

Lab: Branching and Merging (Web UI)

Estimated time: 15 minutes

Objectives

After completing this lab, you will be able to:

1. Create a branch
2. Commit changes to a child branch
3. Open a pull request
4. Merge a pull request into the main branch

Prerequisites

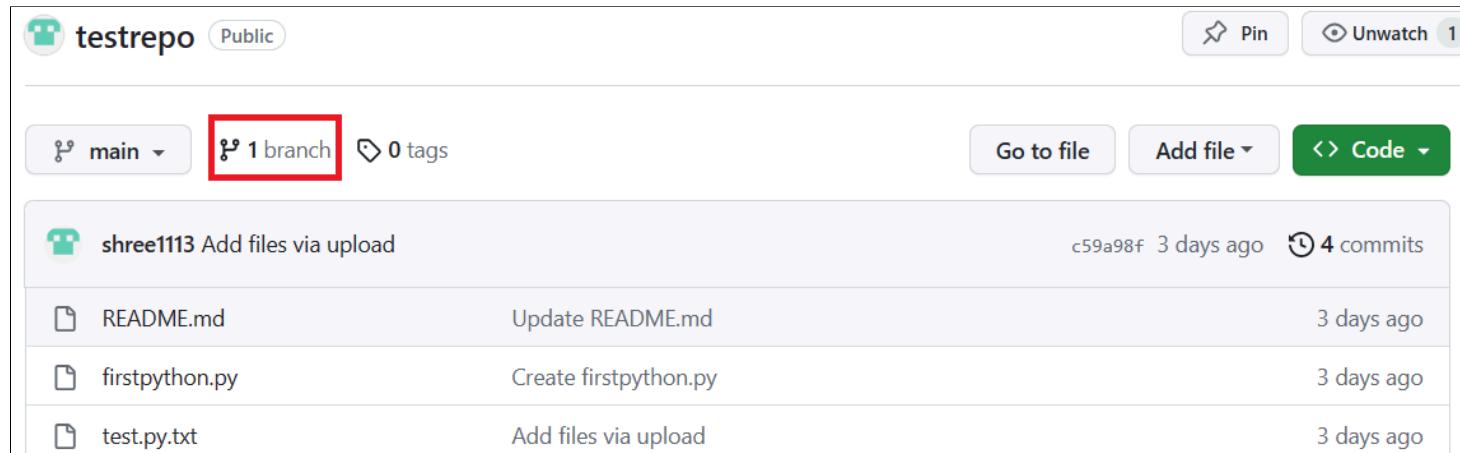
This hands-on lab requires you to have created a GitHub account with a repository in it, as covered in [Getting started with GitHub](#) lab.

NOTE: In the past the default branch in your GitHub repo used the name `master`. Effective Oct 1, 2020, all new GitHub repositories will use the more inclusive term `main` as the name of the default branch instead of `master`.

1. Create a branch

You can create or delete branches using your repository's GitHub web page. To add a branch to your repository, complete the following steps:

1. Go to your repository's main page. Note that when you created your repository, the one branch named `main` was created for you.



The screenshot shows a GitHub repository page for 'testrepo'. At the top, there is a dropdown menu labeled 'main' with a red box around it. Next to it is a button labeled '1 branch' with a red box around it. Below the dropdown are buttons for 'Go to file', 'Add file', and 'Code'. The main content area displays a list of commits by user 'shree1113'. The first commit is 'Add files via upload' with hash 'c59a98f' and timestamp '3 days ago'. The second commit is 'Update README.md' with timestamp '3 days ago'. The third commit is 'Create firstpython.py' with timestamp '3 days ago'. The fourth commit is 'Add files via upload' with timestamp '3 days ago'.

2. At the top of the file list, locate the **Branch** drop-down menu. (By default, the menu displays **Branch: main**.) Click the drop-down menu, type the name of the branch you want to create, or click on the `Create branch from main` or hit `Enter` on your keyboard.

