Software Dev. & Problem Solving I GCIS-123

Git Basics Homework 1.2

Goals of the Assignment

- The use of version control is an essential part of good software engineering practice, and this semester we will be using Git during every lecture and for every assignment. The primary goal of this homework assignment is to give you more practice with Git so that you will become more comfortable with its use.
- You will practice using several Git commands including `clone` (optional), `status`, `add`, `commit`, `push`, and `log`.
- You will see how Git handles deleted files and empty directories.
- You will use a browser to connect to your GitHub repository to verify that changes that you have pushed are correctly represented in the repository.

Tips for Success

- ***Do not procrastinate***. Working on this assignment while the material is fresh in your mind will help you to retain the new information. Furthermore, the more that you procrastinate, the more otherwise small problems will become insurmountable obstacles.
- ***Read the entire assignment carefully*** before seeking help from the course staff!
- Spend a few minutes skimming the lecture slides before attempting any of the problems in this assignment. It's a good idea to refresh your memory.
- Keep the slides open in a browser or PDF viewer while you work on the assignment so that you can quickly and easily refer back to them. If you get stuck on any of the problems in this assignment, find the matching slides or activities from the lecture!
- You are *strongly encouraged* to use the command line to complete all of the activities in the assignment. Proficient users of the command line find it a lot faster and more efficient than using a graphical tool like VS Code for version control. Of course, the command line may seem slower *at first*, but with practice, you will get more comfortable.
- Making and overcoming mistakes is expected and a great learning experience! You will learn more and retain the information better if you take a few minutes to try to find the answers to your own questions before asking for help. The lecture slides are a great resource and contain everything that you need to complete this assignment!
- Don't forget that you can post questions that you may have to the assignment channel on the course Discord server and/or ask an on-duty Course Assistant for help during their scheduled mentoring hours.

Activities

1. If you are working on a computer that is different from the one that you used in class, you will need to clone your repository onto the new computer:

    a. Open the repository in your browser (if you need to, you can find it by clicking the original GitHub Classroom Invitation again).

    b. Create a directory in your user directory: `SoftDevI\Unit01\`

    c. Change into the directory and use `git clone` with your repository URL to download the repository to your new computer.

2. Download and save this PDF file in the repository.

3. Use notepad to create a new file called “`questions.txt`” and type your full name on the first line of the file. As you edit the file, make sure to include a blank line each time you add text to the file (e.g. answer a question) so that it is easy to tell when one answer ends and the next begins. Make sure to save the file each time you type an answer.

4. Run `git status` and copy/paste the output into a file called “`status.txt`”.

5. Use the Git workflow (`status`, `add`, `commit`, `push`) to push the new files into your repository. Note that, because the PDF file name contains spaces, you will need to use quotes to stage it, e.g. `git add “A filename with spaces.txt”` *Hint*: use tab to complete the filename.

    a. Remember, if you forget to use `-m` when committing, you will need to use ***vim*** to enter a comment. Type **`i`** (to insert) and type your comment on the first line. Then press **`ESC`** to open the vim command prompt (at the bottom of the window). Type **`:x`** into the prompt to save and quit.

6. Use your browser to open your repository on https://www.github.com and verify that the files were pushed.

7. Create a new, empty directory inside the `repository` directory using your first name, e.g. "`SoftDevI\Week01\assignment-1-2-harry\Harry`".

8. Use the Git workflow to push the empty directory to GitHub, and check your repository in a browser. What happened? Edit your “`questions.txt`” and type your answer. Remember to leave a blank line after your name.

9. Move (do not copy) your “`questions.txt`” file into the directory that you created in the previous step. Use the Git workflow to push the moved file to GitHub, and check your repository in a browser. What happened? Edit your “`questions.txt`” and type your answer. Is your new directory present? How many copies of the “`questions.txt`” file are present? Explain why you think it happened (feel free to Google).

10. Change into the top directory for your Assignment 1.2 repository (the one with the .git directory and most of your files) and check the status of your files. You should see that the original “`questions.txt`” file has been deleted but is listed under “Changes not staged.” This is the way that Git works: even when you ***delete*** files, you must use the Git workflow to delete them from your repository. Do so now, and then verify that the old “`questions.txt`” file has been removed from your repository on GitHub.

11. Don’t forget to push ***all*** of your files to GitHub. Use a browser to open your repository on https://www.github.com and verify that all of your files are there.

Submission Instructions & Grading

The in-class activities and assignments in Software Development & Problem Solving have been designed to give you an opportunity to practice what you have learned in class and to identify potential gaps in your understanding so that you can seek help from your instructor or the Course Assistants.

Future activities and assignments will be created with the assumption that you have completed earlier assignments, and will therefore build on the learning outcomes from previous assignments. For this reason, it is important that you keep up with the timeline of assignments. Falling behind may result in finding the assignments and activities that seem to be more difficult than they are intended to be,

See the course syllabus for the rubric that is used for grading homework assignments.