Title: GitHub

Submission Date: 20th May 2020

Type of Submission: Share the GitHub Repository to the Demonstrator of the subject.

Lab Sheet Covered: Lab sheet 12 **Overview:**

GitHub is a code hosting platform for version control and collaboration. It lets you and others work together on projects from anywhere.

This tutorial teaches you how to use GitHub:

- To create a new repository
- To create branches and perform operations (commit, pull, merge) □ To clone a repository locally

Step 01: In order to work with GitHub, you need to have GitHub account. If you don‟t already have an account, you can go to https://github.com/ and fill the **sign up** form to create a new account otherwise you sign in your account.

Step 02:

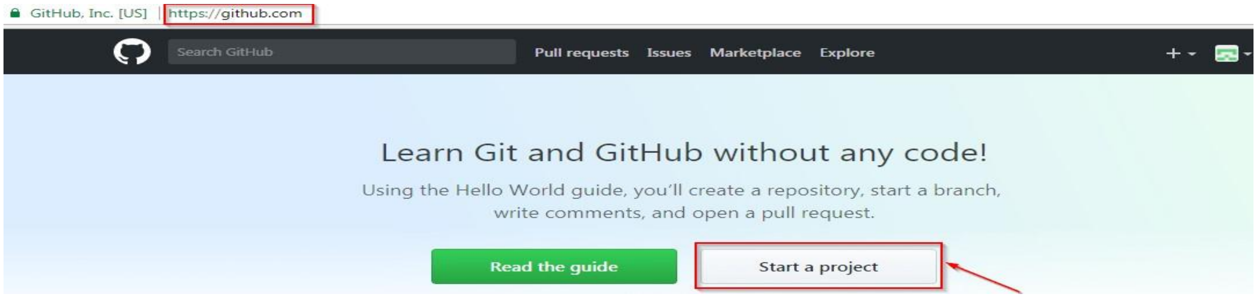
Creating a new repository

A repository is a storage space where your project lives. It can be local to a folder on your computer, or it can be a storage space on GitHub or another online host. You can keep code files, text files, images or any kind of a file in a repository. You need a GitHub repository when you have done some changes and are ready to be uploaded. This GitHub repository acts as your remote repository.

Follow these steps to create a GitHub repository:

- Once you signed up for github, then click on **Start a project.**

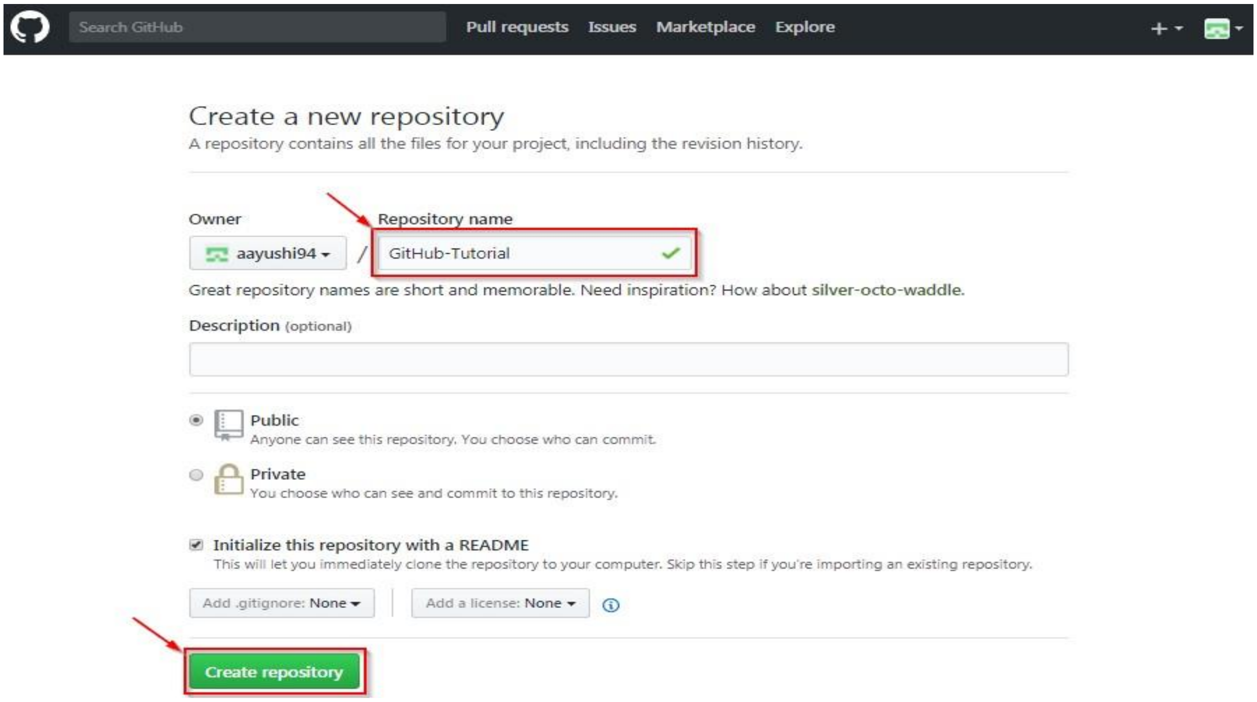
See the below screenshot to get a better understanding:


GitHub, Inc. [US] | https://github.com

Learn Git and GitHub without any code!

Using the Hello World guide, you'll create a repository, start a branch, write comments, and open a pull request.

- If you click on **Start a project,** it will display **create a new repository** page. Enter any repository name and click on "Create Repository". You can also give a description to your repository (optional).


Create a new repository

A repository contains all the files for your project, including the revision history.

Owner: aayushi94 / Repository name: GitHub-Tutorial ✓

Great repository names are short and memorable. Need inspiration? How about silver-octo-waddle.

Description (optional)

● Public
Anyone can see this repository. You choose who can commit.

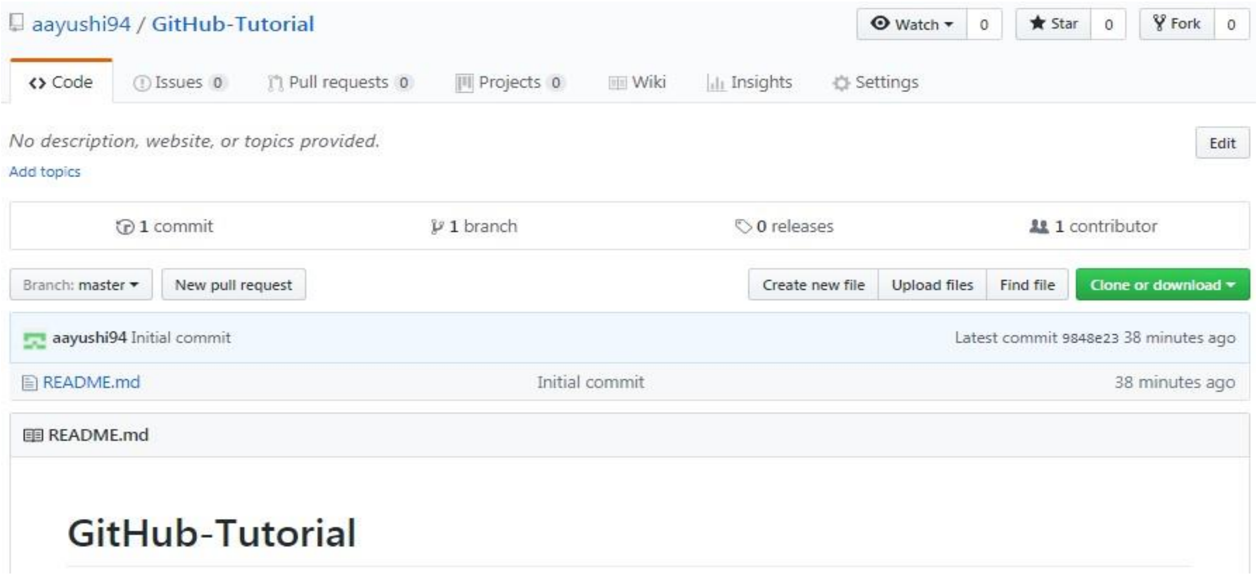
○ Private
You choose who can see and commit to this repository.

