Hands-on Lab: Getting Started with GitHub

Effort: 20 min

In this lab, you will get started with GitHub by creating a GitHub account and project and adding a file to it using its Web interface.

Objectives

After completing this lab, you will be able to:

1. Describe GitHub
2. Create a GitHub account
3. Add a Project / Repo
4. Edit / Create a file
5. Upload a file & Commit

GitHub Overview

First, let us introduce to GitHub. GitHub in simple words is a collection of folders and files. It is a Git repository hosting service, but it adds many of its own features. While Git is a command-line tool and a server needs to be hosted and maintained via command line as well, GitHub provides this Git server for you and a Web-based graphical interface. It also provides access control and several collaboration features, such as wikis and basic task management tools for every project. GitHub provides cloud storage for source code, supports all popular programming languages, and streamlines the iteration process. GitHub includes a free plan for individual developers and for hosting open source projects.

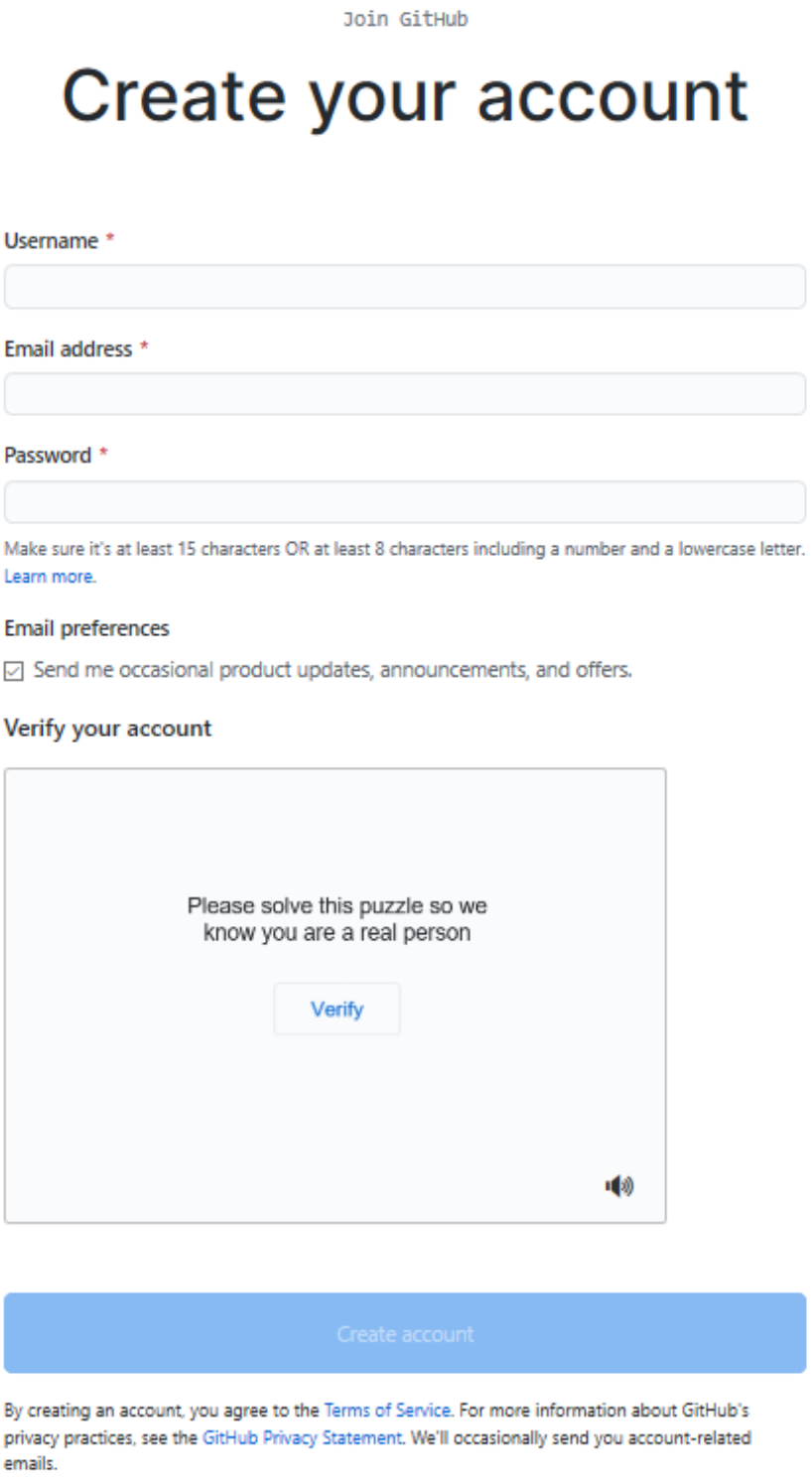
Exercise 1: Creating a GitHub Account

Please follow the steps given below to create an account in GitHub:

Step 1: Create an account: https://github.com/join

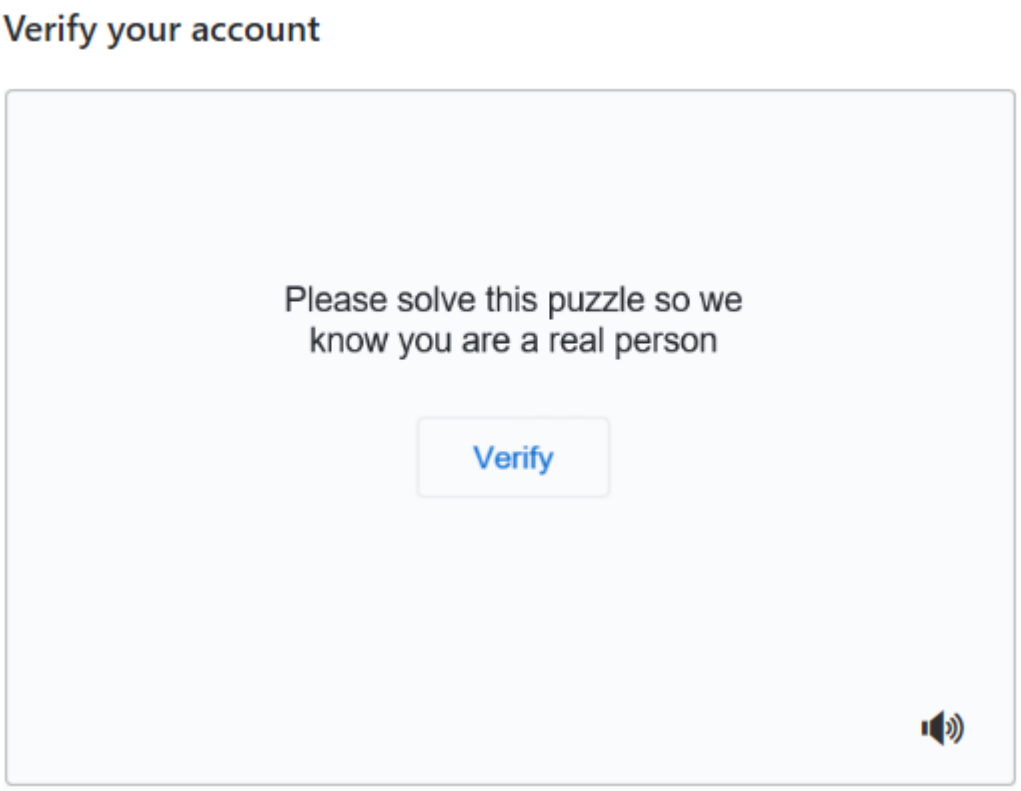
NOTE: If you already have a GitHub account, you can skip this step and simply login to your account.

Step 2: Provide the necessary details to create an account as shown below:

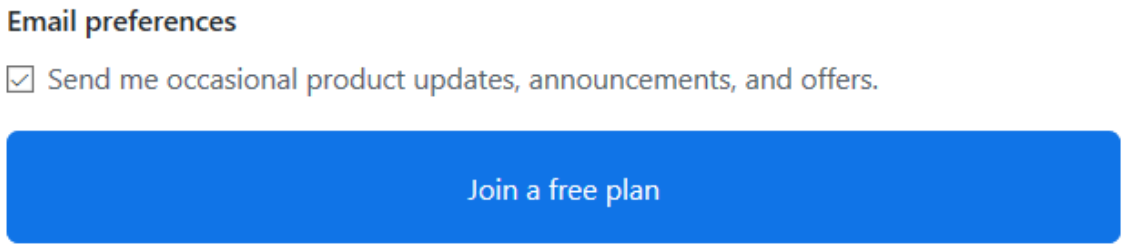


and click `Create account`.

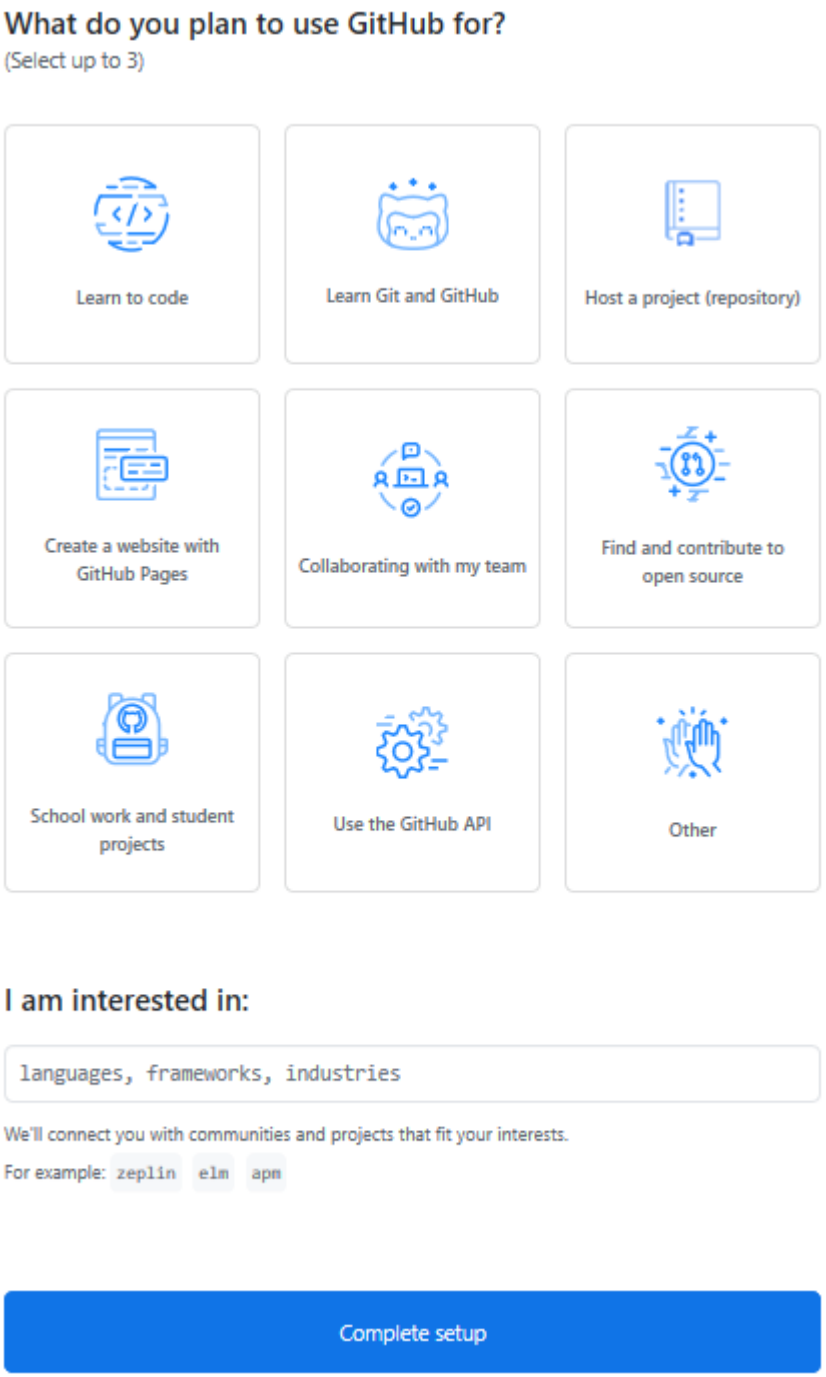
Step 3: Click `Verify` to verify the account and click `Done`



