

# Lab: GitHub Sign Up and Create Repo

Estimated time: 30 minutes

In this lab, you will get started with GitHub by creating a GitHub account and creating a new repository. You will then add a file to the repository 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 command-line-based version-control system for software development. While Git is a command-line tool and you also need to host and maintain a server on which you can use Git for versioning.

GitHub is a repository hosting service that uses Git. GitHub provides a web-based hosting service with a graphical user interface(GUI) and git command line interface (CLI). 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:

The screenshot shows the GitHub sign-up page. At the top, there is a navigation bar with links for Product, Solutions, Open Source, and Pricing. On the right side of the navigation bar are a search bar, a help icon, and a 'Sign in' button. Below the navigation bar, the main heading reads 'First, let's create your user account'. The form contains four input fields: 'Username \*' (filled with a redacted value), 'Email address \*' (filled with a redacted value), and 'Password \*' (filled with a redacted value). Below the password field is a note: 'Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter.' followed by a link 'Learn more...'. At the bottom of the form is a section for 'Email preferences' with a checked checkbox: 'Send me occasional product updates, announcements, and offers.'

3. Verify the account, and then click **Submit**.

### Verify your account

Use the arrows to rotate the object to face in the direction of the hand. (1 of 2)

Match this angle

Submit

Audio

Restart

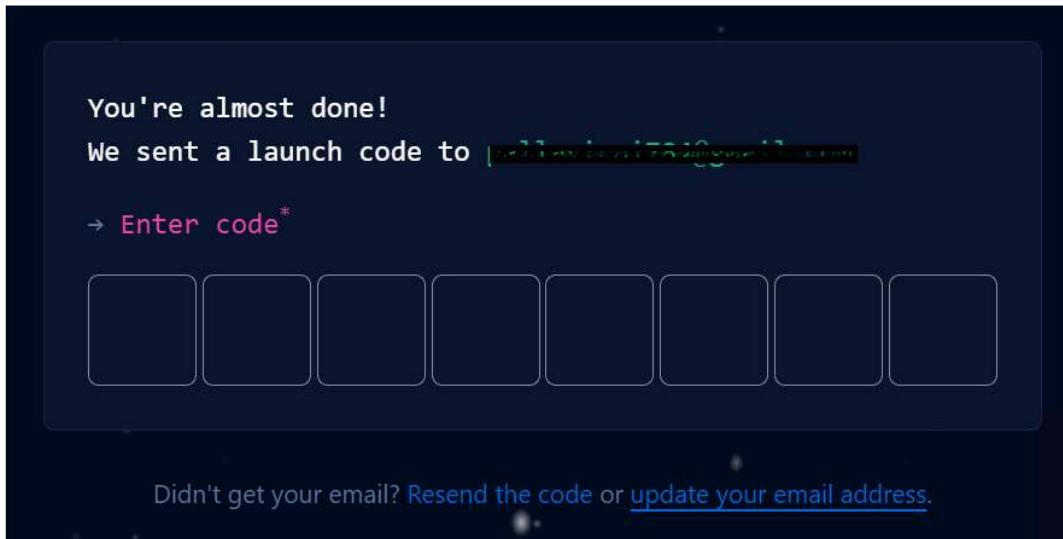
4. When you have finished, click **Create account**.

