**PART 3**

**Question 1:** What is GitHub?

**Answer:** Git is a free and open source distributed version control system designed to handle everything from small to very large projects with speed and efficiency.

It is also known as social media Geeks/ Programmers.

**Question 2:** When was GitHub created?

**Answer:** October 2007.

**Question 3:** Why GitHub was created?

**Answer:** In 2007 they began working on GitHub as a side project.

**Question 4:** Who created GitHub?

**Answer:** 1) Chris Wanstrath

2) PJ Hyett

3) Tom Preston-Werner

**Question 5:** What similar platforms exist?

**Answer:** Subversion, CVS, Perforce, and ClearCase

**Question 6:** Why would you use such a platform?

**Answer:** GitHub is a social media for Programmers. Every programmer is on GitHub a share his/her work on GitHub. We can look at what they are working on and can easily learn, use or modify their code and can make suggestions or changes.

**PART 4**

Syntax 1: >git init

O/p: Initialized empty Git repository in /.git/

Syntax 2: git status (GIVES CURRENT STATUS)

o/p: # On branch master

# Initial commit

nothing to commit (create/copy files and use "git add" to track)

Syntax 3: git add FileName (ADDS FILE TO REPOSITORY)

Syntax 4: git commit -m "Message/Info about changes"

Syntax 5: git add '\*.txt' (add all files of the same type)

Syntax 6: git log (It give log of all the changes committed so far, in the order of commit)

Syntax 7: git remote add origin https://github.com/targit\_repository. (Useto push/move local repository to repository at GitHub server)

Syntax 8: git push -u origin master (push local changes to our origin repo)

Syntax 9: git pull origin master (Use to pull down any new changes to the code)

Syntax 10: git diff HEAD (Display the difference beterr last commited file to new commited file)

Syntax 11: git diff --staged (is use for looking at changes within files that have already been staged/complited.)

Syntax 12: git reset location/filename (It is been use to reset/undo previous changes)

Syntax 13: git checkout -- location/filename (Files will be changes to previous changes before commiting )

Syntax 14: git branch Branch\_Name (Use to create a new branch)

Syntax 15: git checkout <Branch\_Name> (use for switching between branches using the git checkout)

Syntax 16: git rm '\*.txt'(Use for removing all files)

Syntax 17: git merge <Branch\_Name> (use to merge changes)

Syntax 18: git branch -d <Branch\_Name> (Use for deleting branch or files)

**PART 5**

* Repository : Repository is a kind of folder/directory in which we can store our project.
* Commit : Use to put final changes to repository.
* Push : Push is use to send committed changes to a remote repository
* Branch : It is a similar to a repository
* Fork : It is a copy of repository which allow to freely change without affecting the project.
* Merge : Use for joining two or more development history together.
* Clone : It is a copy of a repository that stays on computer instead on server.
* Pull : It is used to fetch from/and integreate with other repositories or local branches.
* Pull request : It allows you to tell others about the changes that have pushed into GitHub repository.

Step

**PART 7**

Steps for creating a new repository and adding files to it:

1. Git init – Git repository was initialized
2. Git clone – Created the copy of repository
3. Git add . – file is added to repository by sending file from local directory to staging phase
4. Git commit -m “Comment” – saved the changes to the depositories
5. Git push origin master - it us use to push all the files into repositories.
6. Git status – Gives the status of the repository.