**Overview of Git/Github**

In this video, you’ll get an overview of Git and GitHub, which are popular environments

among developers and data scientists for performing version control of source code files and projects

and collaborating with others.

You can’t talk about Git and GitHub without a basic understanding of what version control is.

A version control system allows you to keep track of changes to your documents.

This makes it easy for you to recover older versions of your document if you make a mistake,

and it makes collaboration with others much easier.

Here is an example to illustrate how version control works.

Let’s say you’ve got a shopping list and you want your roommates to confirm the things

you need and add additional items.

Without version control, you’ve got a big mess to clean up before you can go shopping.

With version control, you know EXACTLY what you need after everyone has contributed their ideas.

Git is free and open source software distributed under the GNU General Public License.

Git is a distributed version control system, which means that users anywhere in the world

can have a copy of your project on their own computer; when they’ve made changes, they

can sync their version to a remote server to share it with you.

Git isn’t the only version control system out there, but the distributed aspect is one

of the main reasons it’s become one of the most common version control systems available.

Version control systems are widely used for things involving code, but you can also version

control images, documents, and any number of file types.

You can use Git without a web interface by using your command line interface, but

GitHub is one of the most popular web-hosted services for Git repositories.

Others include GitLab, BitBucket, and Beanstalk.

There are a few basic terms that you will need to know before you can get started.

The SSH protocol is a method for secure remote login from one computer to another.

A repository contains your project folders that are set up for version control.

A fork is a copy of a repository.

A pull request is the way you request that someone reviews and approves your changes

before they become final.

A working directory contains the files and subdirectories on your computer that are associated

with a Git repository.

There are a few basic Git commands that you will always use.

When starting out with a new repository, you only need create it once: either locally,

and then push to GitHub, or by cloning an existing repository by using the command "git init".

"git add" moves changes from the working directory to the staging area.

"git status" allows you to see the state of your working directory and the staged snapshot

of your changes.

"git commit" takes your staged snapshot of changes and commits them to the project.

"git reset" undoes changes that you’ve made to the files in your working directory.

"git log" enables you to browse previous changes to a project.

"git branch" lets you create an isolated environment within your repository to make changes.

"git checkout" lets you see and change existing branches.

"git merge" lets you put everything back together again.

To learn how to use Git effectively and begin collaborating with data scientists around

the world, you will need to learn the essential commands.

Luckily for us, GitHub has amazing resources available to help you get started.

Go to try.github.io to download the cheat sheets and run through the tutorials.

In the following modules, we'll give you a crash course on setting up your local environment

and getting started on a project.