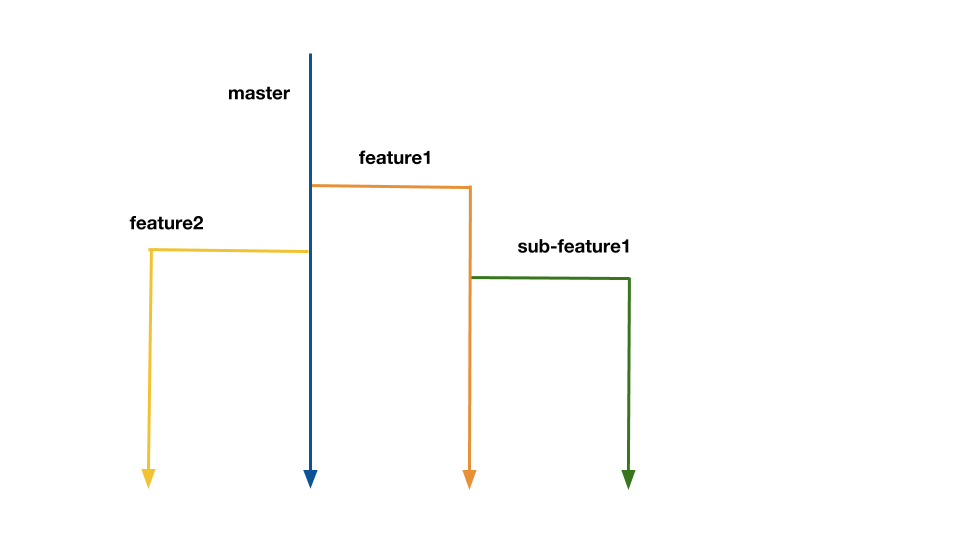
**Using GitHub**

Learning Goals

* **Understand what GitHub is and for what it is used**
* **Familiarize with basic functions of GitHub**
* **Set up GitHub for project work**
* **Display project work on GitHub**
* **Elements of a data analyst portfolio**
* **Preparing a Data Analytics Portfolio:**
  + **Starting a Project**
  + **Gathering all the project pieces – Excel reports, presentations, GitHub**

Introduction

Imagine you are working on a project and at some point you realize some of the changes you made to your code aren’t working out like you thought they would. Now you’ve got to go through your code line by line to make changes. Think about the headaches of working on a collaborate project and having your work overwritten by someone else, or having to wait on teammates to build on your work and send what they’ve done back to you to merge the changes into your copy. There is a solution to these problems and it is called Git. Git is a widely used version control system that allows you to log and track changes you make to your files so that you can restore an earlier version of your file at any time. Git is free and available for download here: <https://git-scm.com/downloads>. Whether you work alone or on a team, being able to revert to a previous version of your project gives you the freedom to experiment or try new ideas without committing to them. Git uses a branching model that allows the user to make independent branches within the code so that you can test your ideas and easily merge or delete a branch from your master project, or “trunk” from which the branches grow in keeping with the tree metaphor.



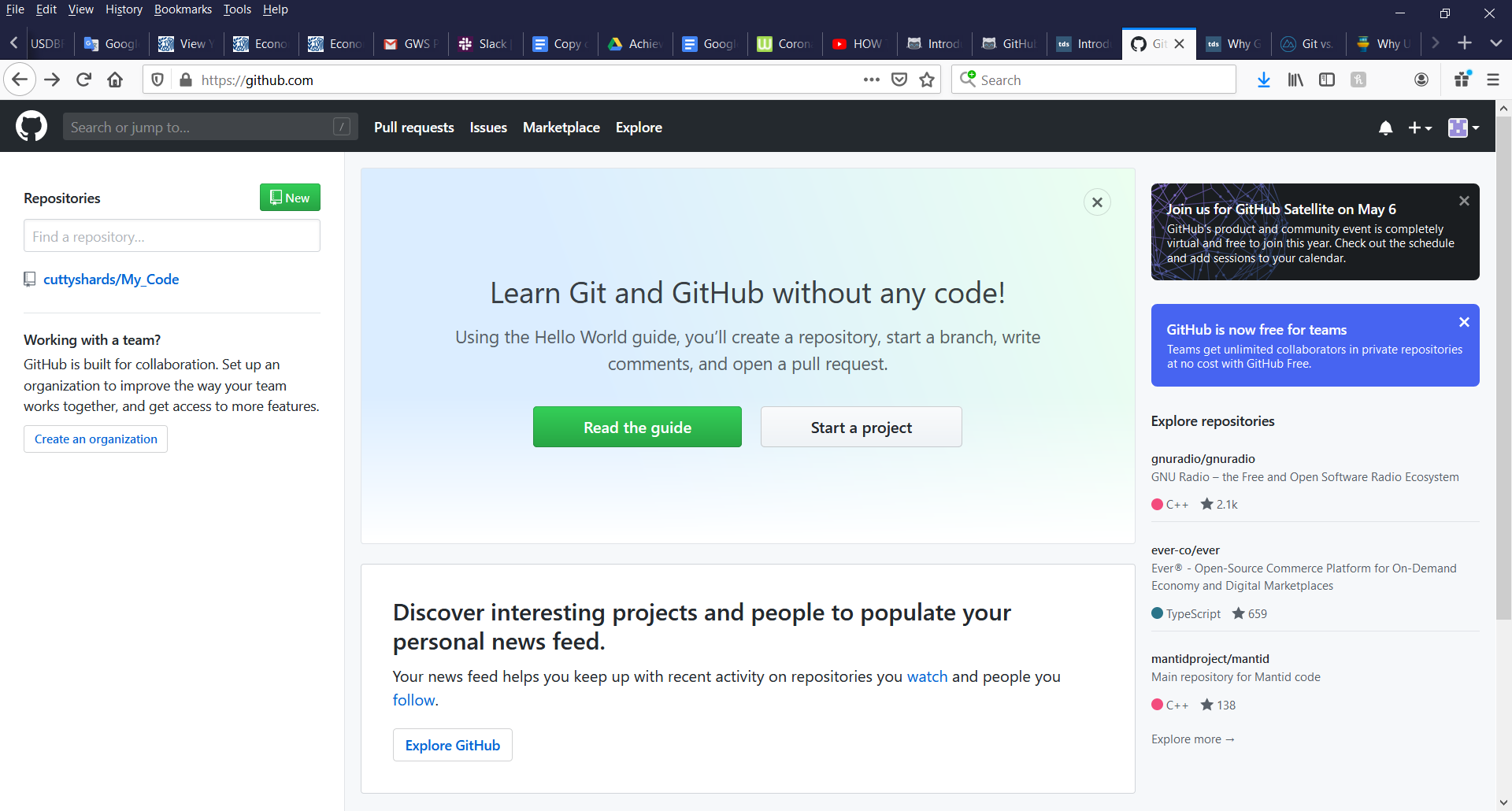
So now that you know what Git is, what is GitHub? Unlike a lot of names and concepts used in the data analytics industry, this one is actually intuitive. GitHub is a repository hosting service for Git. In other words, it is a cloud-based hub where data analysts from anywhere can share code and collaborate on Git version control projects. All one needs to use GitHub is to have Git on their computer and create an account here <https://github.com/join>. When you create a repository on GitHub, you allow authorized users to access your work and build on it, and you can do the same with theirs. Because of Git’s version control system, you don’t have to worry about screwing up someone else’s work or vice-versa because you can always revert back to a previous version. These functions make GitHub a very popular option for open-source projects. GitHub also offers a Marketplace service so that additional features can be implemented. GitHub makes it possible for analysts from anywhere in the world to task organize a project and collaborate together in real-time. Git and GitHub are truly remarkable ways for data analysts to not just work together, but learn from each other.

