

# Hands-on Lab: Getting Started with GitHub

Estimated time: **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 and repo
4. Edit and create a file
5. Upload a file and Commit

## GitHub Overview

First, let's introduce you to GitHub. GitHub is a collection of folders and files. It is a Git repository hosting service, but it adds many of its own features. Git is a command-line tool. It hosts and maintains a server via command line. GitHub provides this Git server and a Web-based graphical interface for you. It also provides access control and collaboration features, such as wikis and basic task management tools for every project. In addition, 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 hosting Open Source projects.

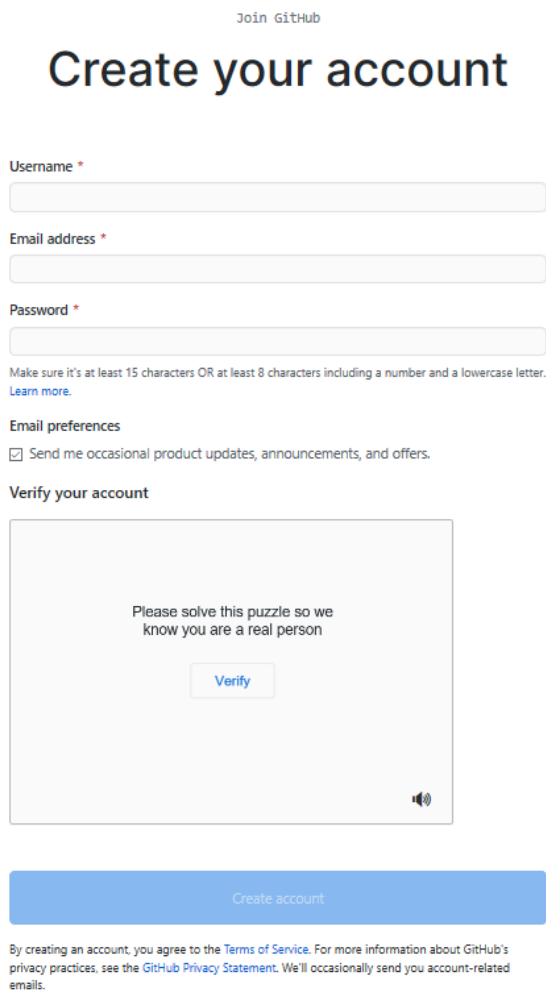
## Exercise 1: Creating a GitHub Account

Please use the following steps to create an account on 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 log in to your account.

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

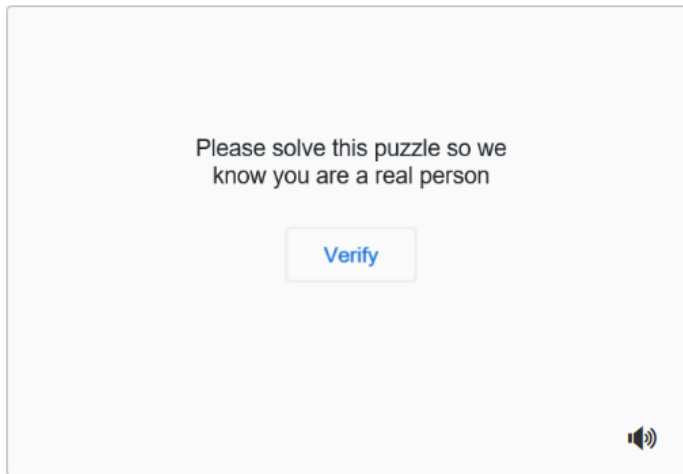


The screenshot shows the GitHub 'Create your account' page. At the top, there is a link 'Join GitHub'. The main heading is 'Create your account'. Below this, there are three input fields: 'Username \*', 'Email address \*', and 'Password \*'. Below the password field, there is a note: 'Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter. [Learn more.](#)'. Below the input fields, there is a section for 'Email preferences' with a checkbox labeled 'Send me occasional product updates, announcements, and offers.' which is checked. Below this, there is a section for 'Verify your account' which contains a large box with the text 'Please solve this puzzle so we know you are a real person' and a 'Verify' button. At the bottom of the form, there is a large blue button labeled 'Create account'. Below the button, there is a small disclaimer: 'By creating an account, you agree to the [Terms of Service](#). For more information about GitHub's privacy practices, see the [GitHub Privacy Statement](#). We'll occasionally send you account-related emails.'

