

Practice Project: Part 1 - GitHub UI

Estimated Time: 30 minutes

Overview

In this lab you will start your journey with the GitHub implementation for your organisation by creating a repository in your GitHub account and then initialising the repository with a README.md file and a License. Following this, you'll update the README file to include a Code of Conduct and Contribution Guidelines markdown files. After making these changes, you'll commit the files to your repository.

Next, you'll create a new branch and add code for Shipping logistics calculation. To wrap up, you'll merge this branch into the main branch using a Pull Request.

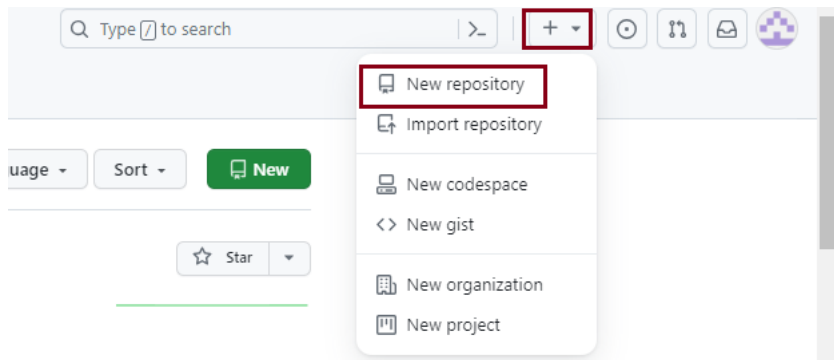
Objectives

After completing this lab, you will have demonstrated that you can:

1. Create a repository with a README.md file and a license.
2. Update the existing files.
3. Commit the files to your repository.
4. Create a new branch and add files to it.
5. Merge this branch to the main branch using a Pull Request.

Task 1: Create a repository with a README file and a license.

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



2. Create a new GitHub repository called `LogisticsShippingRates` and make sure that it is **Public**.
3. Select the **Add a README** file field. Click on the **Choose a license** field and select `Apache License 2.0` from the drop down. Finally, click on **Create repository**.

A screenshot of the GitHub 'Create new repository' form. The 'Owner' is set to the user's profile. The 'Repository name' is 'LogisticsShippingRates', which is highlighted with a red box. Below it, a green checkmark indicates 'LogisticsShippingRates is available.' The 'Description' field is empty. Under the 'Configuration' section, 'Choose visibility' is set to 'Public'. 'Start with a template' is set to 'No template'. 'Add README' is checked (highlighted with a red box). 'Add .gitignore' is set to 'No .gitignore'. 'Add license' is set to 'Apache License 2.0' (highlighted with a red box). At the bottom right, the 'Create repository' button is highlighted with a red box.

4. After successfully creating a new repo, you will see the following screenshot.

LogisticsShippingRates

Type / to search

<> Code

Issues

Pull requests

Actions

Projects

Wiki

Security

Insights

Settings

LogisticsShippingRates

Public

Pin

Watch 0

Fork

main 1 Branch 0 Tags

Go to file

Add file

<> Code

About

Initial commit c9e61b1 · 1 minute ago 1 Commit

LICENSE Initial commit 1 minute ago

README.md Initial commit 1 minute ago

README Apache-2.0 license

LogisticsShippingRates

No description

Readme

Activity

0 stars

0 watchir

0 forks

Releases

No releases pub

Create a new re

Your repository is created and includes the README and LICENSE files. Now, you are ready to update your repository files to include useful information for your community.

Task 2: Update the README file.

