

GitHub tutorial

For Beginners



Contents

[GitHub 2](#_Toc479281393)

[GitHub online 2](#_Toc479281394)

[To create a new repository 2](#_Toc479281395)

[To create a branch 2](#_Toc479281396)

[Make and commit changes 2](#_Toc479281397)

[Open a Pull Request 2](#_Toc479281398)

[Merge your Pull Request 3](#_Toc479281399)

[To add a Collaborator 3](#_Toc479281400)

[GitHub for Windows 3](#_Toc479281401)

[Downloading and Installing Windows GitHub: 3](#_Toc479281402)

[Authenticating to GitHub 4](#_Toc479281403)

[Adding a repository from your local computer to GitHub Desktop 4](#_Toc479281404)

[Cloning Repositories 4](#_Toc479281405)

[Creating a Branch 4](#_Toc479281406)

[Committing and reviewing changes 5](#_Toc479281407)

[Reverting a commit 5](#_Toc479281408)

[Sending a pull request 5](#_Toc479281409)

# **GitHub**

# **GitHub online**

GitHub is a code repository, version control and a collaboration tool. It lets you and others to work together on projects from anywhere.

A repository is usually used to collaborate a single project developed by a team members.

Here is the link to create account in GitHub.

<https://github.com>

## **To create a new repository**

* In the upper right corner, next to your avatar, click and then select New repository.
* Name your repository

## **To create a branch**

Branching is the way to work on different versions of a repository at one time.

By default, your repository has one branch named master which is considered to be the definitive branch.

To create a new branch, follow the below steps

* Go to your new repository.
* Click the drop down at the top of the file list that says branch: master.
* Type a branch name, into the new branch text box.
* Select the blue Create branch box or hit “Enter”.

## **Make and commit changes**

On GitHub, saved changes are called commits. Each commit has an associated commit message, which is a description explaining why a particular change was made. Commit messages capture the history of your changes, so other contributors can understand what you’ve done and why.

To make and commit changes follow the below steps

* Click the branch you created.
* Click on the file on which you want to commit.
* Click the pencil icon in the upper right corner of the file view to edit.
* In the editor, write a bit about yourself.
* Write a commit message that describes your changes.
* Click Commit changes button

## **Open a Pull Request**

Pull Requests are the heart of collaboration on GitHub. When you open a pull request, you’re proposing your changes and requesting that someone review and pull in your contribution and merge them into their branch. Pull requests show differences, of the content from both branches. The changes, additions, and subtractions are shown in green and red.

You can even open pull requests in your own repository and merge them yourself.

To pull request follow the below steps

* Click the Pull Request tab, then from the Pull Request page, click the green New pull request button
* Select the branch you made, to compare with master (the original).
* Look over your changes in the diffs on the Compare page, make sure they’re what you want to submit.
* When you’re satisfied that these are the changes you want to submit, click the Create Pull Request button.
* Give your pull request a title and write a brief description of your changes.
* When you’re done with your message, click Create pull request!

## **Merge your Pull Request**

* Click the green Merge pull request button to merge the changes into master.
* Click Confirm merge.

## **To add a Collaborator**

* Navigate to the repository on GitHub you wish to share with your collaborator.
* Click on the "Settings" link in the right-side menu.
* On the new page, click the "Collaborators" menu item on the left side of the page.
* Start typing the new collaborator's GitHub username into the text box.
* Select the GitHub user from the list that appears below the text box.
* Click the "Add" button.

# **GitHub for Windows**

## **Downloading and Installing Windows GitHub**

Step 1: Visit the  [GitHub Desktop download page](https://desktop.github.com/)…

Step 2: Choose to download for windows

Step 3: In your computer's Downloads folder, double-click GitHub Desktop.

Step 4: In the pop-up window, Click Install.

Step 5: After the program has been installed, click Run.

## **Authenticating to GitHub**

Step 1: In the upper-right corner of the app, click on tools and options symbol.

Step 2: Choose Options.

Step 3: Under Accounts, click Add account.

Step 4: Under Log in, choose GitHub depending on which product you are using.

Step 5: Type your credentials, then click Log in.

## **Adding a repository from your local computer to GitHub Desktop**

Step 1: In the top left corner of the app, click +.

Step 2: At the top of the menu, click **Add.**

Step 3: Click **Browse** and navigate to the local repository you want to add.

Step 4: Click **Add repository**.

## **Cloning Repositories**

Step 1: In the top left corner of the app, click **+**.

Step 2: At the top of the menu, click **Clone.**

Step 3: In the left sidebar, you can select to work with a personal repository or Organization repository.

Step 4: Click on the repository which you like to clone.

Step 5: At the bottom of the menu, click Clone Repository.

## **Creating a Branch**

## Step 1: At the top of the app, click onC:\Users\prade\AppData\Local\Microsoft\Windows\INetCache\Content.Word\branch.png

Step 2: In the Name field, type a meaningful name for your branch

Step 3: Click Create new branch.

## **Committing and reviewing changes**

* Choosing a branch and making changes

At the top of the app, [create a new branch](https://help.github.com/desktop/guides/contributing/creating-a-branch-for-your-work), or next to the icon, choose an existing branch to compare your changes against.

* Selecting changes to include in a commit

As you make changes to files in your text editor and save them locally, you also see the changes in GitHub.

To add all changes in all files to a single commit, keep the checkbox at the top of the list selected.

To add all changes in one or more files to a single commit, deselect the files you don't want included, leaving only the files you want in the commit.

* Creating a partial commit

If one file contains multiple changes, but you only want some of those changes to be included in a commit, you can create a partial commit. The rest of your changes will remain intact, so that you can make additional modifications and commits. This allows you to make separate, meaningful commits, such as keeping line break changes in a commit separate from code changes.

When you review the diff of the file, the lines that will be included in the commit are highlighted in blue. To exclude the change, click the changed line so the blue disappears.

* Write a commit message and sync your changes

Once you're satisfied with the changes you've chosen to include in your commit, write your commit message and sync your changes

At the bottom of the list of changes, in the Summary field, type a short, meaningful commit message. Optionally, you can add more information about the change in the Description field.

Under the Description field, click Commit to BRANCH.

When you're ready to sync your commits with the remote repository, in the upper-right corner of the app, click Sync

## **Reverting a commit**

Step 1: On the comparison graph, click a commit.

Step 2: In the commit history list, click the commit you'd like to review.

Step 3: Click Revert.

## **Sending a pull request**

When you're ready to propose changes to your project, you can open a pull request from GitHub Desktop.

Step 1: At the top of the app, click Pullrequest.

Step 2: In the Title field, type a descriptive title for your pull request.

Step 3: In the Description field, type more information about the pull request, such as:

What's being updated, what kind of review you're looking for, People or teams who you want to notify about the pull request.

Step 4: At the bottom of the pull request pane, click Send pull request.