**GIT?**

Git is a popular version control system.

It is used for:

* Tracking code changes
* Tracking who made changes
* Coding collaboration

Important Features of GIT

* Manage projects with **Repositories**
* **Clone** a project to work on a local copy
* Control and track changes with **Staging** and **Committing**
* **Branch** and **Merge** to allow for work on different parts and versions of a project
* **Pull** the latest version of the project to a local copy
* **Push** local updates to the main project

GIT And GITHUB both are different, Git is a software while GITHUB is a server

GitHub essentials are:

* Repositories
* Branches
* Commits
* Pull Requests
* Git (the version control software GitHub is built on)

GitHub Repository

A GitHub **repository** can be used to store a development **project**.

It can contain **folders** and any type of **files** (HTML, CSS, JavaScript, Documents, Data, Images).

A GitHub repository should also include a **licence** file and a **README** file about the project.

A GitHub repository can also be used to store ideas, or any resources that you want to share.

Branch

A GitHub branch is used to work with different **versions** of a repository at the same time.

By default a repository has a **master** branch (a production branch).

Any other branch is a **copy** of the master branch (as it was at a point in time).

New Branches are for bug fixes and feature work separate from the master branch. When changes are ready, they can be merged into the master branch. If you make changes to the master branch while working on a new branch, these updates can be pulled in.



Commits

At GitHub, changes are called commits.

Each commit (change) has a description explaining why a change was made.



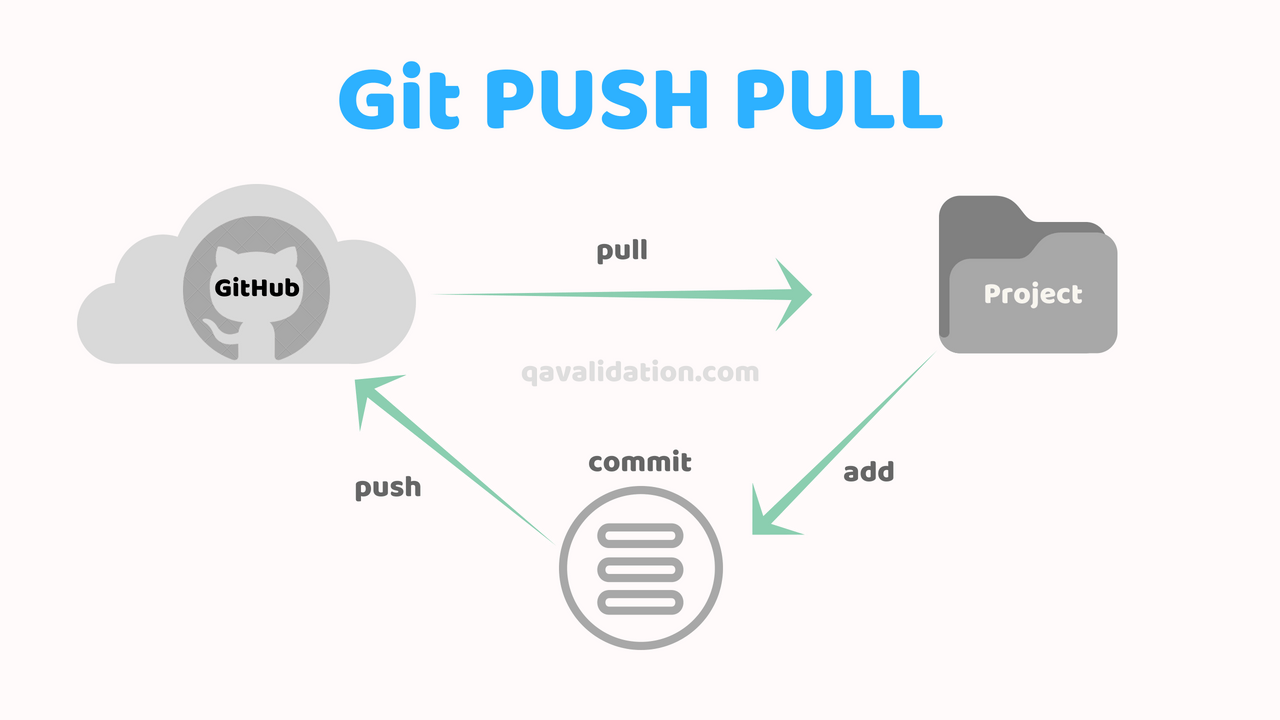
Pull Requests

Pull Requests are the heart of GitHub **collaboration**.

With a pull request you are **proposing** that your changes should be **merged** (pulled in) with the master.

Pull requests show content **differences**, changes, additions, and subtractions in **colors** (green and red).

As soon as you have a commit, you can open a pull request and start a discussion, even before the code is finished.

  
What is GitHub pull request?

Pull request is a process for a developer to notify team members that they have completed a feature. Once their feature branch is ready, the developer files a pull request via their remote server account. Pull request announces all the team members that they need to review the code and merge it into the master branch.

GitHub fork

A fork is a rough copy of a repository. Forking a repository allows you to freely test and debug the 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.

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.

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 a fork.