Step 4: After verification, click `Join a Free Plan`

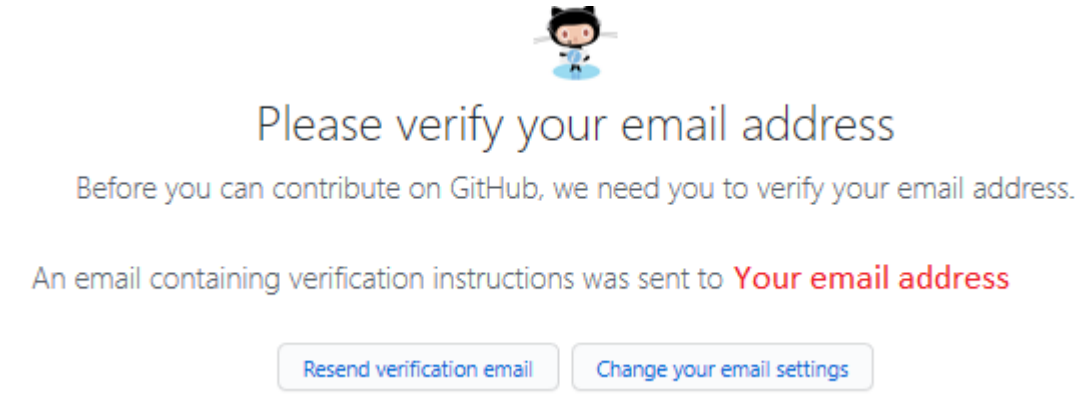


Step 5: Select the details as shown below and click `Complete Setup`

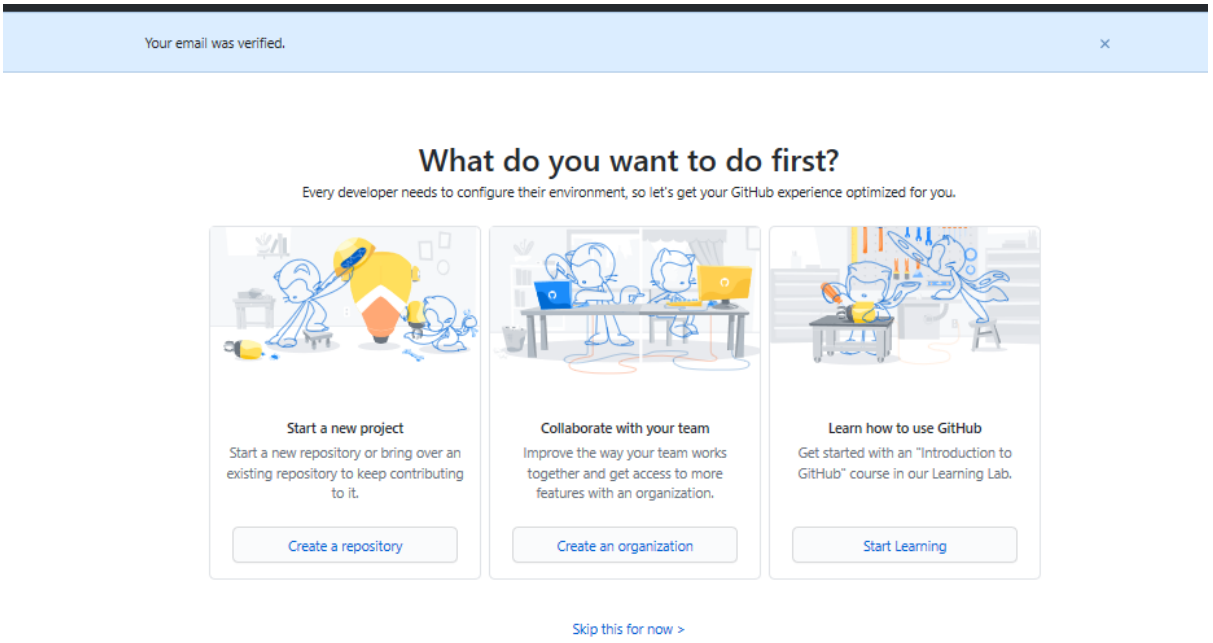


Step 6: Go to your email, find the verification email from GitHub, and click on the link/button in that email to verify your email.

NOTE: If you do not receive verification email, click `Resend verification email`.

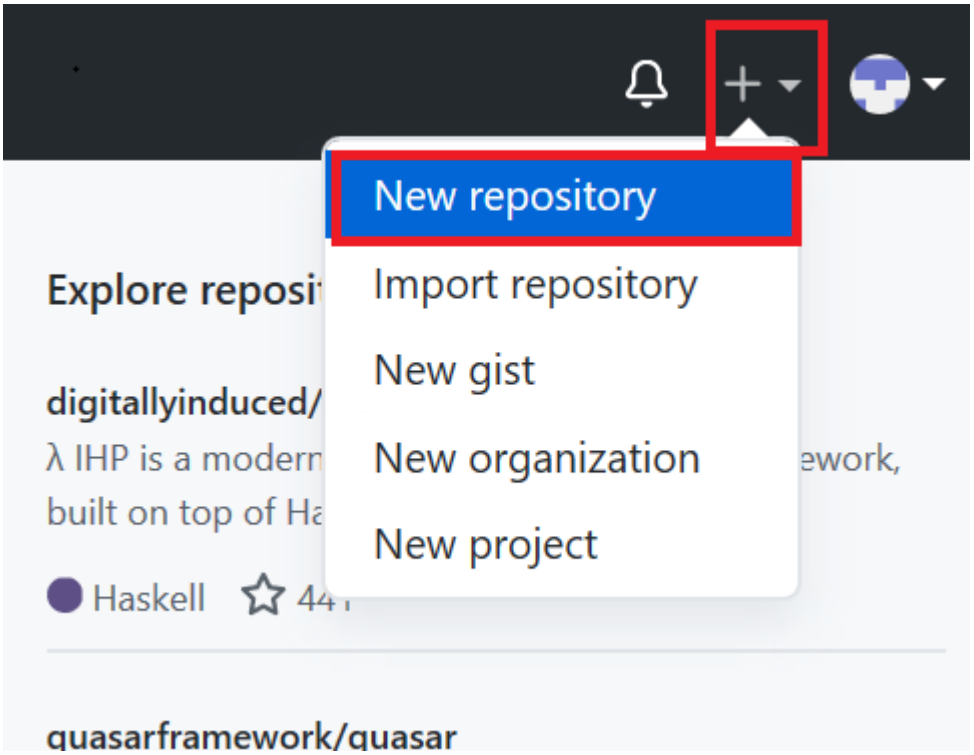


Email is verified

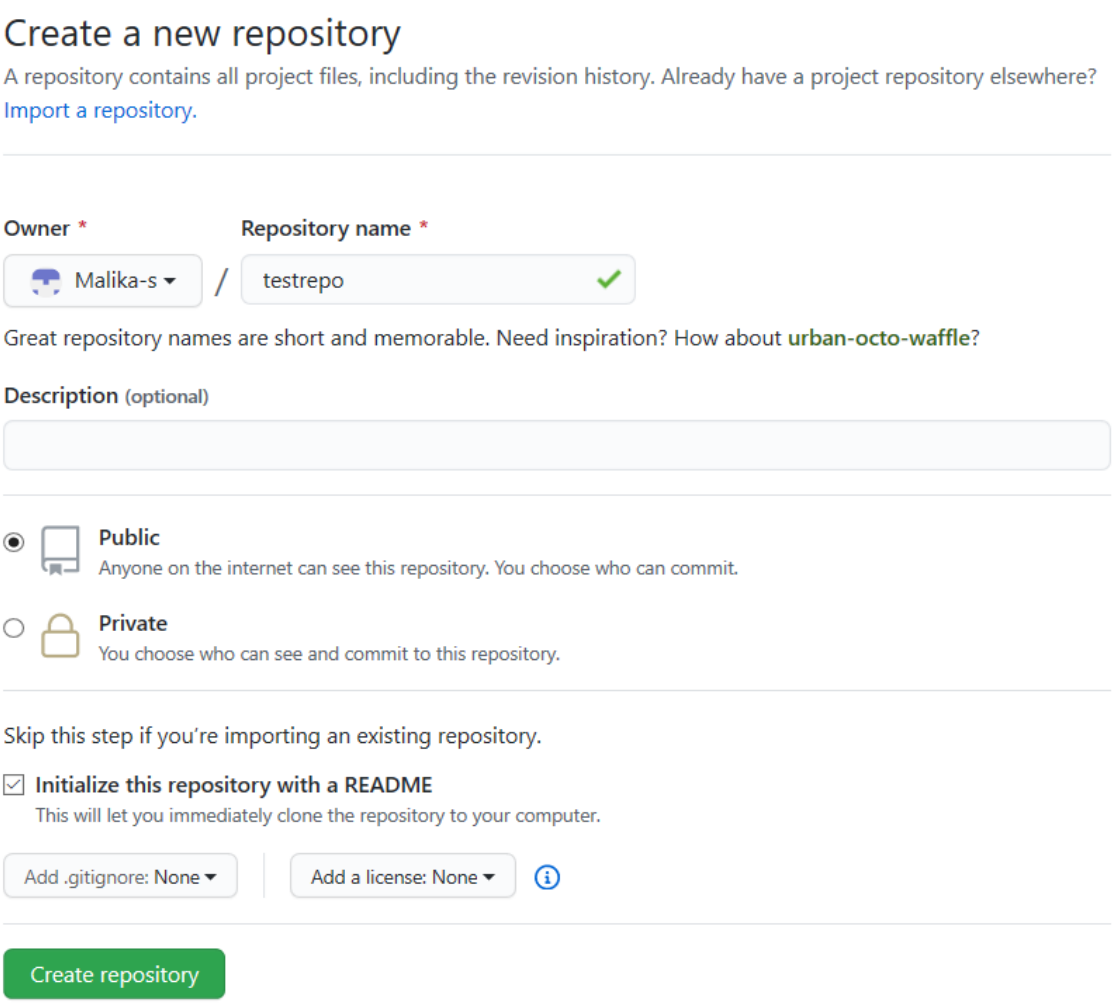


Exercise 2: Adding a Project / Repo

Step 1: Click on the `+` symbol and click `New repository`.



