CS101 - Algorithms & Programming I

Fall 2024 - Lab 1

Due: Week of September 30, 2024

*Remember the **honor code** for your programming assignments.*
*For all labs, your solutions must conform to the CS101 style **guidelines**!*
All data and results should be stored in variables (or constants where appropriate) with meaningful names.

The objective of this lab is to set up & use a Java IDE, and write some basic Java programs & debugging. Remember that analyzing your problems and designing them on a piece of paper *before* starting implementation/coding is always a best practice.

0. Setup Workspace

In VSC, you organize your programs into workspaces. A Visual Studio Code workspace is basically a collection of one or more folders that are opened in a VS Code window (instance). For CS101 labs, we would like you to create a single workspace where you collect all lab solutions. First, create a folder named `labs` in your chosen folder on your hard disk. Then start VSC and open this folder (File > Open Folder...). Now in VSC, under labs, create a new folder named `lab1`. Then, save the current workspace as `labs_ws` under the folder `labs` (File > Save Workspace As...). Now you have a workspace, within which you can create folders (one for each lab) and classes for your Java programs under each folder as needed.

In this lab, you are to have three Java classes/files (under `labs/lab1/` folder) as described below.

1. Fix Compile Errors

First, download the provided `Lab01_Q1.java` from Moodle under the `lab1/` directory. When you try to compile this program, it will give errors and will fail.

Your task here is to fix the compile errors in this program to get an output *exactly* as below:

```
Hi there! Welcome to Lab 01 of CS101 in Fall 2024!
Here are some details about lab grades:

There are 9 lab sessions this semester.
Labs make up 15.0% of your total grade.
This lab:       1.6666666666666667      points
All labs:       15.0    points
Don't let the low weight fool you. Labs are extremely important!

Best of luck!
```

2. Calculate Simple Expressions

Now create a new/empty file of your own under the `lab1` folder named `Lab01_Q2.java` with a class with the same name that calculates these expressions:

$$\frac{(4.3 + 22)\times(5.1 - 7.7)}{32.2 \div 17 - 22} \qquad \frac{2^3 \times (18 - 3.5 \times 4.66)}{2^4 - 34} \qquad 3 \times (24 - 0.222)^{-1/6}$$

and outputs the results as below:

```
Expression 1 evaluates to: 3.4009947337624347
Expression 2 evaluates to: -0.7511111111111101
Expression 3 evaluates to: 3.0
```

Make sure that the type of values in these expressions are **coded exactly as provided**.

3. Calculate Areas in the World

Finally, create a third file named `Lab01_Q3` under the `lab1` directory. Based on the information below,

- The total area of the world is 510,072,000 km^2.
- 29.2% of this area is dry land and the rest is covered by water.
- The world's total area is 6,510.42 times the total area of Turkey.
- The dry land in Turkey is 76 times the water in Turkey in terms of area.

write a Java program in this file that uses the information above to answer the following questions:

- How much of the area in Turkey is dry land and how much is water?
- How much of the area of the world is dry land and how much is water?
- What percent of all dry land in the world is in Turkey?
- What percent of the water is outside Turkey?

The output should look the same (except for perhaps some minor difference due to rounding) as the following sample run:

```
Turkey has 77329 km2 dry land and 1017 km2 water.
Earth has 148941024 km2 dry land and 361130976 km2 water.
Turkey has 0.05191920796784639 percent of the Earth's dry land.
Percentage of Earth water outside Turkey is 99.99971838472256 percent.
```

Assume the total areas of the earth and Turkey are constant values and declare them accordingly. However, the percentage of dry land or water may change over time (during the execution of your program in a future version) so they should be declared using variables for any potential changes.

Also, notice that integers cannot hold very large numbers; `long` variables might not be sufficient to hold certain values in this program either. So you might have to use `double`s. Finally, be careful of typecasting as we are not interested in the precision of the land area and long integers cannot hold fractions.