**Part 3**

**GitHub:**

GitHub is a web-based hosting service for projects that use the Git revision control system. It is written in Ruby by Tom Preston-Werner, Chris Wanstrath and PJ Hyett. Unlike Git, which is strictly a command-line tool, GitHub provides a web-based graphical interface and desktop as well as mobile integration.

**Date of creation:**

Development of GitHub started on October 1, 2007 and within 7 months writers launched it in April 2008.

**Similar Platforms:**

SourceForge, Bitbucket, Redmine, GitLab, Launchpad and many more.

**Why to use GitHub :**

I use it because I found many interesting things about GitHub like developers can store and share their projects very easily. Moreover developers can store others repository and can push and pull their changes to and from that repository. GitHub takes this a step further by encouraging developers to fork a project's repository and then use that as their own centralized repository. Github’s help section and guide is intresting with diagrams and videos.

**Part 4**

**> git init** (initialize a gitrepository)

**$ git status** (To check the current status of repository)

**$ git status** (To check the current status of repository)

**$ git add octocat.txt** (to add file to the staging area)

**$ git status** (To check the current status of repository)

**$ git commit -m "Add cute octocat story"** (to store staged changes and m is for message)

**$ git add '\*.txt'** (to add more than one file into the directory)

**$ git commit -m "Add all the cotocat txt files"** (to store staged changes and m is for message)

**$ git log** (to show all the changes made in the file orderly)

**$ git remote add origin** [**https://github.com/try-git/try\_git.git**](https://github.com/try-git/try_git.git) (to add file into the GitHub server)

**$ git push -u origin master** (to push local changes to the repository in GitHub)

**$ git pull origin master** (to pull changes made by others)

**$ git diff HEAD** (to see what is different than last commit)

**$ git add octofamily/octodog.txt** (to add file to the staging area)

**$ git diff –staged** (to see changes just staged)

**$ git reset octofamily/octodog.txt** (to un-stage the files)

**$ git checkout -- octocat.txt** (to change files how they were before last commit)

**$ git branch clean\_up** (to create branch named clean\_up)

**$ git checkout clean\_up** (to switch to branch clean\_up)

**$ git rm '\*.txt'** (to remove all text files from clean\_up branch)

**$ git commit -m "Remove all the cats"** (to commit after changes)

**$ git checkout master** (to switch to branch named master)

**$ git merge clean\_up** (merger clean\_up branch with master branch)

**$ git branch -d clean\_up** (to delete clean\_up branch)

**$ git push** (to push everything to remote repository at the end)

**>**

**Part 5**

Define the following terms:

* **Commit:** is used to take the snapshot of your repository after making any changes in repository.
* **Push:** is used to display the commits, online on the GitHub.
* **Branch:** is used to make new branch of your own in which you can make any type of changes.
* **Fork:** is a copy of repository which allows you to make changes without affecting the original project.
* **Merge:** is used to merge new branch with master branch after the work done on new branch.
* **Clone:** means you download whole code of repository.
* **Pull:** is used when you are working on local computer and want latest up-to-date repository.
* **Pull request:** is used when you want to make changes to other’s repository.

**Part 7**

1. Click on README.md file and fork it by clicking on Fork button.
2. Copy HTTPS clone URL from GitHub.
3. On Terminal write following commands:
4. **git clone** [**https://github.com/krinapatel/courses.git**](https://github.com/krinapatel/courses.git)**.**

(Clone from the URL)

1. **cd courses**

(Get into courses)

1. **ls**

(Gives list of all the files in courses)

1. Open README.md file in text editor and edit it.
2. On terminal write following commands:
3. **git status**
4. **git add README.md**
5. **git commit –m “Any message”**
6. **git status**
7. **git log**
8. **git push origin master**

6. On GitHub website do Pull Request

1. **git remote –v**
2. **git remote add upstream** [**https://github.com/krinapatel/courses.git**](https://github.com/krinapatel/courses.git)**.**
3. **git remote –v**
4. **git fetch upstream**
5. **git merge upstream/master**
6. **git push origin master**