

Lesson 04 Demo 02

Creating a Pull Request in Git

Objective: To create a pull request in GitHub to facilitate code review

Tools required: GitHub

Prerequisite: None

Steps to be followed:

1. Create a Fork
2. Clone your Fork
3. Sync fork with original repository
4. Generate personal access token
5. Push your changes
6. Create a pull request

Step 1: Create a Fork

- 1.1 Log in to your GitHub account. Then, from your browser, navigate to the following repository:

<https://github.com/GithubResources1/Source-repository.git>



Sign in to GitHub

Username or email address

Password

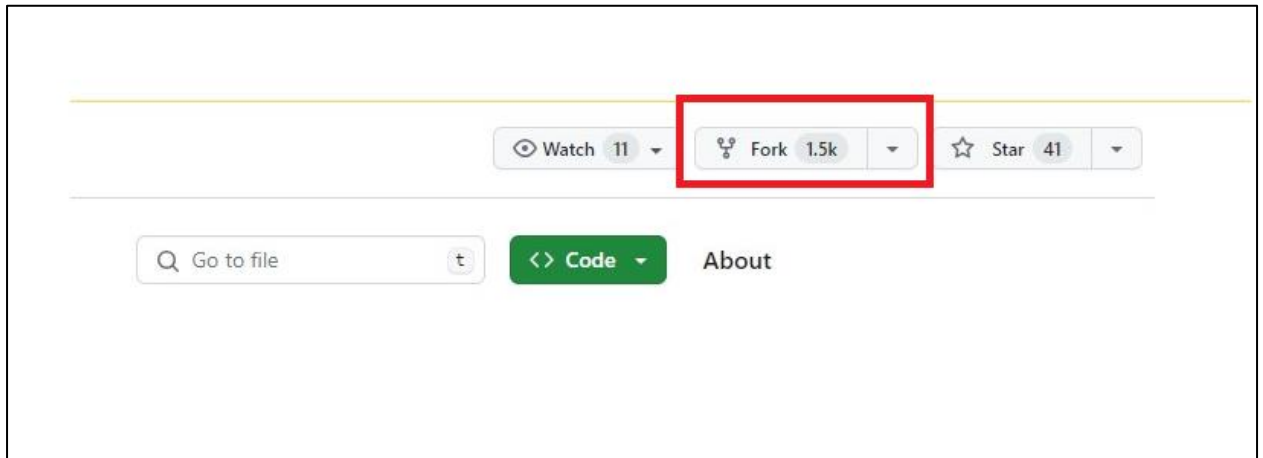
[Forgot password?](#)

Sign in

[Sign in with a passkey](#)

New to GitHub? [Create an account](#)

1.2 In the top-right corner of the page, click on the **Fork** button




1.3 Create a new fork by writing **Repository name** as **sample repository** and click on **Create fork**

Create a new fork

A fork is a copy of a repository. Forking a repository allows you to freely experiment with changes without affecting the original project. [View existing forks.](#)

Required fields are marked with an asterisk ().*

Owner *


AniketJainSL

Repository name *

sample repository

✔ Your new repository will be created as sample-repository.
The repository name can only contain ASCII letters, digits, and the characters `.`, `-`, and `_`.

By default, forks are named the same as their upstream repository. You can customize the name to distinguish it further.

Description (optional)

