Q1- What is GitHub?

Ans:- GitHub is a web-based Git or version control repository and Internet hosting service. It offers all of the distributed version control and source code management (SCM) functionality of Git as well as adding its own features. It provides access control and several collaboration features such as bug tracking, feature requests, task management, and wikis for every project.

Q2- When was it created?

Ans:- The development of GitHub platform began on October 2007 and site was launched in April 2008.

Q3- By who?

Ans:- Tom Preston-Werner, Chris Wanstrath, and PJ Hyett

Q4- What similar platforms exist?

Ans:- Bitbucket, SourceForge, GitLab, Kiln, Codeplane, CodePlex, Beanstalk

Q5- Why would you use such a platform?

Ans:- To collaborate with the developers worldwide, to save my work and share my code and remotely be part of a team with whom i could create different projects and also allow others to learn and reuse it to upgrade or enhance the performance of the softwares to be built with the codes shared by me.

Q-Define the following terms in the context of Git

* Repository

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

* Commit

A commit, or "revision", is an individual change to a file (or set of files). It's like when you save a file, except with Git, every time you save it creates a unique ID (a.k.a. the "SHA" or "hash") that allows you to keep record of what changes were made when and by who. Commits usually contain a commit message which is a brief description of what changes were made.

* Push

Pushing refers to sending your committed changes to a remote repository such as GitHub.com. For instance, if you change something locally, you'd want to then push those changes so that others may access them.

* 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. When you've made the changes you want to make, you can merge your branch back into the master branch to publish your changes.

* Fork

A fork is a personal copy of another user's repository that lives on your account. Forks allow you to freely make changes to a project without affecting the original. Forks remain attached to the original, allowing you to submit a pull request to the original's author to update with your changes. You can also keep your fork up to date by pulling in updates from the original.

* 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. A merge can be done automatically via a Pull Request via the GitHub.com web interface if there are no conflicting changes, or can always be done via the command line. For more information, see "Merging a pull request."

* Clone

A clone is a copy of a repository that lives on your computer instead of on a website's server somewhere, or the act of making that copy. With your clone you can edit the files in your preferred editor and use Git to keep track of your changes without having to be online. It is, however, connected to the remote version so that changes can be synced between the two. You can push your local changes to the remote to keep them synced when you're online.

* 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. For more information, see "[About pull requests](https://help.github.com/articles/about-pull-requests)".

* Screen Shots of the Git Tutorial

































