**Part 3:**

Answer the following questions:

**What is GitHub?**

GitHub is an open source code management system. It offers distributed version-control platform and centralized, cloud-based location where users can store, share, publish, test and collaborate on project development.

**When was it created?**

It was created in 2008.

**Why?**

GitHub lowers the barriers to collaboration by allowing granular recording of changes and programmers. At the heart of GitHub is Git. It revolutionized version control—crucial when you have more than one developer working on a project. GitHub has added a staging platform that lets developers share prototypes, collaborate on projects, edit and build upon each others’ code, and give team member a chance to take ongoing work on a test run.

**By who?**

Linus Torvalds created it.

**What similar platforms exist?**

Bitbucket, SourceForge, GitLab, Cloud Source, Beanstalk

**Why would you use such a platform? (Answer between 5 and 10 lines)**

Such alternative platforms are useful for people who have particular requirements that GitHub doesn’t meet. For example, some teams require security of an in-house system or prefer better pricing options. In addition, there are version control languages other than Git, which are not supported by GitHub.

**Part 4:**

Go through the Git tutorial here: <https://try.github.io>. While doing the tutorial, save your work on the file.

Press enter to submit commands

> 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

>

**Part 5:**

Define the following terms in the context of Git (2 lines maximum):

**Repository** - A repository, or a repo, is like a folder that contains all the files for a specific project. Each repo has a unique URL.

s**Commit** - A commit is a change to a file or set of files. Each time a commit is made, it is given a unique ID that allows the user to keep track of what changes have been made.

**Push** - Pushing sends the recent commit history from the local repository to GitHub.

**Branch** - A branch in Git is a movable pointer to a particular commit. The default branch is the master branch. With every commit, the pointer moves forward automatically.

**Fork** - Forking is creating a new project based off of another existing project. This is helpful if you want to contribute to a project.

**Merge** - Merging combines multiple commits into one unified history. Merging is often used to combine two branches.

**Clone** - Cloning allows the user to create a local copy of a remote repository. So, it creates a copy of a repository stored in GitHub on the user’s computer.

**Pull** - Pulling grabs any changes from the Github repository and merges them into your local repository.

**Pull request** - A pull request is used to notify others about changes you push to a repository on GitHub. Others can review and even push follow-up commits to the pull request.