
1 Assignment 5 Introduction

This is a noncoding tutorial assignment, and **will not be graded**. However, you are still expected to complete this assignment before 11:59pm on the listed date, as there is a flash quiz associated with this assignment that **will be graded**. The flash quiz will be released on the day this assignment is due.

Assignments due on Mondays generally involve materials covered in lectures from the previous week, whereas assignments due on Thursdays involve materials covered in lectures from the running week. So be sure to watch the lectures and go over the reading materials before attempting the assignments.

Good luck!

2 Version Control using git on GitHub (0 points)

Introduction:

One of the most important tools every software developer uses on a daily basis is version control. Version control, also known as source control, is the practice of tracking and managing changes to software code base (i.e., repositories). Version control systems are software tools that help manage changes to source code over time in a collaborative setting. The most popular of the version control systems is ‘git’, and GitHub is the most popular online hosting site for git repositories.

Upon completing this tutorial assignment, you will have learned the basics of creating, modifying, and managing your own git repositories.

Initial setup:

- First, make a free GitHub account on [GitHub.com](https://github.com), preferably using your MSU email.
- Afterwards, complete the ‘Hello World’ introductory tutorial up to Step 1. If you have done this step correctly, you should see your repository under the ‘Dashboard’ on the main page.

Cloning a repository:

- Create a new **private** repository.
- Enter your repository, click the green <> **code** button, and select the **HTTPS** option.
- Copy the **HTTPS URL** to your clipboard for later use.
- Switch to your local command line, and install git (if you do not have it installed already) following the **Git Installation Guide**.
- Create and **cd** into the directory where you want a local copy of your repository.
- Run **git clone https...** from the command line, where the link is the **HTTPS URL** you copied earlier from your GitHub repository.
- You should be prompted for your GitHub username and a personal access token. Your personal access token is **not** your GitHub password. You can obtain your personal access token by following the **Guide on Managing Access Tokens**. Be sure to enable ‘repo’ permissions when creating your personal access token.

Configuring git:

- Run `git config --global push.default simple`.
- Run `git config --global user.name "Your Name Here"`, where 'Your Name Here' should be replaced with your name.
- Run `git config --global user.email "your_id_here@msu.edu"` where 'your_id_here@msu.edu' should be replaced with your email.

Making changes to a repository:

- Edit the `README.md` file (or create it, if it does not exist), and add your name to the file as the author. Save your changes.
- Run `git status` to see any changes made to the repository. It should inform you that the `README.md` file has been changed.
- Run `git add README.md` to instruct git to track that change.
- Run `git commit -m "Changed the README"` to instruct git to save the changes to history. You can replace the commit message with anything you like.
- Run `git push` to instruct git to push the changes to the remote repository on GitHub.
- If you refresh the repository page on the GitHub website, you should now see the changes you made to the `README.md` file.
- Run `git fetch` to instruct git to sync any changes that have been made on the remote repository.

Adding code to repository:

- Write a 'Hello World' program, and save it as `main.cpp`.
- Put the `main.cpp` file inside your local repository.
- Run `git status` to confirm your `main.cpp` file was detected by git.
- Run `git add main.cpp`, and then run `git commit -m "Added main.cpp"` to add and commit your `main.cpp` file to the repository.
- Run `git push` to push the changes to the remote repository on GitHub.
- If you refresh the repository page on the GitHub website, you should now see your `main.cpp` file there.

3 Assignment 5 Submission Process

- Even though this assignment will not be graded, you are still required to submit a dummy zip file in order to have access to the flash quiz. You can create a dummy zip file by simply compressing an empty folder or an empty text file.
- Submit the zip file through D2L.