Hello Git World sample training repository

☒ Copy the `master` branch only

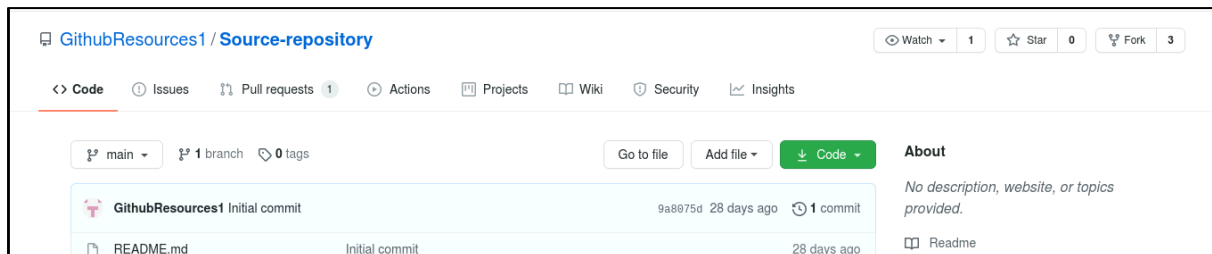
Contribute back to githubtraining/hellogitworld by adding your own branch. [Learn more.](#)

🕒 You are creating a fork in your personal account.

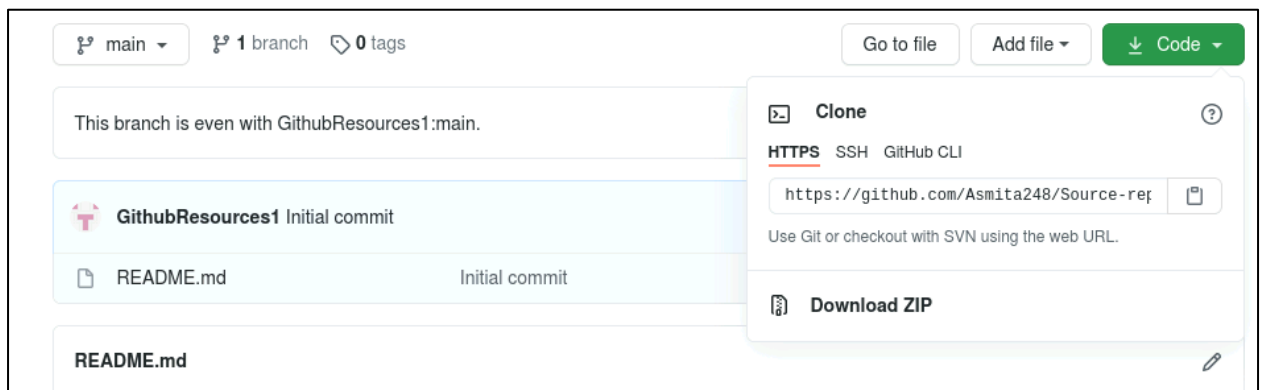
Create fork

Step 2: Clone your Fork

2.1 On GitHub, click on the **Code** button on your repository



2.2 Copy the URL from the **HTTPS** option



2.3 Type the following command on your terminal:

git clone [the copied HTTPS URL]

```
@ip-172-31-71-23:~$ git clone https://github.com/GithubWorkstation/Source-repository.git
Cloning into 'Source-repository'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.
@ip-172-31-71-23:~$
```

Note: Replace [the copied HTTPS URL] in the above command with the URL you copied from the GitHub page.

Therefore, the clone command creates a local git repository from your remote fork on GitHub.

Step 3: Sync fork with the original repository

- 3.1 Navigate to the directory where the fork has been cloned using the following command in the terminal:

cd Source-repository

```
simply@ip-172-31-71-23:~$ cd Source-repository
```

- 3.2 Execute the following command to see the configured remote repository for your fork:

git remote -v

```
simply@ip-172-31-71-23:~$ cd Source-repository
simply@ip-172-31-71-23:~/Source-repository$ git remote -v
origin  https://github.com/GithubWorkstation/Source-repository.git (fetch)
origin  https://github.com/GithubWorkstation/Source-repository.git (push)
simply@ip-172-31-71-23:~/Source-repository$
```

- 3.3 Use **git remote add upstream** command and paste the URL you copied in Step 2 as shown below:

git remote add upstream [the copied HTTPS URL]

```
simply@ip-172-31-71-23:~/Source-repository$ git remote add upstream https://github.com/GithubWorkstation/Source-repository.git
git
```

