INTRODUCTION TO JAVA, MVN, AND GIT: LOC Counting

Assignment Kit

Requirements

Program 1 requirements

Use Maven, Git, JUnit, and GITHUB.

Write a program to Count the Lines of Code of a source file. The Lines Of Code (LOC) of a program are often used as software metric to compare complexity and to measure productivity. For example, suppose you spend 6 hours to complete this assignment, including design, tests, documentation, repository management and delivery. No suppose that your code has 60 LOC. Then your productivity for this project is 10 LOC/h. Note that this metric considers the full development cycle.

Register the time expend in this project and compute the LOC/h.

Your program should be a command line program that receives a three letters parameter and the name of a file with the source code. It then outputs the number of lines of the source program.

For example, a typical invocation will be:
> countlines phy countline.java

If the parameter of the invocation is "phy" the program prints the physical lines of the source code.

If the parameter is "loc" the program will print the Lines of Code found. This is the physical lines without the comments and white lines.

Modify your program to take into account wildcards and recursive selection of files within a directory.

Follow the coding principles found in (https://google.github.io/styleguide/javaguide.html).

Thoroughly test the program. Test the program with two source files and write a report of tests including the original files. Use Junit Tests.

Document your design using the Design Metaphor document. This document includes at Least:

- Title
- Author
- Date
- Class diagram
- Description of class diagram

Document your project in the Readme.md file in the repository using this guide:
https://gist.github.com/PurpleBooth/109311bb0361f32d87a2

Submitting your assignment

Submit the link to the project on Github.

The repository should contain:

- The source code
- The Readme describing the project. Include the LOC/h.

- Test report
- Your design
- The Javadoc.
