"Motivated" Practice Exercises [Part 2]

Start Assignment

Due Sep 19 by 11:59pm **Points** 18 **Submitting** a website url **Available** after Sep 6 at 12am

Overview

This assignment is intended to give you further practice with creating IntelliJ projects and writing (very) simple programs.

Directions

You **shall** create solutions for Exercises 6 though 23, inclusive, as listed on the **Programming Exercises** page.

You **shall** create one GitHub repository for *all* exercises and ensure that your solution code has been pushed to this repository prior to the submission of this assignment.

- Your GitHub repository **shall** follow the naming convention of `<lastname>-a02`

You **shall** create one baseline solution for each exercise which incorporates *all* associated constraints

- Each baseline solution **shall** be created as an IntelliJ project (using Gradle with JDK 16)
- Each baseline project **shall** be named `exercise<nn>`
- Each baseline project **shall** contain a class named `Solution<nn>` that serves as the application entry point
  - As an example, for exercise 8 I would create a GitHub repo named `hollander-a02`, a project named `exercise08` and an application class called `Solution08`
  - As an example, for exercise 18 I would create a GitHub repo named `hollander-a02`, a project named `exercise18` and an application class called `Solution18`

You **may** create one additional solution for each challenge component.

- Each challenge solution **shall** be in the same repository as the associated exercise, but within its own project.
- Each challenge solution **shall** be created as an IntelliJ project (using Gradle with JDK 16)
- Each challenge project **shall** be named `exercise<nn>-challenge<uu>`, where `uu` is replaced by the challenge number with a leading 0.
- The baseline project **shall** contain a class named `Challenge<nn>` that serves as the application entry point
  - As an example, for exercise 10, challenge 1, within the GitHub repo named `hollander-a02`, I would create a project named `exercise10-challenge01` and an application class called `Challenge01`

You **shall** push at least 3 commits to your GitHub repository with at least 1 commit pushed per day over a 3 day period.

- You **shall** push your files to GitHub using an appropriate git client.

    - The "Add file" button on the GitHub website is **not** an appropriate client.
    - Attempting to manually add files to a repository will most likely result in de-synchronizing your local and remote repos and cause you unnecessary headache.
- This requirement can only be satisfied if you spend at least 3 days working on this assignment.
    - *Example*: You complete on problems 6-11 on Tuesday and push them to GitHub; you complete problems 12-17 on Thursday and push them to GitHub; finally you complete problems 18-23 on Sunday and push them to GitHub.
    - My personal suggestion is to complete on 3 problems per day for 6 consecutive days.

You **shall** create a **.gitignore** file to ensure that your **build** folder and your **.gradle** folder are not stored within git.

You **shall** include the following comment at the top of *each* of your *.java files (with the appropriate name substituted in place):

```
/*
 *  UCF COP3330 Fall 2021 Assignment 1 Solution
 *  Copyright 2021 first_name last_name
 */
```

You **shall** write your solution in both pseudocode *and* java

- Your pseudocode **shall** be provided within each corresponding .java file

You **shall** follow good programming practices when solving these exercises.

- You **shall** use "clean" identifiers (refer to **Module 04 - The Anatomy of a Class**)
- You **shall** decompose your solution into multiple "clean" methods (refer to **Module 04 - The Anatomy of a Class**)
    - Points will be deducted if your entire solution exists within a `main` method
- You **shall** follow one of the following Java Style Guides with only minor deviations when it makes sense:
    - **https://google.github.io/styleguide/javaguide.html (https://google.github.io/styleguide/javaguide.html)**
    - **https://www.oracle.com/java/technologies/javase/codeconventions-contents.html (https://www.oracle.com/java/technologies/javase/codeconventions-contents.html)**
- You **shall** install the *SonarLint* plugin for IntelliJ and modify your code to remove any major warnings or style issues
    - SonarLint rules **shall** override your chosen style guide
    - Some exceptions are allowable; e.g. you do not need to replace print statements with a logger

You **should** try and incorporate automated unit testing into your solutions with the help of JUnit 5. You will most likely need to do a bit of research to figure out how to do this. This is intentional, as it will better prepare you employment after graduation (as well as make your future classes a bit easier). The TAs should also be able to help you if you visit them during office hours.

You will find the following links helpful for working with Git and Gradle inside of IntelliJ:

- **https://www.jetbrains.com/help/idea/gradle.html (https://www.jetbrains.com/help/idea/gradle.html)**
- **https://www.jetbrains.com/help/idea/using-git-integration.html (https://www.jetbrains.com/help/idea/using-git-integration.html)**

- https://www.jetbrains.com/help/idea/github.html (https://www.jetbrains.com/help/idea/github.html)

Submission

You will submit a link to your GitHub page (e.g. https://github.com/drhollander (https://github.com/drhollander)). Your repositories should be private prior to the assignment due date, but **must** be public within 24 hours of the assignment due date.

If we cannot access your repositories, or if you provide an invalid link, you will receive a 0 - *please double check your submission once it has been made*.

Late submissions will receive a 0 as per the syllabus.

Grading Criteria

This assignment will be graded strictly as pass or fail. You will receive 1 point per **working** solution that is present in your GitHub repository.

- The graders will run a random subset of your exercises, either as a normal Java program or against a set of test cases.
- If you fail to satisfy any of the "shall" statements, points will be deducted from your total grade.
- If your exercise code fails to compile or generates an incorrect answer (minor formatting differences aside), you will not receive credit for that solution *and* the grader will then look at the rest of your exercises for further problems.