The screenshot shows a GitHub repository named "testrepo" which is public. The main branch is "main". There are 2 branches and 0 tags. A modal window titled "Switch branches/tags" is open, showing a search bar with "Child_Branch" and a button to "Create branch Child_Branch from main". Both the search bar and the create button are highlighted with red boxes.

Your repository now has two branches: **Main** and **Child_Branch**. You can click the drop-down menu to see your branches.

This branch is up to date with main.

shree1113 Add files via upload c59a98f 3 days ago 4 commits

README.md	Update README.md	3 days ago
firstpython.py	Create firstpython.py	3 days ago
test.py.txt	Add files via upload	3 days ago

Any files that were in the **main** branch are reflected in the **Child_Branch**. Note that when you add or edit a file in Child_Branch, that change will not automatically reflect in the main branch

2. Add a file to a branch

To add a file to your new branch, ensure that the name you gave the new branch (which in the case of the example showcased is **Child_Branch**) is displayed in the **Branch** drop-down menu and complete the following steps:

1. Click **Add file > Create new file** to create a file in the repository.

Child_Branch 2 branches 0 tags

Go to file Add file ▾ Create new file

This branch is up to date with main.

shree1113 Add files via upload c59a98f 3 days ago 4 commits

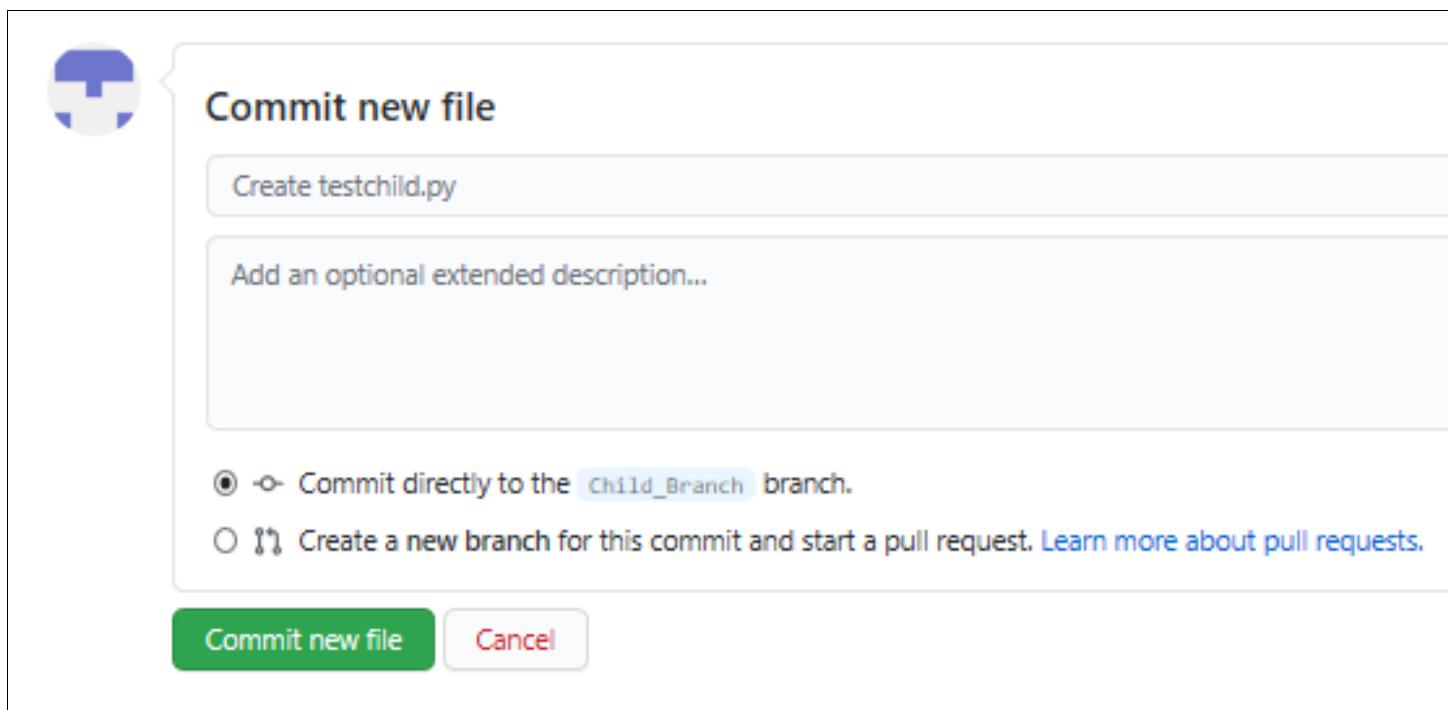
README.md	Update README.md	3 days ago
firstpython.py	Create firstpython.py	3 days ago
test.py.txt	Add files via upload	3 days ago