1. Click on README.md file to open it and click the pencil icon to edit it.

LogisticsShippingRates / README.md

commit

Blame

1 lines (1 loc) · 24 Bytes

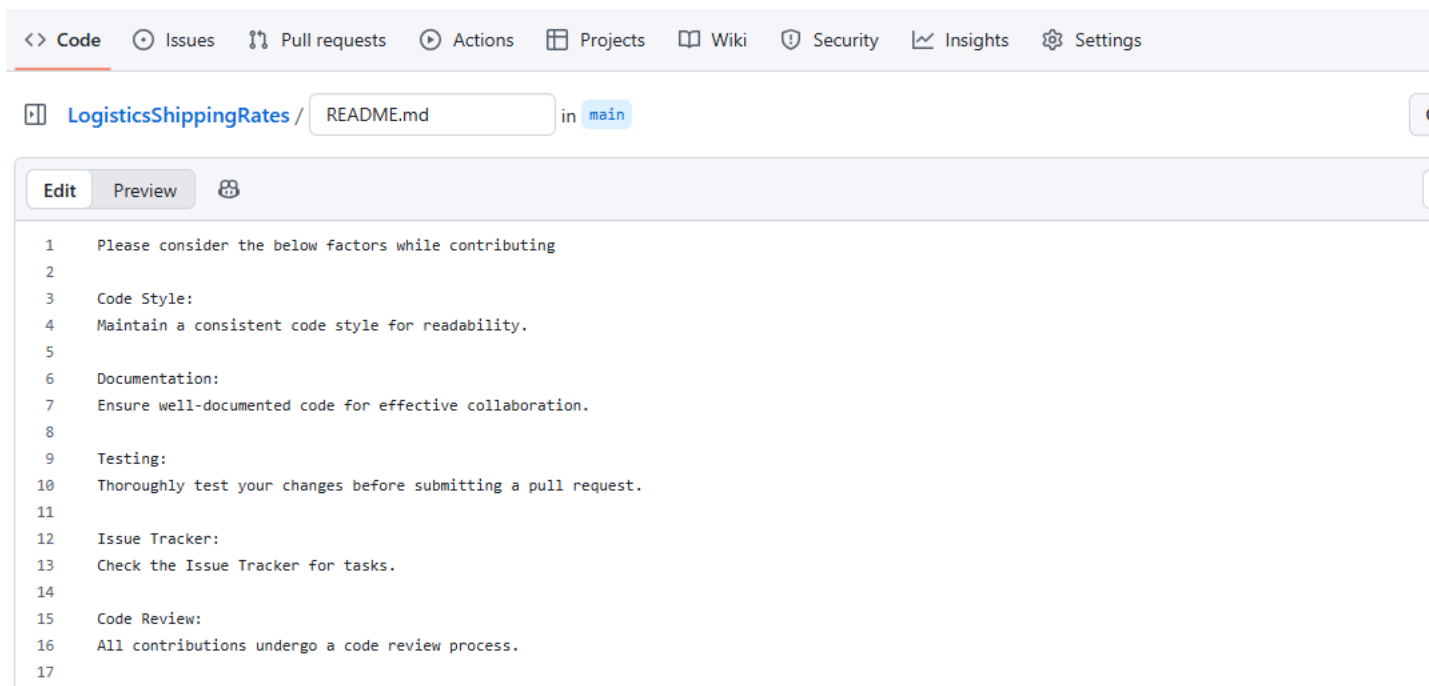
Code 55% faster with GitHub Copilot

LogisticsShippingRates

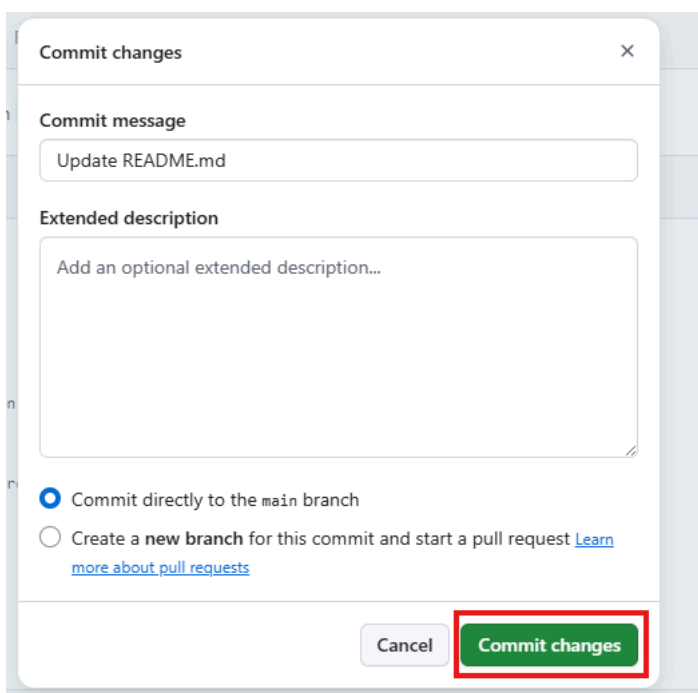
2. Add the following information to the file:

```
Please consider the below factors while contributing
Code Style:
Maintain a consistent code style for readability.
Documentation:
Ensure well-documented code for effective collaboration.
Testing:
Thoroughly test your changes before submitting a pull request.
Issue Tracker:
Check the Issue Tracker for tasks.
Code Review:
All contributions undergo a code review process.
Licensing:
Contributions are licensed.
```

