Assignment 2.1

1) Explain the key difference between Git and Github.

→ Git is a version control system that allows developers to track changes in their changes in their code. Github is a web-based hosting service for git repositories. In simple terms, you can use git without github, but you cannot use github without git.

Git	Github
- Used for version control	- Used for hosting git repositories
- Installed locally on computer	- Cloud-based services
- Tracks changes made to a file	- Provides a web interface to view file changes.
- You can use git without github.	- You cannot use github without git

2) Describe to core concepts of Git: repository, working directory, staging, area and commit.

→ Repository : A git repository stores a

all the versions of files within
a project, enabling developers to
track changes, collaborate, and
easily revert to previous versions
if needed.

- Working directory : The git directory
is where git stores the metadata
and object database for your project.
This is the most important part of
git and it is what is copied when
you clone a repository from another
computer.

- Staging area : the middle ground
between what you have done to
your files (also known as the
working directory) and what you
had last committed (the HEAD commit).
As the name impiles, the staging area
gives you space to prepare the changes
that will be reflected on the next commit.

- Commit : the git commit command
is one of the core primary function
of git. Prior use of the git add
command is required to select
the changes that will be staged
for the next commit.

3) what are different types of Github repositories (public, privates)?

→ public repositories are accessible to everyone on the internet.

— private repositories are only accessible to you, people you explicitly share access with, and, for organization repositories, certain Organization members.

4) Describe the steps involved in pushing your local changes to a remote repository on github.

→ 1. Creating a new repository

2. open your Git Bash

3. create your local project in your desktop directed towards a current working directory.

4. Initilalize the git repository
   — Use git init

5. Add the file to the new local repository
   - git add.

6. commit the files staged in your local repository by writing a commit message.
   - git commit -m 'your message'

7. Copy your remote repository's URL from Github

8. Add the URL copied, which is your remote repository to where your local content from your repository is pushed
     - git remote add origin 'link'

9. Push the code in your local repository to Github
     - git push -u origin master

10. View your files in your repository hosted on Github.