DAYWISE PRESENTATION EXERCISE

TOPIC 2:

**GIT AND GIT COMMANDS**

Git is a popular version control system. It was created by Linus Torvalds in 2005, and has been maintained by Junio Hamano since then.

It is used for:

* Tracking code changes
* Tracking who made changes
* Coding collaboration

### What does Git do?

* Manage projects with ****Repositories****
* ****Clone**** a project to work on a local copy
* Control and track changes with ****Staging**** and ****Committing****
* ****Branch**** and ****Merge**** to allow for work on different parts and versions of a project
* ****Pull**** the latest version of the project to a local copy
* ****Push**** local updates to the main project

### Working with Git

* Initialize Git on a folder, making it a ****Repository****
* Git now creates a hidden folder to keep track of changes in that folder
* When a file is changed, added or deleted, it is considered ****modified****
* You select the modified files you want to ****Stage****
* The ****Staged**** files are ****Committed****, which prompts Git to store a ****permanent**** snapshot of the files
* Git allows you to see the full history of every commit.
* You can revert back to any previous commit.
* Git does not store a separate copy of every file in every commit, but keeps track of changes made in each commit!

### Why Git?

* Over 70% of developers use Git!
* Developers can work together from anywhere in the world.
* Developers can see the full history of the project.
* Developers can revert to earlier versions of a project.

### What is GitHub?

* Git is not the same as GitHub.
* GitHub makes tools that use Git.
* GitHub is the largest host of source code in the world, and has been owned by Microsoft since 2018.
* In this tutorial, we will focus on using Git with GitHub.

## Git Install

You can download Git for free from the following website: [https://www.git-scm.com/](https://git-scm.com/" \t "https://www.w3schools.com/git/_blank)

## Using Git with Command Line

To start using Git, we are first going to open up our Command shell.

For Windows, you can use Git bash, which comes included in Git for Windows. For Mac and Linux you can use the built-in terminal.

There are many different ways to use Git. Git supports many command-line tools and graphical user interfaces. The Git command line is the only place where you can run all the Git commands.

The following set of commands will help you understand how to use Git via the command line.

## Basic Git Commands

Here is a list of most essential Git commands that are used daily.

1. [Git Config command](https://www.javatpoint.com/git-commands" \l "config-command)
2. [Git init command](https://www.javatpoint.com/git-commands" \l "init-command)
3. [Git clone command](https://www.javatpoint.com/git-commands" \l "clone-command)
4. [Git add command](https://www.javatpoint.com/git-commands" \l "add-command)
5. [Git commit command](https://www.javatpoint.com/git-commands" \l "commit-command)
6. [Git status command](https://www.javatpoint.com/git-commands" \l "status-command)
7. [Git push Command](https://www.javatpoint.com/git-commands" \l "push-command)
8. [Git pull command](https://www.javatpoint.com/git-commands" \l "pull-command)
9. [Git Branch Command](https://www.javatpoint.com/git-commands" \l "branch-command)
10. [Git Merge Command](https://www.javatpoint.com/git-commands" \l "merge-command)
11. [Git log command](https://www.javatpoint.com/git-commands" \l "log-command)
12. [Git remote command](https://www.javatpoint.com/git-commands" \l "remote-command)

## 1) Git config command

This command configures the user. The Git config command is the first and necessary command used on the Git command line. This command sets the author name and email address to be used with your commits. Git config is also used in other scenarios

$ git config --global user.name "ImDwivedi1"

$ git config --global user.email "Himanshudubey481@gmail.com"

### 2) Git Init command

This command is used to create a local repository.

****Syntax****

$ git init Demo

### 3) Git clone command

This command is used to make a copy of a repository from an existing URL. If I want a local copy of my repository from GitHub, this command allows creating a local copy of that repository on your local directory from the repository URL.

****Syntax****

$ git clone URL

### **4) Git add command**

This command is used to add one or more files to staging (Index) area.

****Syntax****

To add one file

$ git add Filename

To add more than one file

$ git add\*

### **5) Git commit command**

Commit command is used in two scenarios. They are as follows.

****Git commit -m****

This command changes the head. It records or snapshots the file permanently in the version history with a message.

****Syntax****

$ git commit -m " Commit Message"

****Git commit -a****

### 6)Git status command

The status command is used to display the state of the working directory and the staging area. It allows you to see which changes have been staged, which haven't, and which files aren?t being tracked by Git. It does not show you any information about the committed project history. For this, you need to use the git log. It also lists the files that you've changed and those you still need to add or commit.

****Syntax****

$ git status

### 7) Git push Command

It is used to upload local repository content to a remote repository. Pushing is an act of transfer commits from your local repository to a remote repo. It's the complement to git fetch, but whereas fetching imports commits to local branches on comparatively pushing exports commits to remote branches. Remote branches are configured by using the git remote command. Pushing is capable of overwriting changes, and caution should be taken when pushing.

Git push command can be used as follows.

****Git push origin master****

### 8) Git pull command

Pull command is used to receive data from GitHub. It fetches and merges changes on the remote server to your working directory.

****Syntax****

### 9) Git Branch Command

This command lists all the branches available in the repository.

****Syntax****

$ git branch

### 10) Git Merge Command

This command is used to merge the specified branch?s history into the current branch.

****Syntax****

$ git merge BranchName

### 11) Git log Command

This command is used to check the commit history.

****Syntax****

$ git log

### 12) Git remote Command

Git Remote command is used to connect your local repository to the remote server. This command allows you to create, view, and delete connections to other repositories. These connections are more like bookmarks rather than direct links into other repositories. This command doesn't provide real-time access to repositories.