**Task 5:**

**Git & GitHub Tutorial with Commands**

**Introduction:-**

*Q1. What is Git?*

**Git:** Git is a distributed version control system to track changes in source code during software development. It coordinates work among programmers and can track changes in any set of files. Its primary goals include speed, data integrity, and support for distributed, non-linear workflows.

*Q2. What is GitHub?*

**GitHub:** GitHub is a web-based Git repository hosting service, which offers distributed revision control and source code management functionality of Git , it also adds it’s own function.

***Q3.****State difference between Git and GitHub.*

| **S.No.** | **Git** | **GitHub** |
| --- | --- | --- |
| 1. | Git is a software. | GitHub is a service. |
| 2. | Git is a command-line tool | GitHub is a graphical user interface |
| 3. | Git is installed locally on the system | GitHub is hosted on the web |
| 4. | Git is maintained by linux. | GitHub is maintained by Microsoft. |

***\* To Install :-***

Download Git from: <https://git-scm.com/downloads>

To verify installation: git –version

**Commands:-**

1. ***git init***

To initialize a new Git repository in your project folder.



1. ***git status***

To show the current state of the working directory and staging area.

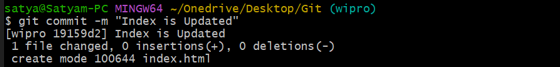
1. ***git add <file>***

To add a specific file to the staging area.



1. ***git commit -m "message"***

To save the staged changes with a message in return.



1. ***git branch***

To list all local branches.



1. ***git checkout <branch\_name>***

To switch a different branch.



1. ***git log***

To show the commit history.



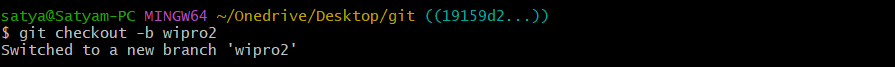
1. ***git branch <branch\_name>***

To create a new branch.



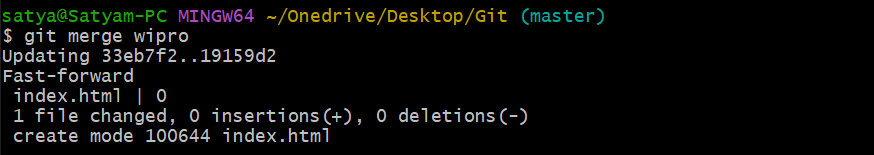
1. ***git checkout -b <branch\_name>***

To create and switch to a new branch.



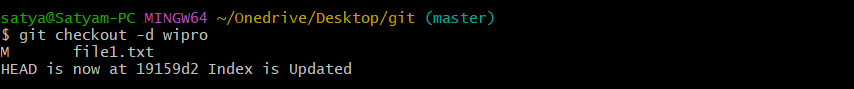
1. ***git merge <branch\_name>***

To merge changes from the specified branch into the current branch.



1. ***git branch -d <branch\_name>***

To delete a branch.

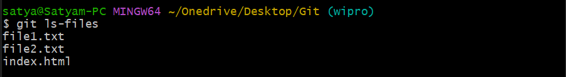


1. ***git rm <file\_name>***

To delete the file from your working directory.

1. ***git ls-files***

Exist in the index/staging area.



1. ***touch filename.txt***

To create filename.txt if it doesn't exist.

If it already exists, then it will update the last modified timestamp (without changing content).

1. ***echo "Hello, Wipro!"***

Prints the message to the terminal.



1. ***echo "This is a test" > file.txt***

Writes "This is a test" into file.txt, overwrites if it already exists.



1. ***echo "Another line" >> file.txt***

Appends "Another line" to the existing file content.

