



## 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:

A screenshot of the GitHub 'Create your account' page. At the top, it says 'Join GitHub' and 'Create your account'. Below this are input fields for 'Username', 'Email address', and 'Password'. A note states: 'Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter. Learn more.' There is a checkbox for 'Email preferences' with the text 'Send me occasional product updates, announcements, and offers.' Below this is a section 'Verify your account' with a box containing the text 'Please solve this puzzle so we know you are a real person' and a 'Verify' button. At the bottom is a large blue 'Create account' button. A footer note says: '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

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

Verify



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` `api`

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

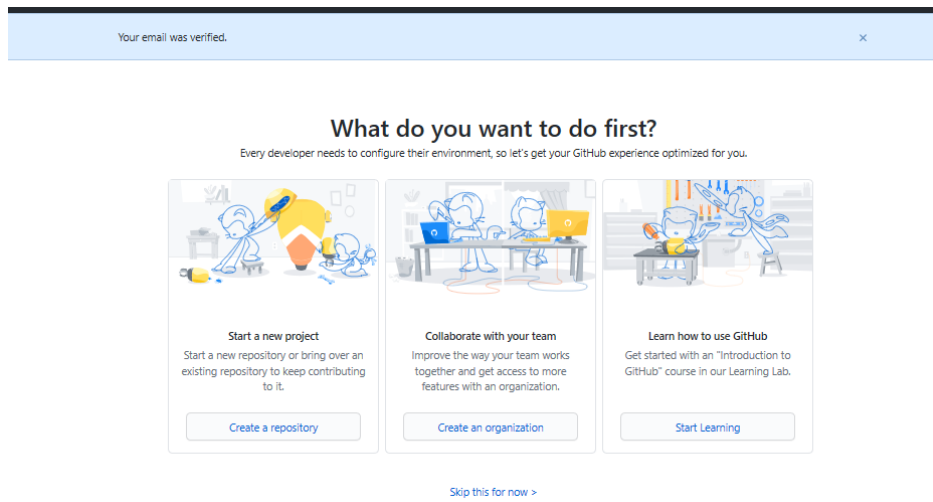
Before you can contribute on GitHub, we need you to verify your email address.

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

Resend verification email

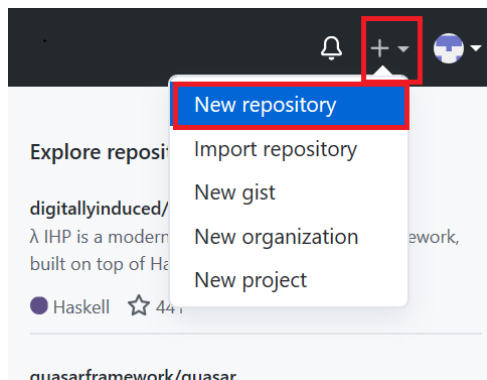
Change your email settings

Email is verified



## 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 \* 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](#)

and click Create repository.

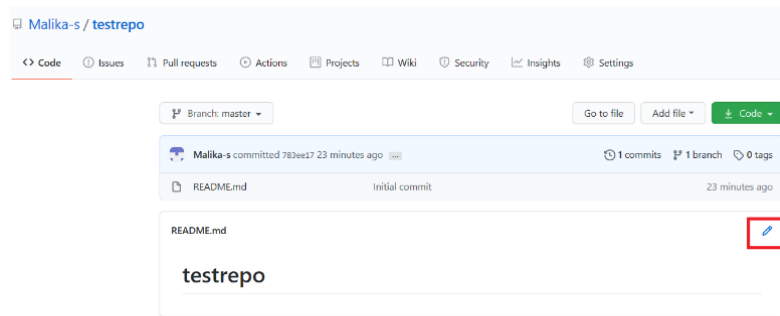
Now, you will be redirected to the repository you have created.

Let's start editing the repository.

