**Assignments 1 Swapnil Gaikwad**

1. What is a Version Control System (VCS) and why is it important?

Answer –

* Version Control System (VCS) / Source Code Management (SCM) is a tool that helps to store files and track the changes happening to files.
* VCS also preserve older and later versions of the code so that at any time we can switch between whichever version we want.
* VCS also keep a track of who is making what kind of changes

1. Explain the difference between Centralized VCS and Distributed VCS.

Answer-

| **Feature** | **Centralized VCS** | **Distributed VCS** |
| --- | --- | --- |
| **Repository Location** | Single central server | Every user has a full copy of the repository |
| **Dependency on Server** | Requires constant server access for most operations | Can work offline; server access needed only for pushing/pulling changes |
| **Failure Tolerance** | Single point of failure; if the server goes down, access to history is lost | No single point of failure; every user has a full copy of the history |
| **Speed** | Slower as operations often require server interaction | Faster since operations are mostly local |
| **Setup Complexity** | Simple to set up and manage | Slightly more complex to set up |
| **Collaboration** | Centralized control; all changes must go through the server | Decentralized; users can collaborate without a central server |
| **Storage Requirements** | Minimal storage on client machines | Larger storage required on each developer's machine due to full repository copies |
| **Examples** | Subversion (SVN), Perforce | Git, Mercurial |

1. What are some examples of Centralized and Distributed VCS?

Answer –

Centralized VCS – SVN

Distributed VCS – Git, Mercurial

1. What are the common operations performed by developers in a VCS?

Answer –

1. Clone: Create a local copy of the repository (specific to distributed VCS).

2. Commit: Save changes to the local or central repository with a descriptive message.

3. Push: Upload local changes to the central/shared repository.

4. Pull/Fetch: Retrieve the latest changes from the central/shared repository.

5. Branch: Create an independent line of development for new features or bug fixes.

6. Merge: Integrate changes from one branch into another.

7. Revert: Undo specific changes in the code-base.

8. Resolve Conflicts: Address issues when changes conflict during merges.

9. Log: View the history of changes in the repository.

10. Diff: Compare changes between versions of files or branches.

1. What are the key phases in the Git lifecycle workflow?

Answer-

Git lifecycle workflow phases.

1. Working directory
2. Staging area
3. Local repository
4. How do you initialize a Git repository locally?

Answer – use command- git init for initialize

1. Explain the purpose of the command git pull.

Answer –

The command git pull is used to fetch and merge changes from a remote repository into your local repository. It combines two operations: git fetch (which retrieves the latest changes from the remote) and git merge (which integrates those changes into your current branch).

1. How can you check the status of your files in a Git repository?

Answer – you can check the status of your file by using below command.

git status

1. What does the command git add . do?

Answer – git add . stages all modified and untracked files in the current directory and its subdirectories for the next commit in a Git repository.

1. Describe the process of committing changes to the local repository in Git.

Answer –

* Make changes in file like edit, delete or create a program
* Stage the file using cmd- git add filename
* Commit the changes using cmd- git commit –m “your message”
* Verify it by cmd – git log

1. How do you link a local Git repository with a remote repository?

Answer –

Open the git bash terminal in the local directory that contains project.

Use cmd- git remote add origin “remote repository url”

Verify it by cmd – git remote -v

1. How do you push code to a remote Git repository?

Answer –

One you are successfully able to link your local repository with remote repository.

Ensure you are in correct branch by cmd- git branch

If not use cmd – git checkout <branch name>

Use cmd – git add .

Git commit -m “your message”

Git push origin main

1. How to configure your Git user name and email in your machine.

Answer – to configure your Git user name and email in your machine use below cmds.

git config --global user.name "Your Name"

git config --global user.email "your.email@example.com"

1. What is the use of windows credential manager?

Answer –

Windows Credential Manager is a feature in the Windows operating system that stores and manages login credentials, such as usernames, passwords, and other authentication details. These credentials are used to access services, applications, websites, and network resources without having to re-enter them each time.