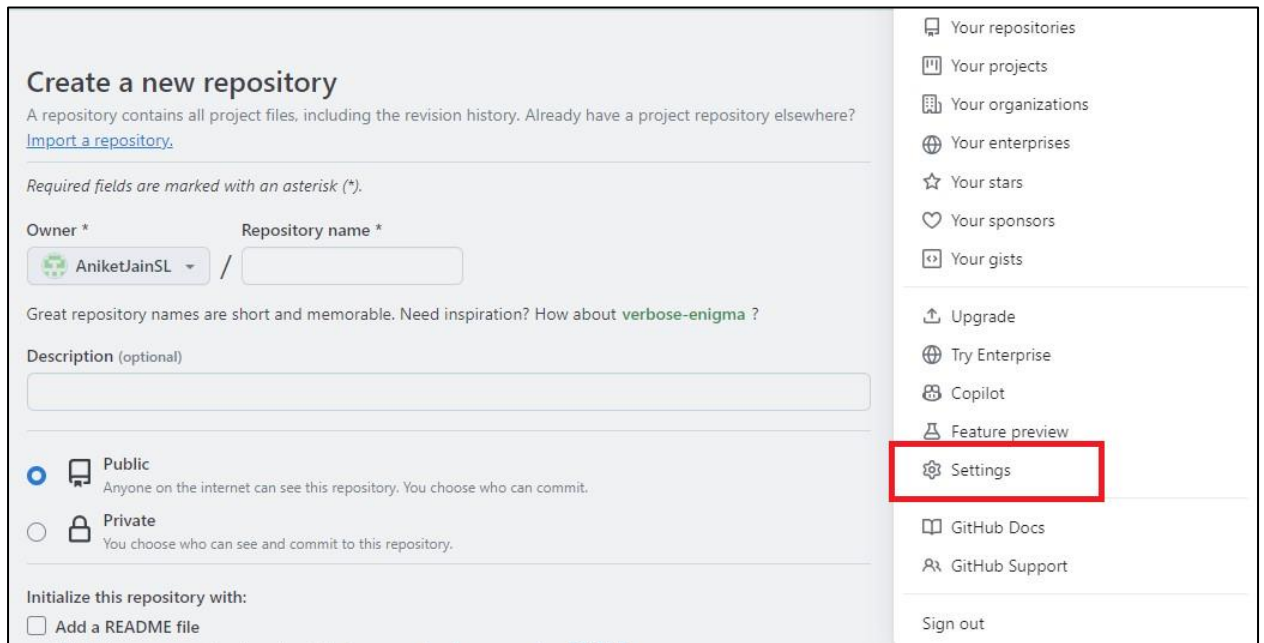
- 3.4 Execute the following command to verify the new upstream repository you have specified for your fork:

git remote -v

```
simply@ip-172-31-71-23:~/Source-repository$ git remote add upstream https://github.com/GithubWorkstation/Source-repository.git
simply@ip-172-31-71-23:~/Source-repository$ git remote -v
origin  https://github.com/GithubWorkstation/Source-repository.git (fetch)
origin  https://github.com/GithubWorkstation/Source-repository.git (push)
upstream https://github.com/GithubWorkstation/Source-repository.git (fetch)
upstream https://github.com/GithubWorkstation/Source-repository.git (push)
simply@ip-172-31-71-23:~/Source-repository$
```

