Experiment 4

GitHub Operations

Lab Objective:
The objective of this experiment is to familiarise participants with essential Git concepts and commands, enabling them to effectively use Git for version control and Collaboration.

Prerequisites:
● Basic understanding of the command-line interface (CLI).
● Familiarity with version control concepts.
● A text editor is installed on your computer.
● Git is installed on your machine.
● Optional: An account on a Git hosting service like GitHub.

Expected Outcomes

By the end of this lab, you will:

1. Understand how to set up a Git repository.
2. Learn how to track and commit changes to files.
3. Explore the history of changes in a Git repository.
4. Gain hands-on experience with branching, merging, and resolving conflicts.
5. Learn how to collaborate using remote repositories on platforms like GitHub

Task 1: Setting Up a Git Repository

Open the command-line interface (CLI) on your computer.

Navigate to the directory where you want to create your Git repository using the cd command.

Initialize a new Git repository in the current directory by running:

git init

This creates a hidden .git folder, marking the directory as a Git repository.

Task 2: Creating and Committing Changes

Create a new text file named example.txt using any text editor.

Add some content to example.txt and save the file.

Check the status of your working directory by running:

git status

Experiment 4

This highlights untracked files.

Stage the file for commit by running:

git add example.txt

Commit the staged changes with a message:

git commit -m "Add content to example.txt"

Task 3: Exploring History

Modify the content of example.txt.

Run the following command to notice the modified file:

git status

View the differences between the working directory and the last commit:

git diff

Display a chronological history of commits:

git log

Task 4: Branching and Merging

Create a new branch named feature:

git branch feature

Switch to the new branch:

git checkout feature

Or use the shorthand:

git checkout -b feature

Make changes to example.txt in the feature branch.

Commit the changes:

git commit -am "Modify example.txt in feature branch"

Switch back to the master branch:

git checkout master

Merge the changes from the feature branch into the master branch:

Experiment 4

git merge feature

Task 5: Collaborating with Remote Repositories

Create an account on a Git hosting service like GitHub (if you don't have one already).

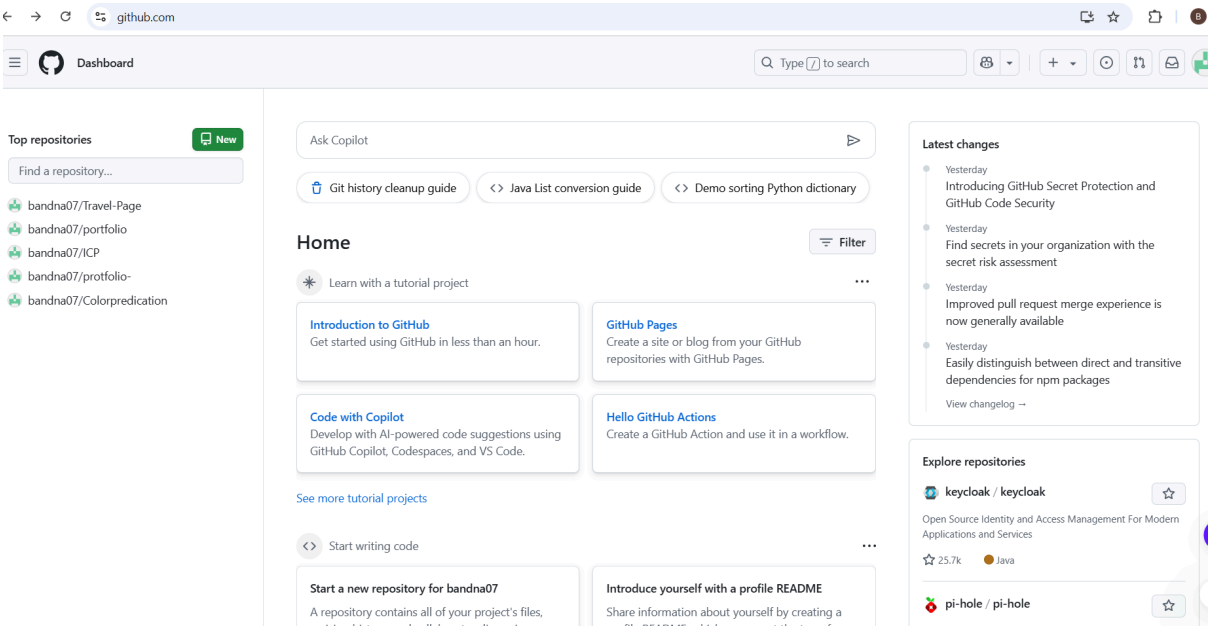
Create a new repository on GitHub.

Link your local repository to the remote repository by running:

git remote add origin <repository_url>

Push your local commits to the remote repository:

git push origin master



Experiment 4