Step 2: Provide a repository a name and initialize with the empty `README.md` file.



and click `Create repository`.

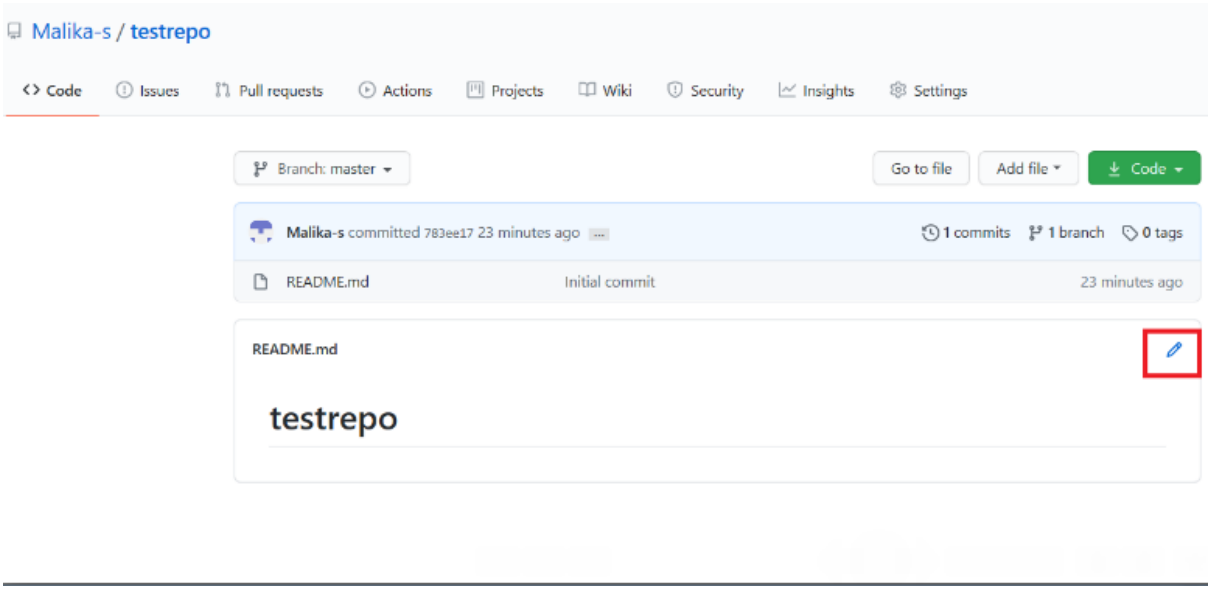
Now, you will be redirected to the repository you have created.

Let's start editing the repository.

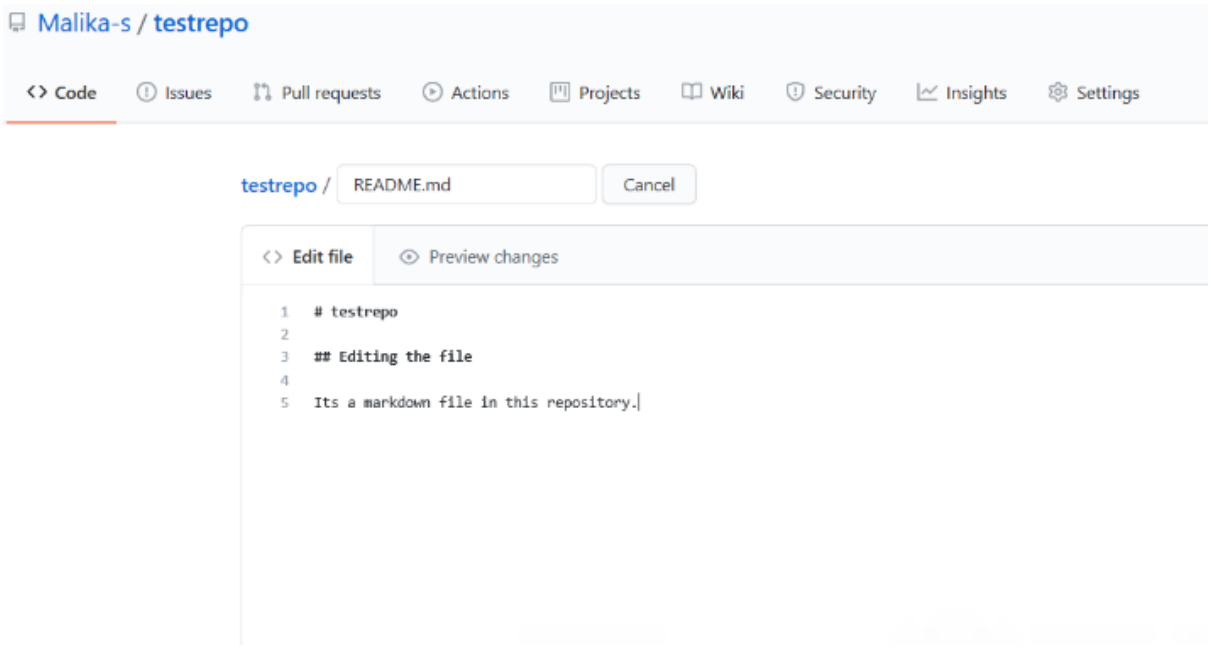
Exercise 3: Create / edit a file

Exercise 3a: Edit a file

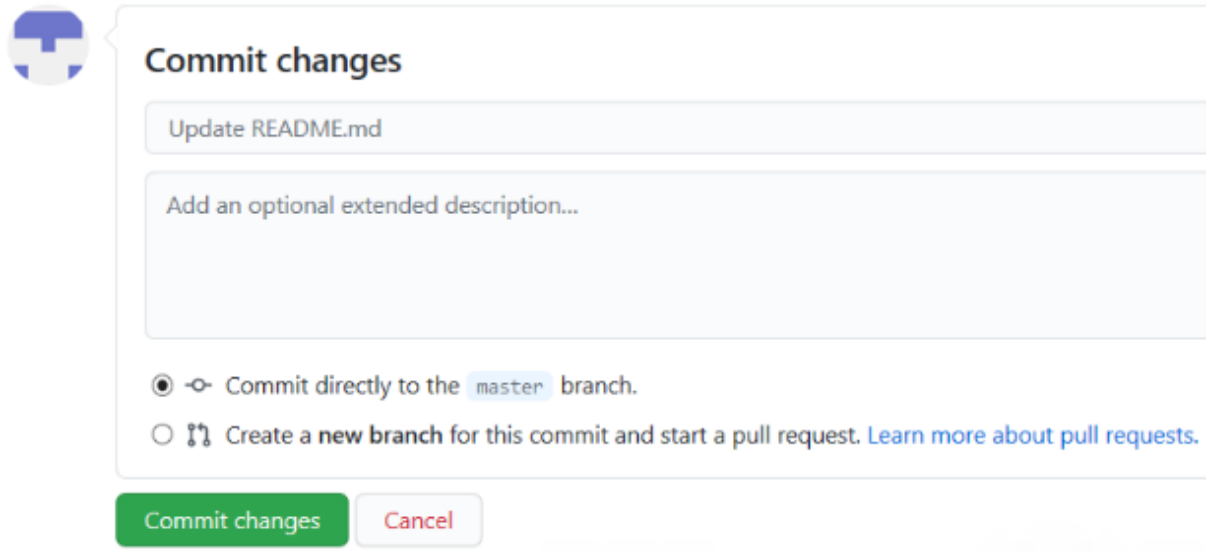
Step 1: Once the repository is created, the root folder of your repository is listed by default and it has just one file `ReadMe.md`. Click on the pencil icon to edit the file.



Step 2: Add text to file.



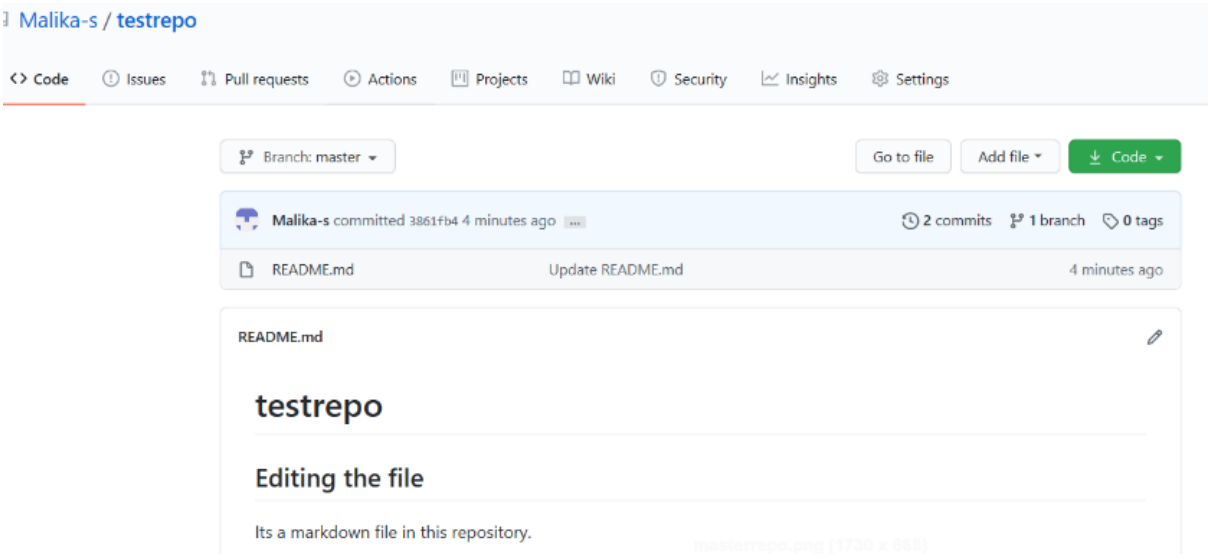
Step 3: Scroll down the page after adding the text and click `Commit Changes`.