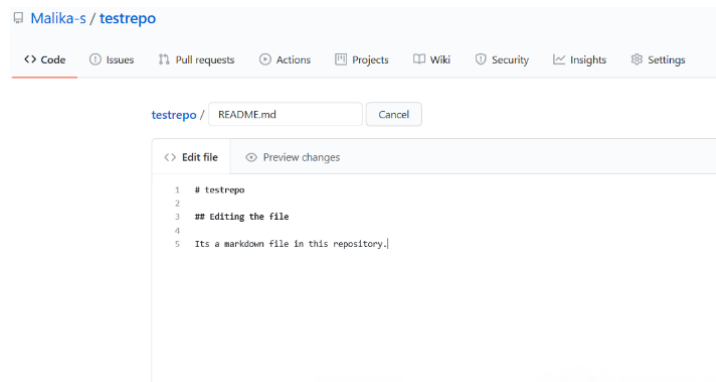
## Exercise 3: Create / edit a file

### Exercise 3a: Edit a file

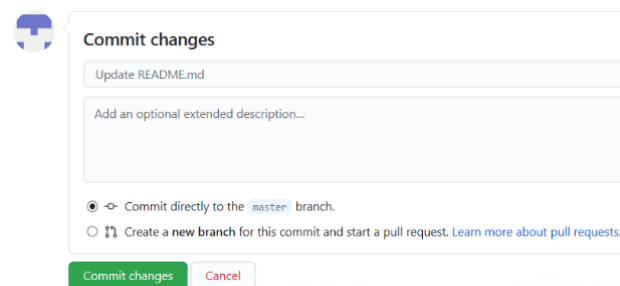
Step 1: Once the repository is created, the root folder of your repository is listed by default and it has just one file `ReadMe.md`. Click on the pencil icon to edit the file.



Step 2: Add text to file.



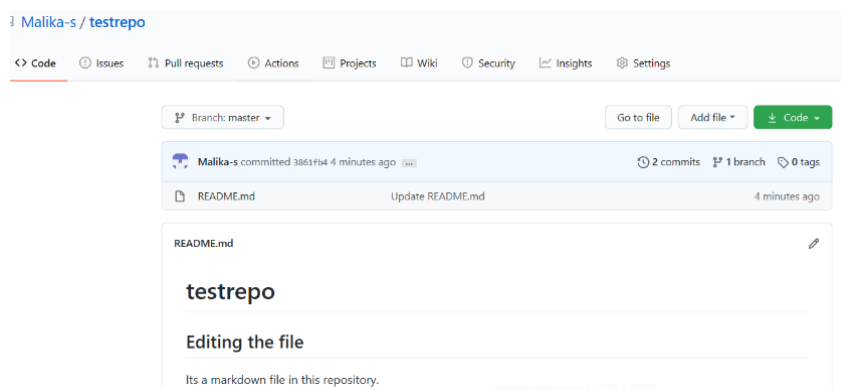
Step 3: Scroll down the page after adding the text and click `Commit Changes`.



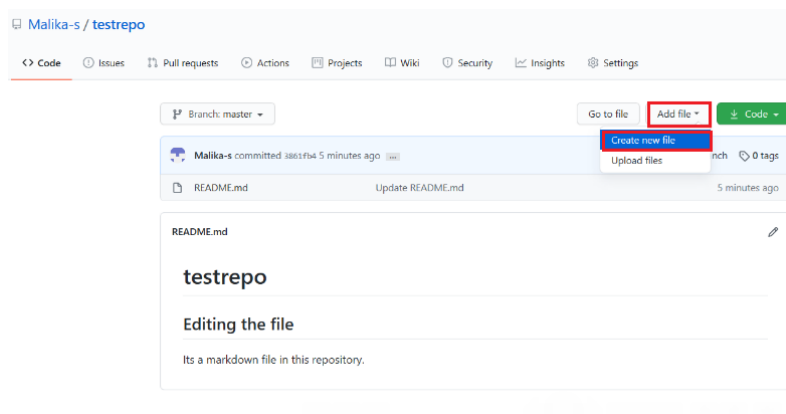
Now, check your file is edited with the new text.