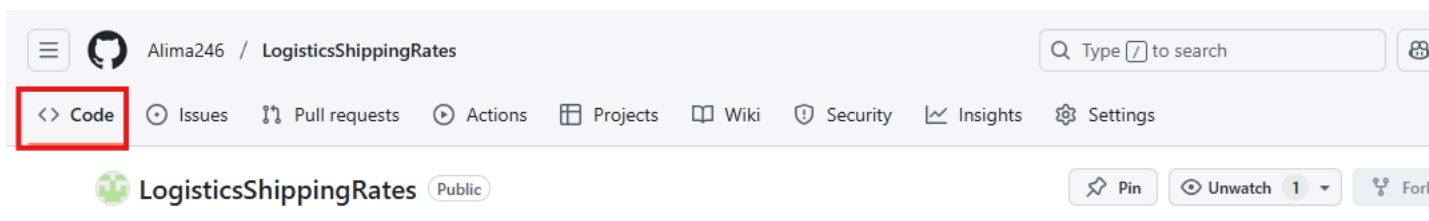
3. Now click on Commit changes



4. Ensure that the radio button to **commit directly to the main branch** is selected. Add a commit message and click on Commit Changes



NOTE: After completing the steps in each of the tasks, if you are unable to see the options depicted in the provided screenshots, click on the Code option as highlighted in below screenshot:

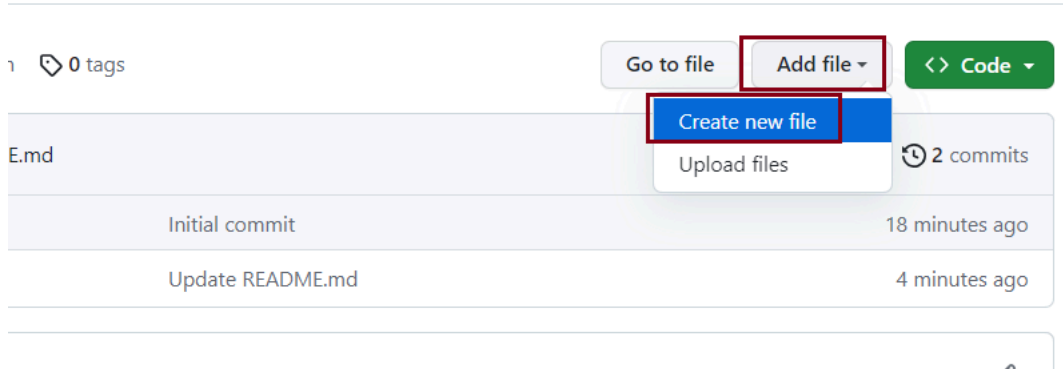


Task 3: Add a code of conduct.

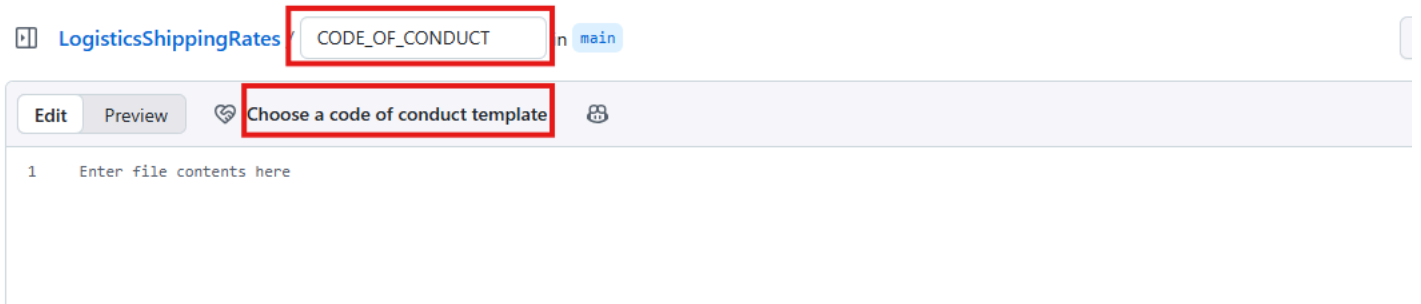
A code of conduct helps set ground rules for the behavior of your project's participants. It defines standards for how to engage in a community.

GitHub provides templates for common codes of conduct to help you quickly add one to your project.