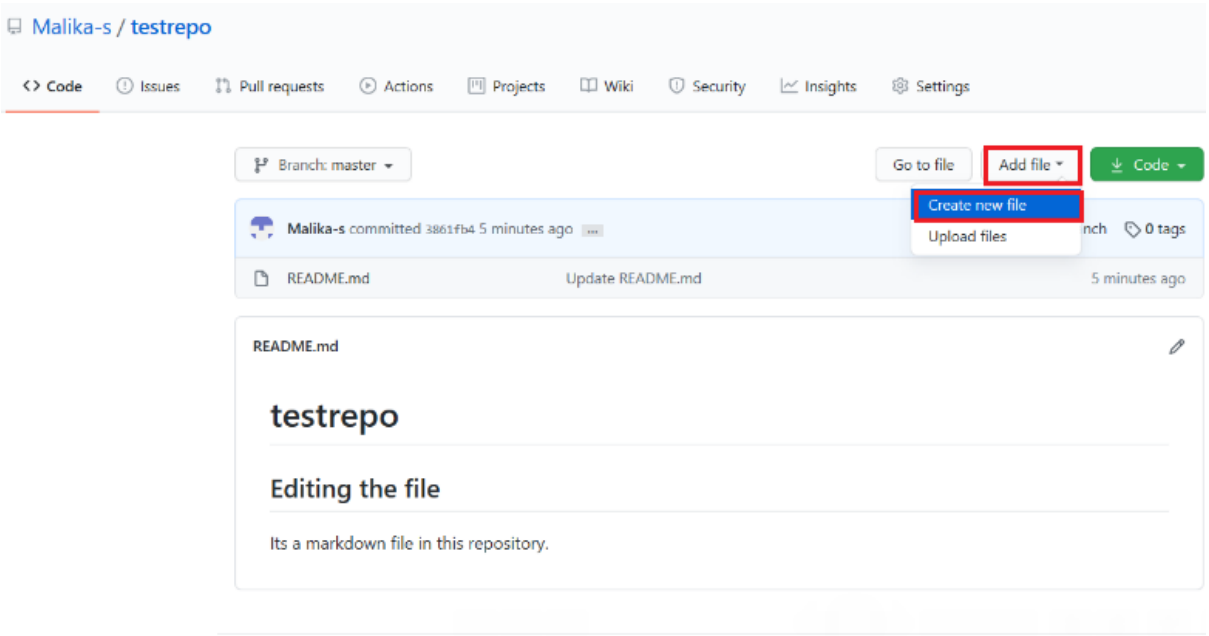
Now, check your file is edited with the new text.

Exercise 3b: Create a new file

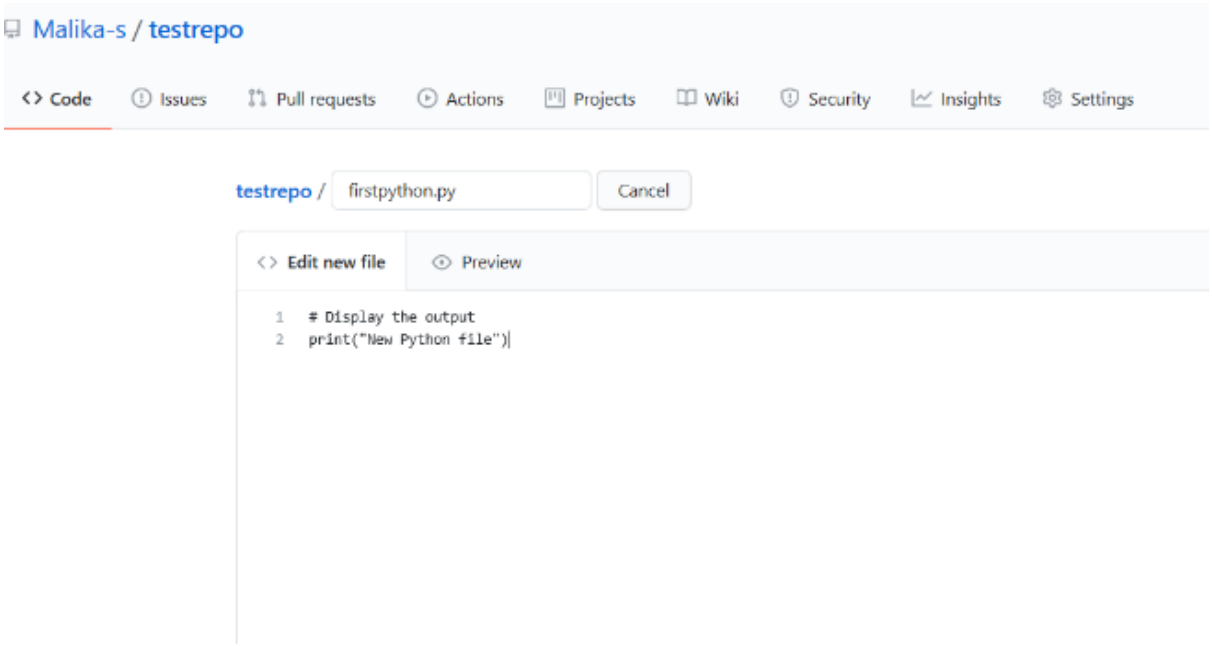
Step 1: Click on the repository name to go back to the master branch like in this testrepo.



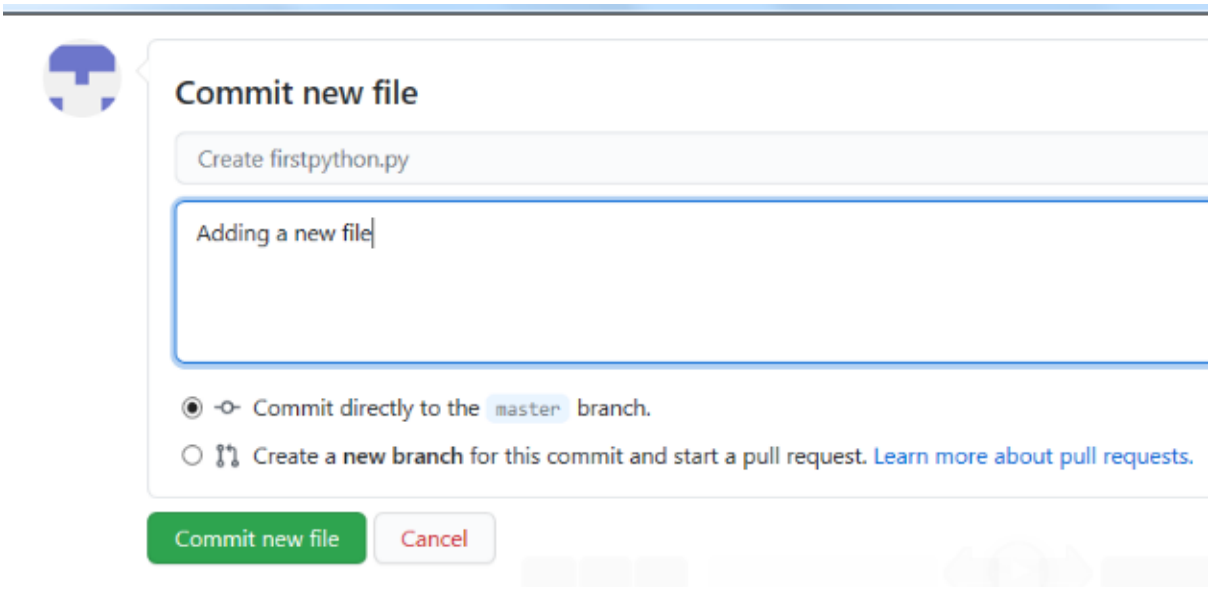
Step 2: Click `Add file` and select `Create New file` to create a file in the repository.



Step 3: Provide the file name and the extension of the file. For example, firstpython.py and add the lines.



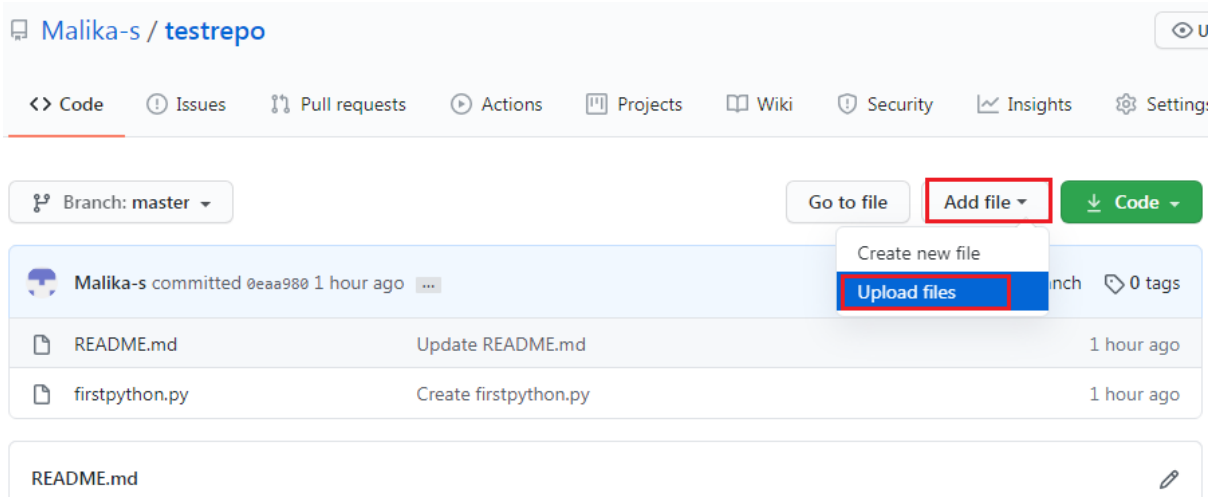
Step 4: Scroll down the page after adding the text. `Add description` of the file (optional) and click `Commit` new file.



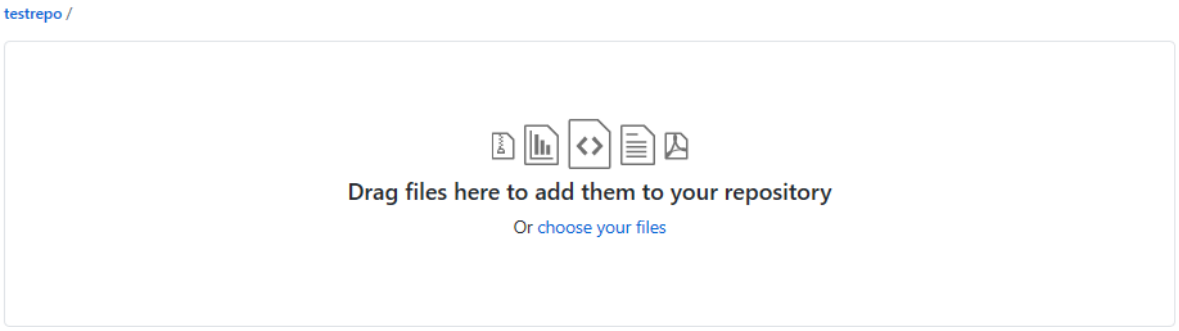
Step 5: Your file is now added to your repository and the repository listing shows when the file was added/changed.