### Exercise 3b: Create a new file

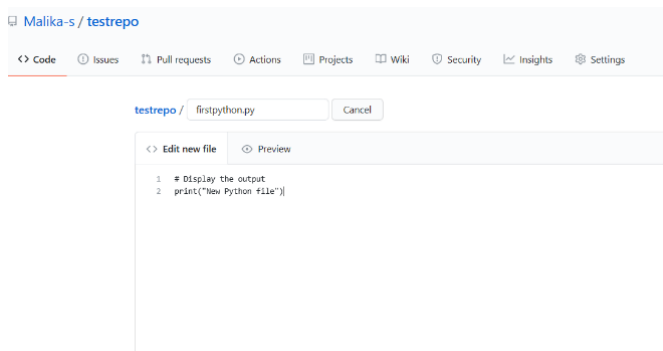
Step 1: Click on the repository name to go back to the master branch like in this testrepo.



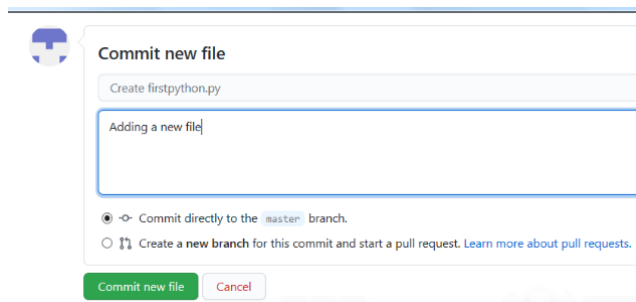
Step 2: Click `Add file` and select `Create New file` to create a file in the repository.



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



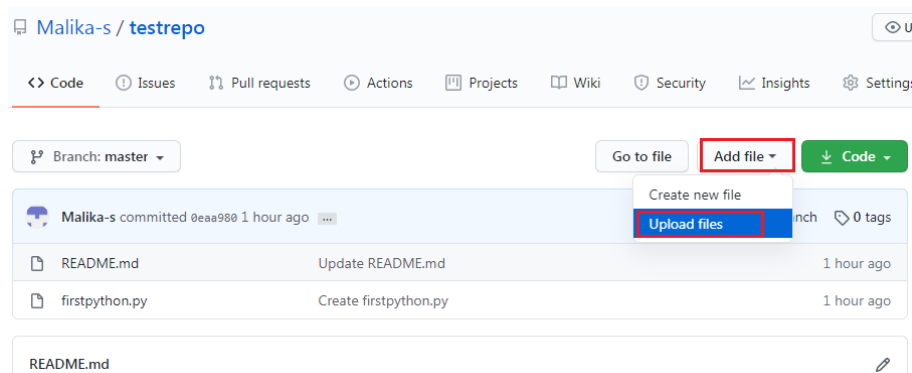
Step 4: Scroll down the page after adding the text. Add description of the file (optional) and click commit new file.



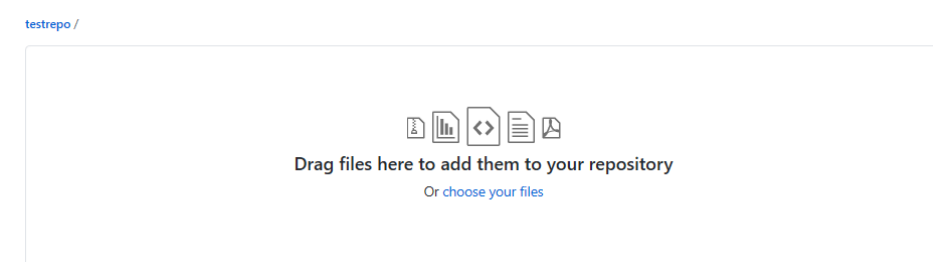
Step 5: Your file is now added to your repository and the repository listing shows when the file was added/changed.

## 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

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.

Branch: master	Go to file	Add file	Code
Malika-s committed resources now <span>4 commits</span> <span>1 branch</span> <span>0 tags</span>			
README.md	Update README.md		2 hours ago
Test.py	Add files via upload		now
firstpython.py	Create firstpython.py		1 hour ago

## 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

## Changelog

Date	Version	Changed by	Change Description
2021-06-03	0.6	Lakshmi Holla	Forked from Previous version
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

© IBM Corporation 2021. All rights reserved.