Using GitHub will not only help you develop experience participating in projects, it can help you market yourself to employers for whom you wish to work. With GitHub you can create a portfolio of the work you did participating in individual or team projects. A well-maintained data analytics portfolio can be shared with colleagues and prospective employers alike. It gives that big tech firm the ability to see for themselves you skills and experience you have. Your portfolio is a place where you can share your code or anything else related to any projects you have completed or participated in. You can start by uploading some of your work from this course, and add to it as you continue to develop yourself as a data analyst. A portfolio in GitHub is fairly easy to create. In this achievement, we will discuss the importance of having a portfolio in today’s job market, how to create a portfolio in GitHub, what should be included in your portfolio.

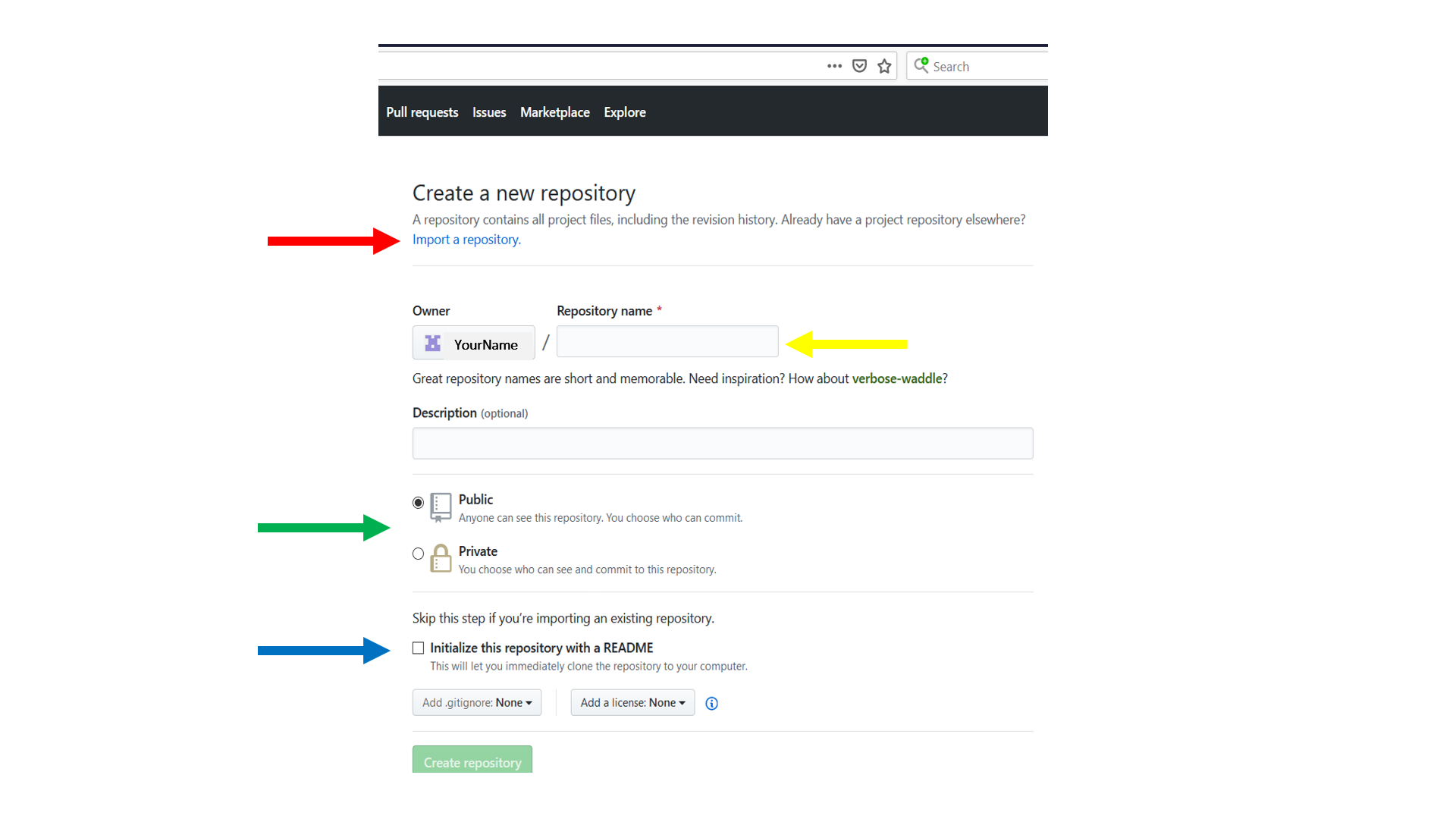
You don’t need to be an expert to use effectively use GitHub as a data analyst. The key learning points discussed in this achievement are to make you aware of GitHub, what it is used for and some of the functions of GitHub to get you started. If you seek to take courses that go into further depth on using GitHub, multiple courses can be found here <https://lab.github.com/>.

Creating a Repository

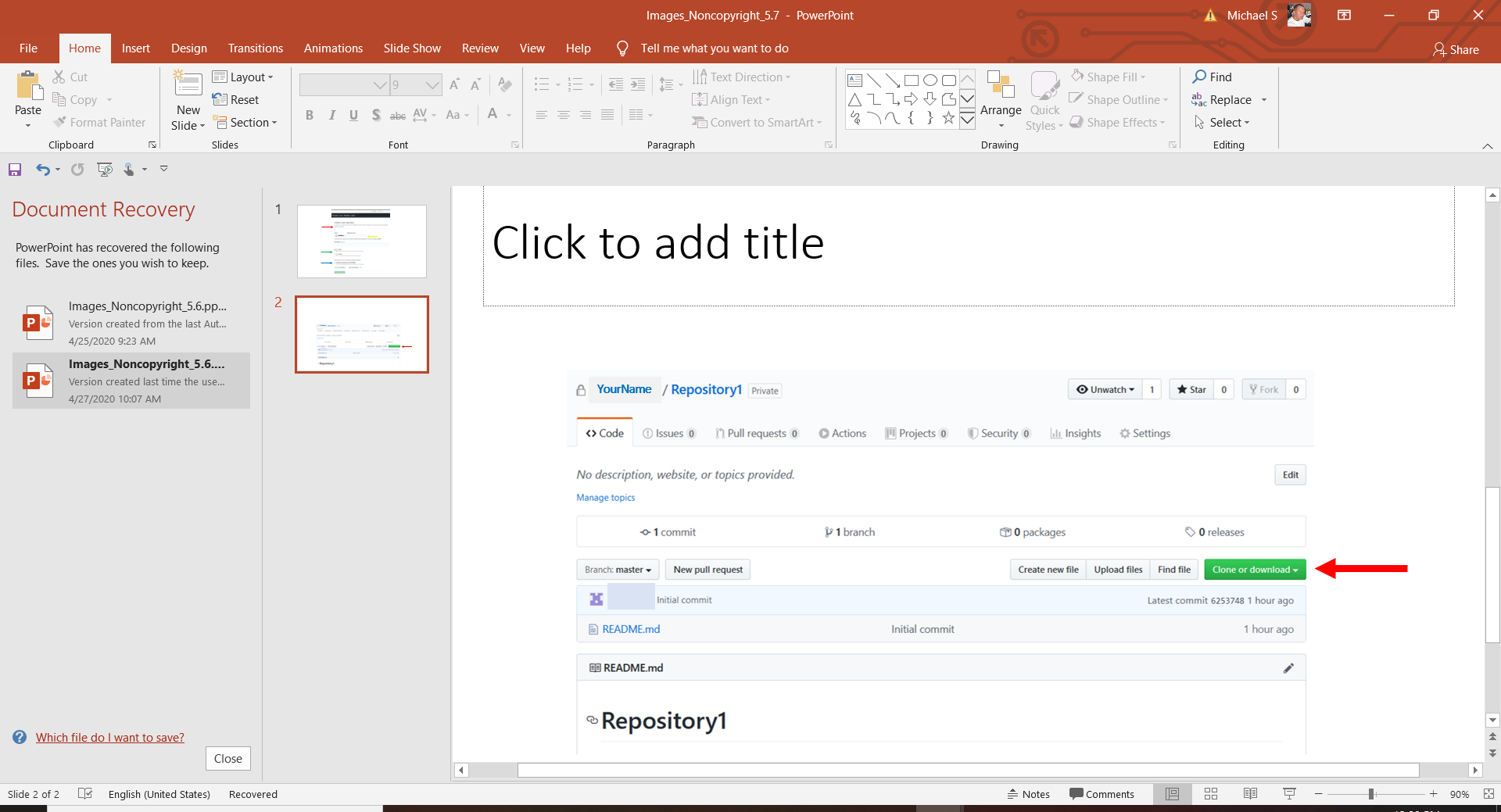
Now that you have downloaded Git and created an account with GitHub, you can begin working. But first you are going to need to create a repository in which to store your work.