Click **Create account**.

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

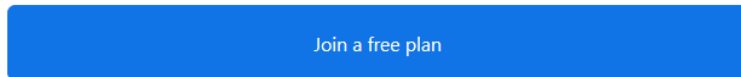
### Verify your account



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

#### Email preferences










☒ Send me occasional product updates, announcements, and offers.



Step 5: Select the details as shown below and click Complete setup.

## What do you plan to use GitHub for?

(Select up to 3)

|   |   |   |
|---|---|---|
| <br>Learn to code                      | <br>Learn Git and GitHub       | <br>Host a project (repository)        |
| <br>Create a website with GitHub Pages | <br>Collaborating with my team | <br>Find and contribute to open source |
| <br>School work and student projects   | <br>Use the GitHub API         | <br>Other                              |

## I am interested in:

We'll connect you with communities and projects that fit your interests.

For example: `zepplin` `elm` `apn`

Complete setup

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

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



## Please verify your email address

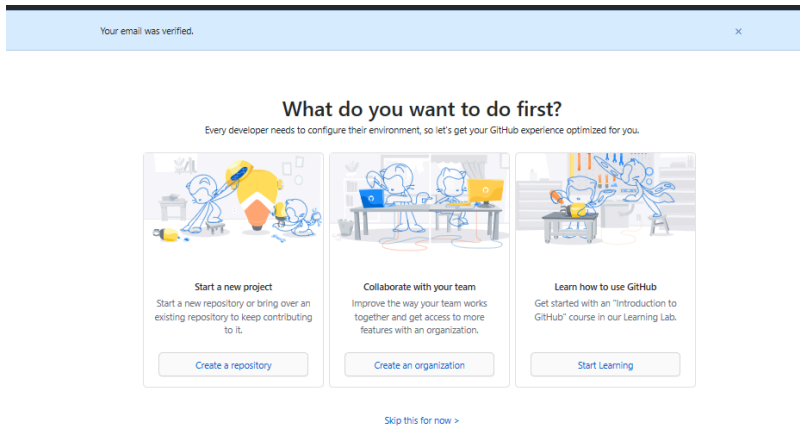
Before you can contribute on GitHub, we need you to verify your email address.

An email containing verification instructions was sent to **Your email address**

[Resend verification email](#)

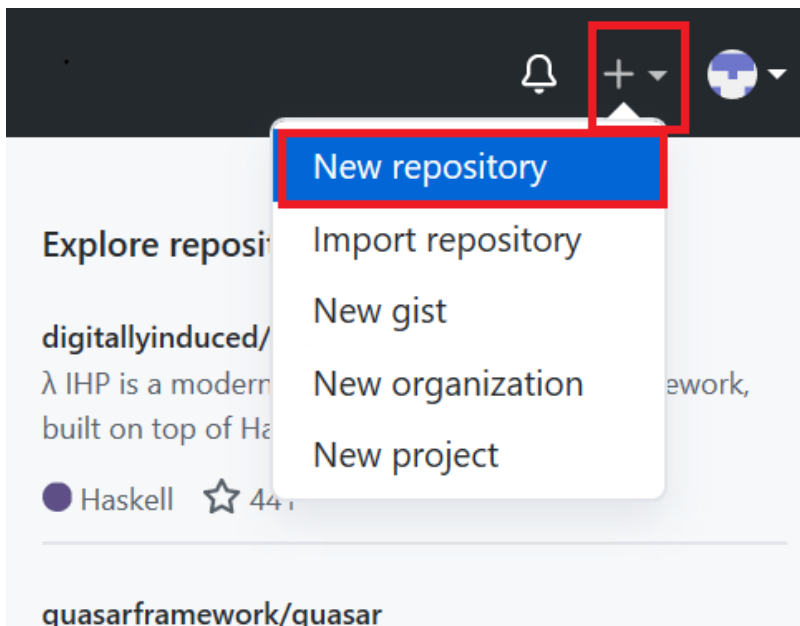
[Change your email settings](#)

Email is verified.



