Question:

What is GitHub? When was it created? Why? By who? What similar platforms exist? Why would you use such a platform? (Answer between 5 and 10 lines)

Answer:

GitHub is a web-based [Git](https://en.wikipedia.org/wiki/Git" \o "Git) [repository](https://en.wikipedia.org/wiki/Repository_(version_control)" \o "Repository (version control)) [hosting service](https://en.wikipedia.org/wiki/Internet_hosting_service" \o "Internet hosting service). It offers all of the [distributed version control](https://en.wikipedia.org/wiki/Distributed_version_control" \o "Distributed version control) and [source code management](https://en.wikipedia.org/wiki/Source_code_management" \o "Source code management) functionality of [Git](https://en.wikipedia.org/wiki/Git" \o "Git) as well as adding its own features. It was created (founded) on February 8, 2008 by [Tom Preston-Werner](https://en.wikipedia.org/wiki/Tom_Preston-Werner" \o "Tom Preston-Werner), Chris Wanstrath, PJ Hyett. We would use GitHub because it offers both plans for private [repositories](https://en.wikipedia.org/wiki/Repository_(version_control)" \o "Repository (version control)), and free accounts[[4]](https://en.wikipedia.org/wiki/GitHub" \l "cite_note-4) which are commonly used to host [open-source](https://en.wikipedia.org/wiki/Open-source" \o "Open-source) software projects as well as several collaboration features such as [bug tracking](https://en.wikipedia.org/wiki/Bug_tracking_system" \o "Bug tracking system), [feature requests](https://en.wikipedia.org/wiki/Software_feature" \o "Software feature), [task management](https://en.wikipedia.org/wiki/Task_management" \o "Task management), and [wikis](https://en.wikipedia.org/wiki/Wiki" \o "Wiki) for every project.

Excercise:

Press enter to submit commands

> git int

> git init

$ git status

$

$ git status

$ git add octocat.txt

$ git status

$ git commit -m "Add cute octocat story"

$ git add '\*.txt'

$ git commit -m 'Add all the octocat txt files'

$ git log

$ git remote add origin https://github.com/try-git/try\_git.git

$ git push -u origin master

$ git pull origin master

$ git diff HEAD

\

$ git add octofamily/octodog.txt

$ git diff --staged

$ git reset octofamily/octodog.txt

$ git checkout -- octocat.txt

$ git branch clean\_up

$ git checkout clean\_up

$ git rm '\*.txt'

$ git commit -m "Remove all the cats"

$ git checkout master

$ git merge clean\_up

$ git branch -d clean\_up

$ git push

>

Define

* Repository

A repository is simply a place where the history of your work is stored. It often lives in a .git subdirectory of your working copy - a copy of the most recent state of the files you're working on.

* Commit

Basically git commit "records changes to the repository" while git push "updates remote refs along with associated objects". So the first one is used in connection with your local repository, while the latter one is used to interact with a remote repository.

* Push

Use git push to push commits made on your local branch to a remote repository. The git push command takes two arguments: A remote name, for example, origin. A branch name, for example, master.

* Branch

A branch is a parallel version of a repository. It is contained within the repository, but does not affect the primary or master branch allowing you to work freely without disrupting the "live" version.

* Fork

To fork a project (take the source from someone's repository at certain point in time, and apply your own diverging changes to it), you would clone the remote repository to create a copy of it, then do your own work in your local repository and commit changes.

* Merge

Merging takes the changes from one branch (in the same repository or from a fork), and applies them into another. This often happens as a Pull Request (which can be thought of as a request to merge), or via the command line.

* Clone

Cloning a git repository means that you create a local copy of the code provided by developer. You can simply do it with a command line: git clone

* Pull

Pull refers to when you are fetching in changes and merging them. For instance, if someone has edited the remote file you're both working on, you'll want to pull in those changes to your local copy so that it's up to date.

* Pull request

Pull requests are proposed changes to a repository submitted by a user and accepted or rejected by a repository's collaborators. Like issues, pull requests each have their own discussion forum.