Basics of Git and GitHub

**Git:**

Git is software that comes pre-installed on Mac or Linux computers, while on Windows, you need to download it using Git Bash. In simple terms, Git acts as a memory card for code. With Git, you regularly enter commands to save your progress locally on your computer. For instance, if you create HTML or CSS files and haven't saved your progress, a file deletion would result in permanent loss. However, using Git allows you to save your work, including the file contents, enabling you to revert to a saved state when needed.

**Few commands:**

• initialising your folder or project to use git: git init

this is like you are putting your memory card to your game system

• create a file: touch <file\_name>

• save a file: git add <file\_name> <another\_filename>

• save everything that you have done since last time you saved: git add .

• commit the changes to memory: git commit -m

-m means to add a message and infornt of that you can describe the changes or save

• remove a file: rm <file\_name>

• ls: list of files in that directory

• number of times you saved at what time can be checked by the command: git log

• create new branch: git checkout -b <branch\_name>

• push changes to GitHub in a new-branch: git push origin <branch\_name>

• pull from GitHub: git pull origin master

• check status of changes: git status

• see your old changes: git checkout <commit\_hash>

**GitHub:**

GitHub is a website where you upload the code saved on your computer to the internet. This enables others to download your code, review it, and make contributions. Understanding GitHub is crucial for collaborative work on projects with people from different parts of the world. You push your progress to GitHub, allowing others to pull your code or simply view it on your profile. To upload your folders, create a repository (folder) on your profile and follow the provided instructions, so simple!!!

Branches in GitHub:

In GitHub, the default branch is usually named "master."However, others can download your code, make changes on a different branch, and you can later merge those changes if needed.

Create new branch: git checkout -b <name\_of\_branch>

To write into the new branch: git branch

You can merge your work with the master branch if you want to. The point of branches is that other people can download your project, make changes in a separate branch, and then say, "Hey, I have made some changes, and I didn't want to mess up what you are doing. You can go through them, and if you like them, you can merge them with what you are doing."

Push: When you have changes on your computer that GitHub doesn't have, you push them up to GitHub.

Pull: When GitHub has changes that you don't have, you pull them down from GitHub.

You need Git for GitHub, but you don't need GitHub just to save your changes, which can be done by Git. However, you probably need GitHub when you want to collaborate with other people in different places, and you might need it for almost every project.