Module 3 Git & GitHub

Friday, 4 July 2025 11:17 AM

Module by Dhaanesh S

Connecting to GitHub via HTTPS

Creating and clonning using HTTPS

1. Copy the repositories HTTPS URL
2. Open terminal

   Mkdir projects
   Cd projects
   Git clone "paste the copied HTTPS url"

   Ls -l
   My first repo

   Cd first-repo

GIT WORKFLOW

3 states called Modified, Staged and Committed

1. Modified - adding, removing, modifying
2. Staged - To track a file
3. Committed - Save the file and snapshot current changes

GIT ADD AND COMMIT

- To check any changes -> git status (It shows nothing to commit and working tree is clean if everything is done)
- Add a file inside my first repo
  - Touch test.txt
- Git status -> shows untracked file
- Use git add to keep track of changes

  Git add test.txt

- To restore back to un tracked stage use restore command

  Git restore --stage test.txt

BRANCHES

- Next step is to create a new branch
- Use git checkout -B feature/lesson (This will create a new branch and moves us into the new branch)
- Git branch feature/lesson (creates only a new branch)
- Push the code

  Git push -u origin feature/lesson

  Git checkout main

  Git pull

REMOTE vs LOCAL

- There might be a centralized server to which we need to get our code from. Eg: Coding project 1 is the central code, so to work on this file the user as to either clone it to their local machine or clone them And make changes and push the code back to the central repository.

Resolving conflicts

- Conflicts generally arise when there is completing changes. Git will normally auto merge, but in case of conflict it will need some confirmation.
- To solve the developer must look at the changes on the server

Developer 1 completes their changes and pushes them to the remote repository. Later, Developer 2 also attempts to push their changes. Since both have modified Feature.js, Git detects a conflict that requires resolution before the merge can proceed.

modified feature.js, Git detects a conflict that requires resolution before the merge can proceed.

Git Diff : To compare changes across your files branches and commits

1. Cd projects -> cd my Coursera repo -> vim README.md -> git diff HEAD README.md
2. It shows the edits that we made to this file with + and -
3. To show all the history of commits, use

    Git log --pretty=oneline

GIT Forking

Let's run through a typical flow of creating a new branch and adding some new content.

Step 1: Clone the repository.

Step 2: Create a new branch.

```
git checkout -b test/forking-example
```

Step 4: Create a new file and commit it to the repository.

```
touch text.txt git add . git commit -m 'chore: testing'
```

Step 5 Push the branch to your remote repository:

```
git push -u origin test/forking-example
```

Step 6: Go to Github and click the Compare & pull request button. If it's not available, click on the branch dropdown menu and select the branch named test/forking-example: