What is GitHub?

GitHub is a code hosting platform for version control and collaboration. It lets anyone work together on projects from anywhere.It offers all of the [distributed version control](https://en.wikipedia.org/wiki/Distributed_version_control) and [source code management](https://en.wikipedia.org/wiki/Source_code_management) (SCM) functionality of Git as well as adding its own features. It provides [access control](https://en.wikipedia.org/wiki/Access_control) and several collaboration features such as [bug tracking](https://en.wikipedia.org/wiki/Bug_tracking_system), [feature requests](https://en.wikipedia.org/wiki/Software_feature), [task management](https://en.wikipedia.org/wiki/Task_management), and [wikis](https://en.wikipedia.org/wiki/Wiki) for every project.

When was it created?

2008

Why was it created?

It was created as a mechanism to securely share private code**.** It was created as a way to work on code in a distributed way**.**And it was also created by viewing the need of having one central location that allows people to collaborate more easily on, well, almost anything.

By who?

[Tom Preston-Werner](https://www.google.com/search?rlz=1C1CHBF_enUS717US717&q=Tom+Preston-Werner&stick=H4sIAAAAAAAAAOPgE-LSz9U3KKosKMhNVwKzM3IrCg1StNSzk630k0qLM_NSi4vhjPj8gtSixJLM_DyrtPzSvJTUIgC4xZOkQwAAAA&sa=X&ved=0ahUKEwjExqLOrLXWAhWi34MKHZSOBIIQmxMIvwEoATAY)

What similar platforms exist?

-SourceForge

-Bitbucket

-Gitlab

Why would you use such a platform?

### Reason 1: A ton of integration options

GitHub can integrate with common platforms such as Amazon and Google Cloud, services such as Code Climate to track your feedback, and can highlight syntax in over 200 different programming languages.

### Reason 2: Have your code reviewed by the community

Fortunately for you, if you post your project on GitHub, the wider community of programmers and hobbyists can download and, as a result, evaluate your work. This means they can give you a heads-up on possible issues such as conflicts or unforeseen dependency issues, etc.

### Reason 3: Collaborate and track changes in your code across versions

Much like using Microsoft Word or Google Drive, you can have a version history of your code so that previous versions are not lost with every iteration.

TERMS:-

* Repository -

A repository is simply a place where the history of our work is stored. It often lives in a .git subdirectory of our working copy - a copy of the most recent state of the files we are working on.

* Commit-

A commit, or "revision", is an individual change to a file (or set of files). It is like when we save a file, except with Git, every time we save it creates a unique ID (a.k.a. the "SHA" or "hash") that allows us to keep record of what changes were made when and by who.

* Push-

Pushing refers to sending our committed changes to a remote repository, such as a repository hosted on GitHub. For instance, if we change something locally, we would want to then push those changes so that others may access them.

* Branch-

Within a repository there are branches, which are effectively forks within our own repository. Our branches will have an ancestor commit in our repository, and will diverge from that commit with our changes. We can later merge our branch changes. Branches let us work on multiple disparate features at once.

* Fork-

A fork is a copy of a repository. Forking a repository allows us to freely experiment with changes without affecting the original project. Most commonly, forksare used to either propose changes to someone else's project or to use someone else's project as a starting point for our own idea.

* Merge-

Merging takes the changes from one branch (in the same repository or from a fork), and applies them into another. A merge can be done automatically via a pull request via the GitHub web interface if there are no conflicting changes, or can always be done via the command line.

* Clone-

Cloning a git repository means that we create a local copy of the code provided by developer.Cloning a repository means that we are downloading a copy of the source code from source control.

* Pull-

A pull grabs any changes from the GitHubrepository and merges them into our local repository.

* Pull request-

Pull requests lets us tell others about changes we have pushed to a GitHub repository. Once a pull request is sent, interested parties can review the set of changes, discuss potential modifications, and even push follow-up commits if necessary.