



## 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:

0. Describe GitHub
1. Create a GitHub account
2. Add a Project / Repo
3. Edit / Create a file
4. 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:

[Join GitHub](#)

# Create your account

Username \*

Email address \*

Password \*

Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter.  
[Learn more.](#)

## Email preferences

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

## Verify your account

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

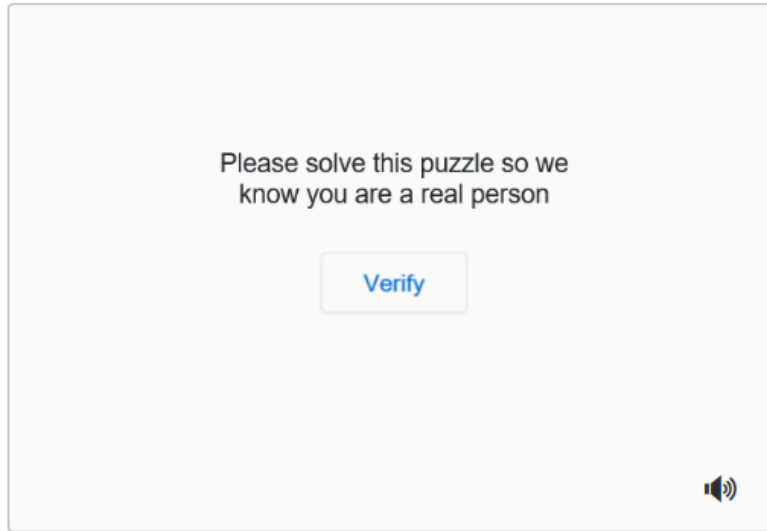
[Verify](#)[Create account](#)

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.

and 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.

Join a free plan

Step 5: Select the details as shown below 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

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`.



## 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](#)

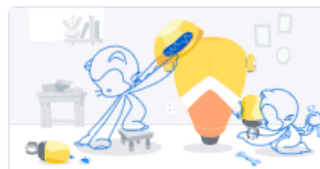
Email is verified

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

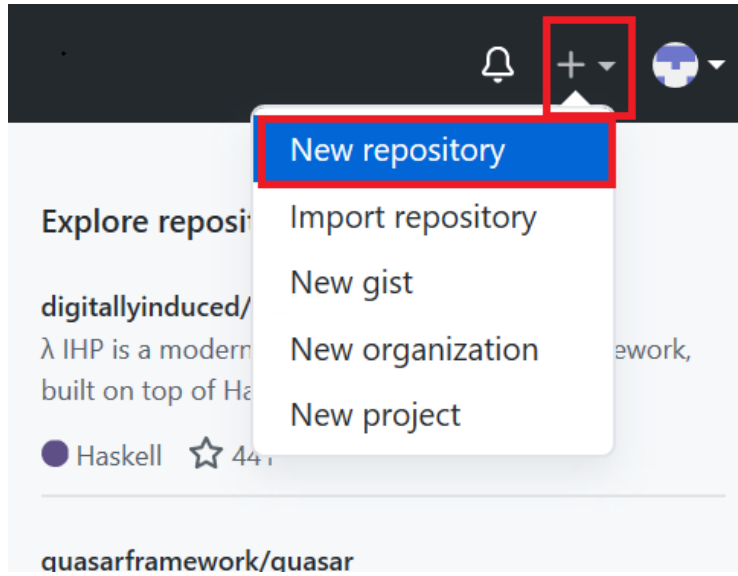
Get started with an "Introduction to GitHub" course in our Learning Lab.

[Start Learning](#)

[Skip this for now >](#)

## 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.


## 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

and click Create repository.

