Q1) What is GitHub? When was it created? Why? By who? What similar platforms exist? Why would you use such a platform?

Answer 1) GitHub is a code-hosting platform for version control and collaboration. It lets you and others work together on projects from anywhere. GitHub is a web based git or version control repository and internet hosting service. It is mostly used for code.it offers more of the distributed version control and source code management functionality. It provides access control and several collaboration features such as bug tracking, feature request, task management for every project. IT is mostly used to host open source projects.

Tom Preston Werner, Chris Werner and PJ Hyett created GitHub in April 2008. Ventures on GitHub can be gotten to and controlled utilizing the standard Git charge line interface and the greater part of the standard Git summons work with it. GitHub additionally permits registered and non-enrolled clients to peruse open stores on the site. GitHub and other outsiders that incorporate with the stage have additionally made various desktop customers and Git modules. A client must make a record keeping in mind the end goal to contribute substance to the website; yet open storehouses can be perused and downloaded by anybody. With an enrolled client account, clients can talk about, oversee, make stores, submit commitments to others' storehouses, and audit changes to code.

Similar platforms that exist are Bit Bucket, Source Forge, Git Lab, Kiln, Code plane and Codeplex

GitHub is a Git vault facilitating administration; however, it includes its very own significant number highlights. While Git is a summon line instrument, GitHub gives a Web-based graphical interface. It likewise gives get to control and a few coordinated effort highlights, for example, a wikis and fundamental undertaking administration apparatuses for each task.

Answer these questions in a Word file called *LastnameFirstnameGitTutorial-mm-dd-yyyy.docx*. Please respect the naming conventions!

Q2) Define the following terms in the context of Git (2 lines maximum):

Answer 2)

* **Repository -** It is typically used to sort out a solitary task. It contain envelopes and documents, pictures, recordings, spreadsheets, and informational indexes anything that our undertaking needs. We suggest including a README, or a record with data about your undertaking. GitHub makes it simple to include one in the meantime you make your new archive. It likewise offers other normal alternatives, for example, a permit record Commit
* **Push -** Fundamentally Git confer "records changes to the storehouse" while Git push "refreshes remote refs alongside related articles". So the first is utilized as a part of association with your nearby archive, while the last one is utilized to interface with a remote storehouse.
* **Branch -** **Expanding is the best approach to take a shot at various adaptations of a store at one time. When you make a branch off the ace branch, you are making a duplicate, or preview, of ace as it was by then. In the event that another person rolled out improvements to the ace branch while you were chipping away at your branch, you could pull in those updates. As a matter of course, your store has one branch named ace, which is thought to be the authoritative branch.**
* **Fork -** A fork is a duplicate of a store. Forking a storehouse enables you to uninhibitedly try different things with changes without influencing the first task. Most usually, forks are utilized either to propose changes to another person's task or to utilize another person's undertaking as a beginning stage for your own thought.
* **Merge -** Consolidating takes the progressions from one branch (in a similar vault or from a fork), and applies them into another. This frequently occurs as a force ask for (which can be thought of as a demand to combine), or by means of the summon line. A consolidation should be possible naturally by means of a draw ask for by means of the GitHub web interface if there are no clashing changes, or should dependably be possible through the charge line.
* **Clone -** clone is a duplicate of a store that lives on your PC rather than on a site's server some place, or the demonstration of making duplicate. With your clone, you can alter the documents in your favored proofreader and utilize Git to monitor your progressions without being on the web. It is, nevertheless, associated with the remote form so changes can be matched up between the two.
* **Pull -** Pull refers to when you are fetching in changes and merging them. For instance, if someone has edited the remote file you are both working on, you will want to pull in those changes to your local copy so that it is 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