**Exercise on Git and GitHub**

**Part 1:**

Submit your GitHub username in the form here:

**User Name:** deshmukh06

**Part 2:**

GitBash is already installed.

**Part 3:**

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

**ANSWER:** GitHub is a provider of [Internet hosting](https://en.wikipedia.org/wiki/Internet_hosting_service) for [software development](https://en.wikipedia.org/wiki/Software_development) and [version control](https://en.wikipedia.org/wiki/Version_control) using [Git](https://en.wikipedia.org/wiki/Git). It offers the [distributed version control](https://en.wikipedia.org/wiki/Distributed_version_control) and [source code management](https://en.wikipedia.org/wiki/Source_code_management) (SCM) functionality of Git, plus its own features. GitHub was created in February 8, 2008; 13 years ago originally as Logical Awesome LLC founded by Tom Preston-Werner, [Chris Wanstrath](https://en.wikipedia.org/wiki/Chris_Wanstrath), [P. J. Hyett](https://en.wikipedia.org/wiki/P._J._Hyett), Scott Chacon and then has been a subsidiary of Microsoft from 2018. There are other such platforms available like CloudForge and BitBucket. But,GitHub provides much more features for version controlling in a Project and thus GitHub is more powerful tool to use

The Word file should be called *LastnameFirstnameGitTutorial-mm-dd-yyyy.docx*. Please respect the naming conventions!

**Part 4:**

Answer the questions below in the *LastnameFirstnameGitTutorial-mm-dd-yyyy.docx* file.

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

* **Repository** – A place on a server where software packages are stored. With GitHub, typically (but not always), you have a repository for each project on your GitHub account.
* **Commit** – This command takes what changes you made and saves them locally; you can add a note with it for future reference. Once you commit you can go back to that point.
* **Push** – Sends changes that have been made locally on a machine up to the remote server, in our case GitHub.
* **Branch** – A branch is when Git makes a new copy of what you are working on, so you can do stuff to it without it changing the main version, within the same repository.
* **Fork** – When you make a copy of a project as the start of a new project. Practically, a fork may be made so you can add something to an open source project before sending a pull request.
* **Merge** – In our context, it’s a command to put together two branches into one branch.
* **Clone** – Copies the data from a remote server to your local git environment. If you’re starting work on an existing repository that isn’t on your local PC yet, you’ll use clone.
* **Pull** – Similar to clone, but for getting your local git environment up to date with whatever is on the server. If you’ve got multiple people working on something you use this a lot.
* **Pull request -** A GitHub feature that allows you to submit your work/changes to a project for the admins to accept or deny.

**Part 6:**

Commands and strategy that I used to do this part of the exercise in the *LastnameFirstnameGitTutorial-mm-dd-yyyy.docx* file and push it to YOUR repository.

1. Figure out about local repositories and remote repositories so that I can push my code through instructed branch.
2. While updating, my branch name was randomly changed with a number as prefix, so I needed to figure out that issue too.
3. Some Git commands didn’t work correctly, so I then researched all the commands using different arguments that suited my repo.