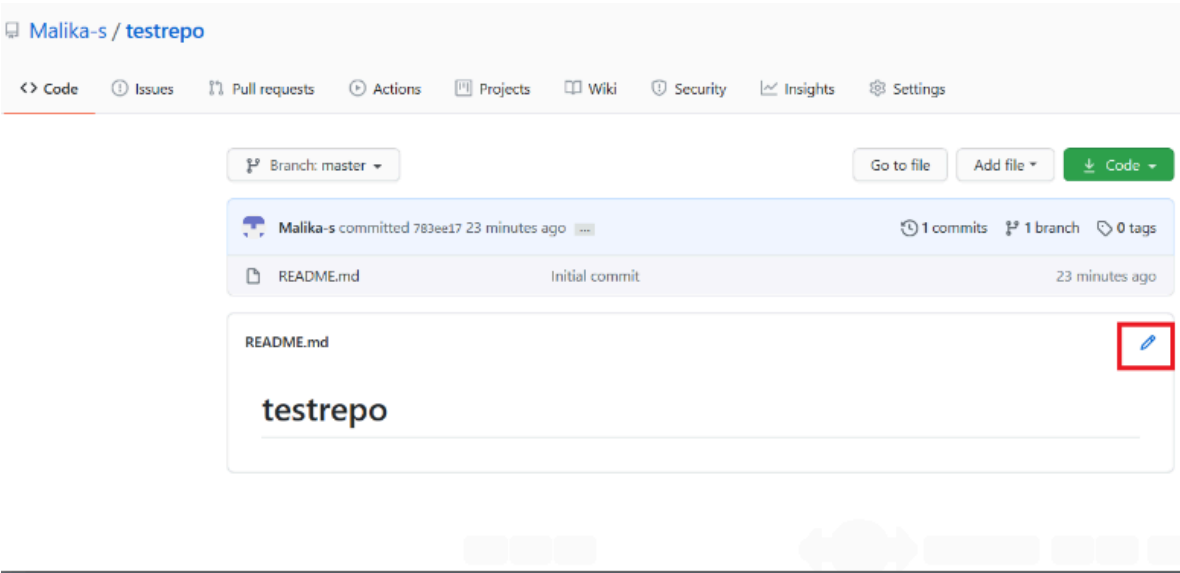
Now, you will be redirected to the repository you have created.

Let's start editing the repository.

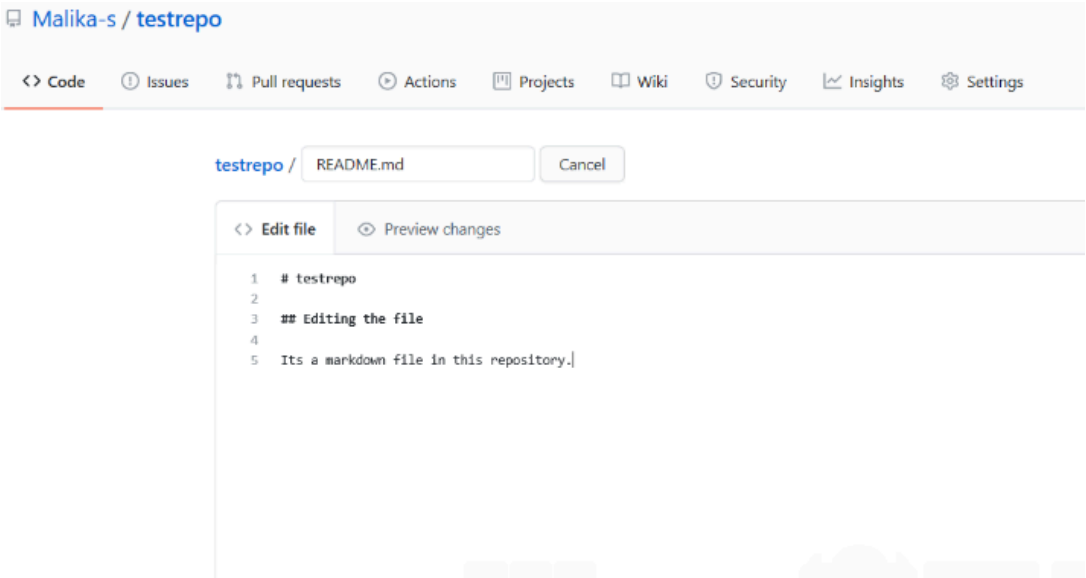
### Exercise 3: Create / edit a file

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.

