Objectives

To continue to learn how to use GitHub to access the files for a practical assignment. Additionally, to keep learning how to use the Ubuntu operating system and programs such as a “terminal window”, the Atom and GVim text editors, and linting tools. You will also continue to practice using Slack to support communication with the teaching assistants and the course instructor. Next, you will practice fixing HTML and CSS files with content that incorrectly creates and styles tabular data. In addition to using the grading tool to assess your progress towards correctly completing the project, you will continue to run a web server and use its output to diagnose source code problems.

Suggestions for Success

• Use the laboratory computers. The computers in this laboratory feature specialized software for completing this course’s laboratory and practical assignments. If it is necessary for you to work on a different machine, be sure to regularly transfer your work to a laboratory machine so that you can check its correctness. If you cannot use a laboratory computer and you need help with the configuration of your own laptop, then please carefully explain its setup to a teaching assistant or the course instructor when you are asking questions.

• Follow each step carefully. Slowly read each sentence in the assignment sheet, making sure that you precisely follow each instruction. Take notes about each step that you attempt, recording your questions and ideas and the challenges that you faced. If you are stuck, then please tell a teaching assistant or instructor what assignment step you recently completed.

• Regularly ask and answer questions. Please log into Slack at the start of a laboratory or practical session and then join the appropriate channel. If you have a question about one of the steps in an assignment, then you can post it to the designated channel. Or, you can ask a student sitting next to you or talk with a teaching assistant or the course instructor.

• Store your files in GitHub. Continuing with this laboratory assignment, you will be responsible for storing all of your files (e.g., JavaScript code and Markdown-based writing) in a Git repository using GitHub Classroom. Please verify that you have saved your source code in your Git repository by using “git status” to ensure that everything is updated. You can see if your assignment submission meets the established correctness requirements by using the provided checking tools on your local computer and in checking the commits in GitHub.

• Keep all of your files. Don’t delete your programs, output files, and written reports after you submit them through GitHub; you will need them again when you study for the quizzes and examinations and work on the other laboratory, practical, and final project assignments.

• Back up your files regularly. All of your files are regularly backed-up to the servers in the Department of Computer Science and, if you commit your files regularly, stored on GitHub. However, you may want to use a flash drive, Google Drive, or your favorite backup method to keep an extra copy of your files on reserve. In the event of any type of system failure, you are responsible for ensuring that you have access to a recent backup copy of all your files.
• **Explore teamwork and technologies.** While certain aspects of the laboratory assignments will be challenging for you, each part is designed to give you the opportunity to learn both fundamental concepts in the field of computer science and explore advanced technologies that are commonly employed at a wide variety of companies. To explore and develop new ideas, you should regularly communicate with your team and/or the teaching assistants and tutors.

• **Hone your technical writing skills.** Computer science assignments require you to write technical documentation and descriptions of your experiences when completing each task. Take extra care to ensure that your writing is interesting and both grammatically and technically correct, remembering that computer scientists must effectively communicate and collaborate with their team members and the tutors, teaching assistants, and course instructor.

• **Review the Honor Code on the syllabus.** While you may discuss your assignments with others, copying source code or writing is a violation of Allegheny College’s Honor Code.

**Reading Assignment**

If you have not done so already, please read all of the relevant “GitHub Guides”, available at https://guides.github.com/, that explain how to use many of the features that GitHub provides. In particular, please make sure that you have read guides such as “Mastering Markdown” and “Documenting Your Projects on GitHub”; each of them will help you to understand how to use both GitHub and GitHub Classroom. To do well on this assignment, you should also read Chapters 1 through 5 in the textbook, paying attention to the class slides and Sections 5.1 and 5.2. Please see the instructor or one of the teaching assistants if you have questions these reading assignments.

**Accessing the Practical Assignment on GitHub**

To access the practical assignment, you should go into the #announcements channel in our Slack team and find the announcement that provides a link for it. Copy this link and paste it into your web browser. Now, you should accept the practical assignment and see that GitHub Classroom created a new GitHub repository for you to access the assignment’s starting materials and to store the completed version of your assignment. Specifically, to access your new repository for this assignment, please click the green “Accept” button and then click the link that is prefaced with the label “Your assignment has been created here”. If you accepted the assignment and correctly followed these steps, you should have created a repository with a name like “Allegheny-Computer-Science-103-Spring-2018/computer-science-103-spring-2018-practical-5-gkapfham”. Unless you provide the course instructor with documentation of the extenuating circumstances that you are facing, not accepting the assignment means that you automatically receive a failing grade for it.