When you are signed in, look to the upper left and you will see the green “New” above a repository search bar. Here you can search for a repository by name, or create a new one by clicking the green “New” button. You should now see the image below where you can add in some additional details about your repository.



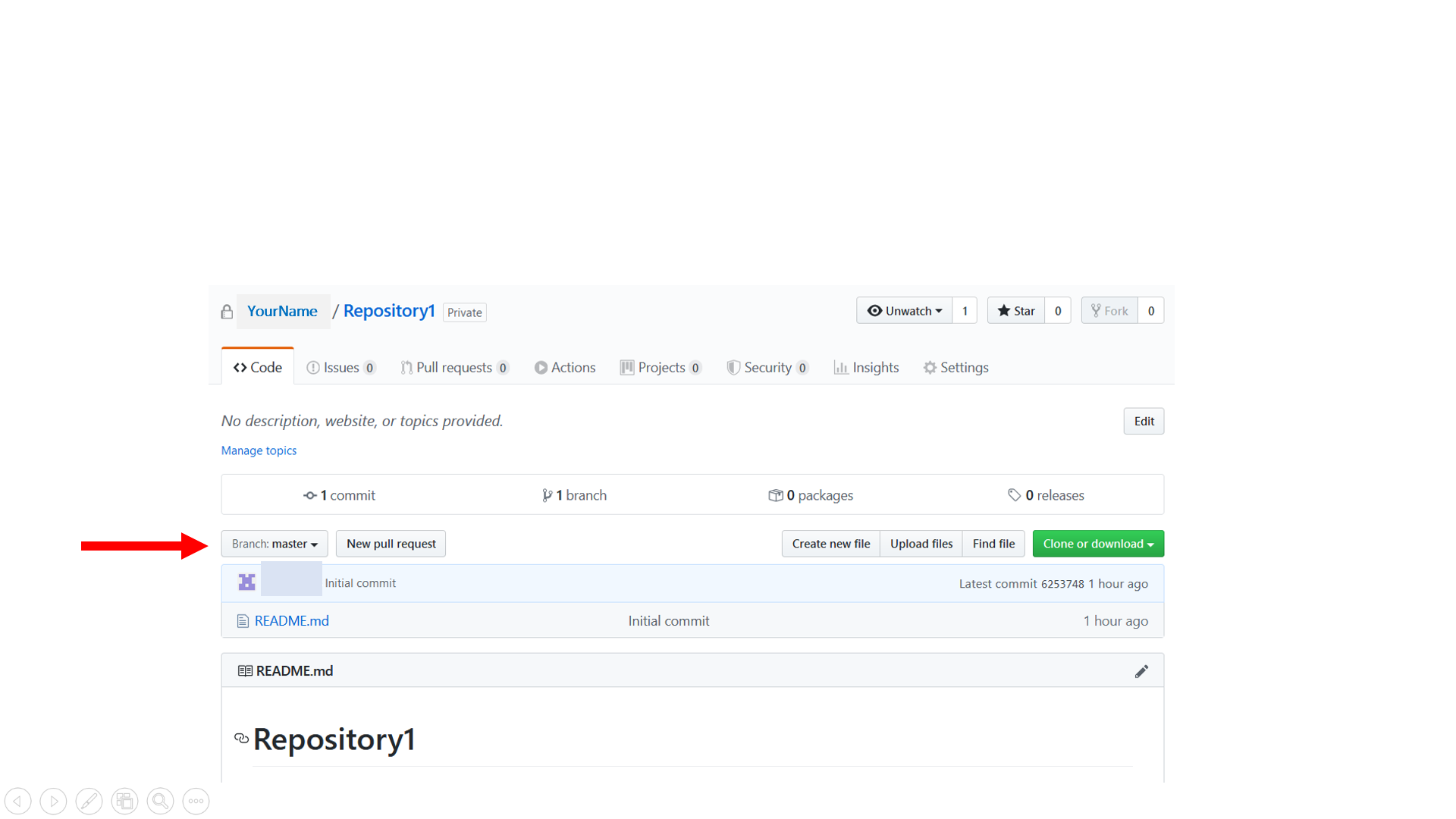
If you are going to import someone else’s repository, you can do that by clicking on the Import a repository link shown by the red arrow. If you are creating your own repository, first enter what you would like to call it under “Repository name” shown above by the yellow arrow. Next you will need to choose a Public or Private option, shown by the green arrow, depending on your preference for the project. Check the README box shown by the blue arrow. Finally, click the green “Create repository” button at the bottom of the screen to conclude the creation of your repository. Your screen should now look like the one below.

