IDEs that will automatically and silently alter package declarations to match the folder structure being used. Check the package declaration at the top of the source code file.

Minus Level

Use these classes to create a table like the one that follows:

```
Celsius  Fahrenheit
   0.0      32.000
  10.0      50.000
  20.0      68.000
  30.0      86.000
  40.0     104.000
```

```
 50.0     122.000
 60.0     140.000
 70.0     158.000
 80.0     176.000
 90.0     194.000
100.0     212.000
```

Add a loop to TempTable.main to generate the Celsius values. Pass that into Converter.c2f to convert the value into Fahrenheit. Use Double.format1 to create String representations of the Celsius values and Double.format3 to create String representations of the Fahrenheit values. You can use these strings to generate the output within the loop in TempTable.main.

Notes:

- The Double.format1 method shall format the double value into a string that is 6 characters wide with one digit after the decimal point.
- The Double.format3 method shall format the double parameter value into a string that is 12 characters wide with three digits after the decimal point.
- For the Minus version you can leave the "dummy implementation" in the three-parameter method Double.format.

Hint: Consider the String.format method.

Here is the formula for Celsius to Fahrenheit conversion:

$$F = 1.8C + 32$$

For reference sake, here is a Python version of the code:

```
def main():
    print('Celsius  Fahrenheit')
    for i in range(11):
        print('%6.1f%12.3f' % (i*10, i*18+32))
```

Package up the three Java source code files into a jar file, from which they can be extracted, compiled, and run.

Check Level

There are two enhancements for the Check Level:

- Make the jar file runnable.
- Complete the Double method. That is, replace the "dummy implementation" within the three parameter Double.format method.

Here are some notes about these enhancements:

Runnable jar
Needless to say, when the jar file is run, it should run TempTable.main.

Double.format
Complete this method in addition to completing the implementations of format1 and format3.
Note: It would be possible to implement the format1 and format3 methods using the three-parameter format method.

Plus work

Additionally, enhance main so that it prints out a blank line, followed by a Fahrenheit to Celsius conversion table, as shown below:

```
Celsius  Fahrenheit
   0.0       32.000
  10.0       50.000
  20.0       68.000
  30.0       86.000
  40.0      104.000
  50.0      122.000
  60.0      140.000
  70.0      158.000
  80.0      176.000
  90.0      194.000
 100.0      212.000

Fahrenheit  Celsius
    32.0      0.000
    52.0     11.111
    72.0     22.222
    92.0     33.333
   112.0     44.444
   132.0     55.556
   152.0     66.667
   172.0     77.778
   192.0     88.889
   212.0    100.000
```

Update the Converter class to include a static method f2c.

```
public static double f2c(double value)
```

Use Double.format to generate String representations for the output. Include appropriate JavaDoc comment with tags.

Submission

One jar file, including both source code (.java) and bytecode (.class) files.

Evaluation Criteria

Minus:
* The source code can be extracted and compiled as submitted.
* The application runs and produces the expected output.
* The application uses Converter.c2f, Double.format1, and Double.format3.

Check:
* Minus criteria, plus:
* The submitted jar file is runnable.
* The application uses Double.format for formatting the output.

Plus:
* Check criteria (including Minus criteria), plus:
* The application has been updated to print out a Fahrenheit to Celsius table.

Back