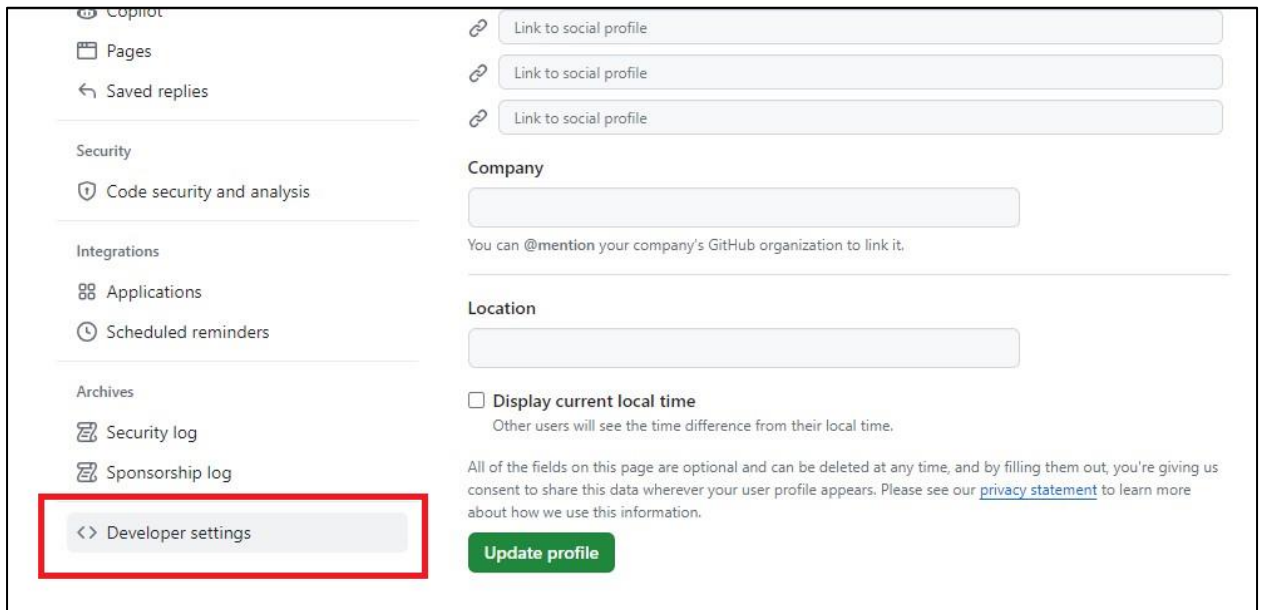
Step 4: Generate personal access token

4.1 Go to **Settings** under your profile



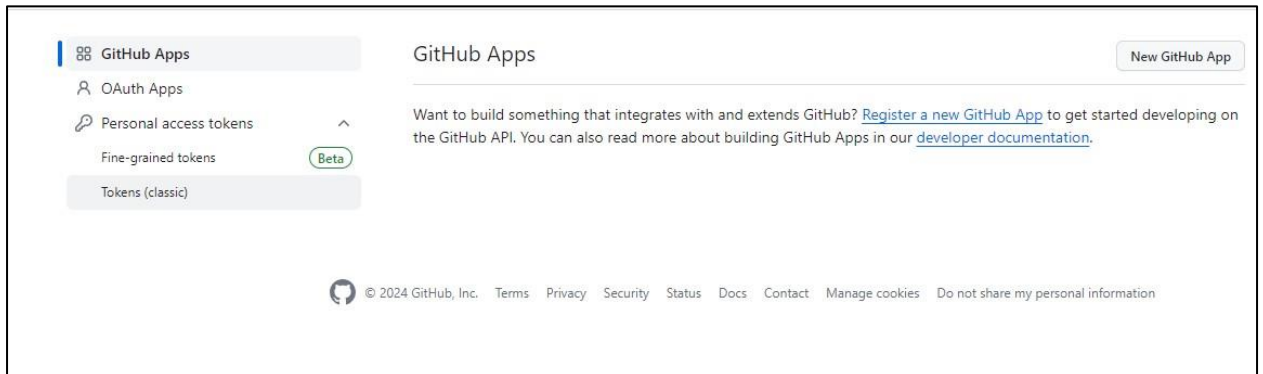
The screenshot shows the GitHub 'Create a new repository' page. The main content area on the left includes the title 'Create a new repository', a subtitle 'A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)', a note 'Required fields are marked with an asterisk (*)', and form fields for 'Owner *' (set to 'AniketJainSL'), 'Repository name *', 'Description (optional)', and visibility options (Public and Private). At the bottom, there is a checkbox for 'Add a README file'. The right sidebar contains a list of navigation links: 'Your repositories', 'Your projects', 'Your organizations', 'Your enterprises', 'Your stars', 'Your sponsors', 'Your gists', 'Upgrade', 'Try Enterprise', 'Copilot', 'Feature preview', 'Settings' (highlighted with a red box), 'GitHub Docs', 'GitHub Support', and 'Sign out'.