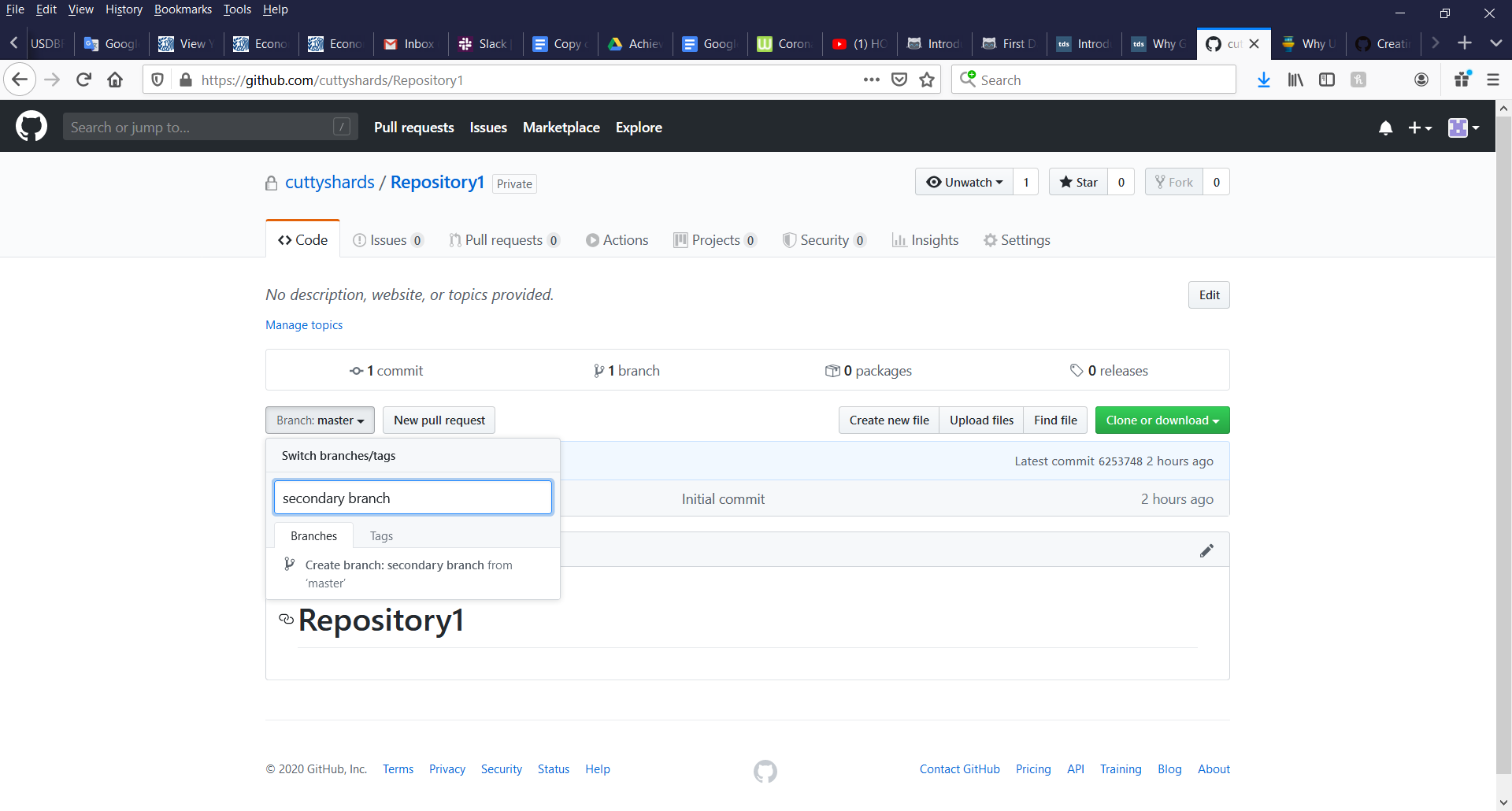


On your repository screen, take note of the green “Clone or download” button midway down on the right shown by the red arrow. Before you can work on your project locally or work on someone else’s, you must clone the repository. A directory will be created when you click “Clone or download” with the same name as your repository. This will serve as your local version.

Branching

You will want to begin work by creating a branch off the master branch in order to make your additions to another project. This way the master branch is preserved and any changes you might add can be merged in after they have been reviewed.

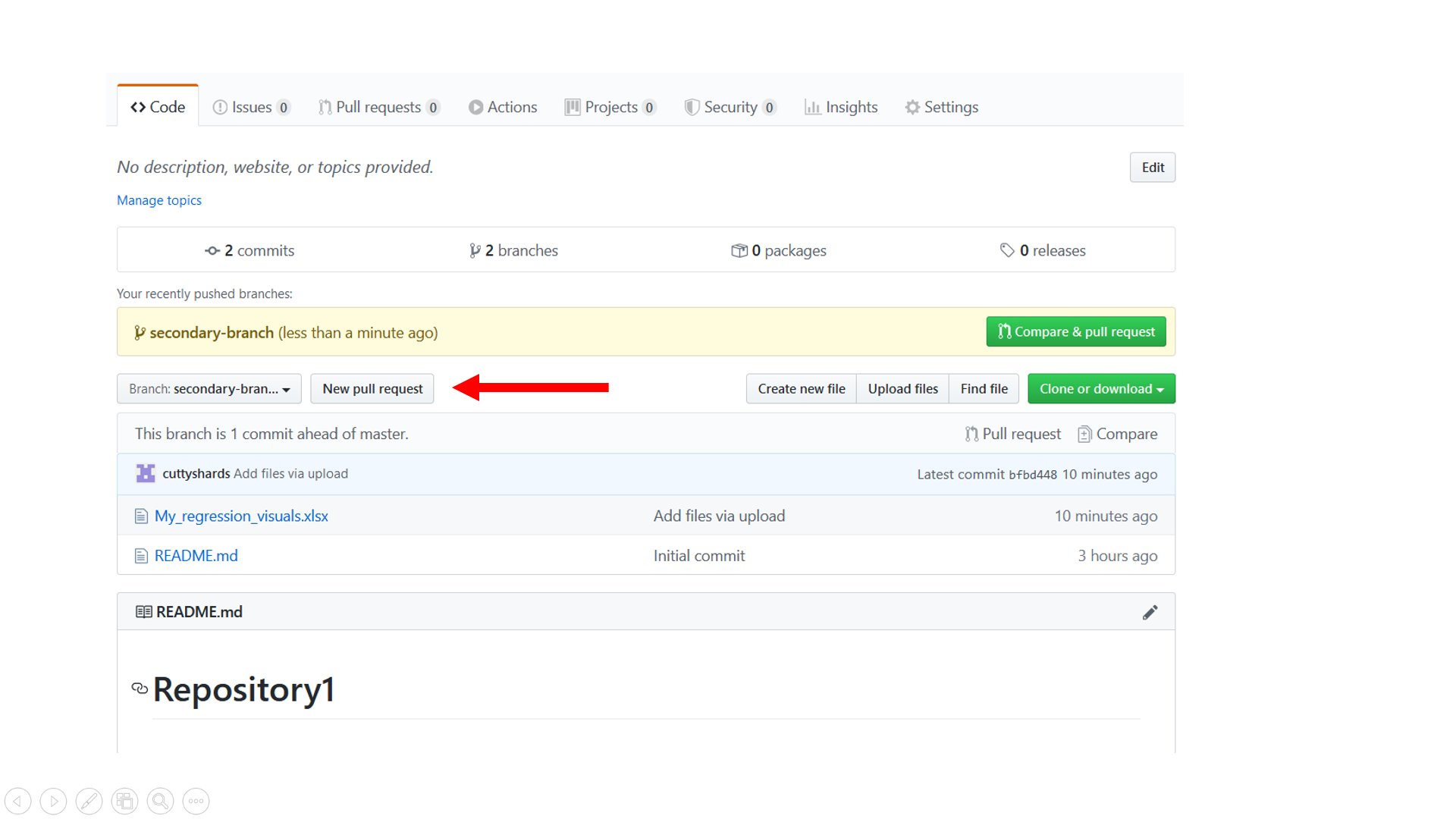


Be sure you are in the correct repository and click on the button marked Branch: **master** shown above with the red arrow. You will then see the image below where you can type the name of the new branch in the bar then click on “Create branch:.”

