**MIT xPRO Data Engineering Certificate**

**Git-GitHub**

Git (global information tracker) is an open-source decentralised version control system, created by Linus Torvold (also creator of Linux kernel).

Each change of code (git commit) is captured and tracked throughout the history of a project (git tree).

Some common git terms:

Repository (Repo) - A storage location for your project's code, including the complete history of changes. It can be local (on your machine) or remote (on platforms like GitHub).

Commit - A snapshot of changes in the repository. Each commit records the state of the project at a given point in time and includes a message describing the changes.

Branch - A parallel version of the repository. Branches allow you to work on different features or fixes independently. The default branch is usually called `main` or `master`.

Merge - Combining changes from different branches. Merging integrates the changes from one branch into another.

Clone - Creating a local copy of a remote repository on your machine using `git clone`.

Fork - Creating a personal copy of someone else's repository. This is commonly used in open-source projects for contributing to someone else’s project.

Remote - A reference to a remote repository, like the one hosted on GitHub. The default remote is usually named `origin`.

Pull - Fetching changes from a remote repository and merging them into your local branch using `git pull`.

Push - Uploading your local commits to a remote repository using `git push`.

Staging Area (Index) - A temporary area where changes are gathered before being committed. You add changes to the staging area using `git add`.

HEAD - A pointer that refers to the latest commit in the currently checked-out branch.

Checkout - Switching between different branches or reverting files to a previous state using `git checkout`.

Rebase - Reapplying commits on top of another base tip, which can lead to a linear project history.

Tag - A marker that points to a specific commit, often used for releases or versions.

Diff - A comparison showing the changes between two commits, branches, or between your working directory and the repository.

Fetch - Downloading changes from a remote repository without merging them into your local branch.

Origin - The default name for a remote repository that you cloned from.

Conflict - A situation that occurs when Git cannot automatically resolve differences between two commits or branches.

Blame - A Git command that shows who last modified each line of a file, helpful for tracking down the source of changes.

Revert - Creating a new commit that undoes the changes made in a previous commit.

Reset - Undoing changes by moving the `HEAD` pointer and optionally modifying the staging area and working directory.

Cherry-Pick - Applying a specific commit from one branch to another, useful for applying hotfixes.

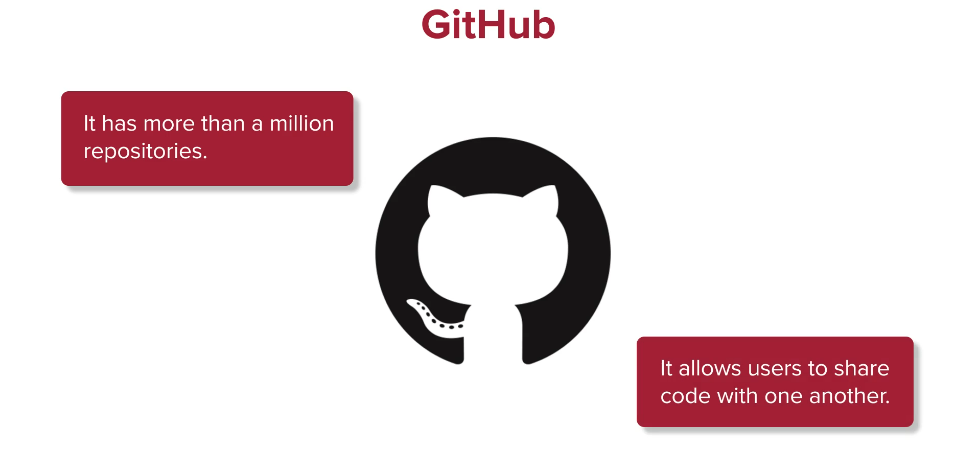
Squash - Combining multiple commits into a single commit, often used during the merge process to simplify history.

