Lab: Getting started with GitHub

Estimated time: 20 minutes

In this lab, you will get started with GitHub by creating a GitHub account and project. You will then add a file to the project using the GitHub web interface.

Objectives

After completing this lab, you will be able to:

1. Describe GitHub
2. Create a GitHub account
3. Add a project / repository
4. Create and edit a file
5. Upload and commit a file

GitHub overview

Before we learn to use GitHub, let's first review Git. *Git* is an open source version-control system for software development. It is simply a place where you can collect all your folders and files for a project. This collection of folders and files is usually called a repository.

GitHub is a Git repository hosting service, but it adds many of its own features. While Git is a command-line tool and you also need to host and maintain a server using the command line, GitHub provides the Git server for you, as well as 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.

1. Creating a GitHub account

To create an account in GitHub, complete the following steps:

1. Go to the Join GitHub page and create an account. **Note**: If you already have a GitHub account, log in now.

2. Provide the necessary details to create an account as shown below. When you have finished, click **Create account**.

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

Verify your account

Please solve this puzzle so we
know you are a real person

Verify

4. After you have verified, click **Join a Free Plan**.

Email preferences

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

Join a free plan

5. Select your reasons for joining GitHub and click **Complete Setup**.

What do you plan to use GitHub for?
(Select up to 3)

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

I am interested in:

languages, frameworks, industries

We'll connect you with communities and projects that fit your interests.
For example: zeplin   elm   apm

Complete setup

6. You will receive a verification email from GitHub. Click the enclosed link to verify your email. **Note**: If you do not receive a 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**

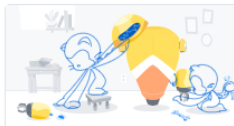[Resend verification email] [Change your email settings]

When you have verified your email, you will see a screen that looks like this:

Your email was verified. ×

What do you want to do first?
Every developer needs to configure their environment, so let's get your GitHub experience optimized for you.

Start a new project
Start a new repository or bring over an existing repository to keep contributing to it.

[Create a repository]

Collaborate with your team
Improve the way your team works together and get access to more features with an organization.

[Create an organization]

Learn how to use GitHub
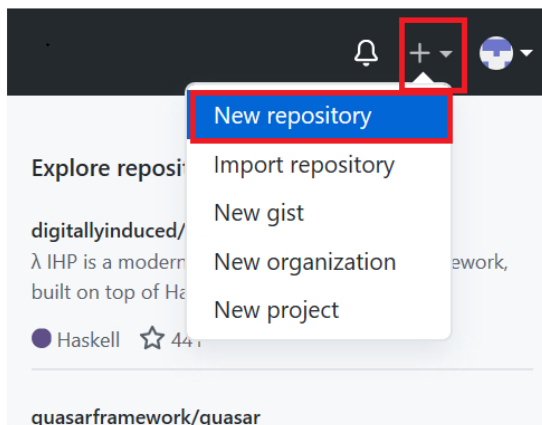Get started with an "Introduction to GitHub" course in our Learning Lab.

[Start Learning]

Skip this for now >

2: Add a project / repository

To add a new repository, complete the following steps:

1. At the top right of the GitHub home page, click on the "+" icon and select **New repository**.

2. Enter a repository name and select the **Initialize this repository with a README** check box.

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere?
Import a repository.

Owner *   Repository name *

🐙 Malika-s ▾   /   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

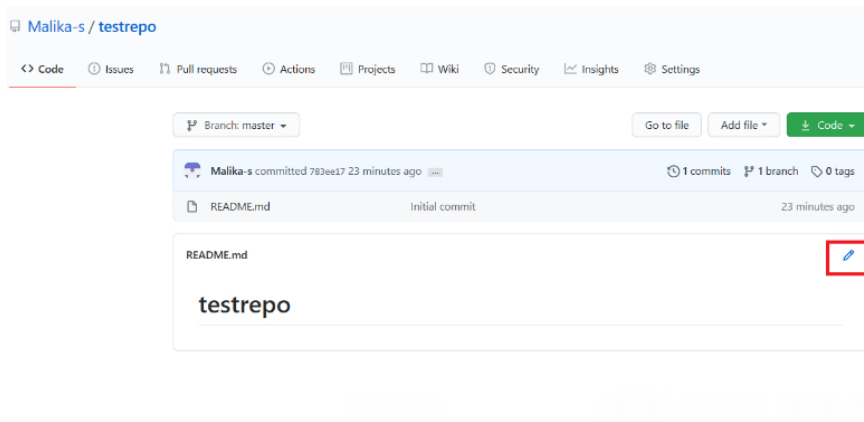3. Click **Create repository**. The repository is created and its home page is displayed.