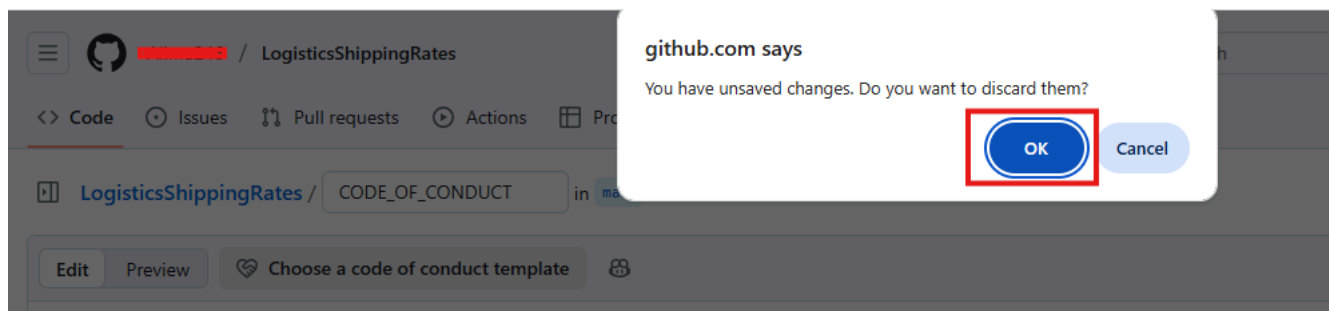
1. To create a new file click on **Add file** and then Create New File.



2. Add a new file named `CODE_OF_CONDUCT.md` to the root folder of the repository. The **Choose a code of conduct template** button is displayed.



3. Click the **Choose a code of conduct template** button. You will see a popup mentioning you may lose unsaved changes. Click on OK.



4. Click on Contributor Covenant to load this template. Then click Review and submit to add the file to your project.

Add a code of conduct to your project

Contributor Covenant
Recommended for projects of all sizes

Citizen Code Of Conduct
Suitable for large communities and events

Contributor Covenant Code of Conduct

Our Pledge

We as members, contributors, and leaders pledge to make participation in our community a harassment-free experience for everyone, regardless of age, body size, visible or invisible disability, ethnicity, sex characteristics, gender identity and expression, level of experience, education, socio-economic status, nationality, personal appearance, race, religion, or sexual identity and orientation.

We pledge to act and interact in ways that contribute to an open, welcoming, diverse, inclusive, and healthy community.

Our Standards

Examples of behavior that contributes to a positive environment for our community include:

- Demonstrating empathy and kindness toward other people
- Being respectful of differing opinions, viewpoints, and experiences
- Giving and gracefully accepting constructive feedback

5. Scroll down the page to read the content. Click on **Commit changes**.

LogisticsShippingRates / CODE_OF_CONDUCT.md in main

Edit Preview Choose a code of conduct template

```
1 # Contributor Covenant Code of Conduct
2
3 ## Our Pledge
4
5 We as members, contributors, and leaders pledge to make participation in our
6 community a harassment-free experience for everyone, regardless of age, body
7 size, visible or invisible disability, ethnicity, sex characteristics, gender
8 identity and expression, level of experience, education, socio-economic status,
9 nationality, personal appearance, race, religion, or sexual identity
10 and orientation.
11
12 We pledge to act and interact in ways that contribute to an open, welcoming,
13 diverse, inclusive, and healthy community.
14
```

Use **Control + Shift + m** to toggle the **tab** key moving focus. Alternatively, use **esc** then **tab** to move to the next interactive element on the page.

Attach files by dragging & dropping, selecting or pasting them.

6. Ensure that the radio button to **Commit directly to the main branch** is selected and click **Commit changes**.

Commit changes

Commit message

Create CODE_OF_CONDUCT.md

Extended description

Add an optional extended description...

☒ Commit directly to the main branch

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

Cancel

Commit changes

If the page is zoomed in, you may see only the **Commit changes** button and not the radio button. Zoom out or scroll down to see the radio button.

7. Your project now contains a code of conduct.

main
LogisticsShippingRates /

Create CODE_OF_CONDUCT.md

Name	Last commit message
CODE_OF_CONDUCT.md	Create CODE_OF_CONDUCT.md
LICENSE	Initial commit
README.md	Update README.md

README.md

Task 4: Establish contribution guidelines.

Contribution guidelines provide clear instructions for individuals looking to contribute to the project. To implement these guidelines, please follow these steps:

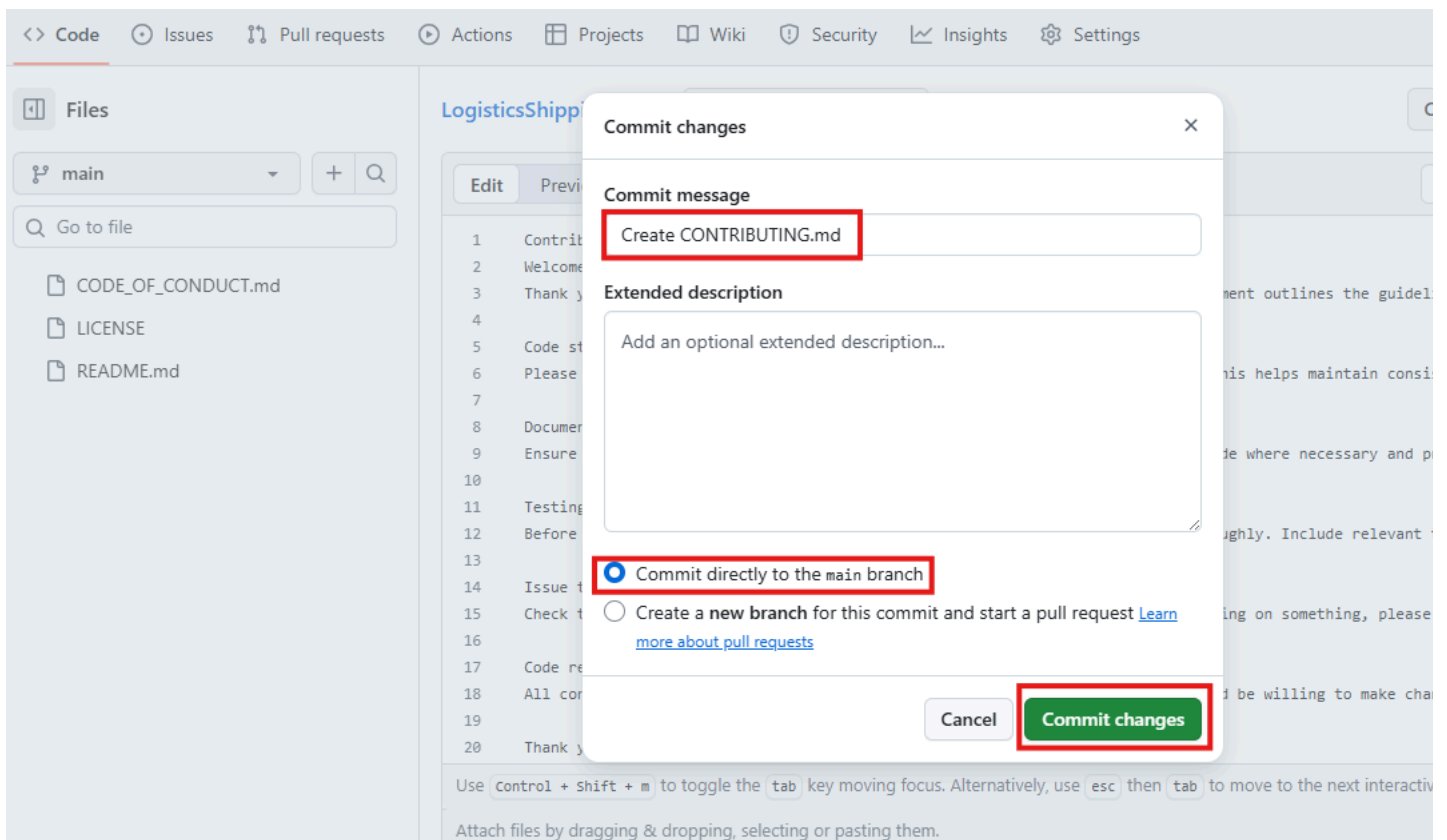
1. Create a new file in the repository's root directory and name it CONTRIBUTING.md.
2. Inside the file, include the following information:

```

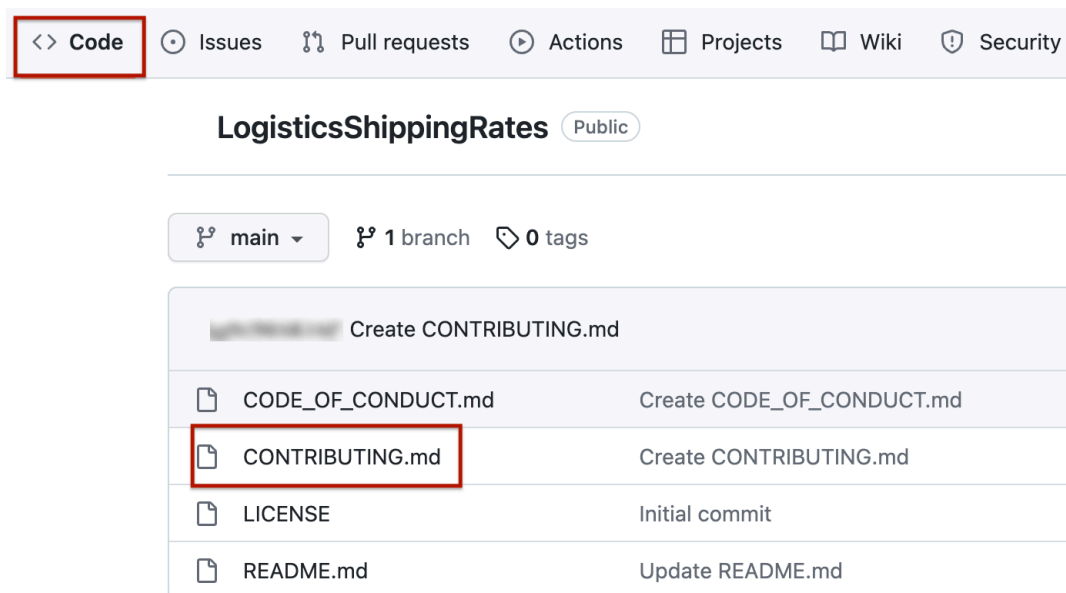
Contribution guidelines
Welcome Contributors!
Thank you for considering contributing to the centralized repository. This document outlines the guidelines for contributing to the development
Code style
Please follow the coding style and conventions used in the existing codebase. This helps maintain consistency across the project.
Documentation
Ensure that your contributions are well-documented. Include comments in your code where necessary and provide a clear and concise description of
Testing
Before submitting a pull request, make sure your changes have been tested thoroughly. Include relevant test cases and ensure that existing tests
Issue tracker
Check the issue tracker for any open issues or feature requests. If you're working on something, please comment on the issue to let others know.
Code review
All contributions will go through a code review process. Be open to feedback and be willing to make changes if necessary. Code reviews help main
Thank you for your contribution!

