alt text

Git / GitHub Workshop-1

Clarusway

Subject: Git Operations

Learning Goals

- Practice using the Git commands.

Introduction

- We've covered some basic Git concepts, but now it's time to put the concepts in to practice. We'll start with Git commands.

Code Along

Part 1 - Create a local repository

1. Open the terminal (Git Bash for Windows user)

- Go to Desktop and create a directory named "my-github" if you do not have already. And, go to "my-github" directory.

```
mkdir my-github
cd my-github
```

- Create another folder named "git-workshop" in the "my-github" folder and go to "git-workshop" directory.

```
mkdir git-workshop
cd git-workshop
```

2. Git configuration

- Configure git with our name and email. This is to identify who has done what on git and github.

```
git config --global user.name <your_user_name>
git config --global user.email <your_email>
```

- Check the setting

```
git  config --list
```

3. Create a local repository

- We can do that by running the "init" command.

```
git init
```

- Check the if ".git" folder is created.

```
ls -a
```

4. If your branch name is "master", change it to "main".

```
git branch -m main
```

Part 2 - Create and connect a remote repository

5. Create a remote repository on GitHub

- Go to your GitHub account and create a repository named "git-workshop".
    - Write a description for your repo

    - select Public
    - add a README.MD file

6. Go to terminal

- Check the connected remote repositories. The 'git remote -v' lists all currently configured remote repositories, which at this point is none.

```
git remote -v
```

- connect to remote repository

```
git remote add origin <remote repo URL>
```

- Verify the new connection

```
git remote -v
```

7. Create a file named "file1.txt"

- check the status of the project folder

```
git status
```

- store the change in the local repo

```
git add file1.txt"
git commit -m "xxxx"
```

8. upload the changes to the remote repo

```
  git push -u origin main
```

- check the files on the github repo.

Part 3 - Cloning a Remote Repo

9. Create a new remote repo named "git-workshop-1" in GitHub.

10. Clone the remote repo

- go the terminal
- clone the "git-workshop-1"

```
git clone <remote repo URL>
```

- Check the files in the "git-workshop-1" and see the README.MD and .git file.

```
ls -a
```

11. Create a file named **test1.txt**

```
touch test1.txt
```

12. Stage **test1.txt**

```
git add test1.txt
```

13. Store it to the local repository.

```
git commit -m "xxxxx"
```

14. Using Vim editor, create a file named **test2.txt**

```
vim test2.txt
```

15. Stage **test2.txt**

```
git add test2.txt
```

16. Unstage **test2.txt**

```
git rm --cached test2.txt
```

17. check the status of the directory

```
git status
```

18. Store the changes to the local repeository

```
git add .
git commit -m "xxxxxx"
```

19. List the commits

```
git log
```

20. switch to the first commit

```
git checkout 'first commit ID'
```

21. switch to the last commit.

```
git checkout main
```

22. Send the changes to the remote repository

```
git push
```

23. Go and check the remote repository, you will see the new files

☺ **Thanks for Attending** ✍

Clarusway