**Placement Empowerment Program**

***Cloud Computing and DevOps Centre***

**Set Up Git Branching**

**Create a new branch in your Git repository for testing. Add a new feature and merge it.**

Name: Kirthika S Department: CSE



**Introduction**

In this Proof of Concept (POC), Git is used for version control to manage the development workflow. Git allows developers to create separate branches for new features, isolate them from the main branch, and merge them back after completion. This ensures organized and collaborative development.

**Overview:**

This POC demonstrates how to:

1. Initialize a Git repository.

2. Create and switch between branches.

3. Commit changes in different branches.

4. Merge feature branches into the main branch.

5. Delete branches after completing the work.

**Objectives**

By the end of this POC, you will:

1. To initialize and set up a Git repository.

2. To create and manage feature branches (e.g., testing-feature).

3. To demonstrate adding, committing, and merging code.

4. To showcase how to delete branches after their purpose is served.

5. To learn how to resolve merge conflicts if any arise during the process.

**Importance of create git branch**

**Isolated Development** – Developers can work on new features or bug fixes without modifying the stable code in the main branch.

**Collaboration** – Teams can work simultaneously on different tasks and later merge their changes.

**Safe Testing** – Changes can be tested in a separate environment before merging into the main branch.

**Version Control** – Keeps the history of changes, making it easy to roll back if needed.

**Code Review & Quality** – Pull requests (PRs) allow for code review before merging, ensuring better quality and fewer errors.

**Step-by-Step Overview**

Step 1:

Open **VS Code**.

Open your **project folder** that is connected to a GitHub repository.

Set the path to the folder created in first step.

Initialize Git by typing this command:

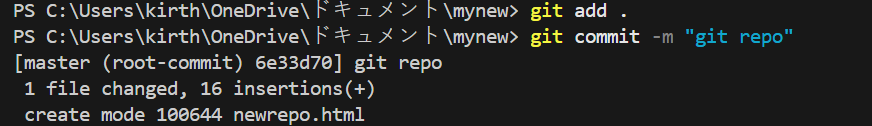
**git init**

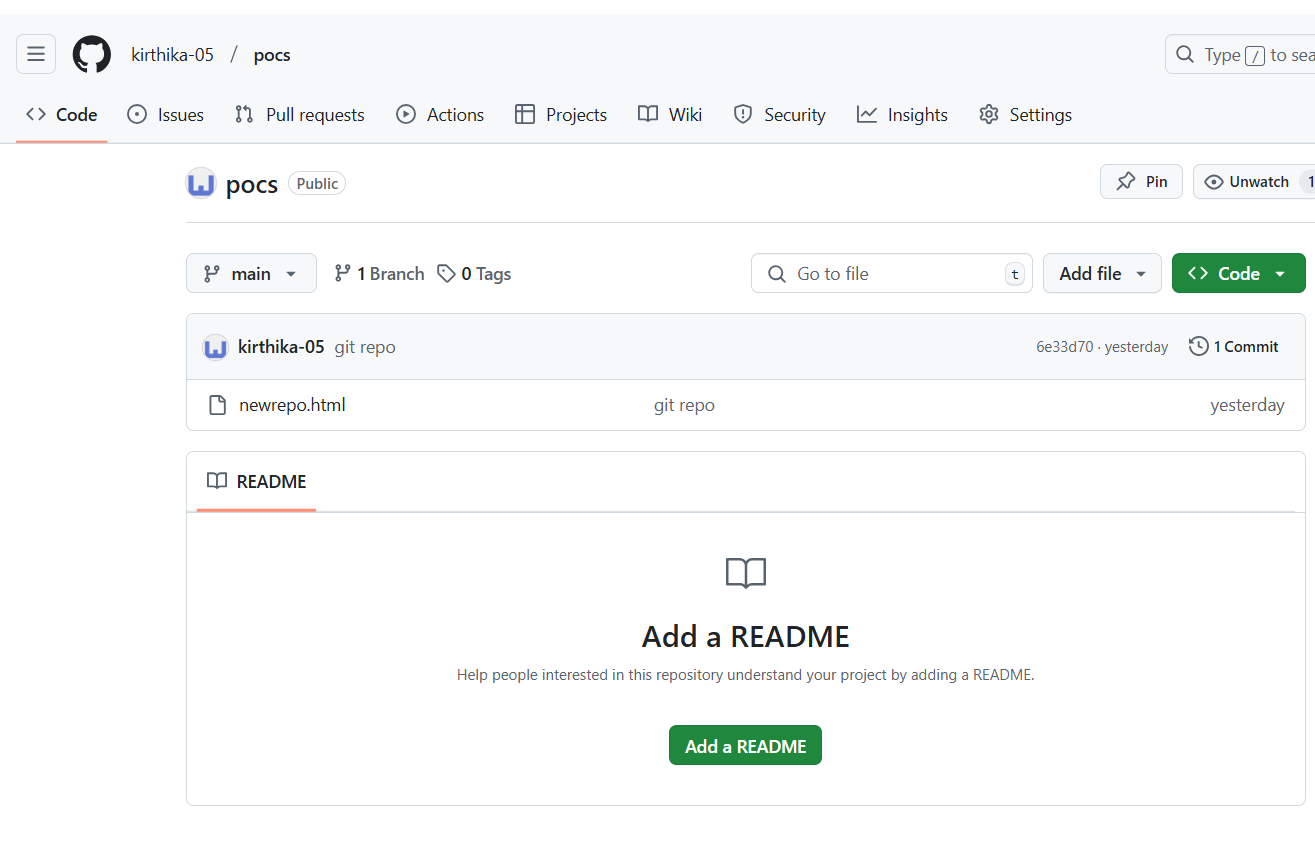
This command will create a .git folder inside your folder, which tells Git to start tracking your files.