☑ Initialize this repository with a README
This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: None | Add a license: None

Now, if you noticed by default a GitHub repository is public which means that anyone can view the contents of this repository whereas in a private repository, you can choose who can view the content. Also, private repository is a paid version. Also, if you refer the above screenshot, initialize the repository with a README file. This file contains the description of the file and once you check this box, this will be the first file inside your repository.

If your repository is successfully created, it will look like the below picture:



No description, website, or topics provided.

aayushi94 Initial commit — Latest commit 9848e23 38 minutes ago

README.md — Initial commit — 38 minutes ago

GitHub-Tutorial

Once you created repository, you are ready to commit, pull, push and perform all the other operations.

Step 03:

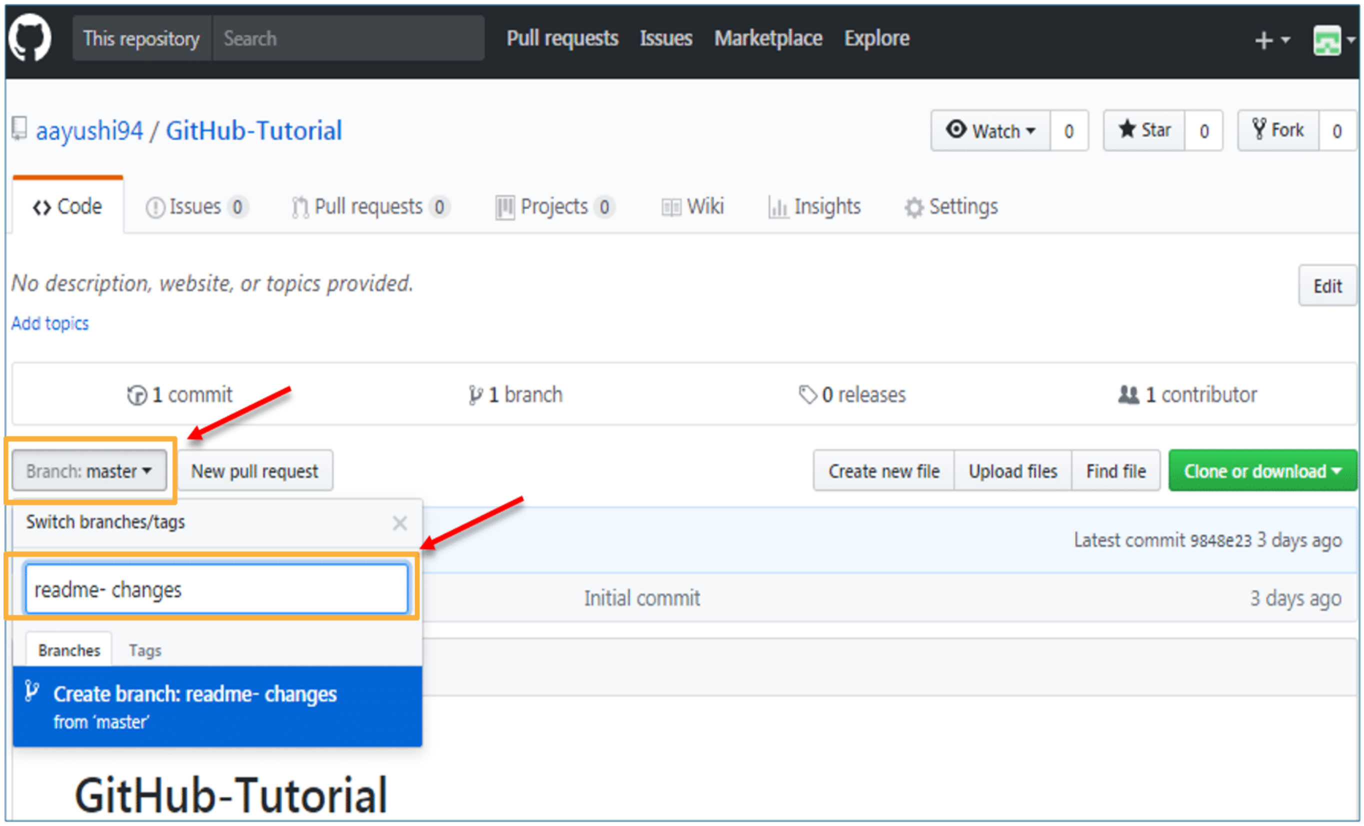
Create branches and perform operations

Branching: Branches help you to work on different versions of a repository at one time. Let‟s say you want to add a new feature (which is in the development phase), and you are afraid at the same time whether to make changes to your main project or not. This is where git branching comes to rescue. Branches allow you to move back and forth between the different states/versions of a project. In the above scenario, you can create a new branch and test the new feature without affecting the main branch. Once you are done with it, you can merge the changes from new branch to the main branch. Here the main branch is the master branch, which is there in your repository by default.