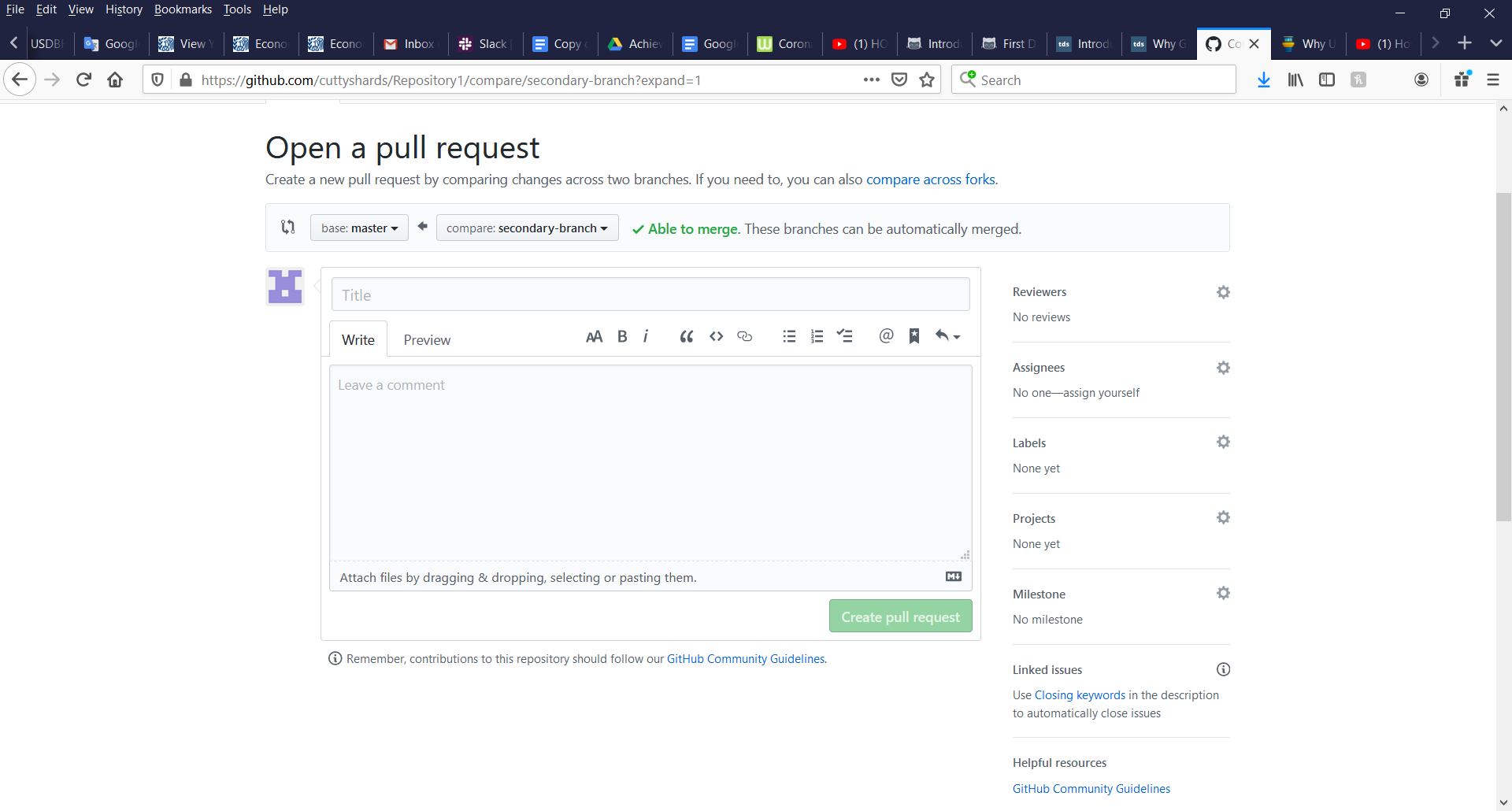
Now you can choose which branch you want to work in. When you are finished with your input, you can then push your work to the master branch.

Pull Request

Before any changes are merged into the master branch, it’s always a good idea to have one or more of your teammates review your work and make any last changes or additions before you accept them. In GitHub, this can be done with the Pull Request. The Pull Request function allows you to post work you have committed and receive feedback before adding to the master branch.



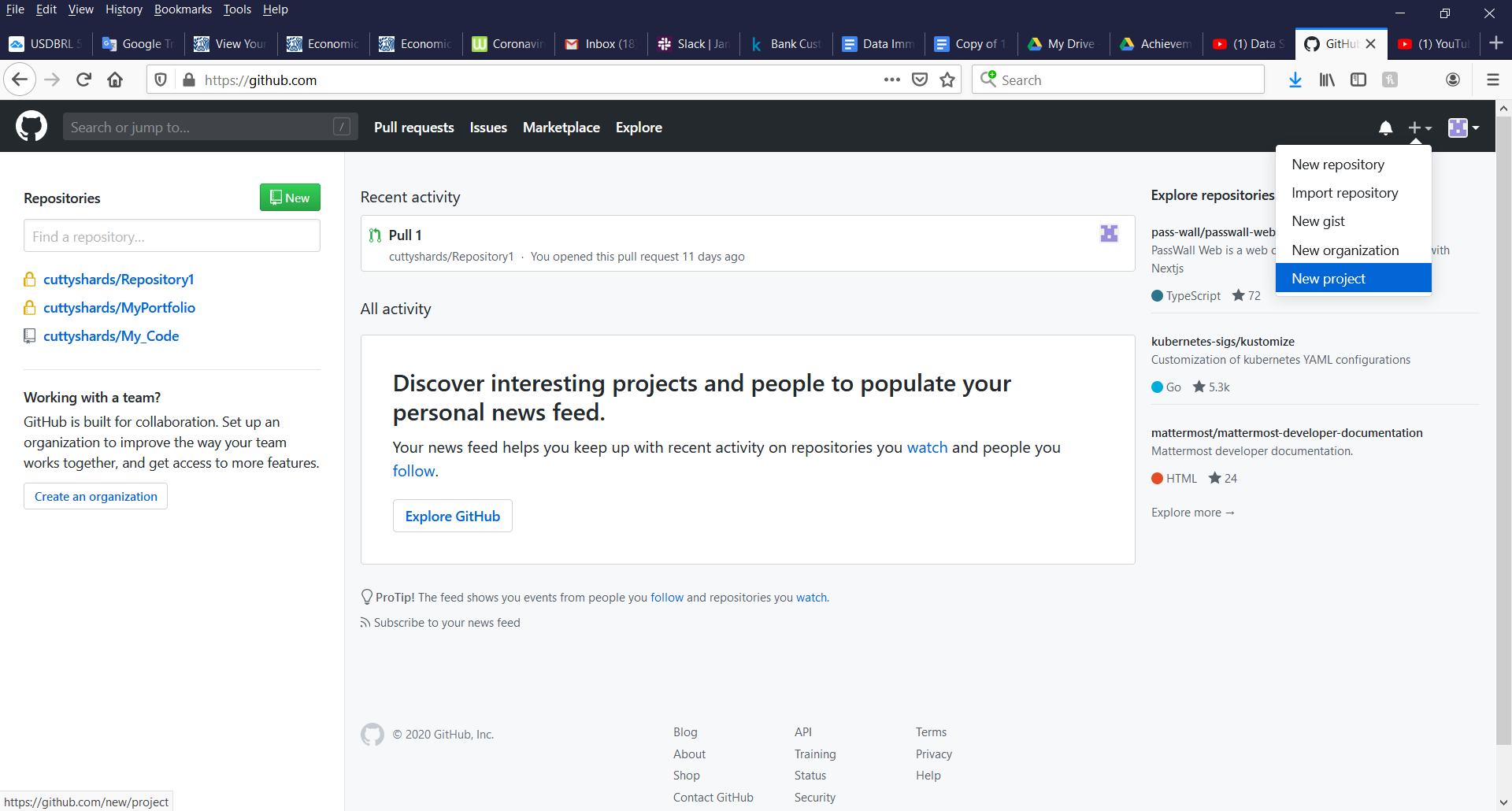
From your branch that you would like to be reviewed, find the “New pull request” button located in the middle of your screen and shown by the red arrow. You should now see the screen below.