4.2 Click on **Developer settings**



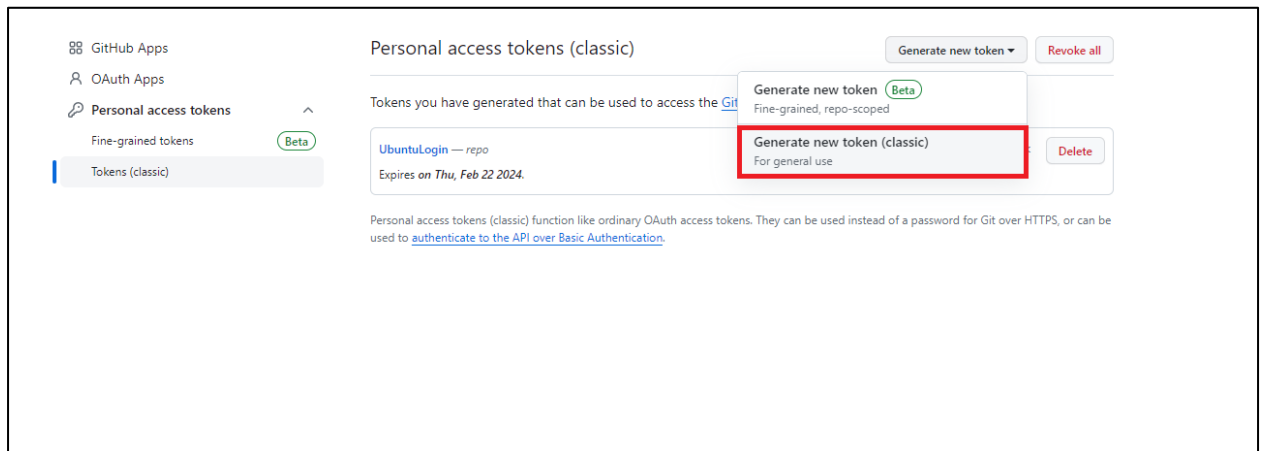
The screenshot shows the GitHub Developer settings page. On the left sidebar, the 'Developer settings' option is highlighted with a red rectangle. The main content area includes sections for 'Link to social profile' (three links), 'Company' (a text input field), 'Location' (a text input field), and a checkbox for 'Display current local time'. Below these is a green 'Update profile' button. A disclaimer at the bottom states: 'All of the fields on this page are optional and can be deleted at any time, and by filling them out, you're giving us consent to share this data wherever your user profile appears. Please see our [privacy statement](#) to learn more about how we use this information.'