```

3. Commit this file to your repository.



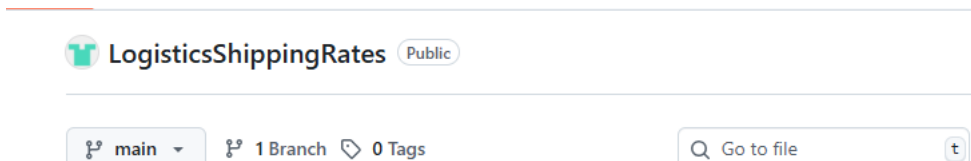
4. Click on Code to go to the repository home page.



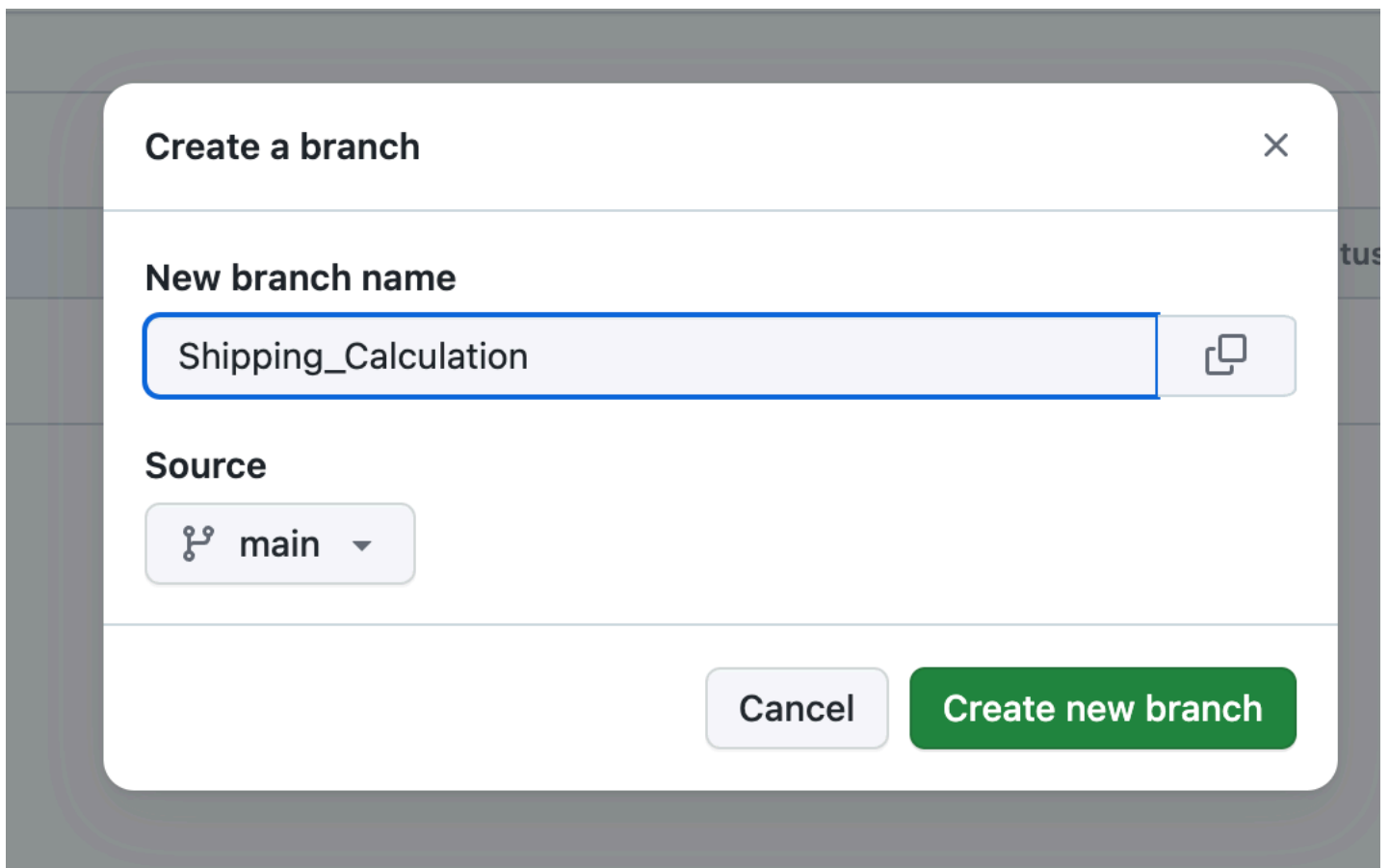
Your project now contains Contribution Guidelines as well.

