WDI Fundamentals Unit 2

Git Cheat Sheet

Git

- An open source program for tracking changes in text files. It was written by the author of the Linux operating system, and is the core technology that GitHub, the social and user interface, is built on top of.

Commit

- An individual change to a file (or set of files).
- It's like when you save a file, except with Git, every time you save it creates a unique ID (a.k.a. the "SHA" or "hash") that allows you to keep record of what changes were made when and by who.
- Commits usually contain a commit message which is a brief description of what changes were made.
- *Synonyms*: a revision

Diff

- A diff is the difference in changes between two commits, or saved changes.
- The diff will visually describe what was added or removed from a file since its last commit.

Remote

- The version of something that is hosted on a server, most likely GitHub.com. It can be connected to local clones so that changes can be synced.

Repository

- The most basic element of Git.
- A repository is a project's folder, containing all of the project files (including documentation), and stores each file's revision history. Repositories can have multiple collaborators and can be either public or private.

Fork

- A personal copy of another user's repository that lives on your account.
- Forks allow you to freely make changes to a project without affecting the original.
- Forks remain attached to the original, allowing you to submit a pull request to the original's author to update with your changes.

Clone

- A copy of a repository that lives on your computer instead of on a website's server somewhere, or the act of making that copy.
- With your clone you can edit the files in your preferred editor and use Git to keep track of your changes without having to be online.
- It is, however, connected to the remote version so that changes can be synced between the two.
- You can push your local changes to the remote to keep them synced when you're online.

Push

- Pushing refers to sending your committed changes to a remote repository such as GitHub.com.
- For instance, if you change something locally, you'd want to then push those changes so that others may access them.

Command	Description
`git init`	Tells Git to start monitoring the current folder you're in. In other words, for my working directory, create a new "timeline" where we can manage our source code.
`git status`	Get the current status of Git. Files that are "staged" (about to be committed), and files that are unstaged (files that have changed since the last commit, but are not about to be committed) will both show up here

Command	Description
`git add path/to/directory/or/file`	Add a file to the "stage". The stage can be thought of as an inbetween state between the last commit and what is ready to be committed. Once the command `git commit` is run, all staged files will be committed to the timeline.
`git commit -m "Commit message"`	Commit all staged files to the timeline. If `-m "Commit messsage"` is ommitted, Git will open your default text editor (typically Vim) to enter a longer message. If for any reason Vim is opened, you can close it by typing `:q`.
`git log`	Visualize the timeline. You can scroll with the arrow keys or `j`, `k`, and it can be exited by typing `q`.
`git diff path/to/directory/or/file`	Show the changes of the given file or directory.
`git clone http://path/to/repo`	Create a new local git repo copied from a remote one
`git push origin master`	Send local changes to tracked remote repository