Exercise 4: Upload a file & Commit

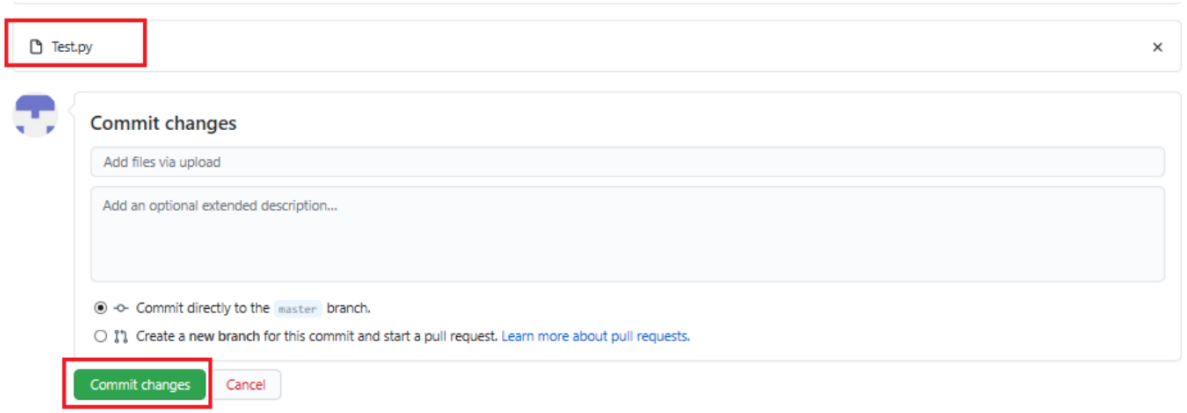
Step 1: Click `Add file` and select `Upload files` to upload a file (Upload any .txt,.ipynb, .png file) in the repository from the local computer.



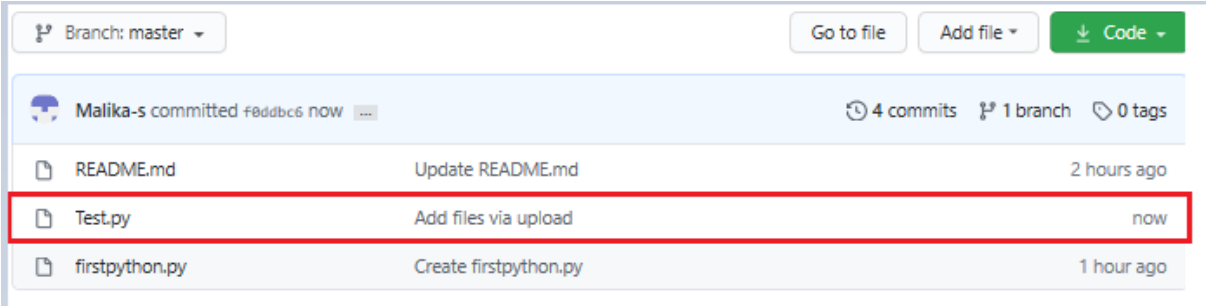
Step 2: Click on choose your files and choose any files from your computer.



Step 3: Once the file finishes uploading, click on `Commit` Changes



Step 4: Now, your file is uploaded in the repository.



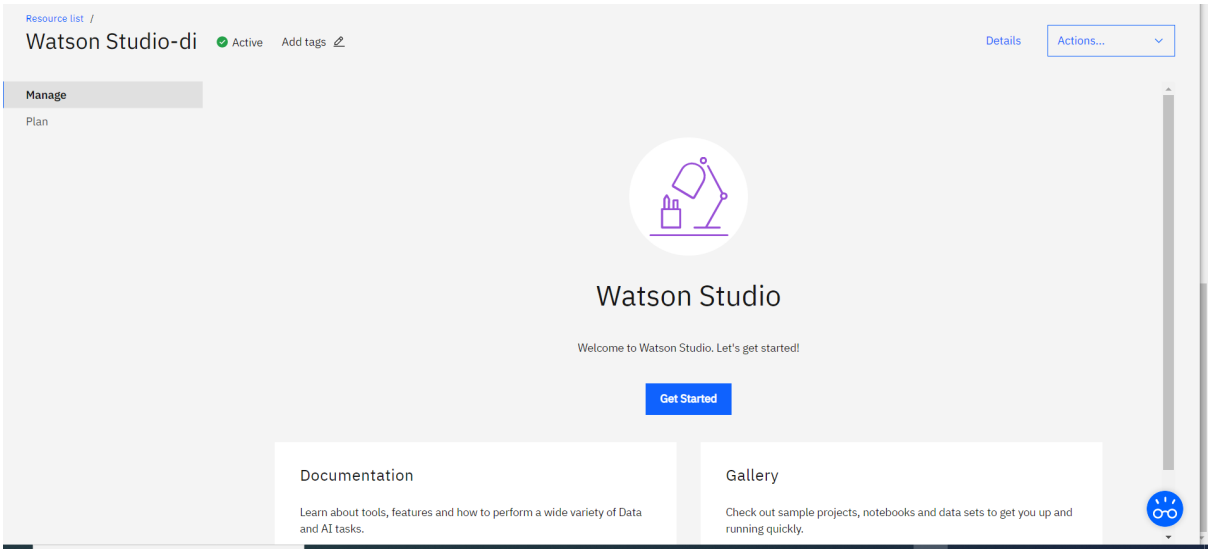
Summary

In this document, you have learned how to create a new repository, adding a new file, editing a file, and uploading a file in a repository and commit the changes.

Hands-on Lab: Publishing notebooks from Watson to Github

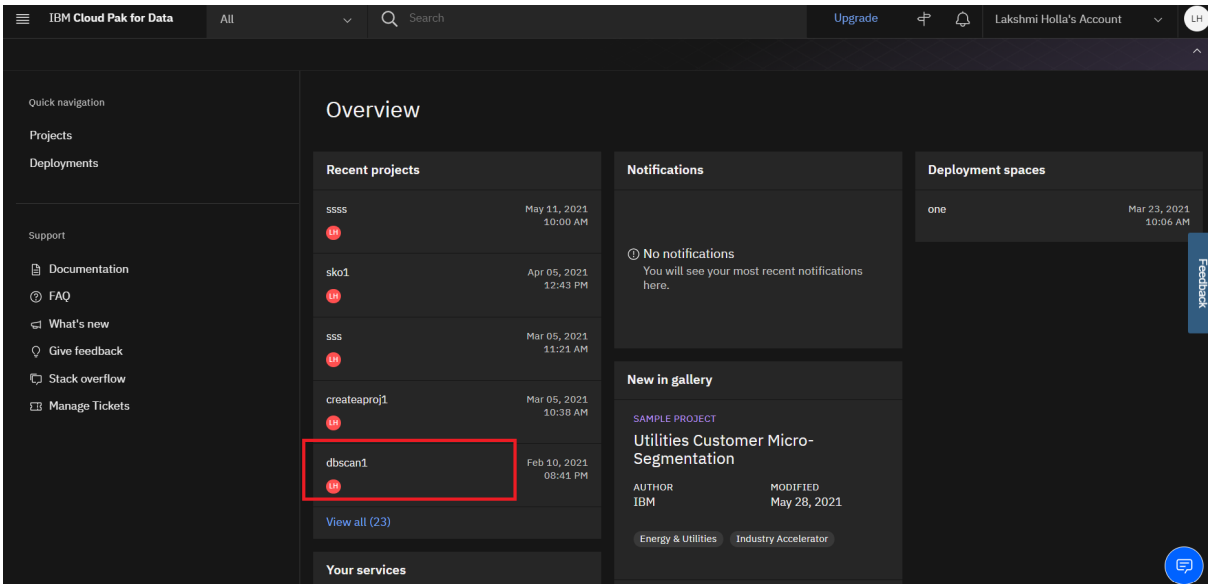
Effort: 20 mins

Step 1: Click on the Get Started Link of Watson Studio to navigate to the page where projects are listed.

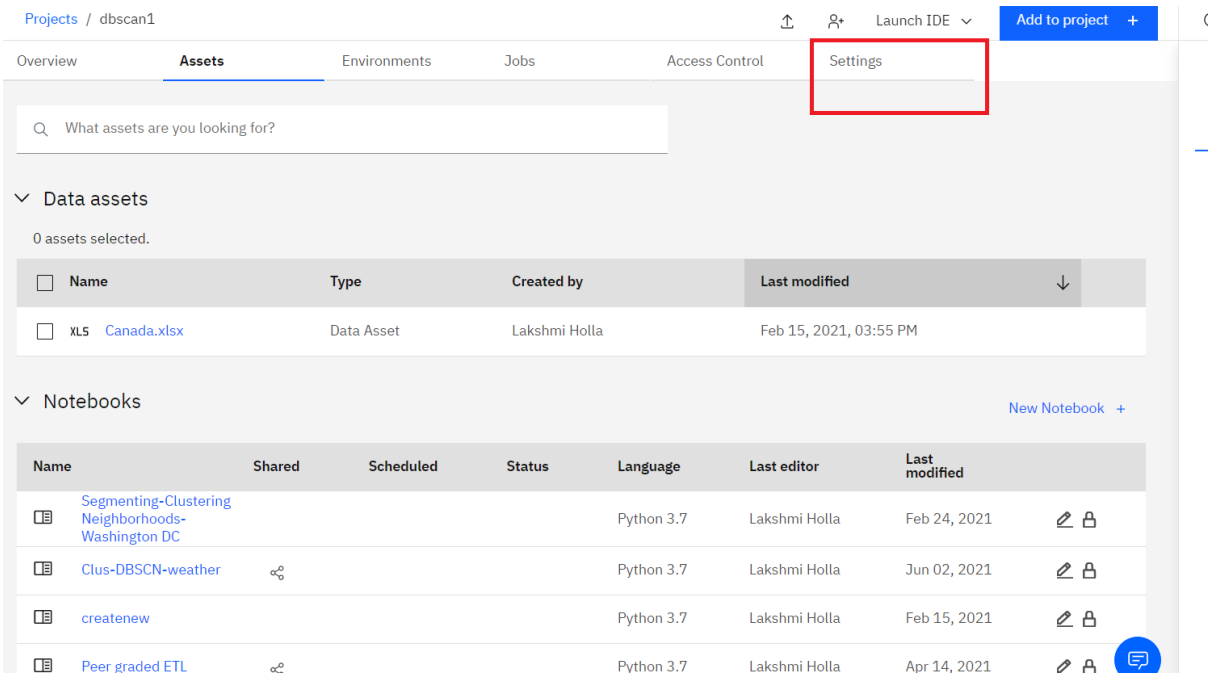


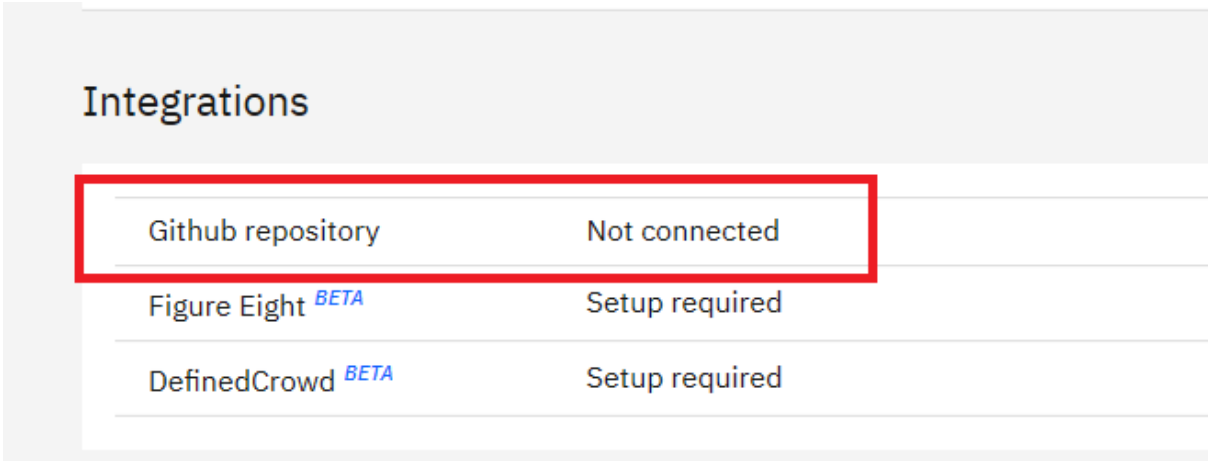
Step 2: Next click on the project which you want to integrate with github. This is the project containing your notebooks to be published on Github.