Task 5: Create a new branch.

1. Click on branch in your repository.



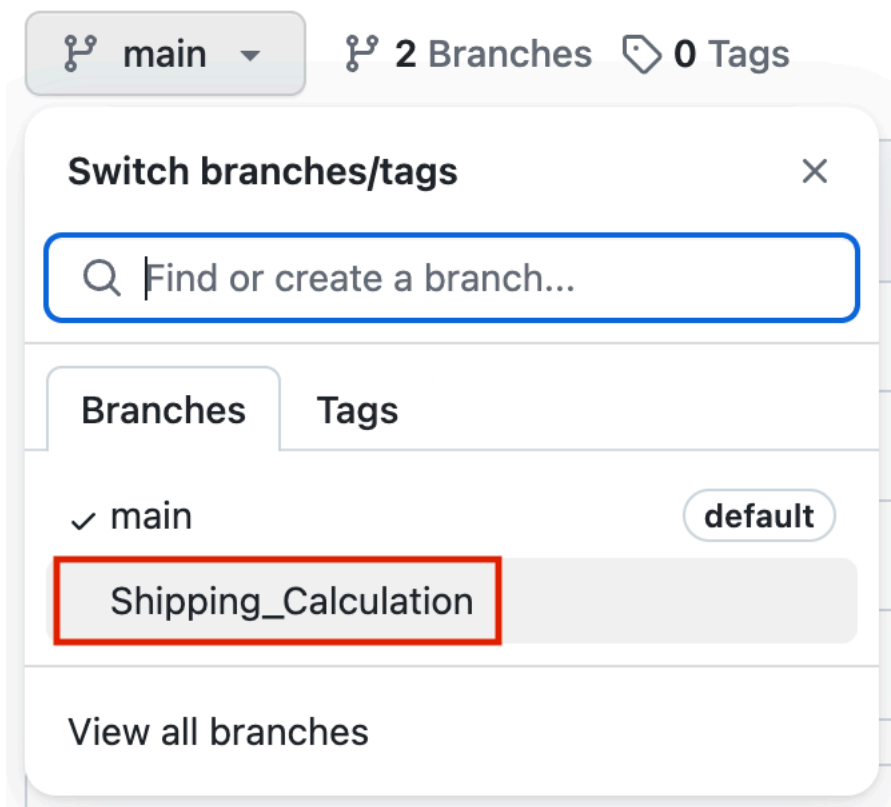
2. Now, click on New Branch and enter the branch name as Shipping_Calculation, then click Create New Branch



This will be used for adding your Shipping Calculation code.

Task 6: Add the Shipping Calculation code in this branch.

1. Change to the Shipping_Calculation branch by selecting it from the dropdown.



2. Create a new file named `Shipping_Cost_Calculator.py`.
3. Add the following code in the new file:


```
# Shipping Cost Calculator
## Input package weight and shipping rate
weight = float(input("Enter the package weight in kilograms: "))
rate = float(input("Enter the shipping rate per kilogram: "))
## Calculate shipping cost
shipping_cost = weight * rate
## Display the result
print(f"Shipping Cost: {shipping_cost} USD")
```

4. Ensure that the radio button to **Commit directly to the Shipping_Calculation branch** is selected and click **Commit changes**.