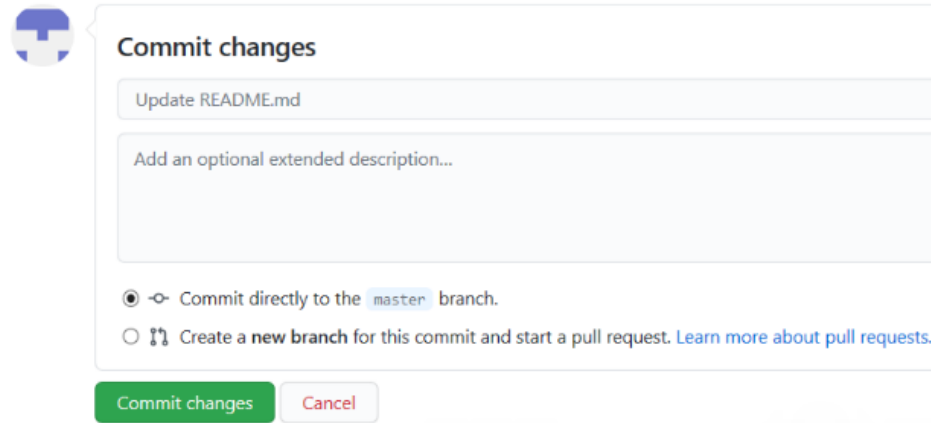


Step 2: Add text to file.



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





Commit changes

Update README.md

Add an optional extended description...

☒ Commit directly to the `master` branch.

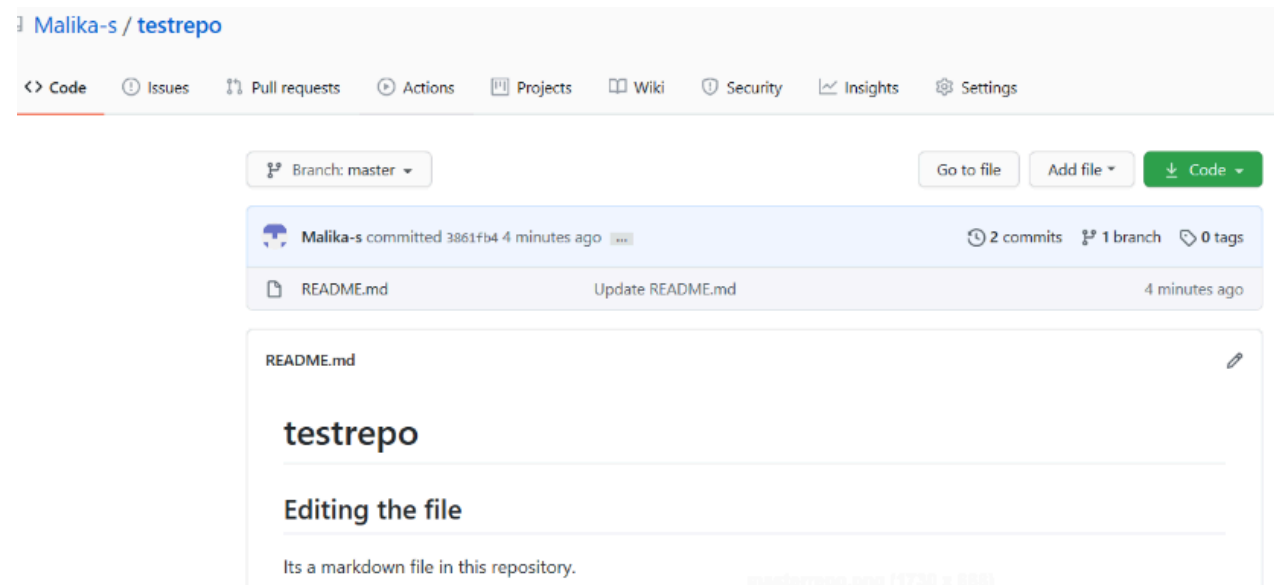
☐ Create a new branch for this commit and start a pull request. [Learn more about pull requests.](#)

Commit changes Cancel

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.



