Git Fork

A fork is a rough copy of a repository. Forking a repository allows you to freely test and debug with changes without affecting the original project. One of the excessive use of forking is to propose changes for bug fixing. To resolve an issue for a bug that you found, you can:

* Fork the repository.
* Make the fix.
* Forward a pull request to the project owner.

Forking is not a Git function; it is a feature of Git service like GitHub.

**When to Use Git Fork**

Generally, forking a repository allows us to experiment on the project without affecting the original project. Following are the reasons for forking the repository:

* Propose changes to someone else's project.
* Use an existing project as a starting point.

Let's understand how to fork a repository on GitHub?

How to Fork a Repository?

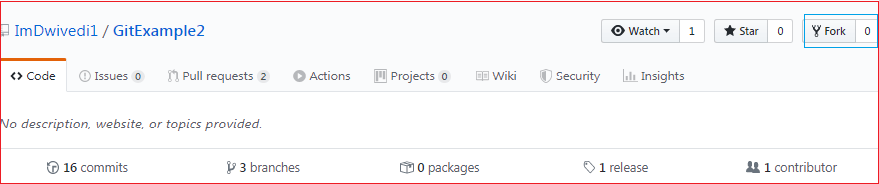
The forking and branching are excellent ways to contribute to an open-source project. These two features of Git allows the enhanced collaboration on the projects.

Forking is a safe way to contribute. It allows us to make a rough copy of the project. We can freely experiment on the project. After the final version of the project, we can create a pull request for merging.

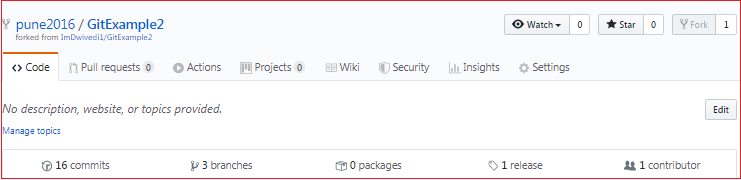
It is a straight-forward process. Steps for forking the repository are as follows:

* Login to the GitHub account.
* Find the GitHub repository which you want to fork.
* Click the Fork button on the upper right side of the repository's page.

We can't fork our own repository. Only shared repositories can be fork. If someone wants to fork the repository, then he must log in with his account. Let's understand the below scenario in which a user pune2016 wants to contribute to our project **GitExample2**. When he searches or put the address of our repository, our repository will look like as follows:



The above image shows the user interface of my repository from other contributors. We can see the fork option at the top right corner of the repository page. By clicking on that, the forking process will start. It will take a while to make a copy of the project for other users. After the forking completed, a copy of the repository will be copied to your GitHub account. It will not affect the original repository. We can freely make changes and then create a pull request for the main project. The owner of the project will see your suggestion and decide whether he wants to merge the changes or not. The fork copy will look like as follows:



As you can see, the forked repository looks like **pune2016/GitExample2**. At the bottom of the repository name, we can see a description of the repository. At the top right corner, the option fork is increased by 1 number.

Hence one can fork the repository from GitHub.

Fork vs. Clone

Sometimes people considered the fork as clone command because of their property. Both commands are used to create another copy of the repository. But the significant difference is that the fork is used to create a server-side copy, and clone is used to create a local copy of the repository.

There is no particular command for forking the repository; instead, it is a service provided by third-party Git service like GitHub. Comparatively, git clone is a command-line utility that is used to create a local copy of the project.

Generally, people working on the same project clone the repository and the external contributors fork the repository.

A pull request can merge the changes made on the fork repository. We can create a pull request to propose changes to the project. Comparatively, changes made on the cloned repository can be merged by pushing. We can push the changes to our remote repository.