2. Type a name and extension for the file, for example, `testchild.py`, and add the following lines to the body of the new file:

The screenshot shows a GitHub code editor interface. At the top, there's a header with a repository icon, the text "testrepo /", the file name "testchild.py", and a location indicator "in child". Below the header, there are two buttons: "Edit" and "Preview". Next to them is a GitHub Copilot icon with the text "Code 55% faster with GitHub Copilot". The main area contains the following code:

```
1 ## Adding a new file in the child branch
2 print ("Inside Child branch")
```

3. Scroll to the bottom of the page, add a description of the file you are about to add (note that the description is optional), and click **Commit**.



The file is added to your child branch.

3. Open a pull request

The file that you added to your child branch is not automatically added to the **main** branch. (You can check this by using the **Branch** drop-down menu to go to the **main** branch; note that there is no `testchild.py` file in the file list).

A screenshot of a GitHub repository page for the **main** branch. The top navigation bar shows "main" with a dropdown arrow, "2 branches", and "0 tags". On the right, there are buttons for "Go to file", "Add file", and a green "Code" button. Below the navigation, a commit from a user is shown: "committed 92fe776 1 minute ago". The commit details show three files: "README.md" (Update README.md), "test.py.txt" (Add files via upload), and "firstpython.py" (Create firstpython.py). Each file has a green "View diff" button next to it.

You can also compare the two branches and open a *pull request*, which will enable you to merge the changes that you've made in the child branch – in this case, adding a new file – to the **main** branch.

1. In **Child_Branch**, click the **Compare & pull request** button.

A screenshot of a GitHub repository page for the **Child_Branch**. The top navigation bar shows "Child_Branch" with a dropdown arrow, "2 branches", and "0 tags". On the right, there are buttons for "Go to file", "Add file", and a green "Code" button. A yellow banner at the top says "Child_Branch had recent pushes 8 minutes ago". To the right of the banner is a green "Compare & pull request" button, which is highlighted with a red border.

2. Scroll to the bottom of the page and note that there is **1 changed file** listed and the changes are highlighted in green.

A screenshot of a GitHub diff view. The header says "Showing 1 changed file with 2 additions and 0 deletions." There are "Unified" and "Split" buttons. The diff shows a single file, "testchild.py", with two additions. The additions are highlighted in green:

```

@@ -0,0 +1,2 @@
+ ## Adding a new file in child branch
+ print ("Inside Child branch")

```

3. Scroll up and note that GitHub is comparing the **main** and **Child_Branch** branches and that there are no conflicts between the two. Optionally, you can add a comment to the pull request. Click **Create pull request**.

The screenshot shows the GitHub interface for creating a pull request. At the top, there are dropdown menus for 'base: main' and 'compare: Child_Branch'. To the right of these, a green checkmark icon indicates that the branches are 'Able to merge'. Below this, there is a text area labeled 'Child branch' with a 'Write' tab selected. A large blue-bordered text input field is available for adding a description, with placeholder text 'Add your description here...'. Below the input field, there is a note: 'Attach files by dragging & dropping, selecting or pasting them.' At the bottom right, a green button with the text 'Create pull request' is highlighted with a red box. A small informational message at the bottom left reminds users to follow GitHub Community Guidelines.