## Exercise 2: Adding a project and repo

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



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

### Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner \* Malika-s / Repository name \* testrepo ✓

Great repository names are short and memorable. Need inspiration? How about [urban-octo-waffle?](#)

Description (optional)

☒ Public  
Anyone on the internet can see this repository. You choose who can commit.

☐ Private  
You choose who can see and commit to this repository.

Skip this step if you're importing an existing repository.

☒ Initialize this repository with a README  
This will let you immediately clone the repository to your computer.

Add .gitignore: None Add a license: None ⓘ

[Create repository](#)

Click [Create repository](#).

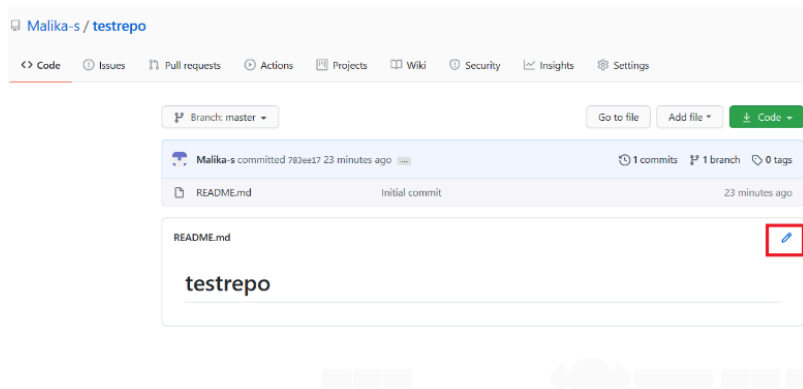
Now, you will be redirected to the repository you have created.

Let us start editing the repository.

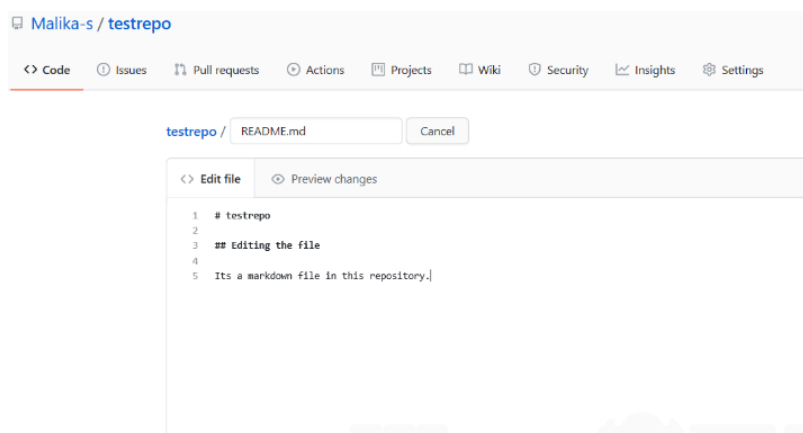
## Exercise 3: Create and edit a file

### Exercise 3a: Edit a file

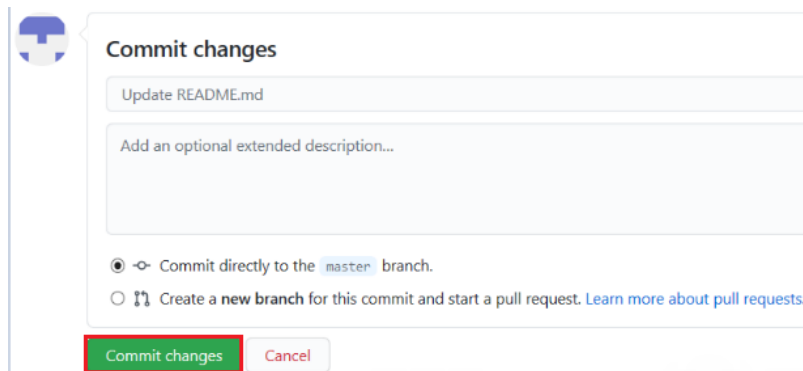
Step 1: Once the repository is created, the root folder of your repository is listed by default, and has just one file, `ReadMe.md`. Click the pencil icon to edit the file.