Note:This is just an example of a project . The project names will differ according what you have created.

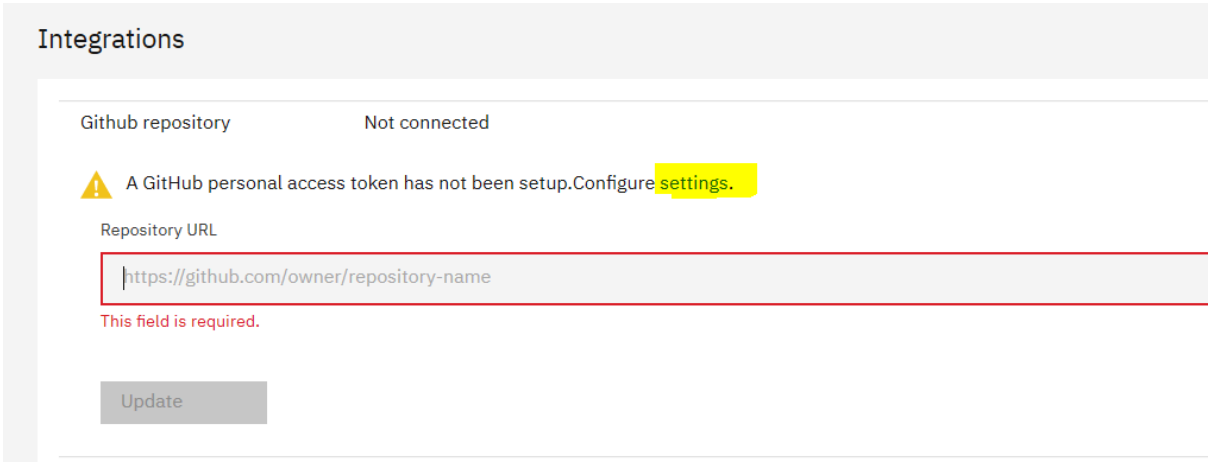


Step 3: Next click on the Settings Tab and scroll down to check the Integrations section.



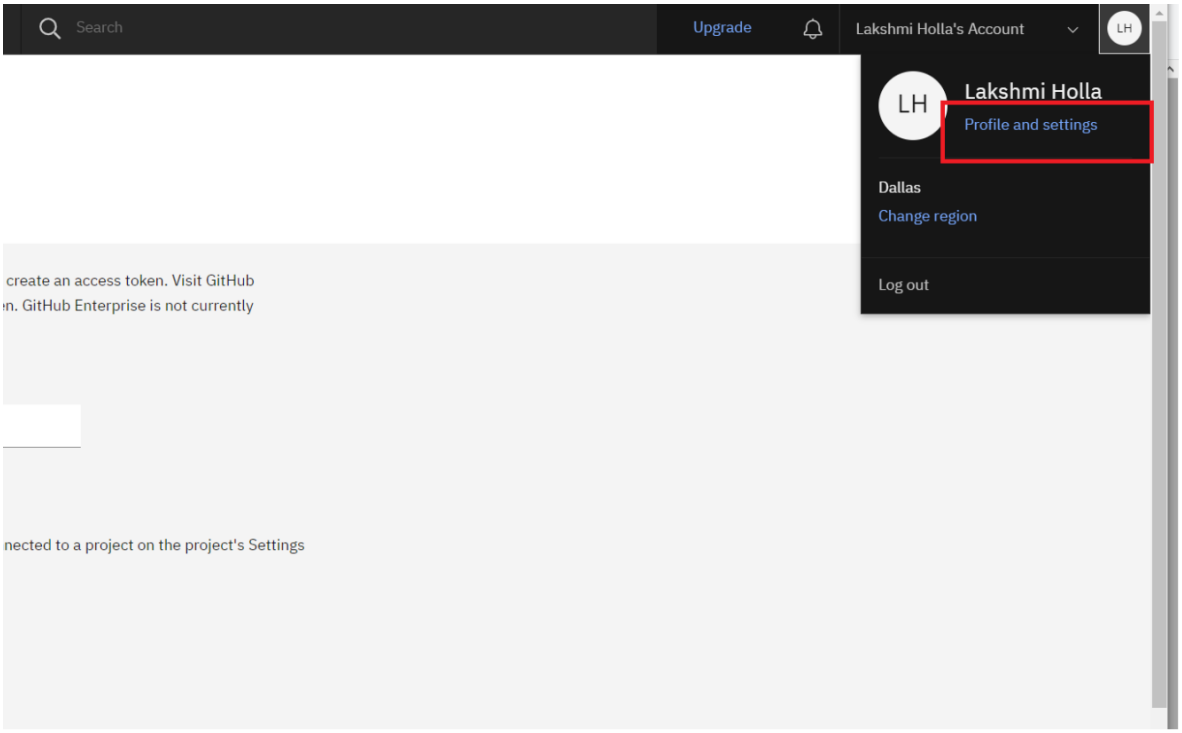


Step 4:Click on the GitHub repository on the integrations section .When you click on the textbox under the Repository url you will get the error as shown below:

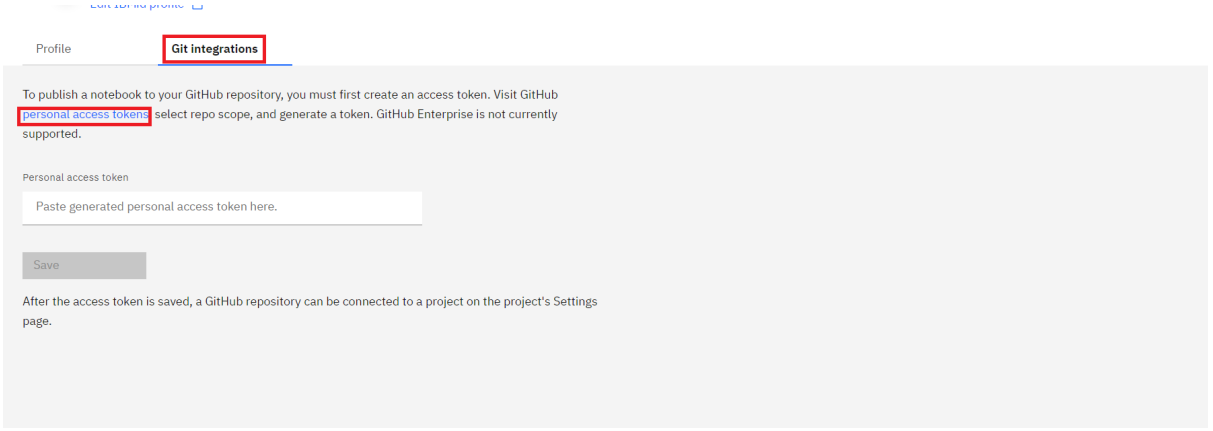


To create and configure the personal access token follow the subsequent steps.

Step 5:Next click on Profiles and Settings right below your initials on the top right screen.

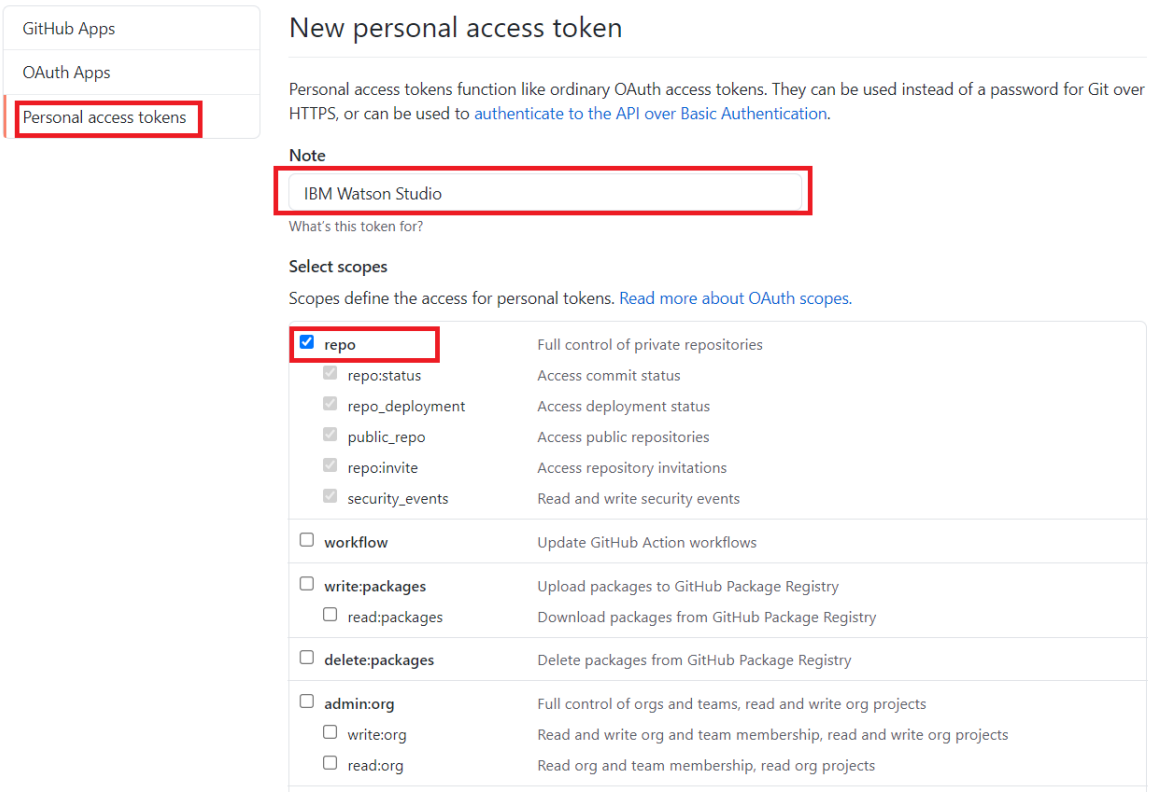


Step 6:Click on the gitIntegrations tab.