To create a branch in GitHub, follow the below steps:

- Click on the dropdown **Branch: master**
- As soon as you click on the branch, you can find an existing branch or you can create a new one. Creates a new branch with a name **readme-changes.**

See the below screenshot for better understanding:



No description, website, or topics provided.

readme- changes

Create branch: readme- changes
from 'master'

Latest commit 9848e23 3 days ago

Initial commit 3 days ago

Once you have created a new branch, you have two branches in your repository now i.e. read-me (master branch) and readme- changes. The new branch is just the copy of master branch.

So let‟s perform some changes in our new branch and make it look different from the master branch.

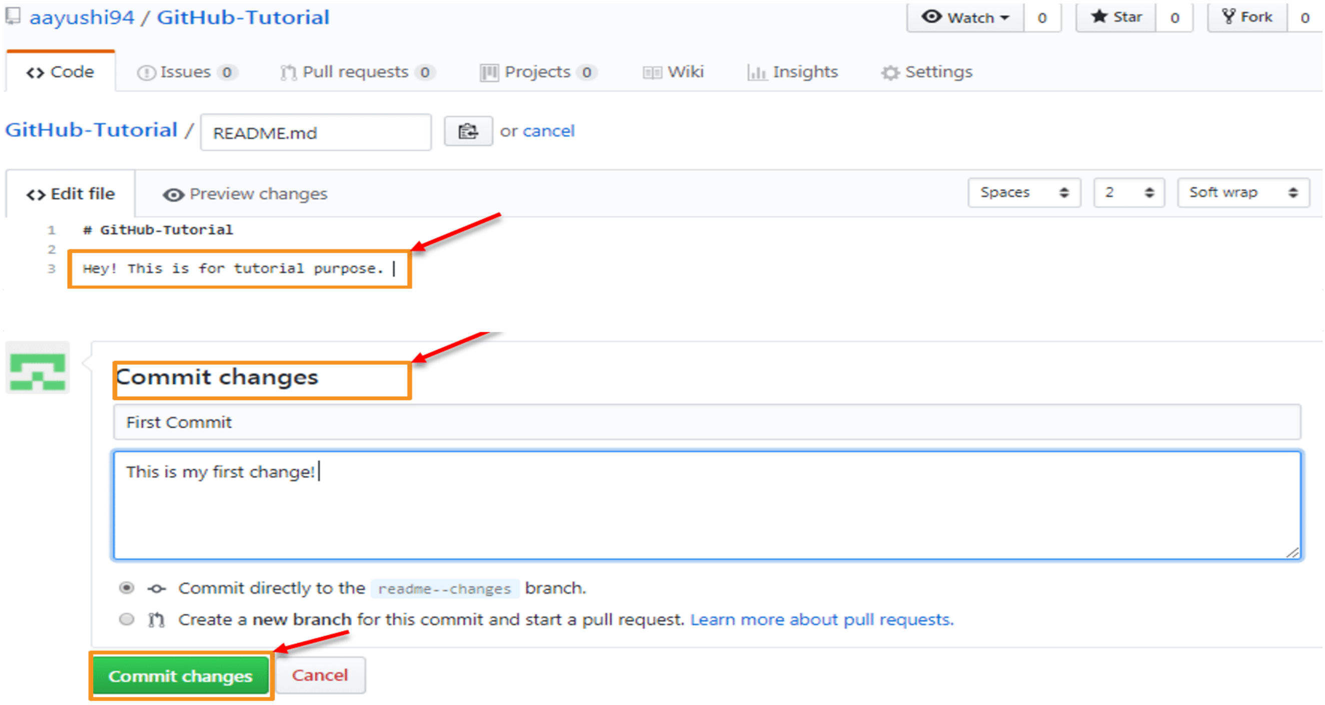
GitHub Operations

- **Commit Command**: This operation helps you to save the changes in your file. When you commit a file, you should always provide the message, just to keep in the mind the changes done by you. Though this message is not compulsory but it is always recommended so that it can differentiate the various versions or commits you have done so far to your repository. These commit messages maintain the history of changes which in turn help other contributors to understand the file better.