Step 2: Add some text to the file.



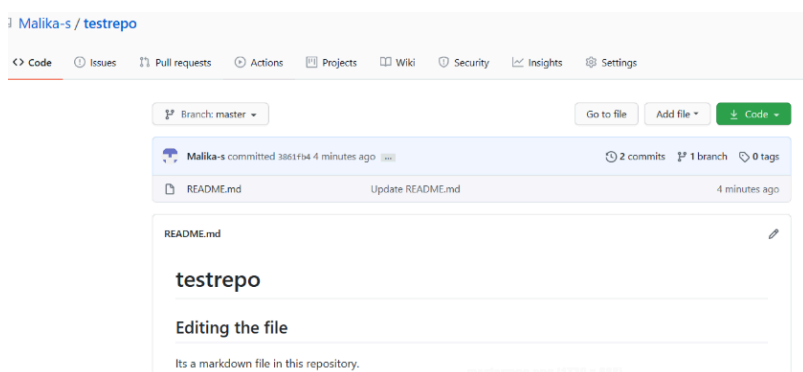
Step 3: Scroll down the page after adding the text and click **Commit Changes**.



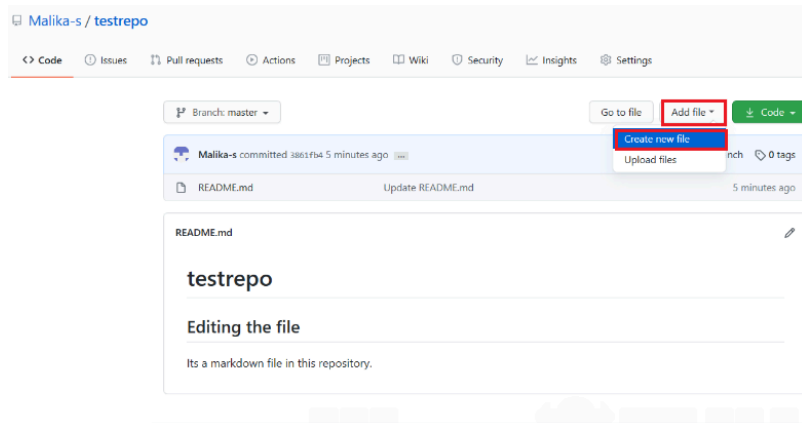
Now, check that your file is edited with the new text.

### Exercise 3b: Create a new file

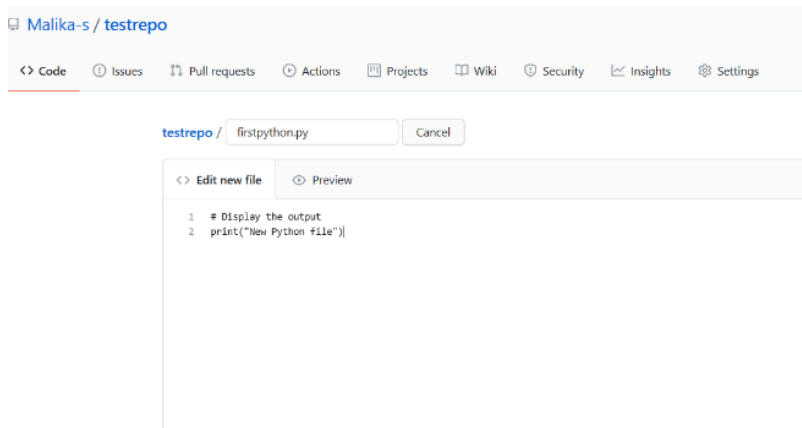
Step 1: Click the repository name to return to the master branch, like in this testrepo.