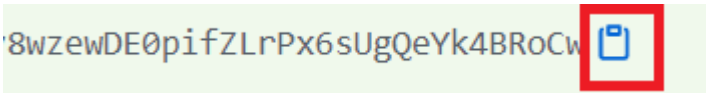
Step 7:Click on personal access token. This will prompt you to enter your github password and login to github.

Later create a token by specifying the name in the Note and choose repo scope.

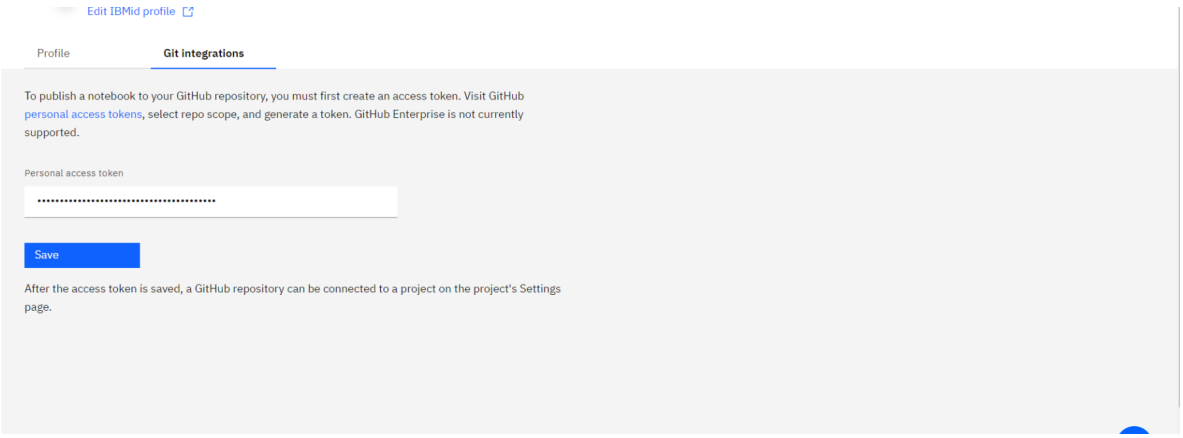


Scroll down and click ‘Generate Token’.

Step 8: Copy the token generated.



Step 9: Paste it under the Personal Access token in the Git Integrations tab of Watson and click on save.

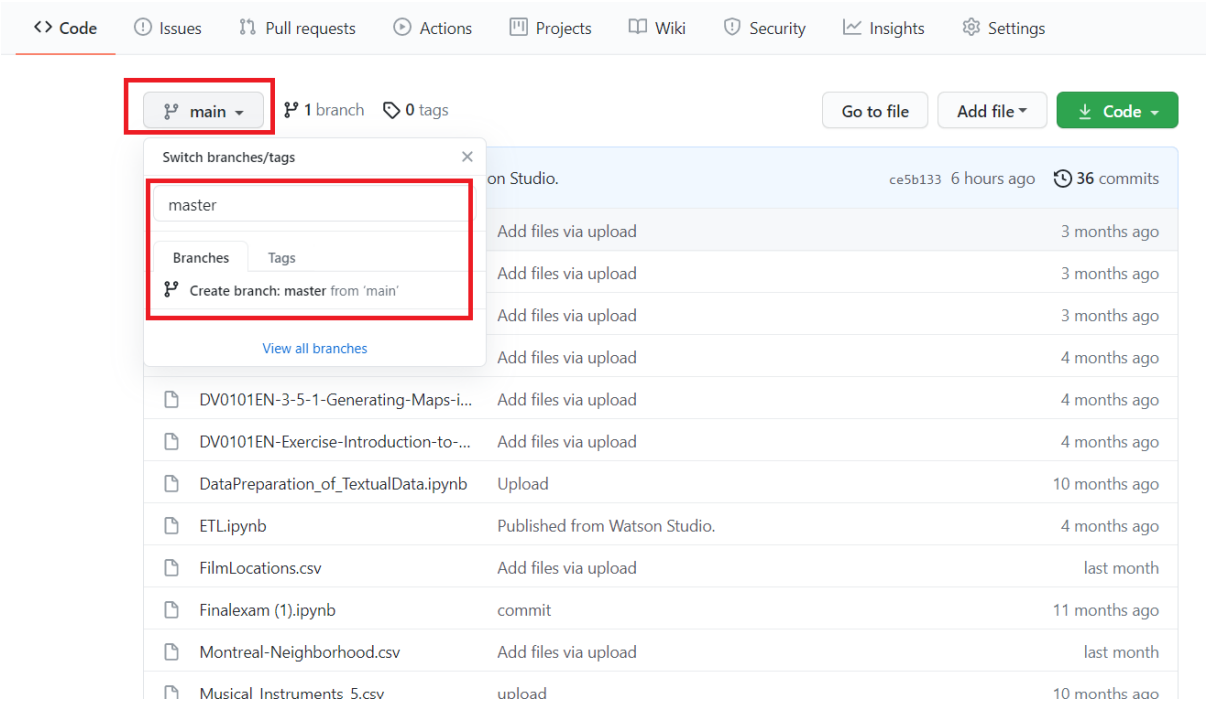


Step 10: Go back to your github repository by clicking this link.
Select your repository on the left pane.

Create a branch called master under your repository.

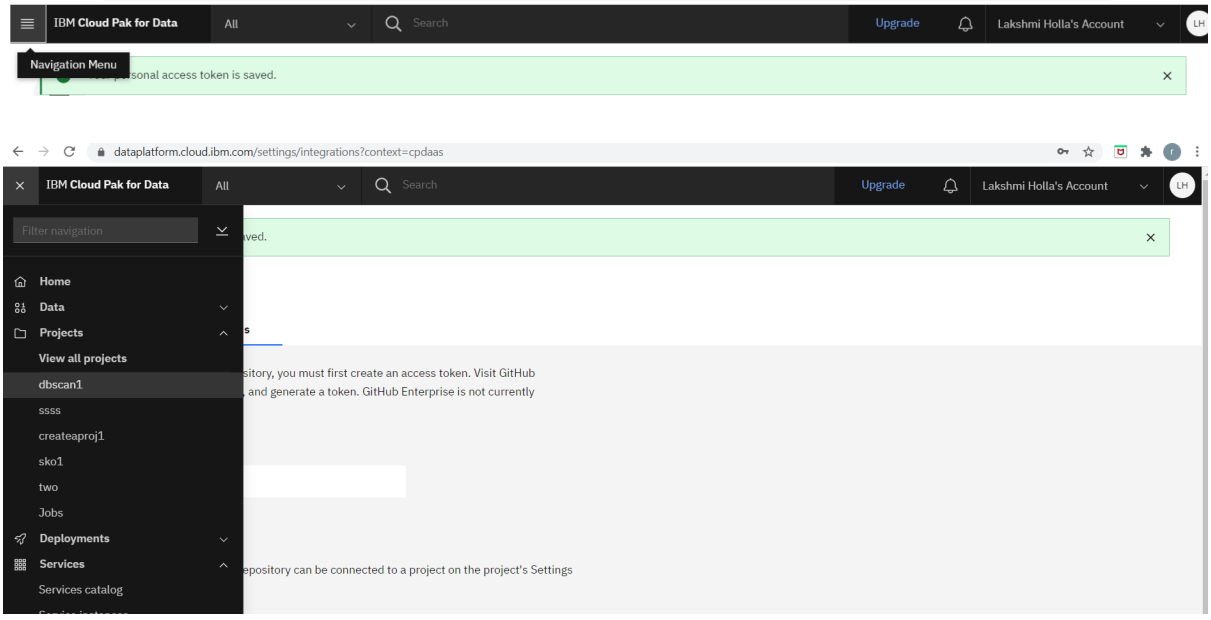
Step 11: Click the button on the left side of your screen that shows the current branch as "main" and has a dropdown arrow.

Step 12: Create a new branch called "master"



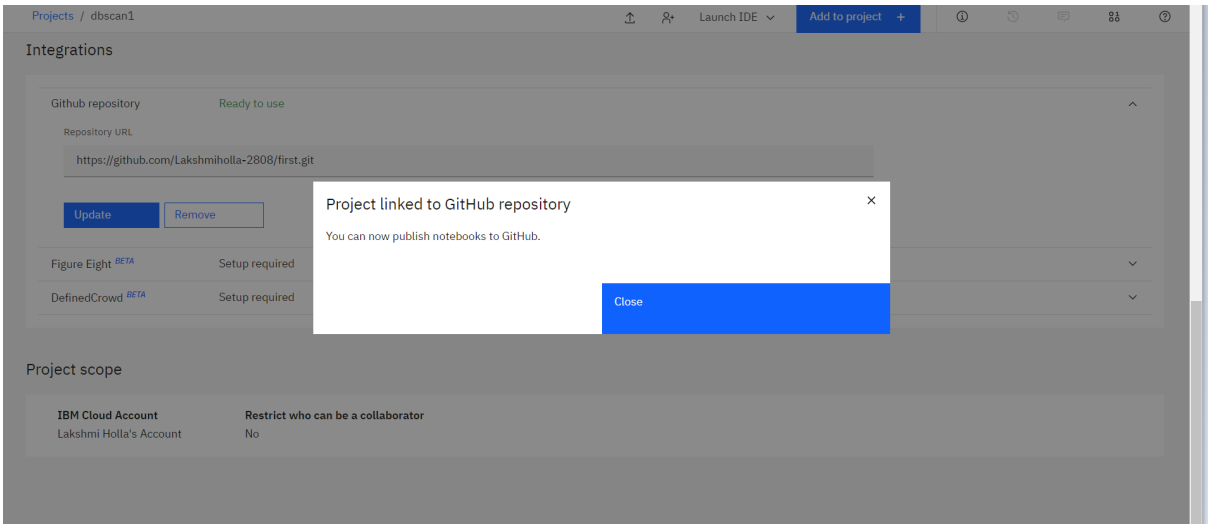
This is necessary because when you connect a repository to a project in Watson Studio and try to publish a notebook to GitHub, it automatically pushes your notebook to the master branch. If you only have a main branch in your repository and not one titled "master," Watson will not have anywhere to push your notebook.

Step 13: Once done with this step navigate to your project on watson by clicking the Navigation menu.