### Verify your account

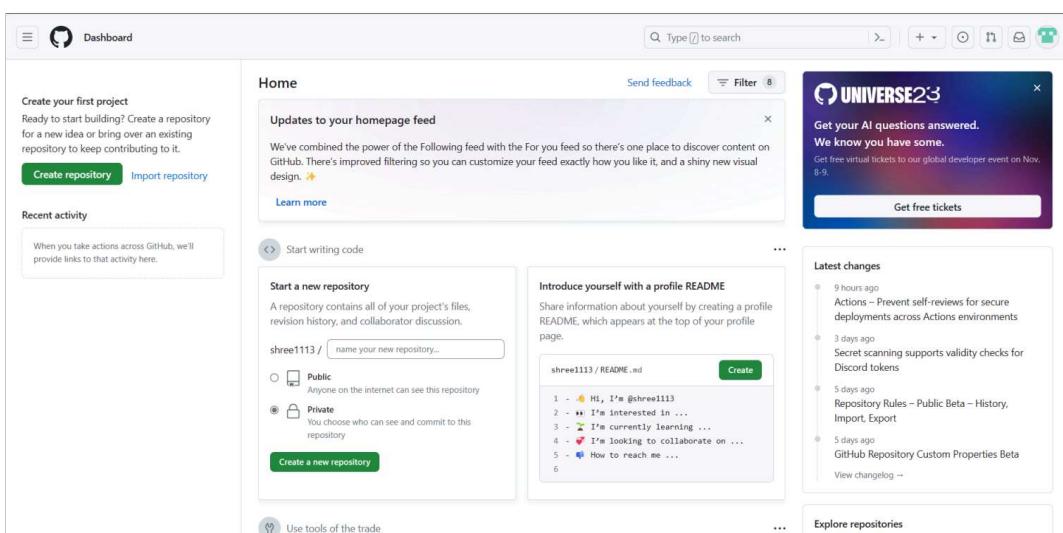
Create account

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.

5. You will receive a verification email from GitHub. Click the enclosed link to verify your email or enter the provided verification code. **Note:** If you do not receive a verification email, click **Resend the code**.



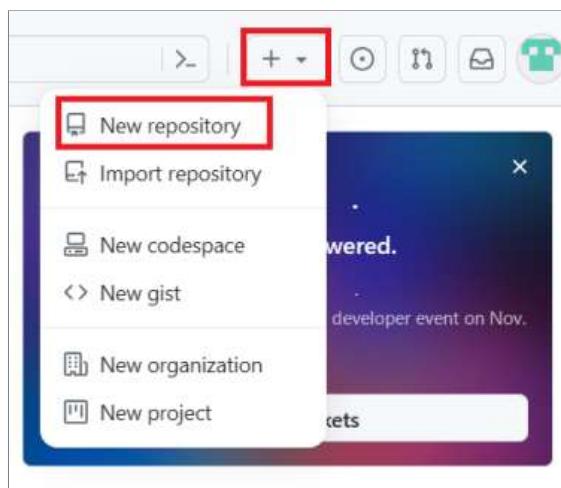
6. After entering the provided verification code, you will see a screen that looks like this. This indicates that you are logged into your account and you are ready to start using it.



## 2: Add a project/repository

To add a new repository, complete the following steps:

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



- Enter a repository name and select the **Add a README file** check box under the **Initialize this repository with:** option.

## Create a new repository

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

*Required fields are marked with an asterisk (\*).*

**Repository template**

Start your repository with a template repository's contents.

---

**Owner \***



**Repository name \***

 testrepo is available.

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

**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.

---

**Initialize this repository with:**

**Add a README file**  
This is where you can write a long description for your project. [Learn more about READMEs.](#)

**Add .gitignore**

3. Click **Create repository**.

**Initialize this repository with:**

**Add a README file**  
This is where you can write a long description for your project. [Learn more about READMEs.](#)

**Add .gitignore**

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

**Choose a license**

A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

This will set  `main` as the default branch. Change the default name in your [settings](#).

 You are creating a public repository in your personal account.

4. The repository is created and its home page is displayed.

The screenshot shows a GitHub repository named 'testrepo' which is public. The repository has 1 branch and 0 tags. The README.md file is the only file present, with a commit message 'Initial commit' and timestamp 'now'. The file content is '# testrepo'. On the right side of the page, there is an 'About' section with the note 'No description, website, or topics provided.' It also shows 1 commit, 0 stars, 0 forks, and 0 watching. Below that is a 'Releases' section with a link to 'Create a new release'. At the bottom is a 'Packages' section with a link to 'Publish your first package'.

Next, you'll start editing the repository.

### 3: Create and edit a file

#### 3.1 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.

This screenshot is identical to the one above, showing the 'testrepo' repository on GitHub. The README.md file is open in the editor, and the pencil icon in the top right corner of the editor is highlighted with a red box.

2. Add "It's a markdown file in this repository" text to the file and go to top right and click Commit changes

This screenshot shows the GitHub code editor for the 'testrepo/README.md' file. The file content is as follows:

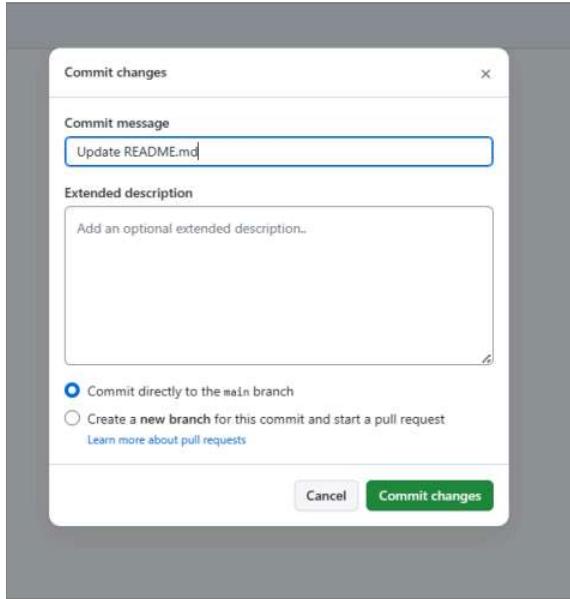
```

1 # testrepo
2
3 ## Editing the file
4
5 Its a markdown file in this repository

```

The 'Commit changes' button in the top right corner of the editor is highlighted with a red box.