Follow the below steps:

- Click on **readme- changes** file which we have just created.
- Click on the **pencil icon** in the rightmost corner of the file.
- Once you click on that, an editor will open where you can type in the changes or anything.
- Write a commit message which identifies your changes.
- Click **commit changes** in the end.

See the below screenshot for better understanding:



```
# GitHub-Tutorial

Hey! This is for tutorial purpose.
```

First Commit

This is my first change!

● Commit directly to the `readme--changes` branch.

○ Create a **new branch** for this commit and start a pull request. Learn more about pull requests.

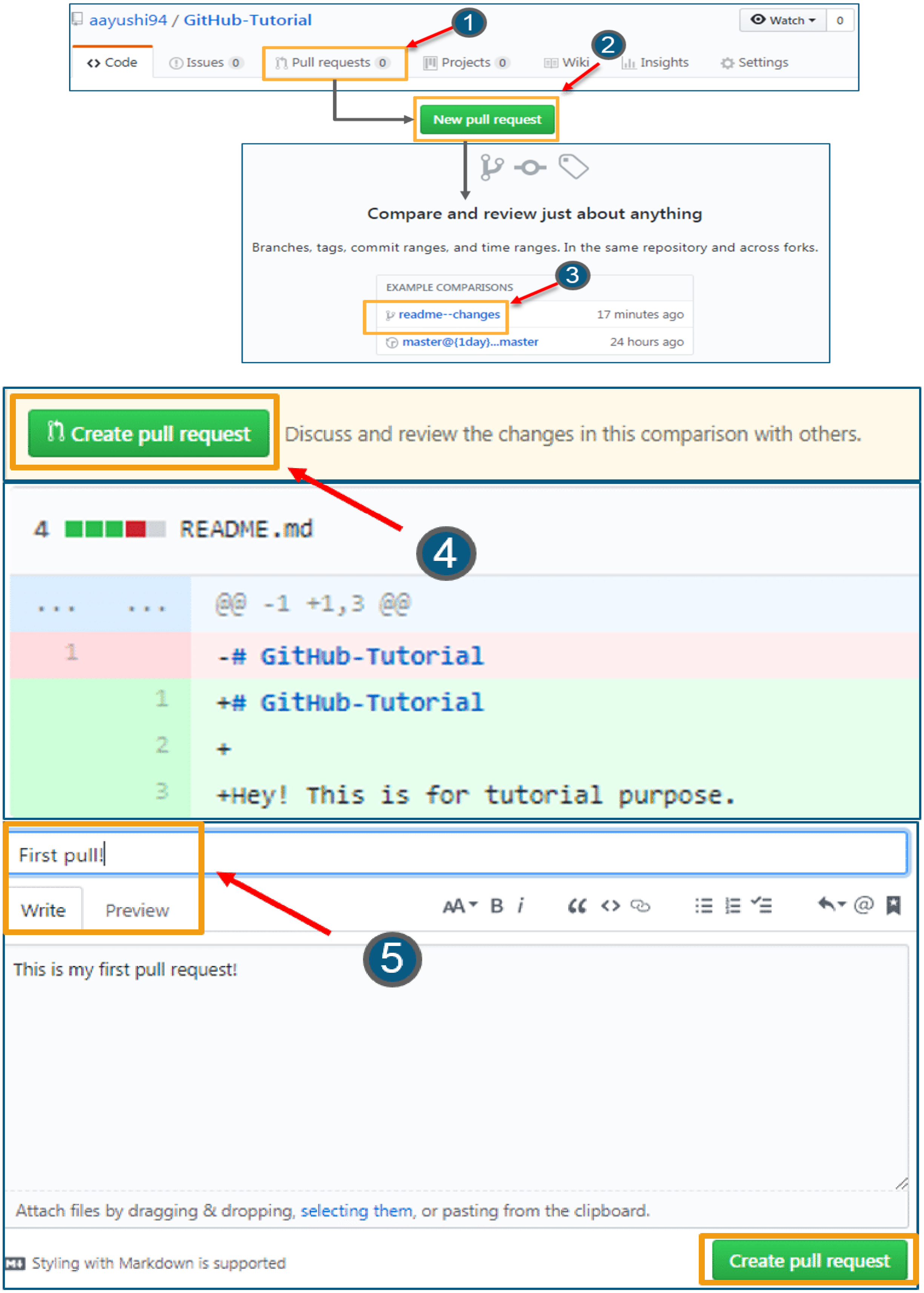
❖ Now you have successfully made your first commit. Now this “readme- changes” file is different from the master branch.