Malika-s / testrepo

<> Code Issues Pull requests Actions Projects Wiki Security Insights Settings

Branch: master Go to file Add file Code

Malika-s committed 3861fb4 4 minutes ago 2 commits 1 branch 0 tags

README.md Update README.md 4 minutes ago

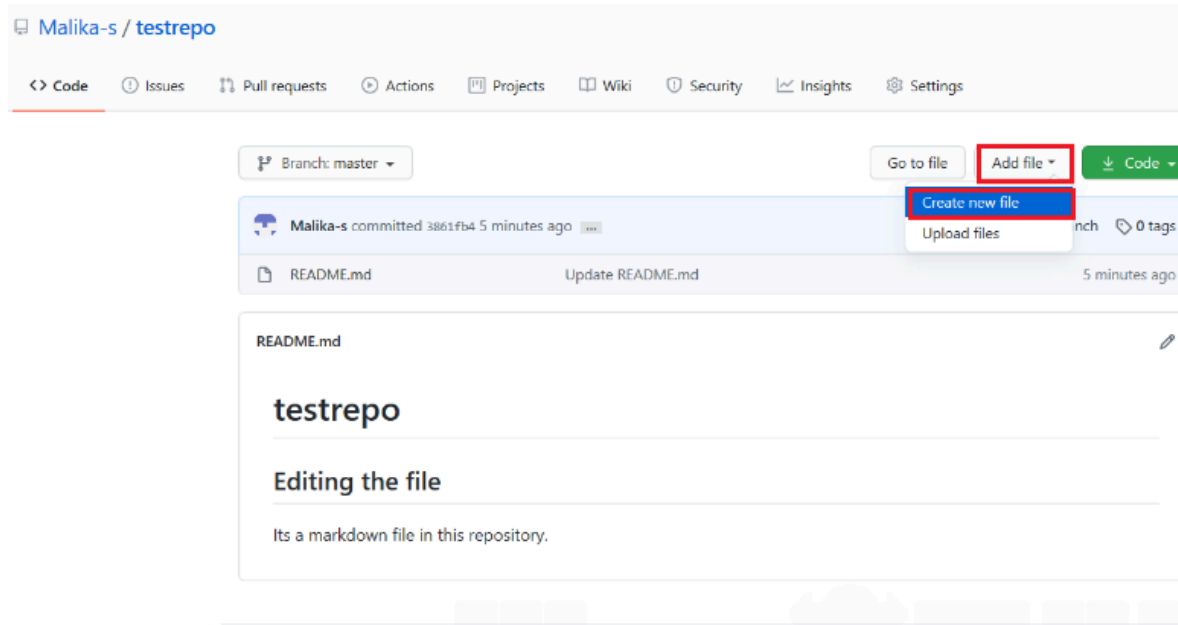
README.md