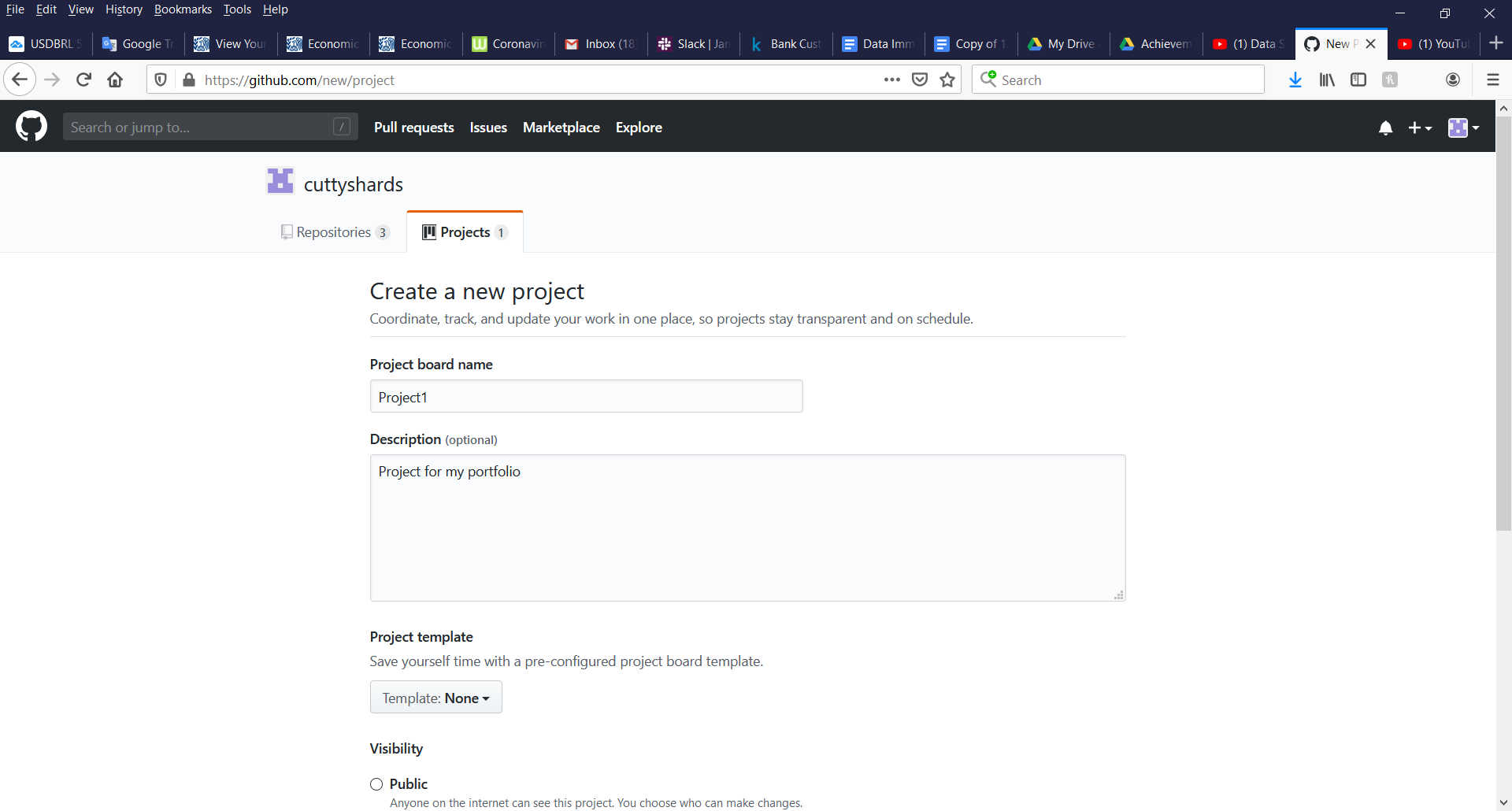
Give your pull request a title, then simply drag and drop the files you wish to be reviewed into the body. When you have dropped the file or files, click “Create pull request” to submit.