The pull request has been successfully created and is now ready to be merged by a repository administrator. For all the repositories that you create, you automatically have administrative rights.

4. Merge a pull request

To merge a pull request into a project, complete the following steps:

1. Click the **Pull requests** tab. A list of pending pull requests is displayed.

The screenshot shows a web interface for managing pull requests. At the top, there are navigation links: Issues, Pull requests (with 1 item), Actions, Projects, Wiki, Security, Insights, and Settings. The 'Pull requests' link is highlighted with a red box. Below the navigation is a search bar with the query 'is:pr is:open'. To the right of the search bar are buttons for 'Labels 9', 'Milestones 0', and 'New pull request'. Underneath the search bar are filtering options: '1 Open' (unchecked) and '1 Closed' (checked). Further down are dropdown menus for 'Author', 'Label', 'Projects', 'Milestones', 'Reviews', 'Assignee', and 'Sort'. The main content area displays a single pull request card. The card has a checkbox, a green icon, and the title 'Create testchild.py'. Below the title is the text '#2 opened 2 minutes ago by [redacted]'. A 'ProTip!' message at the bottom suggests using '-label:bug' to exclude certain pull requests. The entire pull request card is also highlighted with a red box.

2. Click the pull request that you want to merge into the main project. Review the changes, click **Merge pull request** to accept the pull request and merge the updates. (You can optionally add a comment.)

Child branch #1

[Open](#) Malika-s wants to merge 3 commits into `master` from `Child_Branch`

Conversation 0 Commits 3 Checks 0 Files changed 1

Malika-s commented 2 minutes ago
Want to change in the master branch.

Malika-s added 3 commits 25 minutes ago

- o Create testchild 9767921
- o Delete testchild 48b4479
- o Create testchild.py eb5ced2

Add more commits by pushing to the `Child_Branch` branch on [Malika-s/testrepo](#).

Continuous integration has not been set up
[GitHub Actions](#) and [several other apps](#) can be used to automatically catch bugs and enforce style.

This branch has no conflicts with the base branch
Merging can be performed automatically.

Merge pull request
You can also open this in [GitHub Desktop](#) or view [command line instructions](#).

3. When you click **Merge pull request**, a **Confirm merge** button is displayed. Click that button to complete the merge.

Child branch #1

[Open](#)Malika-s wants to merge 3 commits into `master` from `Child_Branch` [Conversation 0](#)[Commits 3](#)[Checks 0](#)[Files changed 1](#)

Malika-s commented 4 minutes ago

Owner



...

Want to change in the master branch.



Malika-s added 3 commits 27 minutes ago



Create testchild

Verified

9767921



Delete testchild

Verified

48b4479



Create testchild.py

Verified

eb5ced2

Add more commits by pushing to the `Child_Branch` branch on [Malika-s/testrepo](#).Merge pull request #1 from [Malika-s/Child_Branch](#)

Child branch

[Confirm merge](#)[Cancel](#)The pull request has now been merged successfully. Note that you can delete the child branch because your changes have been incorporated into the **main** branch.

Pull request successfully merged and closed

You're all set—the `Child_Branch` branch can be safely deleted.[Delete branch](#)Check the list of files in the **main** branch to confirm that it now includes the file that you added in the pull request.

main ▾ 2 branches 0 tags

Go to file Add file ▾ Code ▾

committed 7a7c576 3 minutes ago ... 8 commits 2 branches 0 tags

File	Description	Time Ago
README.md	Update README.md	3 hours ago
test.py.txt	Add files via upload	1 hour ago
firstpython.py	Create firstpython.py	3 hours ago
testchild.py	Create testchild.py	21 minutes ago

Summary

Congratulations! You've now learned how to create a branch, edit and commit changes in that branch, open a pull request, and merge the pull request into your main project. We encourage you to continue to experiment with branches and pull requests to become more familiar with the concepts and processes.

Author details

Author:

- Malika Singla

Other contributor:

- Rav Ahuja



Skills Network