- **Pull Command:** Pull command is the most important command in GitHub. It tells the changes done in the file and request other contributors to view it as well as merge it with the master branch. Once the commit is done, anyone can pull the file and can start a discussion over it. Once it‟s all done, you can merge the file. Pull command compares the changes which are done in the file and if there are any conflicts, you can manually resolve it.

Follow the bellow steps to pull request in GitHub:

- Click the **Pull requests** tab
- Click **New pull request**
- Once you click on pull request, select the branch and click „readme- changes‟ file to view changes between the two files present in our repository
- Click **Create pull request**
- Enter any title, description to your changes and click on **Create pull request**

See the below screenshots for better understanding:


aayushi94 / GitHub-Tutorial
Watch 0
Code
Issues 0
Pull requests 0
Projects 0
Wiki
Insights
Settings
1
2
New pull request
Compare and review just about anything
Branches, tags, commit ranges, and time ranges. In the same repository and across forks.
EXAMPLE COMPARISONS
3
readme--changes 17 minutes ago
master@{1day}...master 24 hours ago
Create pull request
Discuss and review the changes in this comparison with others.
4
5
Write
Preview
Attach files by dragging & dropping, selecting them, or pasting from the clipboard.
Styling with Markdown is supported
Create pull request

```
4 README.md
...   ...   @@ -1 +1,3 @@
1           -# GitHub-Tutorial
      1     +# GitHub-Tutorial
      2     +
      3     +Hey! This is for tutorial purpose.
```

First pull!

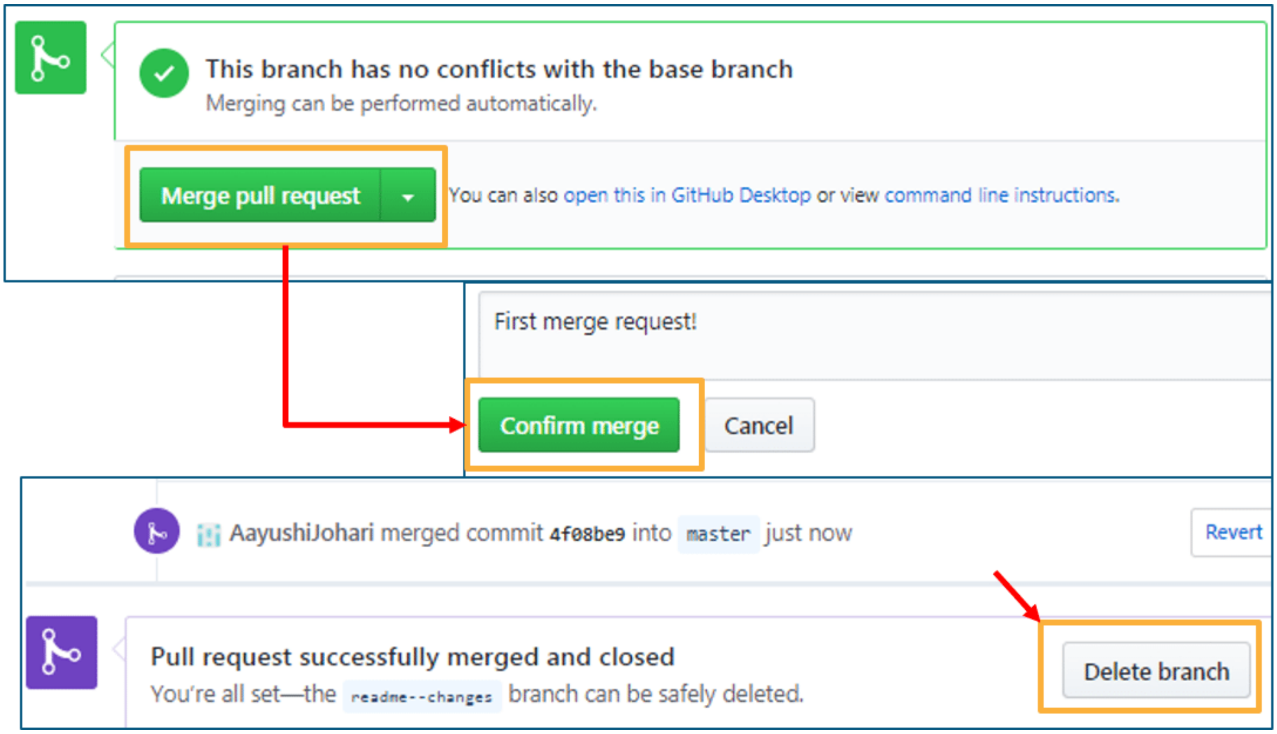
This is my first pull request!