Before you move to the next step of this assignment, please make sure that you read all of the content on the web site for your new GitHub repository, paying close attention to the technical details about the commands that you will type and the output that your program must produce. Now you are ready to download the starting materials to your laboratory computer. Click the “Clone or download” button and, after ensuring that you have selected “Clone with SSH”, please copy this command to your clipboard. To enter your course directory you should now type “cd cs103S2018”. Next, you can type the command “ls” and see that there are files or directories inside of this directory. By typing “git clone” in your terminal and then pasting in the string that you copied from the GitHub site you will download all of the code for this assignment. For instance, if the course instructor ran the “git clone” command in the terminal, it would look like:

```
git clone git@github.com:Allegheny-Computer-Science-103-S2018/computer-science-103-spring-2018-practical-5-gkapfham.git
```
Repairing Mistakes in the HTML and CSS for a Web Site

This practical assignment invites you to repair a web site that contains incorrect HTML and CSS source code. You should use the “htmlhint src/www/index.html” command and the GatorGrader tool to identify and fix all of the mistakes in the provided source code files. Figure 1 gives some examples of how the correct source code should look in the provided files. Please remember that a “linting” tool is unlikely to point out all of the problems in the source code — you will need to study the provided source code and your textbook to carefully determine the root cause of each defect. For example, you should check that the HTML tags match and the CSS rules are correct.

Now, you are ready to practice running a web server to make your web site available in a browser. You can now run your web server by typing the command “serve src/www 4250”. At this point, you can start your web browser and go to the web site http://localhost:4250/. Does your new web site look correct? Can you see the required table? Are all of the styles and fonts correctly applied? If not, then continue to edit and convert it until the files are correct. Since this is a challenging practical assignment and you are still learning how to debug complex HTML and CSS files, don’t become frustrated if you make a mistake. Instead, use your mistakes as an opportunity for learning both about the necessary web development technology and the background and expertise of the other students in the class, the teaching assistants, and the course instructor.

Checking the Correctness of Your Web Site

The Markdown file that contains your reflection must adhere to the standards described in the Markdown Syntax Guide https://guides.github.com/features/mastering-markdown/. Finally, your index.html file should adhere to the HTML standards established by the “HTML linting” tool available at http://htmlhint.com/. Instead of requiring you to manually check that your deliverables adhere to these industry-accepted standards, the GatorGrader tool that you will use in this practical assignment makes it easy for you to automatically check if your submission meets the standards for correctness. For instance, GatorGrader will check to ensure that index.html creates
To get started with the use of GatorGrader, type the command "./gatorgrader.sh --start" in your terminal window. Once this step completes you can type "./gatorgrader.sh --check". If your work does not meet all of the assignment’s requirements, then you will see the following output in your terminal: “Overall, are there any mistakes in the assignment? Yes”. If you do have mistakes in your assignment, then you will need to review GatorGrader’s output, find the mistake, and try to fix it. Once your static web site displays correctly, fulfilling at least some of the assignment’s requirements, you should transfer your files to GitHub using the “git commit” and “git push” commands. For example, if you want to signal that the src/www/index.html file has been changed and is ready for transfer to GitHub you would first type “git commit src/www/answers.md -m "Your descriptive commit message"” in your terminal, followed by typing “git push” and checking to see that the transfer to GitHub is successful. If you notice that transferring your code or writing to GitHub did not work correctly, then please read the messages in your terminal and try to determine why, asking a teaching assistant or the course instructor for assistance, if necessary.

After the course instructor enables “continuous integration” with a system called Travis CI, when you use the “git push” command to transfer your source code to your GitHub repository, Travis CI will initialize a “build” of your assignment, checking to see if it meets all of the requirements. If both your source code and writing meet all of the established requirements, then you will see a green ✔ in the listing of commits in GitHub after awhile. If your submission does not meet the requirements, a red ✗ will appear instead. The instructor will reduce a student’s grade for this assignment if the red ✗ appears on the last commit in GitHub immediately before the assignment’s due date. Yet, if the green ✔ appears on the last commit in your GitHub repository, then you satisfied all of the main checks, thereby allowing the course instructor to evaluate other aspects of your source code and writing, as further described in the “Evaluation” section of this assignment sheet. Unless you provide the instructor with documentation of the extenuating circumstances that you are facing, no late work will be considered towards your grade for this practical assignment.

Summary of the Required Deliverables

Students do not need to submit printed source code or technical writing for any assignment in this course. Instead, this assignment invites you to submit, using GitHub, the following deliverables.

1. A properly formatted and correct version of src/www/index.html that both meets all of the established requirements and produces the correct HTML and the desired static web site.

2. A properly formatted and correct version of all of the CSS files (e.g., site.css and table.css) that correctly style the HTML tags and produce a web site meeting the requirements.

Evaluation of Your Practical Assignment

Using a report that the instructor shares with you through the commit log in GitHub, you will privately received a grade on this assignment and feedback on your submitted deliverables. Your grade for the assignment will be a function of the whether or not it was submitted in a timely fashion and if your program received a green ✔ indicating that it met all of the requirements. Other factors will also influence your final grade on the assignment. In addition to studying the efficiency and effectiveness of your Markdown source code, the instructor will also evaluate the accuracy of both your writing and the constructs in your source code. If your submission receives a red ✗, the instructor will reduce your grade for the assignment while still considering the regularity with which you committed to your GitHub repository and the overall quality of your partially completed work. Please see the instructor if you have questions about the evaluation of this practical assignment.