The screenshot shows the GitHub web interface for a repository named 'LogisticsShippingRates'. The file 'Shipping_Cost_Calculator.py' is being edited. The commit dialog is open, showing the commit message 'Create Shipping_Cost_Calculator.py' and the option to 'Commit directly to the Shipping_Calculation branch' selected. The 'Extended description' field is empty. The 'Cancel' button is visible at the bottom right of the dialog.

The Shipping_Calculation branch now contains the Shipping_Cost_Calculator.py file.

Task 7: Create a pull request to the main branch.

1. Go to the Pull request tab and click on Compare & pull request.

<> Code

⦿ Issues

🔗 Pull requests

▶ Actions

📁 Projects



Label issues and pull requests for new contributors

Now, GitHub will help potential first-time contributors [discover](#)


good first issue

🔗 **Shipping_Calculation** had recent pushes 1 minute ago

2. Add an appropriate commit message and click on Create pull request.

Open a pull request

Create a new pull request by comparing changes across two branches. If you ne



base: main ▾

← ...

compare: Shipping_Calculation ▾

✓ Able to merge.



Add a title

Merge Shipping_Cost_Calculator.py to the main branch


Add a description


Write

Preview

H B I

Add your description here...


 Markdown is supported


 Paste, drop, or click to add files

Task 8: Merge the branch to the main branch.

1. Click on Merge pull request.


Add more commits by pushing to the Shipping_Calculation branch of LogisticsShippingRates.




**Require approval from specific reviewers before merging**

Add rule ×

[Branch protection rules](#) ensure specific people approve pull requests before they're merged.

**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**

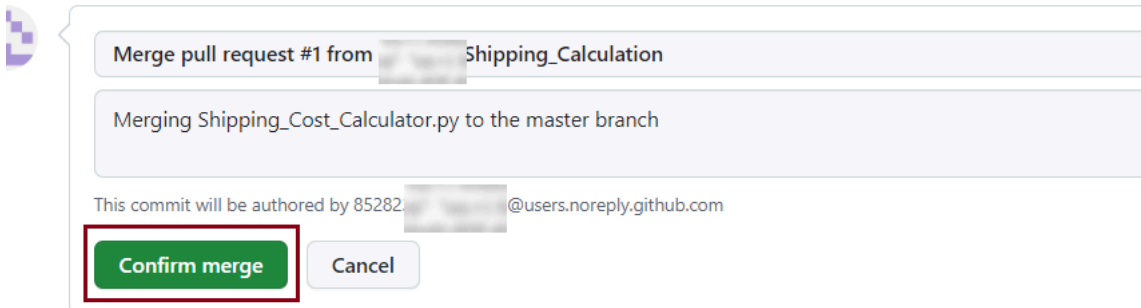
Merging can be performed automatically.

Merge pull request ▾

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

2. Click on Confirm merge.

3. The successfully merged changes will be shown in the Github UI as below:



Merge pull request #1 from Shipping_Calculation

Merging Shipping_Cost_Calculator.py to the master branch

This commit will be authored by 85282 @users.noreply.github.com

Confirm merge

Cancel

4. You will see the below message once the branch is merged.

Merge Shipping_Cost_Calculator.py to

 Merged

lauvshree merged 1 commit into **main** from **Shipping_Calculati**

 Conversation 0

 Commits 1

 Checks 0

 Files



lauvshree commented now

No description provided.



Create Shipping_Cost_Calculator.py



lauvshree merged commit **de4a353** into **main** now

5. Check to see that your project now contains the Shipping_Cost_Calculator.py file as well.

main ▾

2 Branches 0 Tags

Go to



lauvshree Merge pull request #1 from lauvshree/Shipping_Calculation

CODE_OF_CONDUCT.md	Create CODE_OF_CONDUCT.md
CONTRIBUTING.md	Create CONTRIBUTING.md
LICENSE	Initial commit
README.md	Update README.md
Shipping_Cost_Calculator.py	Create Shipping_Cost_Calculator.py

Congratulations

In this lab you have created a repository in your GitHub account with a README.md file and a License. You have then edited the README file and added Code of Conduct, and Contribution Guidelines. Further, you have committed these files to your repository. Finally, you have added the Shipping logistics calculator file to a new branch, and merged it to the main branch using a Pull Request.

Author(s)

Nikesh Kumar

K Sundararajan



Skills Network