Next, you'll start editing the repository.
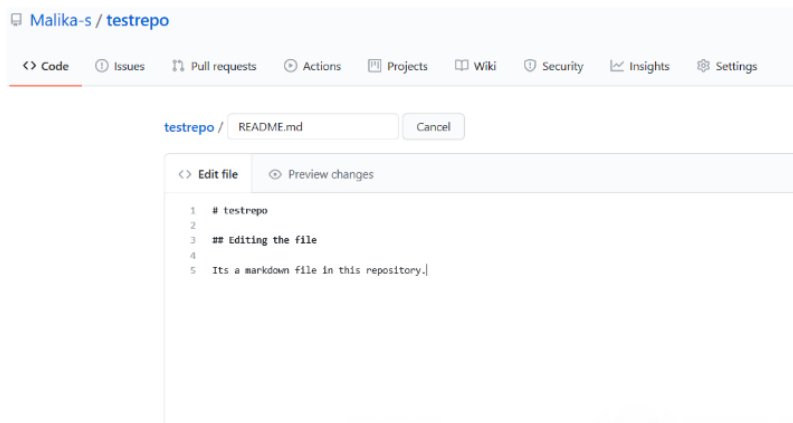
3: Create and edit a file

Edit a file

Although you will normally create a file before you edit it, in this case the `README.md` file has already been created for you. To edit that file, complete the following steps:
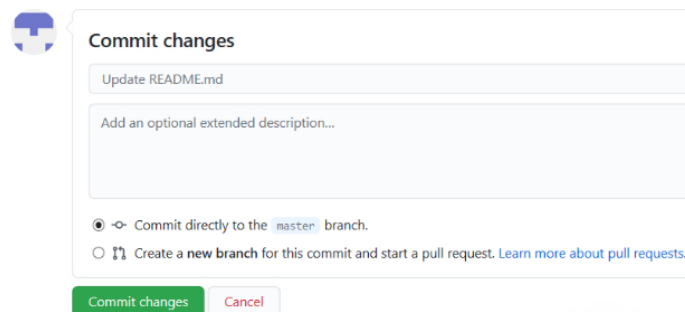
1. Your repository root folder contains just one file: `README.md`. Click the pencil icon at the right to edit the file.
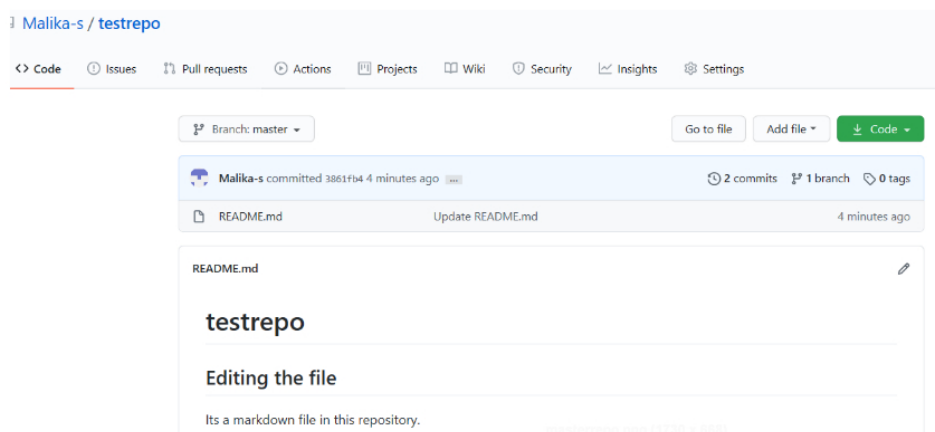

2. Add some text to the file.

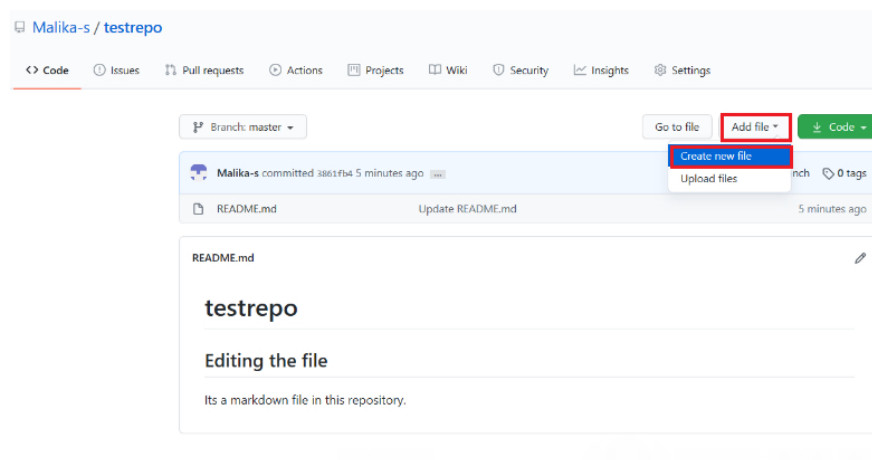3. Scroll to the bottom of the file and click **Commit changes**.

