WEEKLY ASSIGNMENT- 15 JAN 2024

(By: Joel Jacob John
Roll no: STB03-T0003)

Git:

Git is a distributed version control system designed to manage and track changes in source code during software development. Developed by Linus Torvalds in 2005, Git allows multiple developers to collaborate on a project efficiently. It is widely used across the software development industry to maintain a history of changes, track different versions of files, and enable seamless collaboration among team members.

Key Concepts:

1. Repository (Repo): A repository is a collection of files and their version history. It can be local or hosted on a remote server.

2. Commit: A commit is a snapshot of changes made to the repository. Each commit has a unique identifier and a commit message describing the changes.

3. Branch: A branch is a separate line of development. Developers can create branches to work on specific features or bug fixes without affecting the main codebase.

4. Merge: Merging combines changes from different branches. It's a way to integrate features or bug fixes into the main branch.

5. Clone: Cloning is the process of copying a repository from a remote server to a local machine. It establishes a connection between the local and remote repositories.

6. Pull and Push: Pulling retrieves changes from a remote repository, and pushing sends local changes to a remote repository.

GitHub:

GitHub is a web-based platform built on top of Git, providing a centralized location for hosting and collaborating on Git repositories. It adds a layer of features, including a user-friendly web interface, issue tracking, pull requests, and more. GitHub has become a standard platform for open-source and private software development projects, facilitating collaboration among developers worldwide.

Key Features:

1. Repository Hosting: GitHub serves as a hosting platform for Git repositories, making it easy to share code with others.

2. Collaboration: Developers can collaborate by forking repositories, creating branches, and submitting pull requests for code review.

3. Issue Tracking: GitHub includes a robust issue tracking system, enabling teams to manage and prioritize tasks, bugs, and feature requests.

4. Pull Requests: Pull requests provide a mechanism for proposing changes, discussing modifications, and merging code into the main repository.

5. Actions and Workflows: GitHub Actions automate workflows, allowing developers to run tests, deploy applications, and perform other tasks directly from the repository.

6. Community and Social Coding: GitHub fosters a collaborative and social coding environment, allowing developers to follow projects, contribute to discussions, and discover new codebases.

In summary, Git is the version control system, while GitHub is a platform built around Git, offering additional features for collaboration and project management. Together, they form a powerful combination for efficient and organized software development.

Git and GitHub Commands Documentation:

1. Git Basics:

a. Initialize a Repository:

```
git init
```

b. Clone a Repository:

```
git clone <repository-url>
```

c. Add Changes:

```
git add <file>
git add .
```

d. Commit Changes:

```
git commit -m "Commit message"
```

e. View Changes:

```
git status
git diff
```

2. Branching:

a. Create a New Branch:

```
git branch <branch-name>
```

b. Switch to a Branch:

```
git checkout <branch-name>
```

c. Merge Branches:

```
git merge <branch-name>
```

d. Delete a Branch:

```
git branch -d <branch-name>
```

3. GitHub Collaboration:

a. Push Changes to Remote:

```
git push origin <branch-name>
```

b. Pull Changes from Remote:

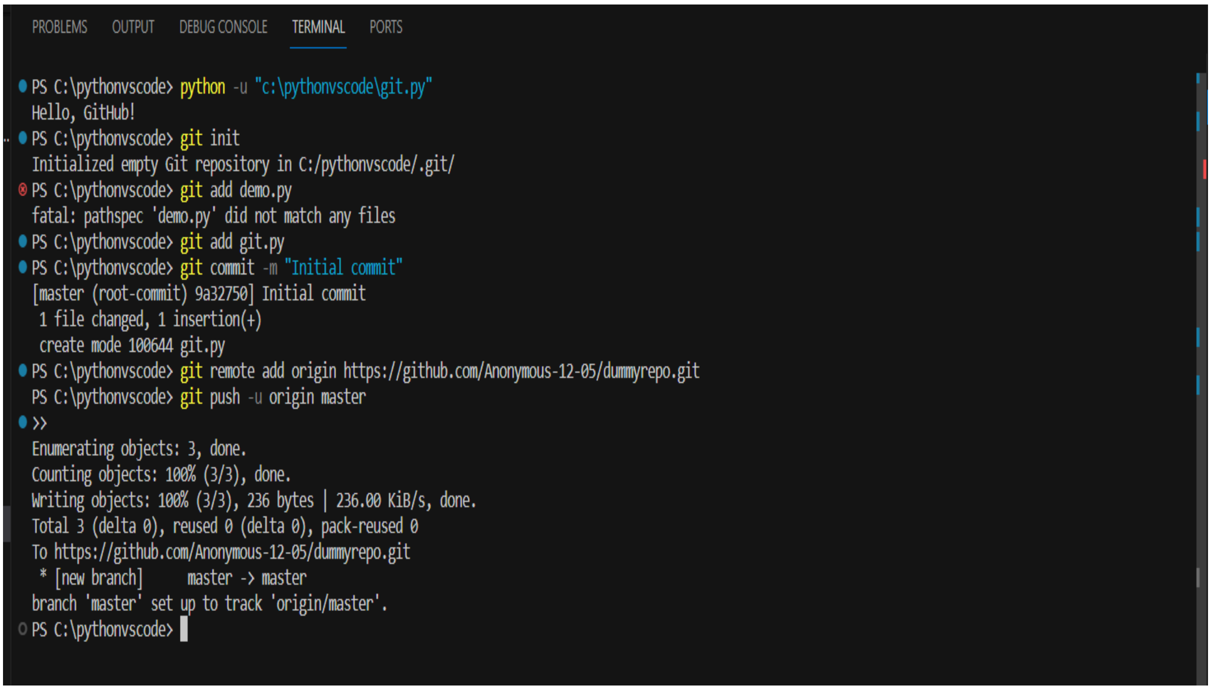
```
git pull origin <branch-name>
```

c. Create a Pull Request:
1. Create a branch and push changes.
2. Open a pull request on GitHub.

d. Fork a Repository:

1. Click "Fork" on GitHub.

Demo run of Git on local device:



```
PS C:\pythonvscode> python -u "c:\pythonvscode\git.py"
Hello, GitHub!
PS C:\pythonvscode> git init
Initialized empty Git repository in C:/pythonvscode/.git/
PS C:\pythonvscode> git add demo.py
fatal: pathspec 'demo.py' did not match any files
PS C:\pythonvscode> git add git.py
PS C:\pythonvscode> git commit -m "Initial commit"
[master (root-commit) 9a32750] Initial commit
 1 file changed, 1 insertion(+)
 create mode 100644 git.py
PS C:\pythonvscode> git remote add origin https://github.com/Anonymous-12-05/dummyrepo.git
PS C:\pythonvscode> git push -u origin master
>>
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 236 bytes | 236.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Anonymous-12-05/dummyrepo.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.
PS C:\pythonvscode>
```