- **Merge Command:** Here comes the last command which merge the changes into the main master branch. Now merge the **readme- changes** file with the master branch/ read-me.

Follow the below steps to merge pull request:

- Click on **Merge pull request** to merge the changes into master branch
- Click **Confirm merge**
- You can **Delete** the branch once all the changes have been incorporated and if there are no conflicts.

See the below screenshots:



This branch has no conflicts with the base branch

Merging can be performed automatically.

You can also open this in GitHub Desktop or view command line instructions.

First merge request!

AayushiJohari merged commit 4f08be9 into master just now

Pull request successfully merged and closed

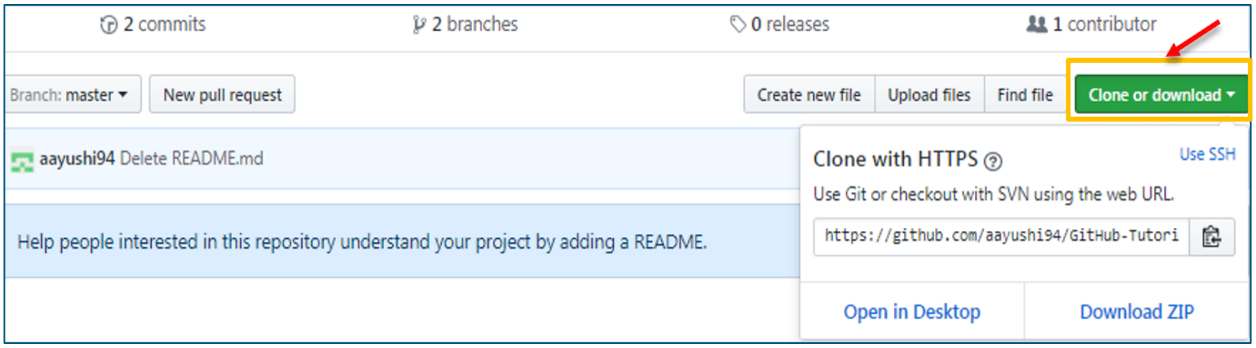
You're all set—the readme--changes branch can be safely deleted.

Step 04:

Cloning a github repository locally

Cloning: Suppose you want to use some code which is present in a public repository, you can directly copy the contents by cloning or downloading. This is the reason why do we need to clone a repository.

See the below screenshot for a better understanding:



aayushi94 Delete README.md

Help people interested in this repository understand your project by adding a README.

Clone with HTTPS

Use Git or checkout with SVN using the web URL.

https://github.com/aayushi94/GitHub-Tutori

Task to be completed:

1. Create a GitHub account.
2. Create a new repository.
3. Push your project (which you have done for Assignment 02) to the repository.
4. Share with me the project.