15.12.24

Expt 21: Cloning Github Repository

Aim:-

To clone an existing github repository to local machine, make a small change to file and save the change.

Tool used:- Git, Github.

Procedure:-

1. Select an existing Public github repository
2. Clone the Repository
3. navigate into the clone Directory
4. Open a file (e.g. "README.md") in text editor and make a small edit, such as fixing typo or adding a line
5. Save the edited.

Result:-

Successfully cloned the repository, made changes to the 'README.md' file and saved the modification, demonstrating basic version control practices using git.

15 12.24

Expt 22: Commit Changes to Cloned Repository

Aim:-

To clone an existing github repository, modify a file and push the changes back to github.

Tool used:- git, github

Procedure:-

1. Choose Public repository and the Run the Command (eg: git clone https://github.com/username/Sample-repo.git
cd Sample-repo)
2. open 'README.md' in text editor and make small edit (Eg: fix a typo or add line)
3. Stage the modified file (git add README.md)
4. Commit the changes with message
5. Push the Commited changes back to the remote repository.

Result:-

Successfully cloned repository, modified file, committed the changes and pushed them back to github, demonstrating effective use of version control with git.

15.12.2024

Expt 23: Pull latest changes from Github Repository

Aim:-

To update local repository by pulling the latest changes made by collaborator from github.

Tool used:- git, github

Procedure:-

1. Clone the Repository
2. Ask a collaborator to make a change in the repository and Push it to github.
3. Use the following command to pull the latest changes from the main branch (git pull origin main)
4. Check the modified files in your local repository to ensure that the collaborator's changes are reflected.

Result:-

Successfully pulled the latest changes from github into the local repository, ensuring that your local copy is up to date with collaborator's modifications.

15.12.24

Expt 24: Create new Branch in github

Aim:-

To create a new branch for implementing a login function, add the function in a 'login.py' file and merge it into the main branch via a pull request.

Tool used: git, github.

Procedure:-

> In your terminal, navigate to your repository and run (git checkout -b feature-login)

> Create a new file named 'login.py' and implement a simple login function.

> Stage the new file and commit the changes.

> Push ~~changes~~ the feature branch to github

> Go to your github repository, navigate to the Pull request tab, and click on new Pull request. Select 'feature-login' as the branch to merge into 'main' and create the Pull request.

> After reviewing the changes, click Merge Pull request to merge 'feature-login' into the main branch.

Result:-

Successfully created a feature branch for implementing a login function, added the functionality in 'login.py', Pushed changes to github, and merged them into the main branch through a pull request.

Expt 25: implement Feature in Forked repository

Aim:- To fork a repository, create a new branch for a feature, implement changes, and submit a pull request to the original repository.

Tool used:- git, github

Procedure:-

1. Navigate to the desired public repository on github and click the fork button to create a copy in your account.
2. Clone your forked repository to your local machine.
3. Create a new branch for your feature.
4. Add a new file (eg 'login.py') with your feature implementation.
5. Stage the new file and commit the changes with descriptive message (git commit -m "implement simple login function")
6. Push changes to your forked repository
7. Go to your forked repository on github, navigate to the Pull request tab, and click on New Pull request. Choose 'feature-login' as the branch to merge into the original repository and create the pull request.

Result:-

Successfully forked a repository, implement a new feature in a separate branch, pushed the changes to github and submitted a pull request for review, facilitating collaboration on the Project.