**Preparing a Data Analytics Portfolio**

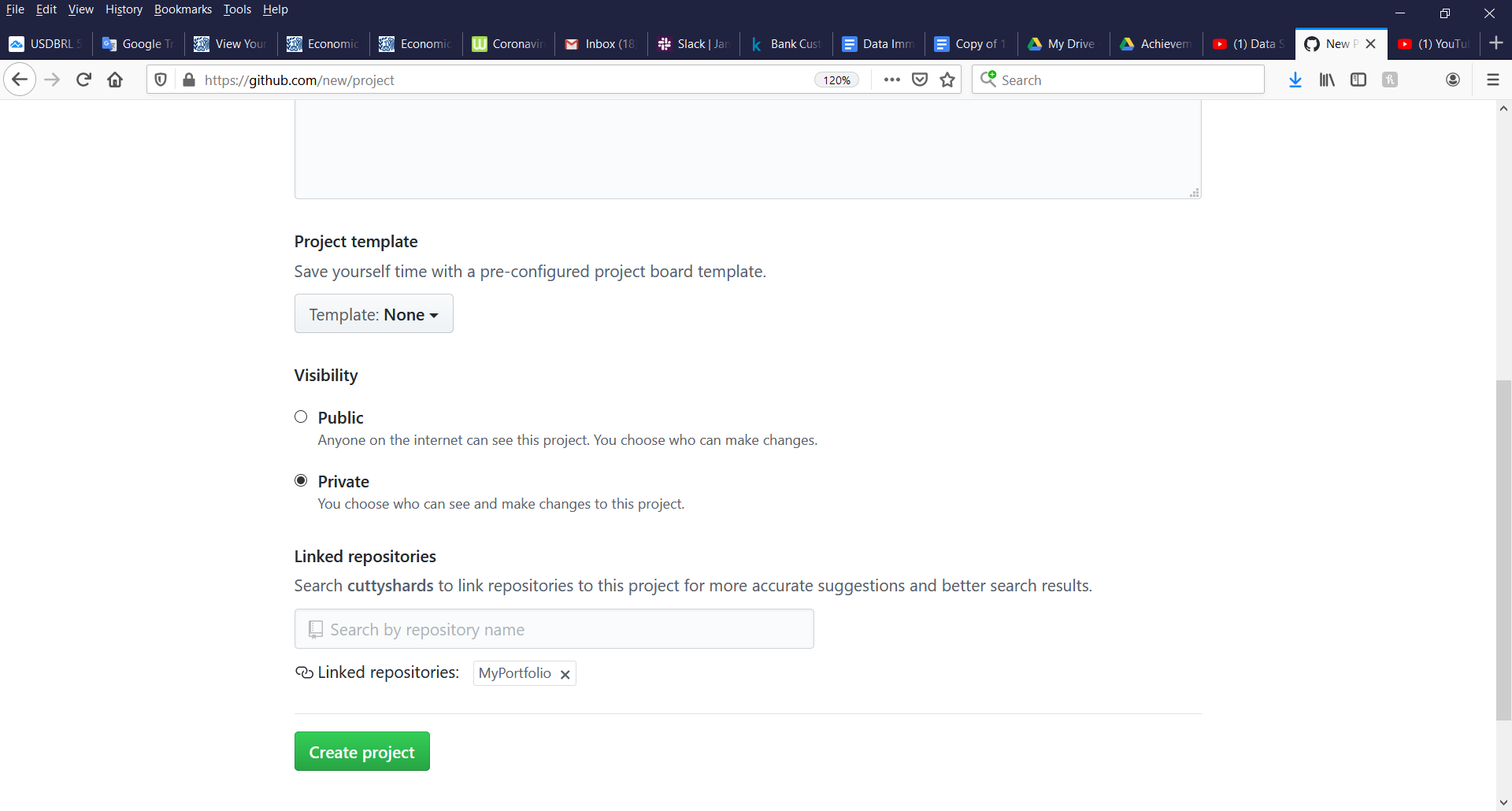
Start a Project

Login to your GitHub account you created as part of achievement 5.7. It’s best for whomever you wish to share your work that you create a repository in which to store your portfolio. This way you can direct whomever you are sharing with exactly where they can find which files you feel best highlight your experiences and skills. Go ahead and click on “New” button and create a repositor, just like you did in 5.7. Name the repository “MyPortfolio.” Now that you have a repository created and opened, the next thing you’ll need to do is start a project. From your homescreen, click on the + sign in the upper righthand side of your screen. Click on “New Project” on the drop down menu as shown in the image below. 

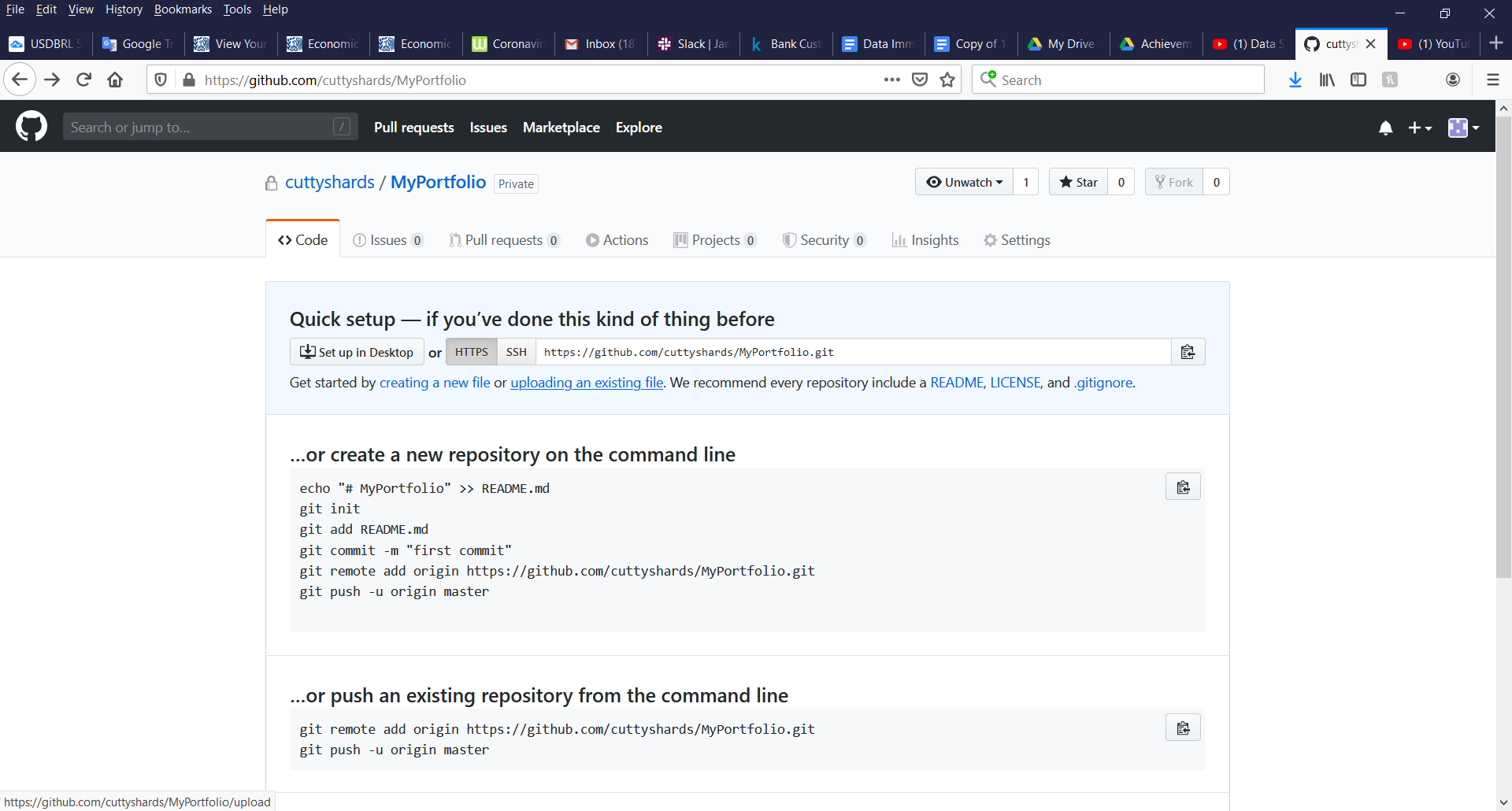
Next, you will need to give your project a name and give it a description. See image below.



