**GIT BASICS**

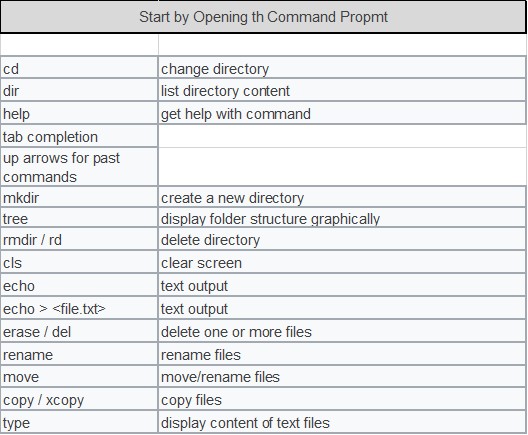
* Git = Version Control System (VCS)
* History documented through commits
* Commits can be linked through branches.
* Git = Language used by process
* GitHub = Website

**GIT COMMANDS**

* <git init> initializes a new repo
* <git clone> creates a local copy of a remote project
* <git add> adds change to staging area
* <git commit> saves snapshot in staging area
* <git status> shows status of changes
* <git push> updates remote repo

**UNIX vs Windows Commands**

Reference the below: <https://www.geeksforgeeks.org/linux-vs-windows-commands/>

**Navigating Directories: Windows**

**TEXT EDITORS**

## [Using Visual Studio Code as your editor](https://docs.github.com/en/get-started/getting-started-with-git/associating-text-editors-with-git#using-visual-studio-code-as-your-editor)

1. Install [Visual Studio Code](https://code.visualstudio.com/) (VS Code). For more information, see "[Setting up VS Code](https://code.visualstudio.com/Docs/setup/setup-overview)" in the VS Code documentation.
2. Open Git Bash.
3. Type this command:

git config --global core.editor "code --wait"

**Git and GitHub Basics**

* git init <name> - creates a new repository within the current directory
* touch readme.md
* git add – upload uncommitted files to staging area
* git status – lets us see status of files in repo(uploaded etc.)
* git diff – shows differences in files between working directory and staging area
* git commit -m “Initial Commit”
* git log – shows comment history

[**https://docs.github.com/en/get-started/getting-started-with-git/associating-text-editors-with-git**](https://docs.github.com/en/get-started/getting-started-with-git/associating-text-editors-with-git)

**Command to configure text editor for VS code**

git config --global core.editor "code --wait"

**Creating repository on website:**

* Hit the “+” in upper-right corner
* Select “New Repository”
* Enter repository with name - Choose to initialize with or without a readme

**Connect local to remote repository**

git remote add origin “*Enter GitHub HTTPS link here”*

git push -u origin master

**Git Hub Flow**

**https://docs.github.com/en/get-started/using-github/github-flow**

**Branches**

* Branches isolate work without affecting other branches
* There’s one default branch, but you can have as many as you like/need

**Branching**

* git branch *BranchName* – creates new branch
* git branch – shows all branches
* git status – shows if branch is up to date with master
* git checkout *BranchName* – switches to branch
* git push origin *BranchName* – pushes to remote branch
* Click on branch dropdown in GitHub to review different branches in repository
* git branch -d *BranchName* – generally do not use to keep versions of code in different branches

**Merging**

* When merging, return to the master branch using git checkout master
* git merge *BranchName* – merges branches from branch name to current selected branch
* git push origin master – uploads repository changes to branch remote branch

**Cloning**

* two ways to create repositories – git init and git clone
* cloning – complete copy of original repository
* git clone <*github.com address>*
* Cloning a repo, you can’t push changes to GitHub like you would with your created repos
* Default *remote* is the cloned repo