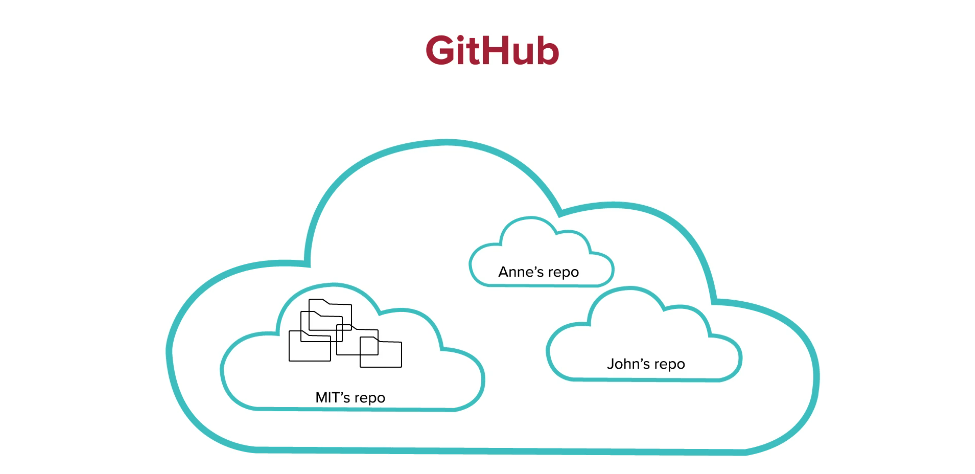
Stash - Temporarily saving changes that are not ready to be committed, allowing you to switch branches or work on something else without losing your progress.

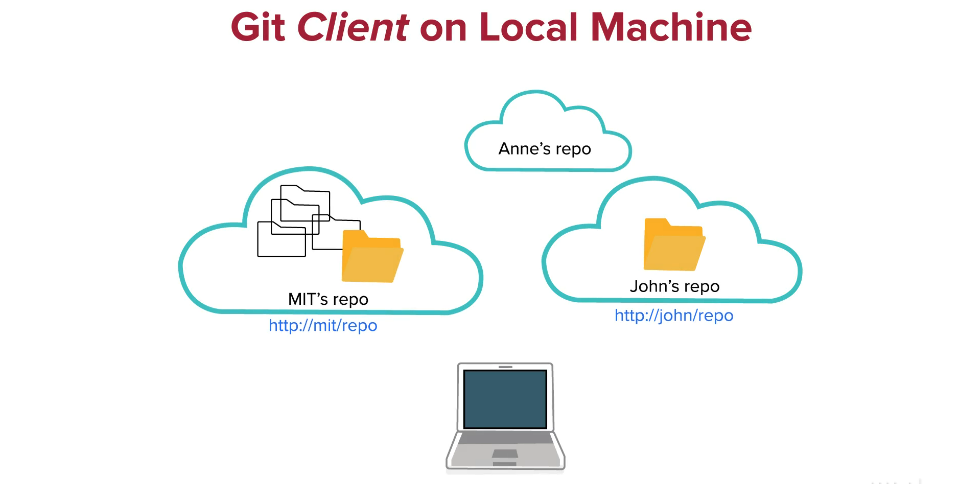
Submodule - A repository embedded inside another repository, allowing you to include external projects.

Pull Request (PR) - A request to merge changes from one branch into another, often used in collaborative workflows.

GitHub is a fully managed cloud service that hosts your version control repositories. Owned by Microsoft. Others are GitLab, BitBucket, SourceForge.







You can ‘fork’ a repo to copy it between GitHub accounts, and ‘clone’ a repo to make a copy on a local machine:

* git clone <repo> e.g. git@github.com/mhtattersall/Markdown.git (passphrase clue: football forest)

You place a GitHub agent on your local machine, which uses a private key and a public key is stored on your GitHub account:

If you update a file locally, you can commit to your local repo then push it to GitHub:

* git add . (or <filename>) add all files from cwd to staging area for next commit
* git status check if branch is up to date
* git commit -m <message> commit staged files to local repo with a message
* git push origin <branch-name> push to remote repo - ie origin main
* git log view commit logs

You can issue a pull request to see if the administrator of a repo wants your update.