Step 2:

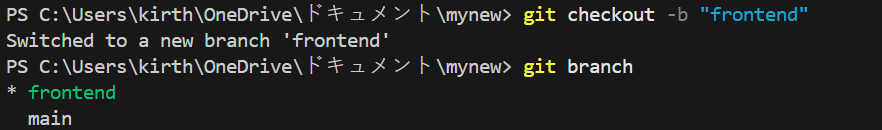
Create a simple file to start the repository and Add the File to Git, Save this change in Git with a commit message





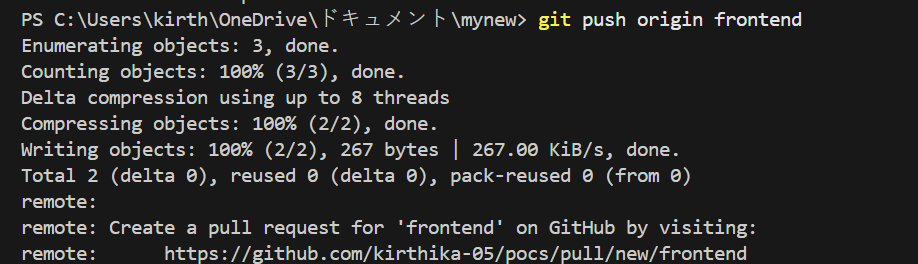
Step 3:

Create and switch to a new branch called **frontend**



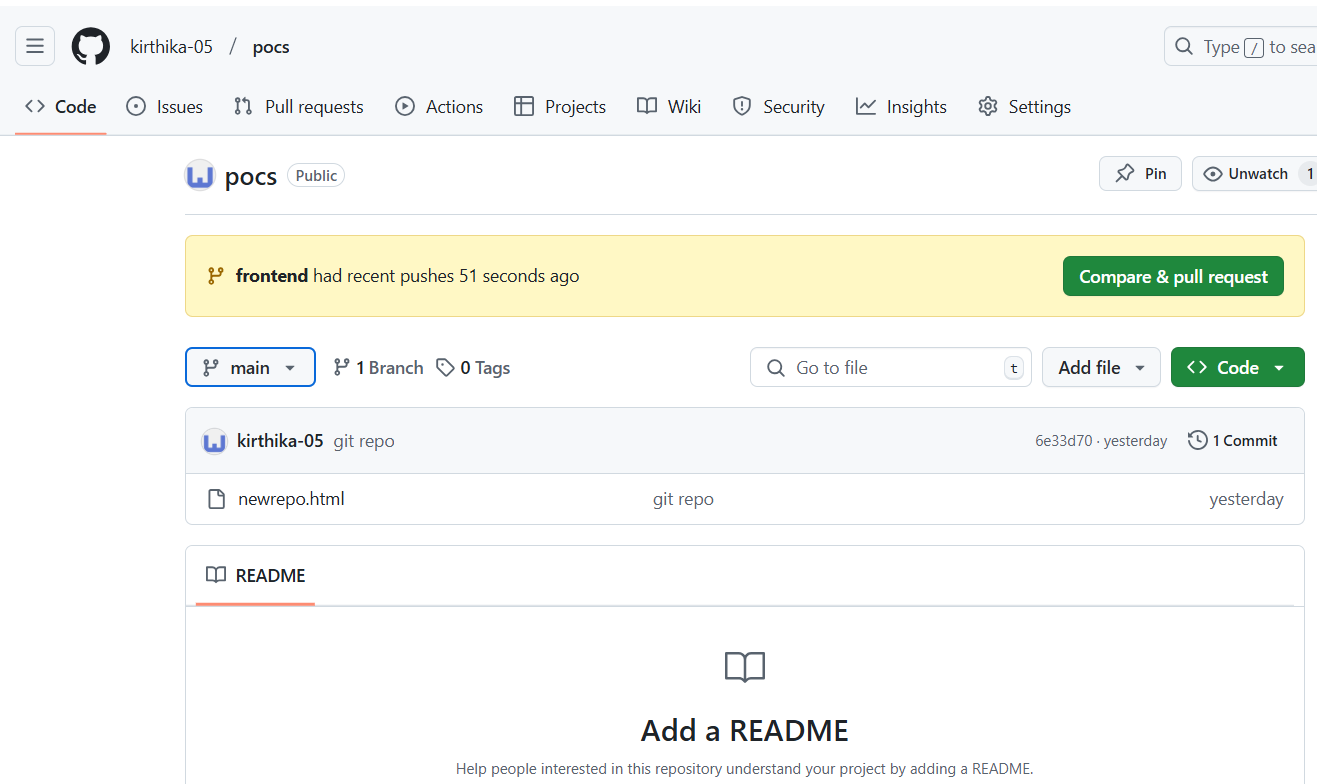
Step 4:

Now push your branch,



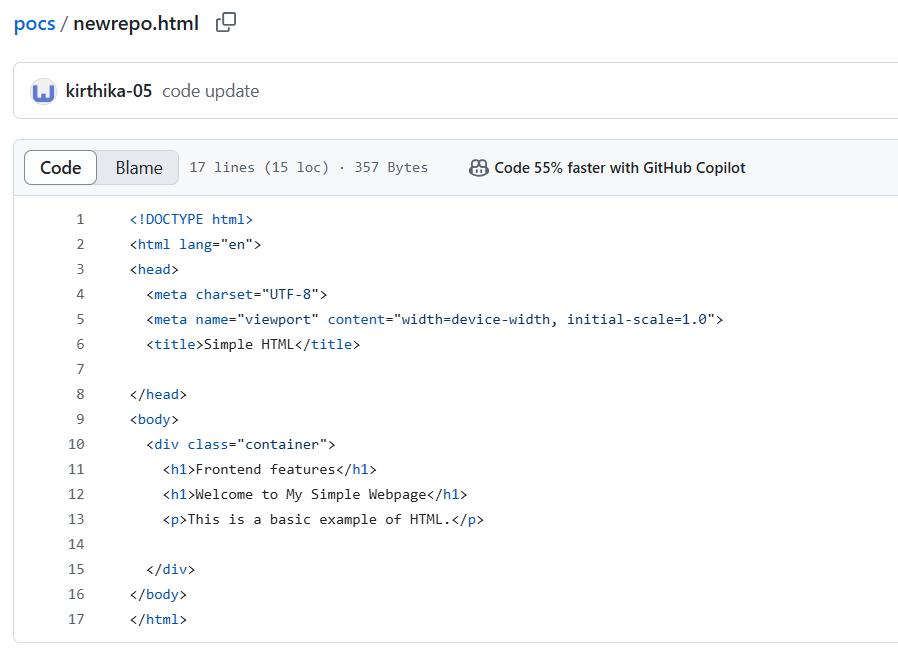
Step 5:

Your branch was created,



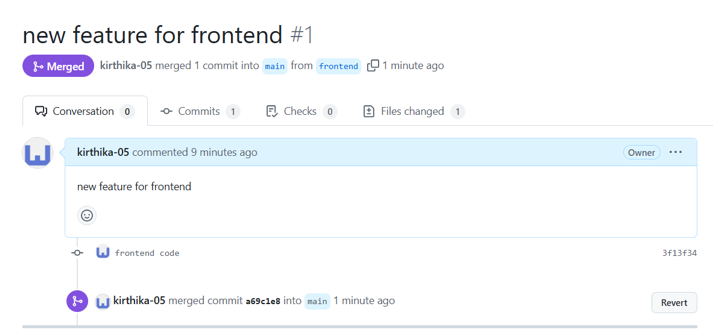
Step 6:

Open the **newrepo.html** in github, you can see the new features added **“Frontend features”**



Step 7:

Merge it,



Step 8:

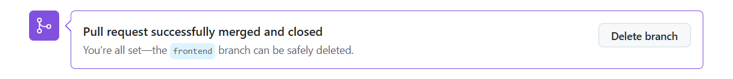
1.Always test your changes before merging.

2.Use descriptive branch names to easily identify the purpose of each branch.

3.Regularly pull updates from the main branch to your feature branch to avoid conflicts.

Step 9:

Once you merge it, you can delete the branch **frontend**



Overview:

By using Git branching effectively, you can streamline your development workflow, minimize conflicts, and work collaboratively with your team. This task demonstrates how to create and merge branches, an essential skill for modern software development.