**Applying branch rule for branches:**

Steps:

1. log into git hub.
2. Go to setting.
3. Select branch in code and automation
4. In side the branch there have 2 option
5. Default Branch
6. Branch protection rule
7. In branch protection rule click add rule button

Step 1: Go to setting.

Graphical user interface, text, application, email

Description automatically generated

Step 2: Select branch in code and automation

Graphical user interface, text, application, email

Description automatically generated

Step 3: In branch protection rule click add rule button

Graphical user interface, text, application, email

Description automatically generated

Step 4: select the options which are selected in picture and click create button.

Graphical user interface, text, application, email

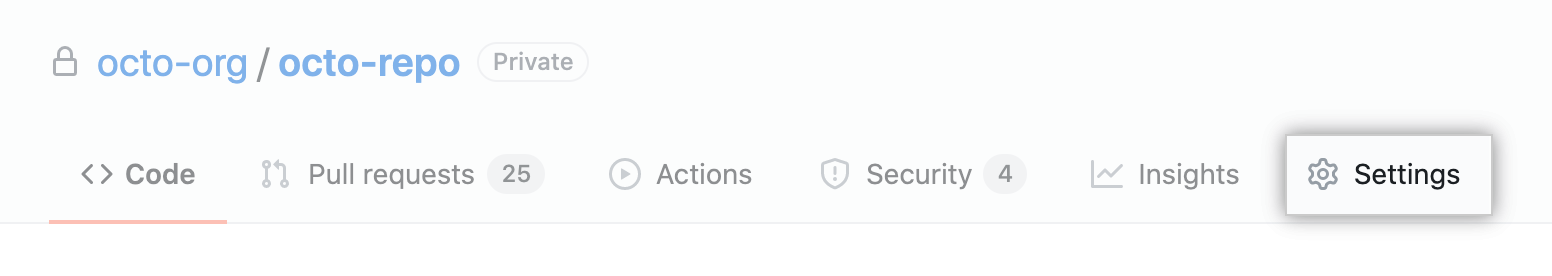
Description automatically generated

Step 5: result

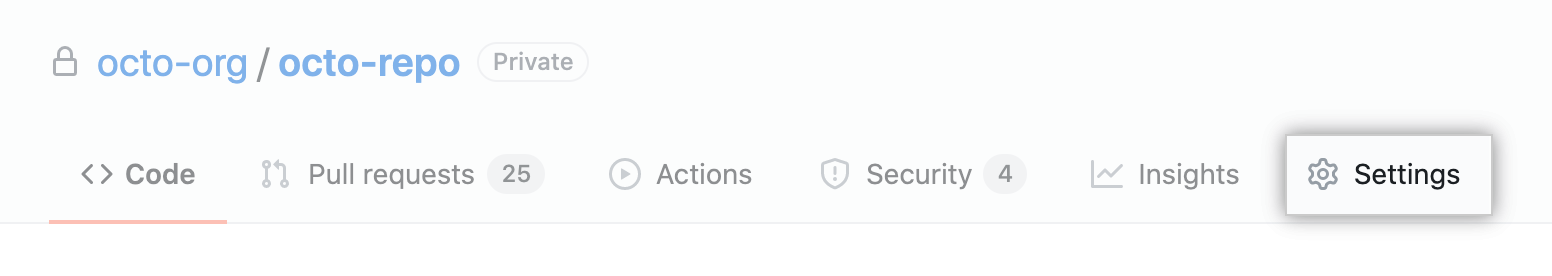
Graphical user interface, text, application, email

Description automatically generated

## Editing a branch protection rule

1. On GitHub.com, navigate to the main page of the repository.
2. Under your repository name, click  **Settings**.
3. In the "Code and automation" section of the sidebar, click **Branches**.
4. To the right of the branch protection rule you want to edit, click **Edit**.Edit button
5. Make your desired changes to the branch protection rule.
6. Click **Save changes**.Save changes button

**Deleting a branch protection rule**

1. On GitHub.com, navigate to the main page of the repository.
2. Under your repository name, click  **Settings**.
3. In the "Code and automation" section of the sidebar, click **Branches**.
4. To the right of the branch protection rule you want to delete, click **Delete**.Delete button

**Git tag**

**Git tags** are used as reference points in your development worflow.

**In order to create a new tag, you have to use the “git tag” command and specify the tag name that you want to create.**

$ git tag <tag\_name>

**To create a Git tag with a message, use the “git tag” command with the “-a” option for “annotated” and the “-m” option for message.**

$ git tag -a <tag\_name> -m "message"