4.3 Inside **Personal access tokens**, click on **Tokens (classic)**

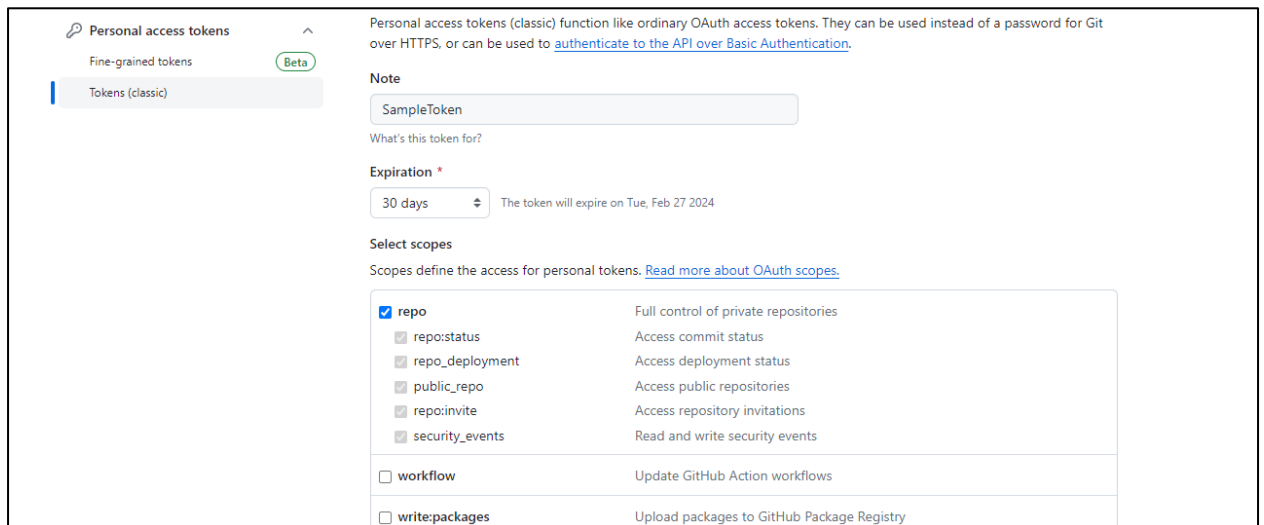


The screenshot shows the GitHub 'Personal access tokens' page. In the left sidebar, 'Tokens (classic)' is selected. The main content area is titled 'GitHub Apps' and includes a 'New GitHub App' button. Below the title, there is a paragraph explaining GitHub Apps and links to 'Register a new GitHub App' and 'developer documentation'. At the bottom, there is a footer with the GitHub logo, copyright information, and links for Terms, Privacy, Security, Status, Docs, Contact, Manage cookies, and Do not share my personal information.

4.4 Inside **Generate new token**, click on **Generate new token (classic)**



4.5 Inside **Note**, write the token name as **SampleLogin** , click on **repo**, and then click on **Generate token**



<input type="checkbox"/> copilot	Full control of GitHub Copilot settings and seat assignments
<input type="checkbox"/> manage_billing:copilot	View and edit Copilot Business seat assignments
<input type="checkbox"/> project	Full control of projects
<input type="checkbox"/> read:project	Read access of projects
<input type="checkbox"/> admin:gpg_key	Full control of public user GPG keys
<input type="checkbox"/> write:gpg_key	Write public user GPG keys
<input type="checkbox"/> read:gpg_key	Read public user GPG keys
<input type="checkbox"/> admin:ssh_signing_key	Full control of public user SSH signing keys
<input type="checkbox"/> write:ssh_signing_key	Write public user SSH signing keys
<input type="checkbox"/> read:ssh_signing_key	Read public user SSH signing keys

[Generate token](#) [Cancel](#)

4.6 Copy the link for the password to push changes

GitHub Apps
 OAuth Apps
 Personal access tokens ^
 Fine-grained tokens Beta
Tokens (classic)

Personal access tokens (classic)

Generate new token ▼ [Revoke all](#)

Tokens you have generated that can be used to access the [GitHub API](#).

Make sure to copy your personal access token now. You won't be able to see it again!

✓ ghp_Nb6Uf8USuGw8JjRmSwaxFtFMdkuRST2suk2s
 [Delete](#)

UbuntuLogin — repo
 Expires on Thu, Feb 22 2024.
 [Delete](#)