4. Confirm that the text you added to the file has been saved.
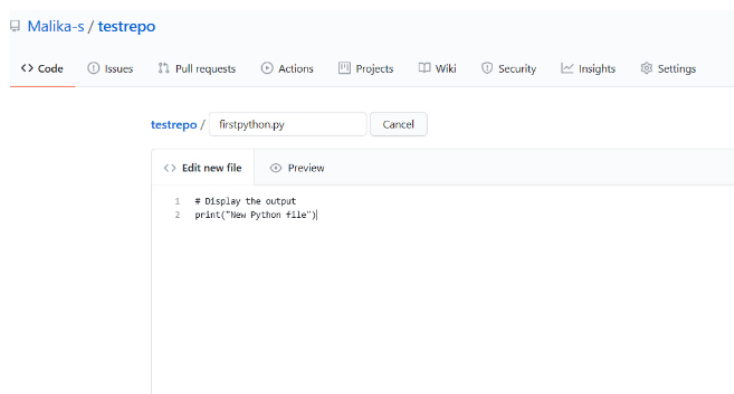
Create a new file

1. Click on the repository name to go back to the main branch, similar to this repository called `testrepo`.

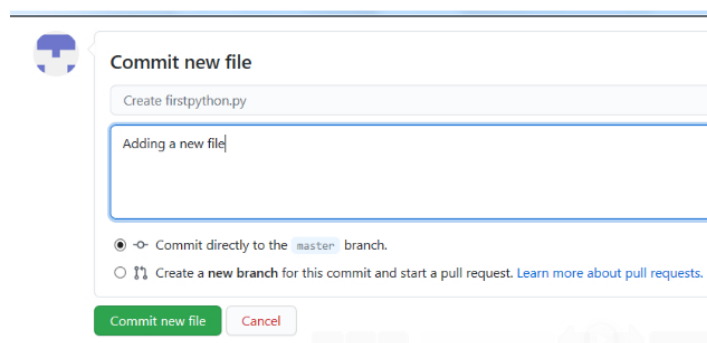
2. Click **Add file** and select **Create new file** to create a new file in the repository.

3. Enter a file name and extension; for example, `firstpython.py`. Add the lines of code displayed in the following image to your file:

4. Scroll to the bottom of the page. You can optionally add a description of your update (for example, "Adding a new file"). Click **Commit** to create your new file.
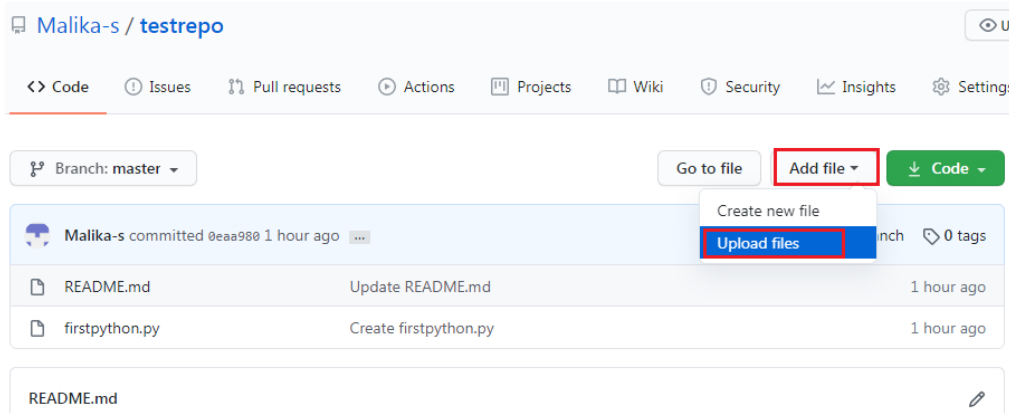

