

Assignment day 5:

Assignment 1:

Initialize a new Git repository in a directory of your choice. Add a simple text file to the repository and make the first commit.

Step 1: Initialize a New Git Repository

1. **Open Terminal (or Command Prompt)**
2. **Navigate to your desired directory:**

3. **Initialize the Git repository:**

Git init

This command creates a new subdirectory named `.git` that contains all of your necessary repository files. This also makes the current directory the working directory.

Step 2: Add a Simple Text File

1. **Create a new text file:**

This command creates a new file named `simplefile.txt` and writes "This is a simple text file." into it. Alternatively, you can use a text editor to create and edit the file.

2. **Check the status:**

```
git status
```

This command shows the status of the repository and lists **simplefile.txt** as an untracked file.

Step 3: Stage and Commit the File

1. Stage the file:

```
Git add . simplefile.txt
```

This command stages **simplefile.txt**, preparing it for the next commit.

2. Commit the file:

```
git commit -m "new commit"
```

This command commits the staged file to the repository with the message "Add simple text file as the first commit".

Summary of Commands

After following these steps, you will have successfully initialized a new Git repository, added a text file, and made your first commit.

1] get to your path

2] git init

3] create new file

4] git status

5] git add . // git add simplefile.txt

6] git commit -m "___"

Assignment 2:

Branch Creation and Switching

Create a new branch named 'feature' and switch to it. Make changes in the 'feature' branch and commit them.

Step 1: Create and Switch to the 'feature' Branch

1. Create a new branch named 'feature':

```
git branch feature
```

2. Switch to the 'feature' branch:

```
git checkout feature
```

Alternatively, you can create and switch to the new branch in one command:

```
git checkout -b feature
```

Step 2: Make Changes in the 'feature' Branch

1. **Make some changes:** Edit the `simplefile.txt` to add some new content. You can do this using a text editor or from the command line.

Using the command line, you can add new content like this:

2. **Check the status:**

```
Git status
```

This command will show that `simplefile.txt` has been modified.

Step 3: Stage and Commit the Changes

1. **Stage the modified file:**

```
Git add simplefile.txt
```

2. **Commit the changes:**

```
Git commit -m "update simplefile.txt in feature"
```

Summary of Commands

```
1] git checkout -b feature
```

```
2] add content in file
```

```
3] git status
```

```
4] git add simplefile.txt
```

```
5] git commit -m "___"
```

After following these steps, you will have created a new branch named 'feature', switched to it, made changes to `simplefile.txt`, and committed those changes.

Assignment 3: Feature Branches and Hotfixes

Create a 'hotfix' branch to fix an issue in the main code. Merge the 'hotfix' branch into 'main' ensuring that the issue is resolved.

Step 1: Create and Switch to the 'hotfix' Branch

1. Ensure you are on the 'main' branch:

```
Git branch feature
```

2. Create a new branch named 'hotfix' and switch to it:

```
Git checkout feature
```

Step 2: Make and Commit the Hotfix

1. Make the necessary changes to fix the issue: Edit the `simplefile.txt` or any other file to apply the hotfix. For example:

2. Check the status:

```
git checkout -b feature
```

This command will show the modified files.

3. Stage the modified file:

4. Commit the changes:

```
Git commit -m "Fix critical issue in the hotfix branch"
```

Step 3: Merge the 'hotfix' Branch into 'main'

1. Switch back to the 'main' branch:

```
git checkout main
```

2. Merge the 'hotfix' branch into 'main':

```
git merge hotfix
```

Step 4: Verify the Merge

1. Check the status and log to ensure the merge was successful:

```
Git status
```

```
Git log --oneline
```

Ensure that the commit from the 'hotfix' branch is present in the 'main' branch.

Summary of Commands

1]git checkout main

2]git checkout -b hotfix

3]add content in fiel

4]git status

5]git add simplefile.txt

6]git commit -m"adding new point"

7]git checkout main

8]git merge hotfix

9] git status

10] git log --online

By following these steps, you will have created a 'hotfix' branch, made the necessary changes to fix the issue, and then merged the 'hotfix' branch back into the 'main' branch, ensuring that the issue is resolved.