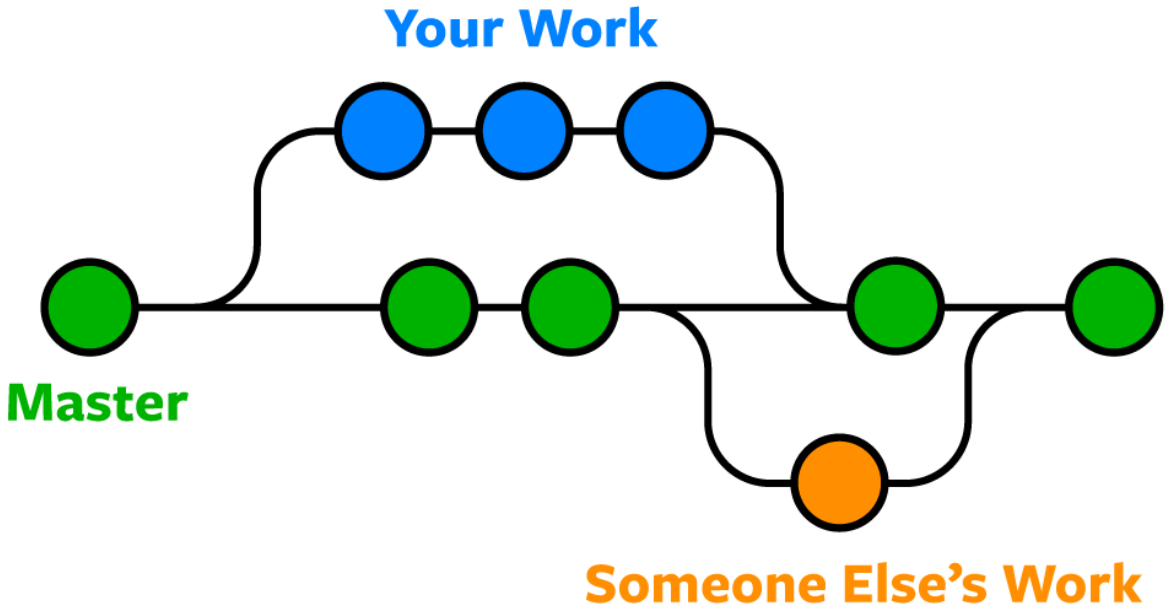
Github Basics

This section covers the basics of how Github works. There are a lot of resources online that can also help you understand. Github is a website that is based on Git. Git is a common version control software.

What is Git

Git was developed to track changes in software over time, with multiple people working on the same code at once. You can download Git onto your PC and use it without worrying about Github at all. The diagram below shows how a project is developed using Git version control.



The project starts in an initial state, called the main (sometimes master) branch. In the diagram above, the main branch is colored green. Each circle is the code at a point in time, referred to as a commit. When a change is made to the code, you must choose to commit the change to the version history. You will also be asked to write a short description of what you changed or added. Uncommitted changes are considered staged, since they are waiting to be accepted or rejected.

For minor fixes, like correcting a typo, you could edit the main branch of code and commit the changes to it directly. But sometimes a change is more involved, and working on it will take time. Furthermore, sometimes you need to make changes to the foundations of a script that render it non-functional, but other people rely on the code to keep running. In this case, Git allows you to create a branch (in blue and orange above). Branches are a copy of the code that is separate from the main, functional code. You can then make edits to your branch without breaking the functional code. When you have finished work on the branch, you can then merge your changes into the main branch. In the diagram above, the blue branch took three commits to finish, whereas the orange commit only took one.

When making a commit, or merging a branch into main, you will have the opportunity to review the old version of the code and the new version. This is so each change is highlighted and looked over before it is added to the version history. The above process allows for complex projects with several contributors to be developed with a clear history of who did what and when, with the ability to go back to previous versions of the project, if need be.

A Git project can consist of one or more files that are located together in a folder. This location, and the files in it are known as a repository, or repo. All the files and subfolders in the repo are included in the project's version control. Git can be used by typing commands into a terminal or command prompt, but other tools (Github, Rstudio, etc.) provide a user interface that can make getting started easier.

Here are the key terms from the description above:

- Version control - Tracking, recording, and organizing changes to a project over time
- Git - a version control software
- Main - The base version of a project
- Branch - a copy of the main project, where changes can be made
- Staged Change - an edit to a file in a repo that has not been committed
- Commit - Officially adding a change to the version history (to the main project, or to a branch)
- Merge - Adding changes from a branch back into the main project
- Repository - A location that contains the files and folders that make up a project

What is Github

As stated above, you can use Git locally on your own computer. In this case, you are working on a local repository, or a project on your own computer. Github is a remote repository; a place to store the code that is not your own computer. This allows the code, with all its version