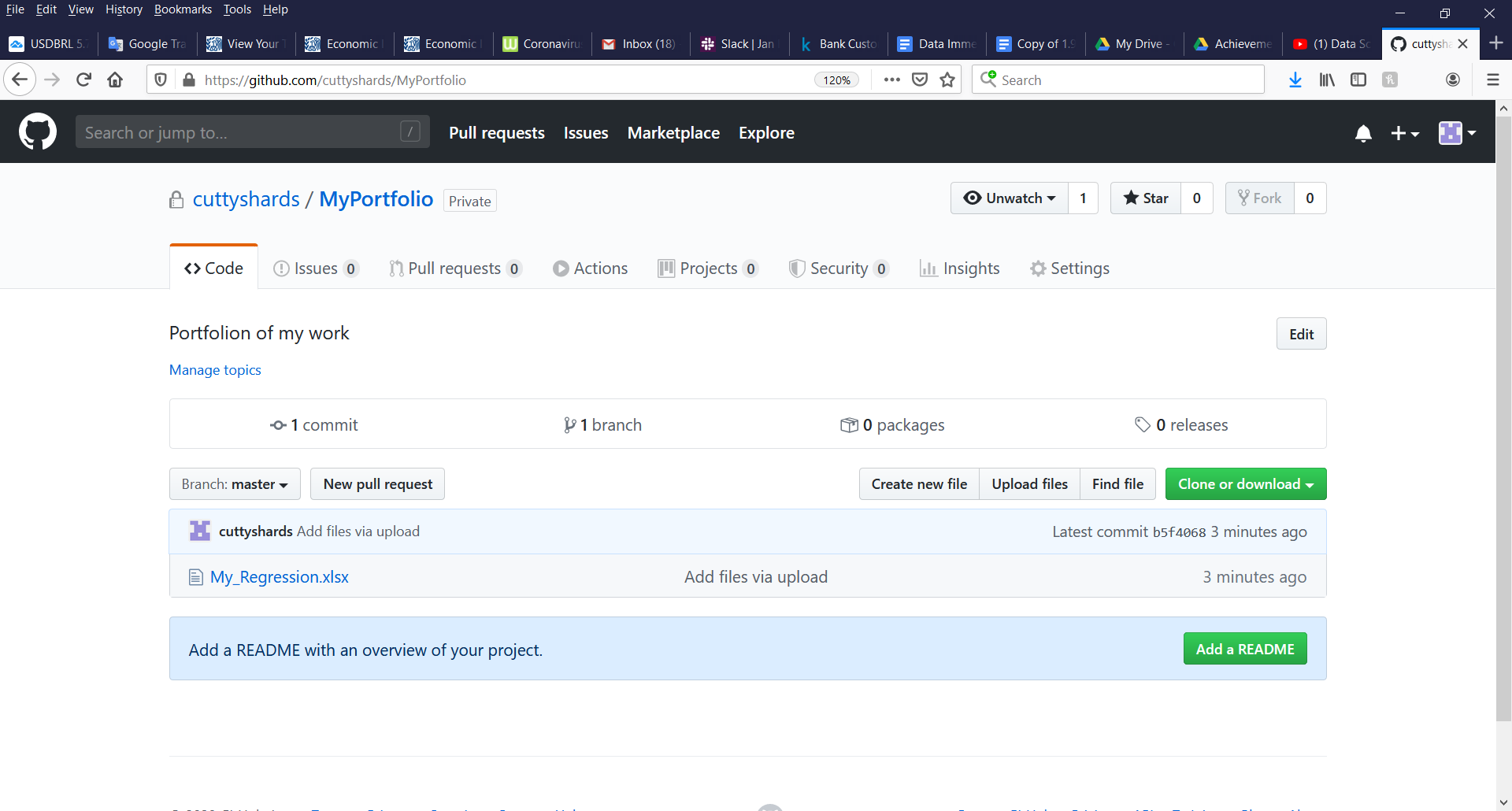
Finally, link your project to the repository you just created. Simply type in the name of the repository and choose it from the auto-suggest list. Once you choose the repository, you will see it below next to where it says “Linked Repositories.”



Now that your project has a name and place to reside, you can begin to work and upload your work. From your home screen, click on the name of your project you will find in the upper left side of your screen. This will take you to the screen shown below. From here you can upload an existing file to your portfolio project by clicking “uploading an existing file” shown by the red arrow.



Drag files into your repository or click on “choose your files” to upload and select the files from your computer you’d like to share. Now you should see your file in your portfolio as shown below.

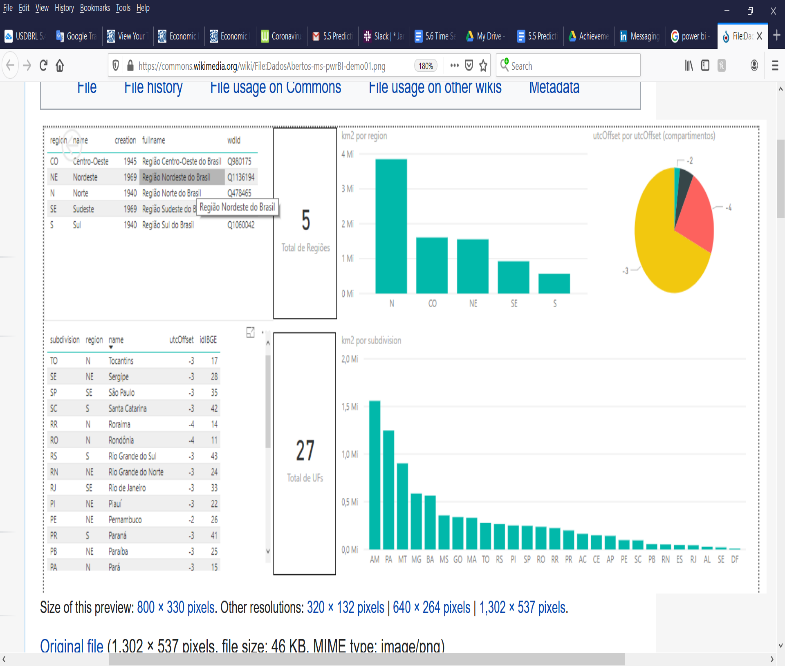


Portfolio Requirements

When it comes to landing a job with a desired company, a well-managed portfolio can be as valuable or more than any certification or degree because it stands as proof of experience. But A prospective employer will likely not want to take the time to open and examine every file in every project you participate in. Your portfolio should be limited to the files that best reflect your experience and skills.

Think about the specific skills that you see desired in the qualifications/skills sections of those job advertisements you have skimmed though. Often, we see SQL, Python, R, quantitative/statistics, Power BI and others. Your portfolio should attempt to show off as many as your data analyst skills as possible. We all know it is not fun nor easy to read someone else’s code. Your files that demonstrate your coding skills should be limited to your best code. Do take the time to “clean” up your code so it is easier to read. Highlight your comments explaining what your code does and use spacing between blocks of code that make is easier on the reader. Remember, when you are marketing yourself, you want to make all of your files look worthy of presentation.

In addition to having some code examples, you will want to include a file or files that reflect your skills in math, statistics in particular. Files that show the recruiter you have gathered a sample of data from an open source and found a p-value, r-squared, or ANOVA for example that show the recruiter you understand the concepts of these statistics and how to apply them to model data.

No portfolio would be complete without some compelling graphics that demonstrate your ability to tell your project’s story. Tools such as Power BI are free to download and can be used to create substantive visuals. Power BI is from Microsoft and there are multiple online tutorials to assist you in learning to create presentation worthy data products.

As you seek out and participate in more projects through GitHub, you can update your portfolio to better reflect your experience.

Conclusion

Your education as a data analyst is continuous. As technology evolves and new software tools are developed and implemented, so to must you keep your skills sharp and education current. Not only will taking advantage of GitHub give you experience collaborating with and learning from other professionals, it will allow you to share your work with prospective employers. In this achievement we have discussed Git technology and how version control makes collaborating on a single project easier. We have also discussed GitHub and creating an account on GitHub. You have also been provided step by step instructions for creating a portfolio and using the functions of GitHub that allow you to create repositories, branches, push changes as well as request reviews from your teammates. There are many more in-depth trainings and useful tools offered in GitHub Lab and marketplace for you to explore and learn how to develop experience and learning through collaboration. With your skills in place and GitHub account created, you are now ready to tackle some real-world projects!

Resources

<https://towardsdatascience.com/introduction-to-github-for-data-scientists-2cf8b9b25fba>

<https://towardsdatascience.com/why-git-and-how-to-use-git-as-a-data-scientist-4fa2d3bdc197>

<https://www.git-tower.com/learn/git/ebook/en/command-line/basics/why-use-version-control>

Images

<https://commons.wikimedia.org/wiki/File:DadosAbertos-ms-pwrBI-demo01.png>