## testrepo

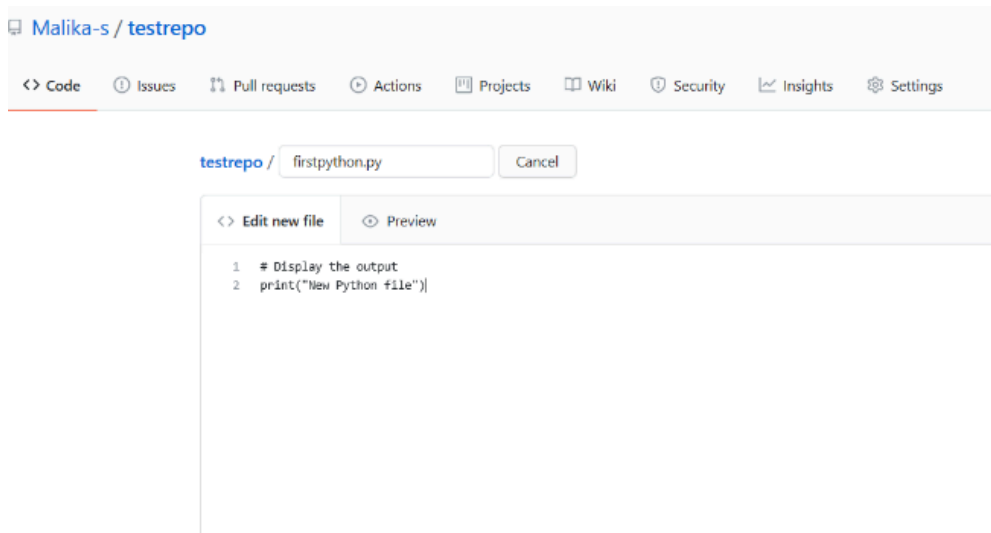
### Editing the file

Its a markdown file in this repository.

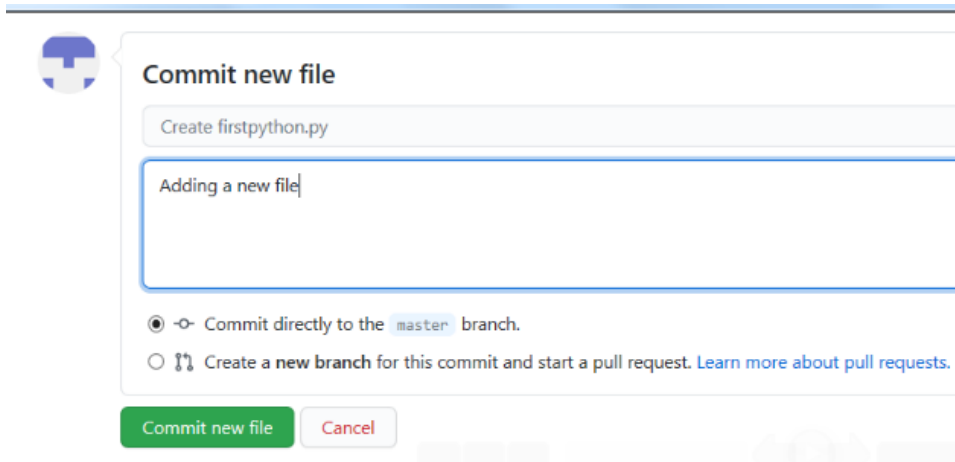
Step 2: Click Add file and select Create New file to create a file in the repository.



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.

