## **GIT BASICS**

### **Introduction**

In this lesson, we’ll cover common Git commands used to manage your projects and to upload your work onto GitHub. We refer to these commands as the basic Git workflow. When you’re using Git, these are the commands that you’ll use 70-80% of the time, so if you can get these down, you’ll be more than halfway done mastering Git!

### **Learning Outcomes**

By the end of this lesson, you should be able to do the following:

* Describe how to copy an existing repository from GitHub onto your local machine.
* Explain the two-stage system that Git uses to save files.
* Describe how to upload your work to GitHub using Git.
* Describe how to check the status of your files and how to view your commit history.

### **Assignment**

Note: As of October 1st, 2020, all new Git repositories will create a default branch named ‘main’ instead of ‘master’.

1. Watch [this video](https://www.youtube.com/watch?v=HVsySz-h9r4) by Corey Schafer for a great overview of some basic Git commands.

### **Cheatsheet**

This is a reference list of the most commonly used Git commands. (You might consider bookmarking this handy page.) Try to familiarize yourself with the commands so that you can eventually remember them all:

* Commands related to a remote repository:
  + git clone git@github.com:USER-NAME/REPOSITORY-NAME.git or git clone https://github.com/user-name/repository-name.git
  + git push origin main
* Commands related to workflow:
  + git add .
  + git commit -m "A message describing what you have done to make this snapshot different"
* Commands related to checking status or log history
  + git status
  + git log

The basic Git syntax is program | action | destination.

For example,

* git add . is read as git | add | ., where the period represents everything in the current directory;
* git commit -m "message" is read as git | commit -m | "message"; and
* git status is read as git | status | (no destination).

### **Conclusion**

You may not feel completely comfortable with Git at this point, which is normal. It’s a skill that you will get more comfortable with as you use it. Therefore, we have a project coming right after this lesson where we’ll walk you through the entire Git workflow, which is the exact same process you would use in a real project.

The main thing to take away from this lesson is the basic workflow. The commands you’ve learned here are the ones you will be using the most often with Git.

Don’t worry if you don’t know all the commands yet or if they aren’t quite sticking in your memory yet. They will soon be seared into your brain as you use them over and over in future Odin projects.

### **Additional Resources**

This section contains helpful links to other content. It isn’t required, so consider it supplemental if you need to dive deeper into something.

* [Learn Enough Git to Be Dangerous](https://www.learnenough.com/git-tutorial) is an introductory guide on Git by [Michael Hartl](http://www.michaelhartl.com/).
* An easy-to-read, pragmatic guide to using Git is available for free on [Kindle](https://www.amazon.com/Rys-Git-Tutorial-Ryan-Hodson-ebook/dp/B00QFIA5OC).
* The [Git Cheat Sheet](https://education.github.com/git-cheat-sheet-education.pdf) from GitHub provides quick instructions for using common commands (you can find a webpage version [here](https://github.github.com/training-kit/downloads/github-git-cheat-sheet/)).
* [Atlassian](https://www.atlassian.com/git/tutorials/what-is-version-control) has a very thorough and well laid out Git tutorial.
* [This video](https://youtu.be/HkdAHXoRtos) by Jeff Delaney has a fast-paced overview of Git.
* For a more in-depth understanding of Git, read the free [ProGit eBook](https://git-scm.com/book/en/v2).

### **Knowledge Check**

This section contains questions for you to check your understanding of this lesson. If you’re having trouble answering the questions below on your own, review the material above to find the answer.

* What is the Git command used to get a full copy of an existing Git repository from GitHub?
* What is the Git command used to check the status of your files?
* What is the Git command used to track files with Git?
* What is the Git command used to remove tracked files with Git?
* What is the Git command used to commit files?
* What is the Git command used to view your commit history?
* What is the Git command used to upload projects into GitHub?
* Explain the two-stage system that Git uses to save files.
* Explain what origin is in git push origin main.
* Explain what main is in git push origin main.