Step 5: Push your changes

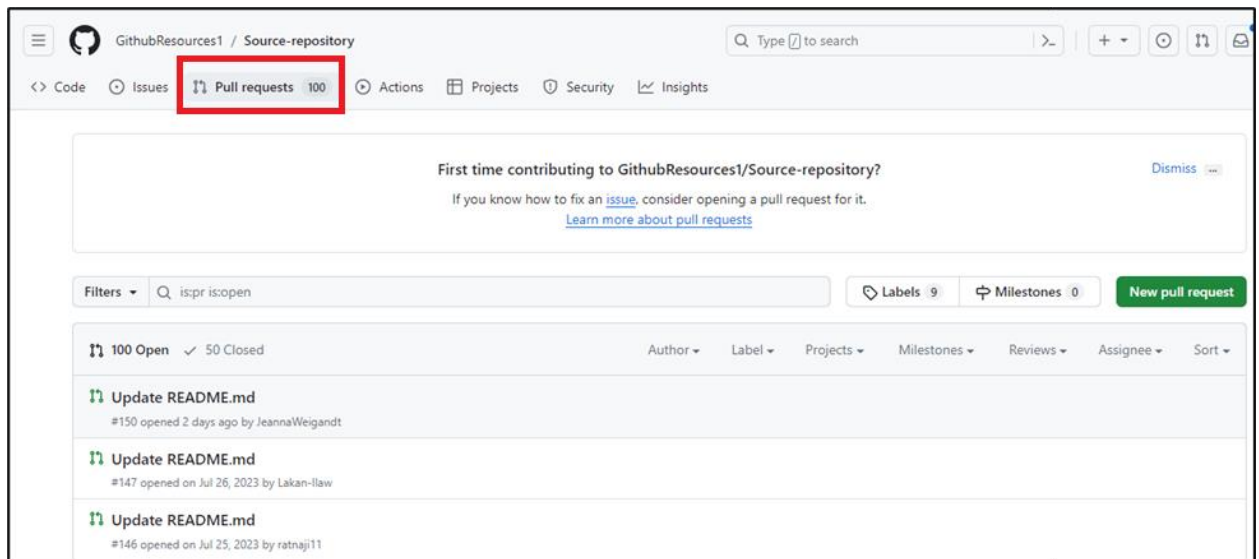
- 5.1 Use the **git push** command to upload the changes to your remote for fork on GitHub and use the copied link for password

```

root@ip-172-31-71-23:~/Source-repository$ git push
Username for 'https://github.com': GithubWorkstation
Password for 'https://GithubWorkstation@github.com':
Everything up-to-date
root@ip-172-31-71-23:~/Source-repository$
  
```

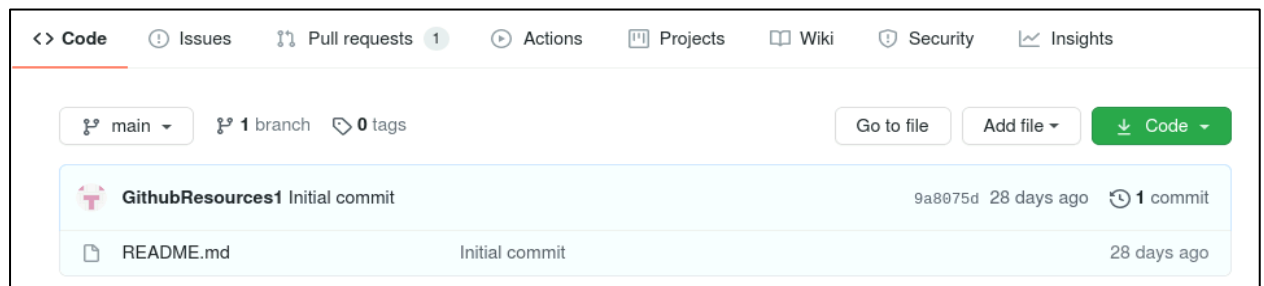
Step 6: Create a pull request

6.1 On the GitHub page of your remote fork, click the **Pull requests** button



Wait for the owner to merge or comment on your changes

6.2 If the owner suggests some changes before merging, you can simply push these changes into your fork by repeating steps 3 and 4; the pull request is updated automatically.



By following these steps, you have successfully created a pull request, proposing changes to a repository and enabling collaborative code review and integration.