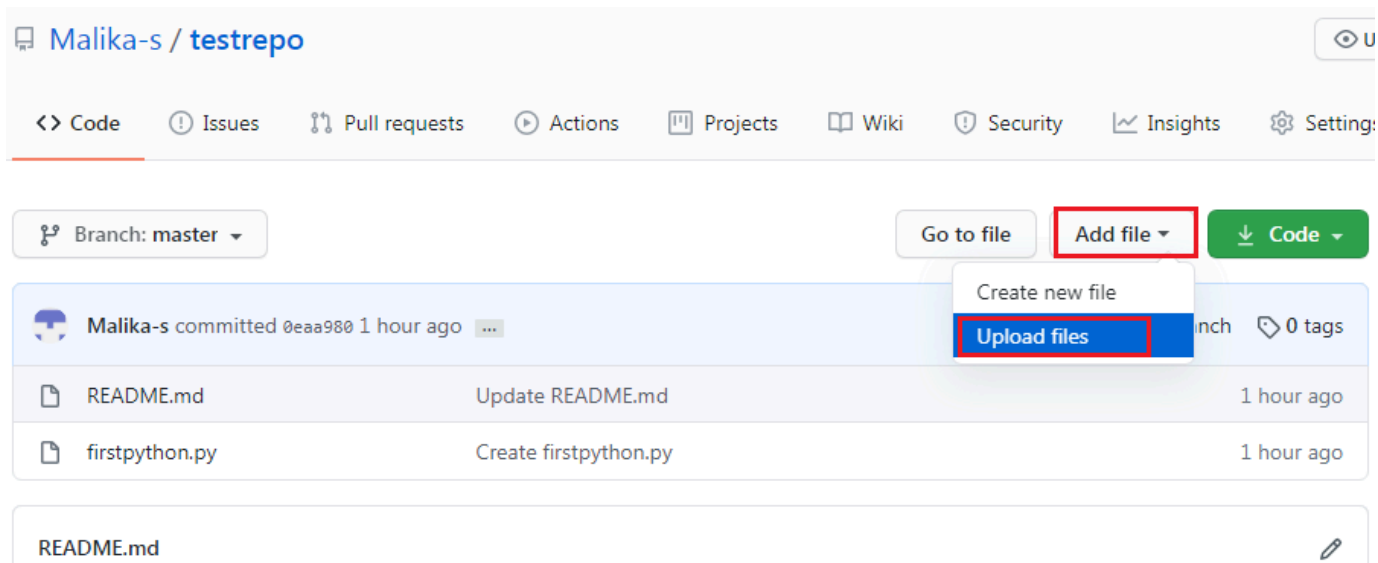


The image shows the 'Commit new file' dialog in GitHub. At the top, there's a text input field containing 'Create firstpython.py'. Below it is a larger text area with the placeholder text 'Adding a new file'. Underneath the text area are two radio button options: the first is selected and says 'Commit directly to the master branch.', and the second says 'Create a new branch for this commit and start a pull request. Learn more about pull requests.' At the bottom left are two buttons: 'Commit new file' (green) and 'Cancel' (white with a red border).

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.



The image shows the GitHub repository page for 'Malika-s / testrepo'. The top navigation bar includes links for Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. Below the navigation bar, there's a 'Branch: master' dropdown, a 'Go to file' button, an 'Add file' button (highlighted with a red box), and a 'Code' button. A dropdown menu is open under 'Add file', showing 'Create new file' and 'Upload files' (highlighted with a red box). Below the 'Add file' dropdown, there's a commit history table. The table has three columns: file name, commit message, and time. The first row shows 'README.md' with the message 'Update README.md' and '1 hour ago'. The second row shows 'firstpython.py' with the message 'Create firstpython.py' and '1 hour ago'. At the bottom, there's a section for 'README.md' with a pencil icon for editing.

File	Commit Message	Time
README.md	Update README.md	1 hour ago
firstpython.py	Create firstpython.py	1 hour ago

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

testrepo /




Drag files here to add them to your repository

Or [choose your files](#)

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

Test.py



### Commit changes

Add files via upload

Add an optional extended description...

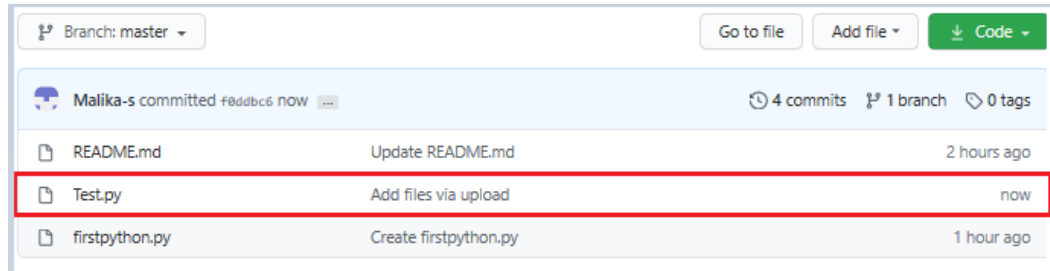
☒ Commit directly to the `master` branch.

☐ Create a new branch for this commit and start a pull request. [Learn more about pull requests.](#)

Commit changes

Cancel

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.

## Author(s)

**Romeo Kienzler**

**Malika Singla**

## Other Contributor(s)

Rav Ahuja

© IBM Corporation 2021. All rights reserved.