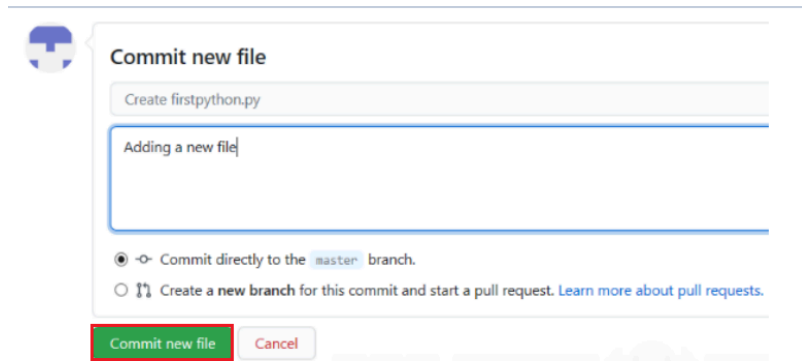
Step 2: Click **Add file** and select **Create new file** to create a file in the repository.



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



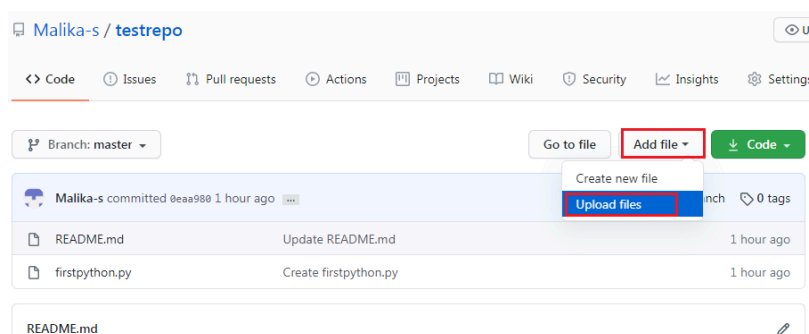
Step 4: Scroll down the page after adding the text. Add description of the file (optional) and click **Commit new file**.



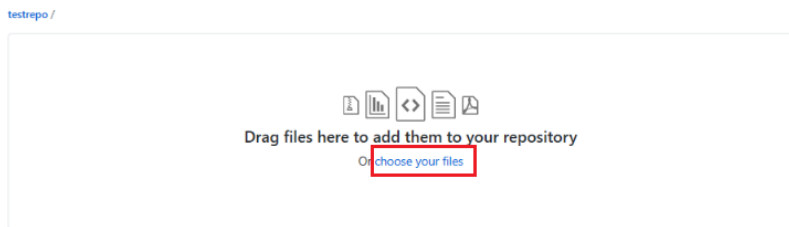
Step 5: Your file is now added to your repository, and the repository listing shows when the file was added and changed.

## Exercise 4: Upload a file & Commit

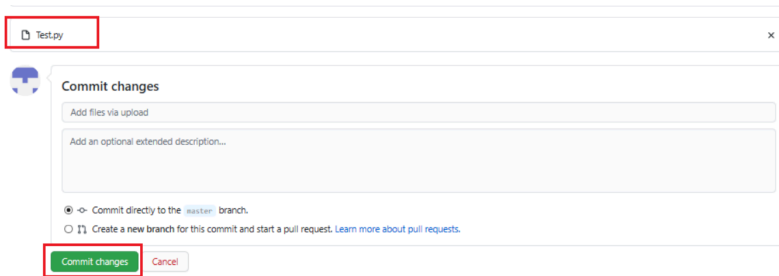
Step 1: Click **Add file** and select **Upload files** to upload a file (any .txt, .ipynb, .png file) in the repository from the local computer.



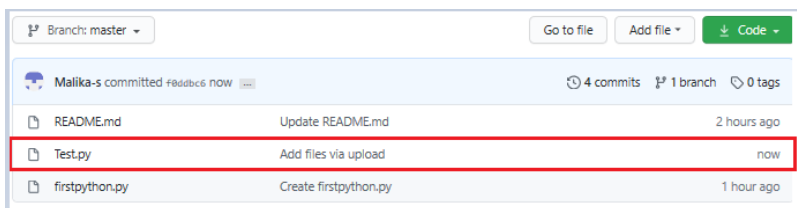
Step 2: Click **choose your files** and select any files from your computer.



Step 3: Once the file finishes uploading, click 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, add a new file, edit a file, upload a file in a repository, and commit the changes.

## Author(s)

Romeo Kienzler

Malika Singla

## Other Contributor(s)

Rav Ahuja



# Skills Network