3. Add some additional description and click **Commit changes**.

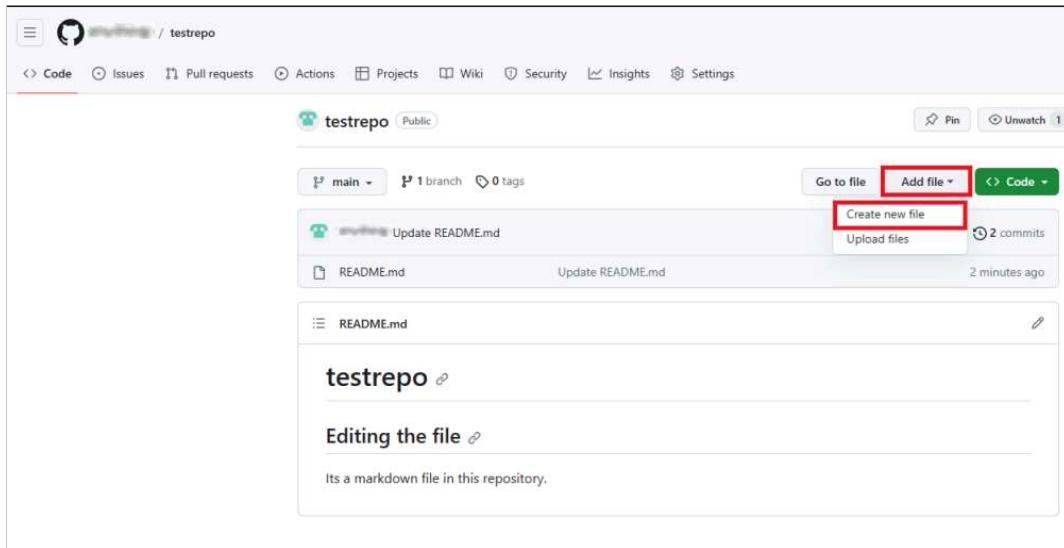


4. Confirm that the text you added to the file has been saved.

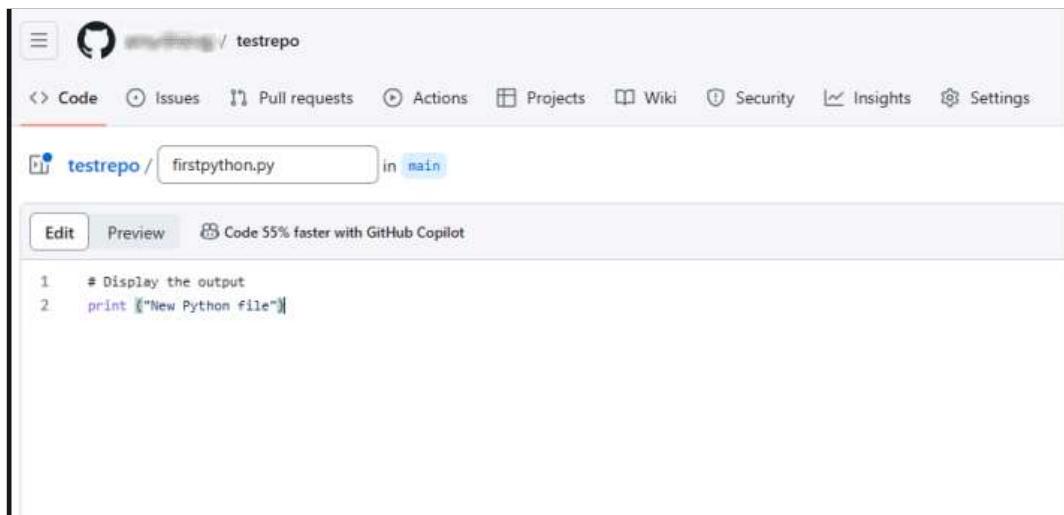
## 3.2 Create a new file

1. Click on the repository name **testrepo** to go back to the main branch.

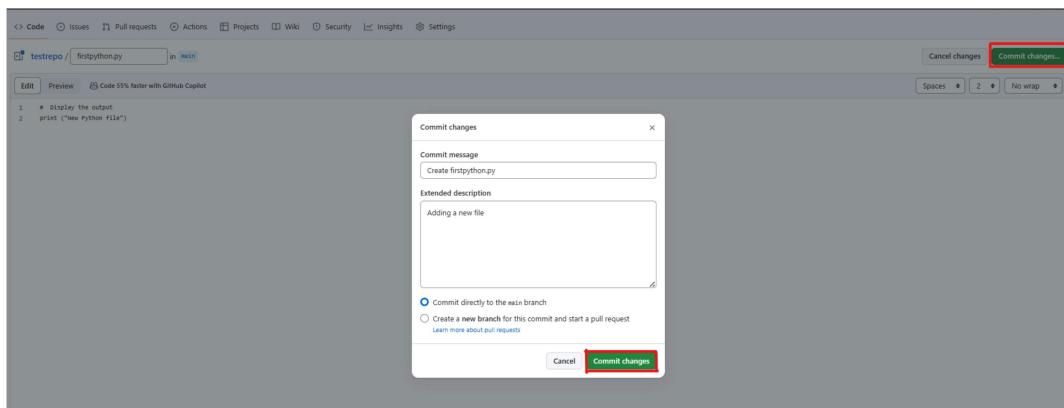
2. Click **Add file** and select **Create new file** to create a new file in the repository.



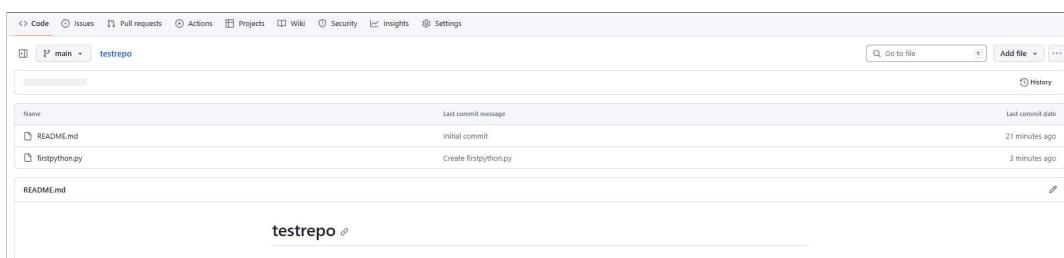
3. Enter a file name and extension; for example, `firstpython.py`. Add the lines of code displayed in the following image to your file:



4. Scroll to the top right of the page and click Commit changes. You can optionally add a description of your update (for example, “Adding a new file”). Click **Commit changes** to create your new file.



5. 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.

The screenshot shows a GitHub repository named 'testrepo'. The repository has 1 branch and 0 tags. It contains a file named 'README.md' with an 'Initial commit' and another file named 'firstpython.py' created 6 minutes ago. The 'Code' dropdown menu is open, showing options like 'Create new file' and 'Upload files'. The 'Upload files' option is highlighted with a red box.

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.

The screenshot shows a GitHub repository page titled 'testrepo'. A large central area has a 'Drop files here to add them to your repository' placeholder with a paper icon. Below it is a link 'Or choose your files'.

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

The screenshot shows a GitHub 'Commit changes' dialog box. It shows a file named 'test.py.txt' selected for upload. The 'Commit changes' button is highlighted with a red box.

4. Your file is uploaded to the repository.

The screenshot shows a GitHub repository page titled 'testrepo'. The file 'test.py.txt' is listed at the bottom of the commit history, highlighted with a red box. The commit message is 'Add files via upload'. The commit was made by 'c59a98f' and is dated 'now'. The commit history also includes 'README.md' updated 29 minutes ago and 'firstpython.py' created 6 minutes ago.

## 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.

## Author details

### Authors:

- Romeo Kienzler
- Malika Singla

**Other contributors:**

- Rav Ahuja
- Upkar Liddar



# Skills Network