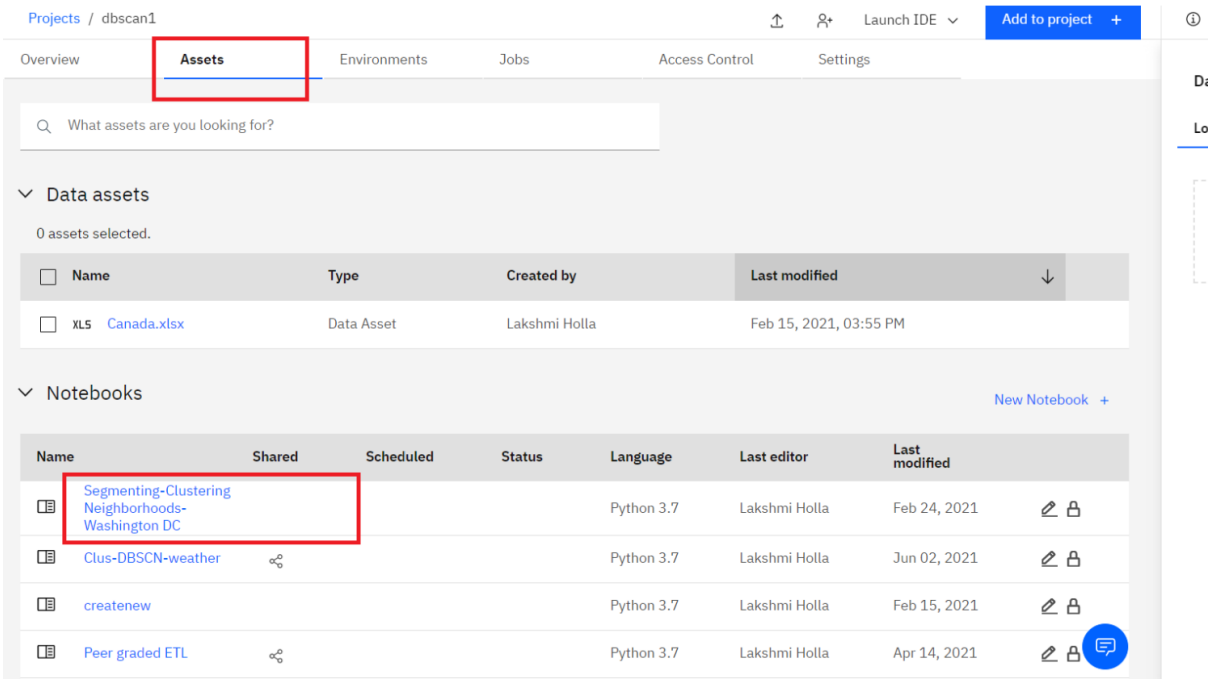
Step 14: Go to the settings tab and provide the github link of your repository.

If you have not yet created the repository follow the steps given in the link Github Repository Creation for creating a repository.



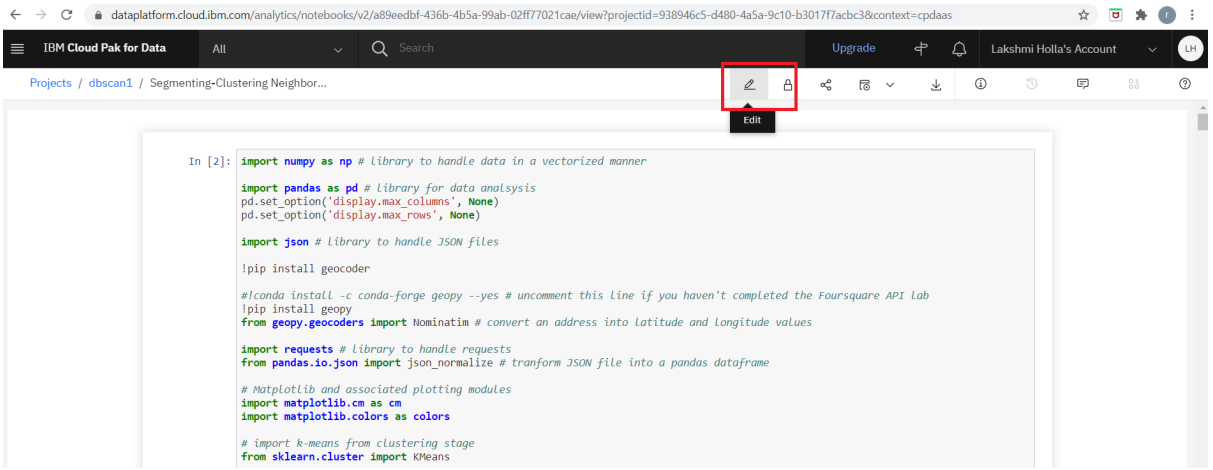
If update is not enabled, click outside text box which will enable it and then click ‘Update’.

Step 15: Navigate to the Assets tab of your project and then click on the notebook which you want to publish.

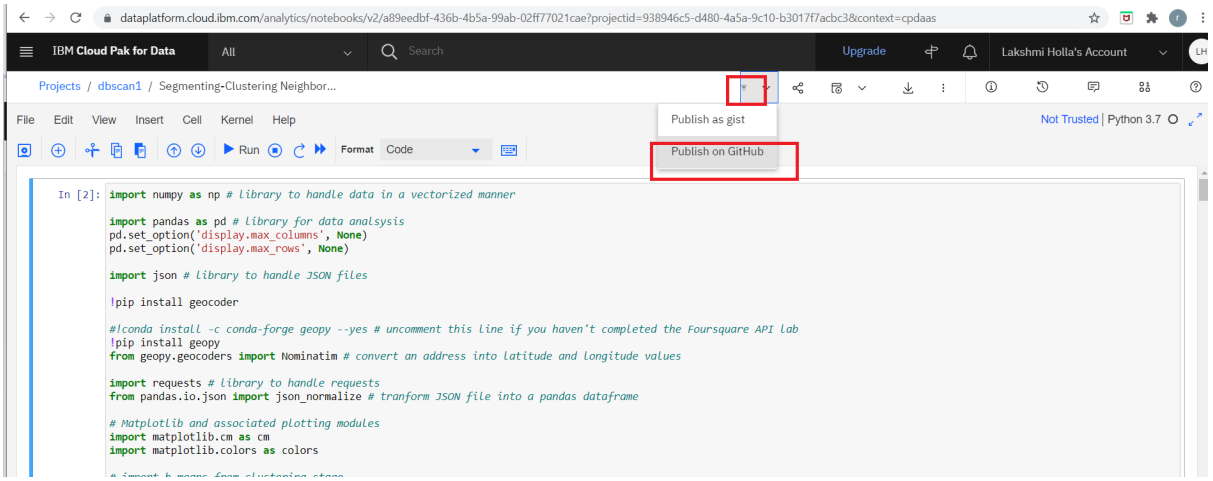


This will open your notebook.

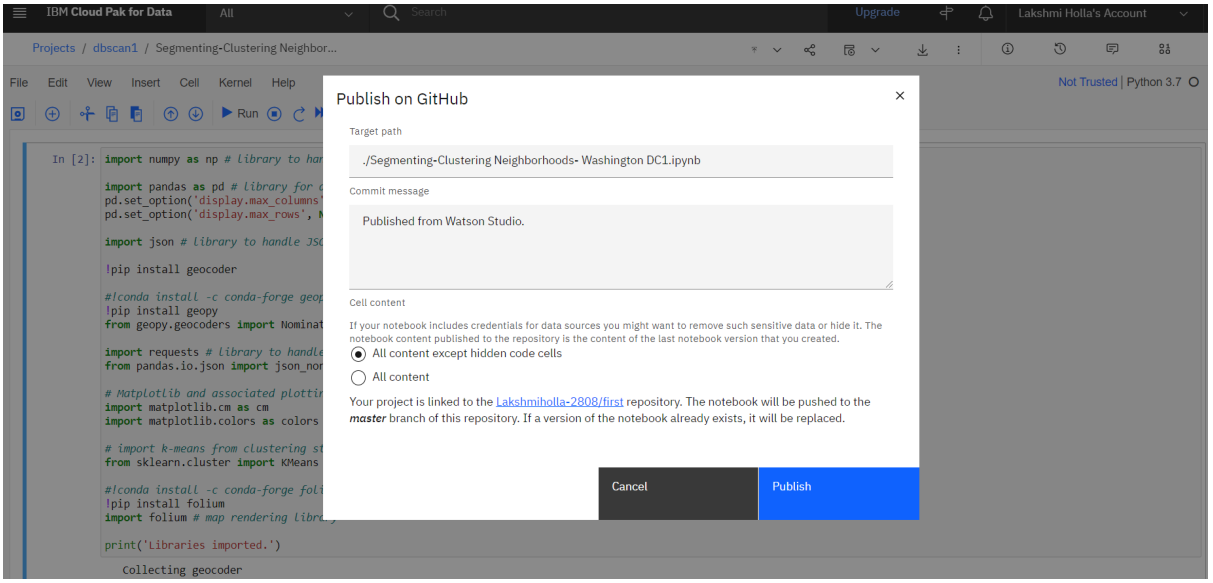
Step 16: Click on the edit option to edit your notebook.



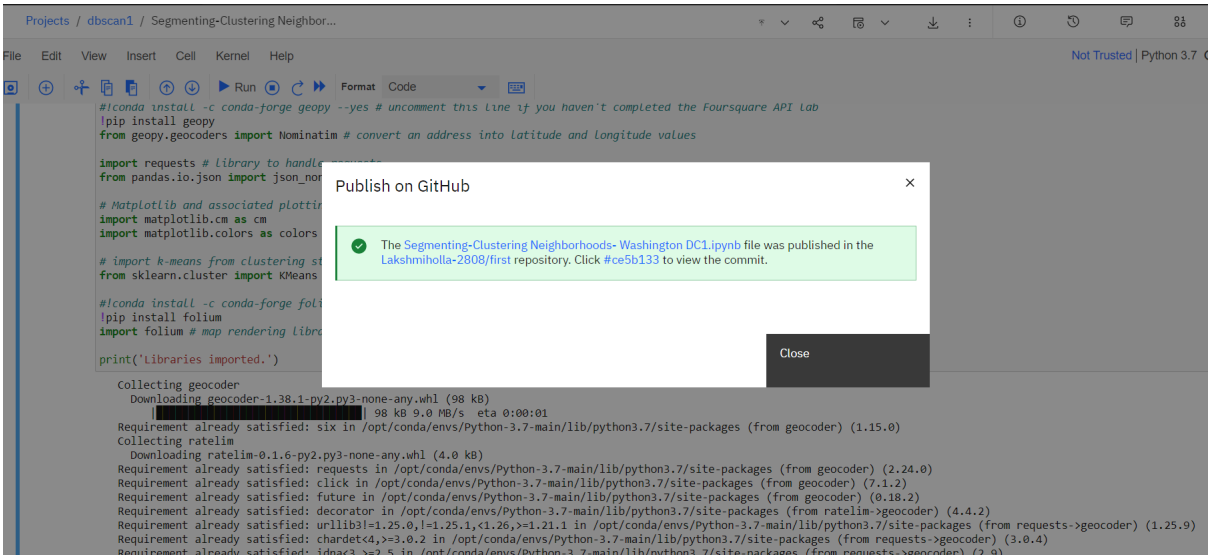
Step 17: Select Publish on github.



Step 18: Next click on Publish to publish your notebook



Step 19: This will publish your notebook to github.



Author(s)