Your file is now added to your repository and the repository listing shows when the file was added or last revised.
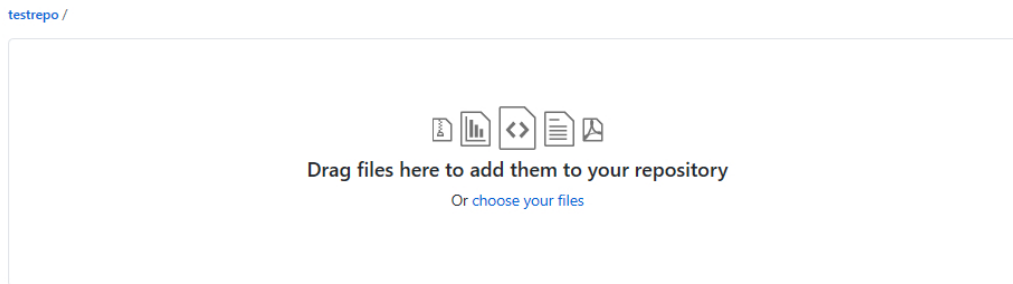
4. Upload and commit a file

To upload a local file and commit it to your repository, complete the following steps:
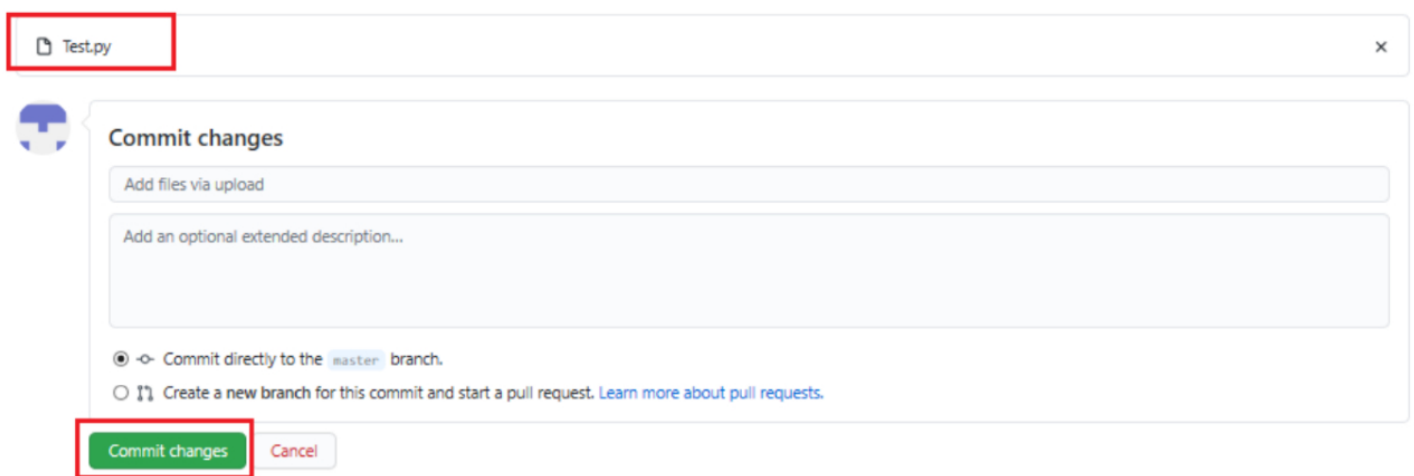
1. On your repository page, click **Add file** and then select **Upload files** to upload a file.
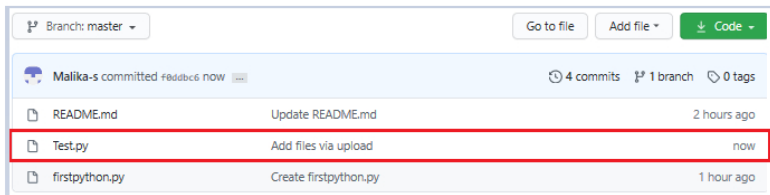

2. Click **choose your files** and choose a file from your computer. You can upload any file (for example, a .txt, .ipynb, or .png file) to the repository.

3. When the file finishes uploading, click **Commit changes**.

Your file is uploaded in the repository.

Summary

Congratulations! In this lab, you have learned how to create a new repository, add a new file, edit a file, upload a file, and commit your changes. We encourage you to continue to update your repository to become familiar with the processes that you have learned.

Tutorial details

Authors:

- Romeo Kienzler
- Malika Singla

Other contributors:

- Rav Ahuja
- Upkar Lidder

Changelog:

Date	Version	Changed by	Change Description
2020-08-18	0.6	Upkar Lidder	Updated GitHub Overview section
2020-07-16	0.5	Malika Singla	Spell check, step number added
2020-07-14	0.4	Rav Ahuja	Changed logo, updated effort, title, intro, objectives, added Authors and Changelog
2020-07-13	0.3	Malika Singla	Added to GitLab and made some formatting changes, added objectives, etc.
2020-07-03	0.2	Malika Singla	Ported to markdown and added GitHub account signup, new screenshots, etc.
2020-06-30	0.1	Romeo Kienzler	Drafted initial version