Git (GLOBAL INFORMATION TRACKER)

Git is a version control system (VCS) that helps keep track of changes in code and allows multiple people to work on the same project without confusion.

🔹 It remembers all changes, so if you make a mistake, you can go back to a previous version easily.

🔹 Git is local, meaning it works on your computer. You don't need the internet to use Git.

Web-Based Git Platforms

These platforms let you work with Git repositories online:

GitHub – Best for collaborators, open-source projects, and beginners.

GitLab – Best for companies needing full DevOps tools (CI/CD, security, automation).

Bitbucket – Best for teams using Atlassian tools (Jira, Trello) and private repositories.

GIT INSTALLATION STEPS:

Download Git for Windows:

Visit the official Git website: https://git-scm.com/

Click the "Download" button for Windows.

Run the Installer

Once the .exe file is downloaded, run it.

Choose the installation options based on your preferences:

Select Components

Adjusting your PATH environment

Choose HTTPS transport backend

Configuring the line ending conversions

Complete Installation:

Continue through the steps and click Install.

Once the installation is complete, click Finish.

Verify Git Installation:

Open Git Bash (you can find it in the Start menu).

Run the following command to check the installation:

git --v

You should see the installed version of Git.

Download Git for Mac:

Go to the official Git website: https://git-scm.com/download/mac.

Click on the "Download" button for macOS. This will download the binary installer for macOS.

Install Git:

Once the .dmg file is downloaded, open it to start the installation.

Follow the on-screen instructions to complete the installation process.

If an Alert Message Appears (during installation):

If macOS prevents you from opening the installer, it will show an alert message.

Go to System Preferences > Security & Privacy.

In the General tab, you’ll see a message that says "Git.pkg was blocked from opening because it is not from an identified developer."

Click "Open Anyway".

Now, go back and click Install to proceed.

You may be prompted to enter your Mac password to complete the installation.

Verify Installation:

Open Terminal (found in Applications > Utilities or search for it via Spotlight).

To verify Git installation, run the following command in the Terminal:

git --version

You should see the installed version of Git.

Setting Git to Use Bash (Optional):

Open Terminal Preferences:

In the Terminal, click on the Terminal menu in the top-left corner of the screen, then select Preferences.

Set Default Shell to Bash:

In the Profiles tab, you can set your default shell. Under General, click on the Default dropdown.

Choose Bash if it’s not already selected.

Start Using Git:

Once Bash is selected, you can now use Git commands in the Terminal.

Creating a Repository in github(One of the Way)

1. Sign In to GitHub

Go to GitHub and log in with your account.

2. Navigate to the Repositories Page

After logging in, click on your profile picture at the top right, and select Your repositories from the dropdown. Alternatively, you can directly go to this URL: github.com/repositories.

3. Create a New Repository

On the repositories page, click the New button (or directly go to https://github.com/new) to create a new repository.

4. Fill in the Repository Details

Repository Name: Choose a name for your repository (e.g., my-awesome-project).

Description: Optional. Add a short description of what the repository is for.

Visibility: Choose between:

Public: Anyone can see this repository.

Private: Only you and people you grant access to can see this repository.

Initialize this repository with:

Optionally, you can select to add a README file (recommended for most projects).

You can also choose a .gitignore template and select a license if needed.

5. Create Repository

Once you've filled in the details, click the Create repository button.

6. Start Adding Files

After creating the repository, you can either upload files directly via the GitHub web interface.

Deployment Process in github(to check the output)

Go to your GitHub repository.

Click on the Settings tab.

Scroll down to the Pages section (on the left sidebar).

Under Source, select the branch to deploy from (usually main or master).

Choose /root for the folder (for simple static sites).

And click on the save

Next , go to actions tab

Click on the pages build and deployment

After completing deploying, click on the link which is generate below the deploy.

Git Stages

Git helps you keep track of changes in your project, Git organizes changes into four stages:

Local/Working Directory – Where you make changes.

Staging Area – Where you prepare changes for saving.

Commit Area – Where changes are saved permanently in your local Git repository.

Remote Directory – Where changes are stored online for collaboration.

1. Local/Working Directory (Untracked or Modified)

The Local/Working Directory is where you create or edit files. Git sees your changes but does not track them yet.

2. Adding/Staging Area – Getting Ready

The Staging Area is like putting your draft in a folder before submitting it. You’re telling Git:

"These changes are important, and I want to save them."

3. Commit Area – Saved Locally

The Commit Area is where changes are saved permanently on your computer.

4. Remote Directory – Sharing Your Work Online

The Remote Directory is an online repository (like GitHub, GitLab, or Bitbucket) where you can store and share your project.

GIT COMMANDS

Process-1 Push the files from local to remote

1. git init (Initialize a git Repository)

2. git add <filename> or git add . (Stage a Specific File or all files)

3. git status (Check Repository Status)

4. git commit -m "message" (Save Changes)

5. git config --global user.name "username" (Set Global Username)

6. git config --global user.email "email@example.com" (Set Global Email)

7. git remote add origin <repo-url> (Connect to Remote Repository)

8. git checkout -M main (Switch Branch to Main)

9. git push -u origin main (Push Changes to Remote)