St. JOSEPH'S
GROUP OF INSTITUTIONS
OMR, CHENNAI - 119

Placement Empowerment Program

Cloud Computing & DevOps Centre

Set up Git Branching: Create a new branch in your Git repo for testing. Add a new feature and merge it.

Name: Krishna bhattad J

Dept: IT

INTRODUCTION

Creating a new branch in your Git repository for testing allows you to work on new features, bug fixes, or experiments without affecting the main branch (often named main or master). This ensures that your primary codebase remains stable while you develop and refine new functionality. Once you've made and tested your changes, you can merge the branch back into the main branch using a merge or rebase strategy. This process helps in maintaining a clean and organized version history while enabling smooth collaboration among developers. Additionally, resolving merge conflicts, if any, ensures that the final integrated code functions correctly. Proper use of branching and merging in Git enhances workflow efficiency, reduces errors, and facilitates continuous integration and deployment (CI/CD) practices.

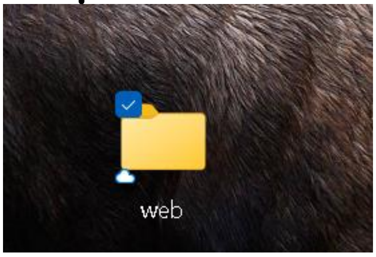
Git Commands used:

1. Git init
2. Git add .
3. Git commit -m " "

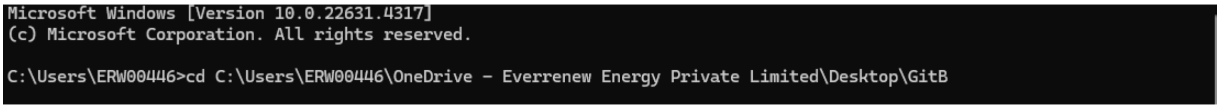
4. Git checkout -b
5. Git merge
6. Git branch -d

Step-by-step process:

Step 1: Create a folder in your deskstop.



Step 2: Open the cmd prompt and set the path to the folder that you created.



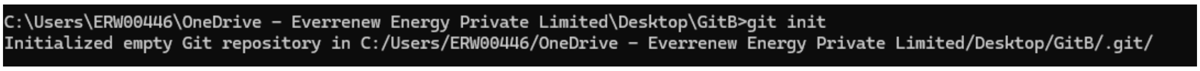
```
Microsoft Windows [Version 10.0.22631.4317]
(c) Microsoft Corporation. All rights reserved.

C:\Users\ERW00446>cd C:\Users\ERW00446\OneDrive - Everrenew Energy Private Limited\Desktop\GitB
```

Step 3: Initialize the git cmds for the process.

Git commands:

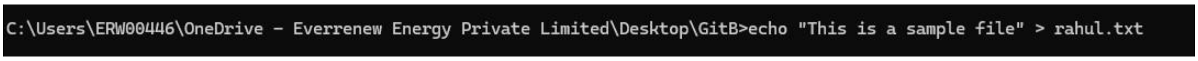
Cmd 1: git init



```
C:\Users\ERW00446\OneDrive - Everrenew Energy Private Limited\Desktop\GitB>git init
Initialized empty Git repository in C:/Users/ERW00446/OneDrive - Everrenew Energy Private Limited/Desktop/GitB/.git/
```

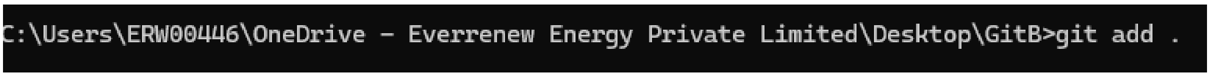
Note:

Before cmd 2 you need to create a text file.



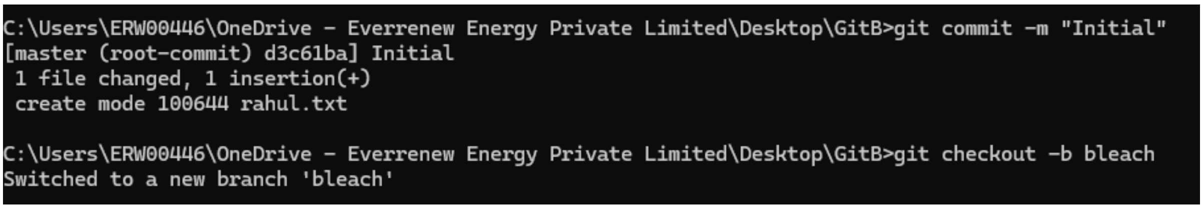
```
C:\Users\ERW00446\OneDrive - Everrenew Energy Private Limited\Desktop\GitB>echo "This is a sample file" > rahul.txt
```

Cmd 2: git add .



```
C:\Users\ERW00446\OneDrive - Everrenew Energy Private Limited\Desktop\GitB>git add .
```

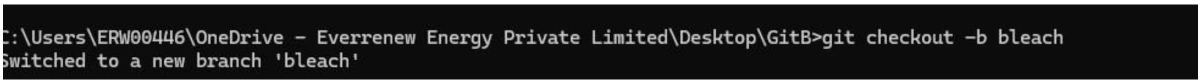
Cmd 3: git commit -m



```
C:\Users\ERW00446\OneDrive - Everrenew Energy Private Limited\Desktop\GitB>git commit -m "Initial"
[master (root-commit) d3c61ba] Initial
 1 file changed, 1 insertion(+)
 create mode 100644 rahul.txt

C:\Users\ERW00446\OneDrive - Everrenew Energy Private Limited\Desktop\GitB>git checkout -b bleach
Switched to a new branch 'bleach'
```

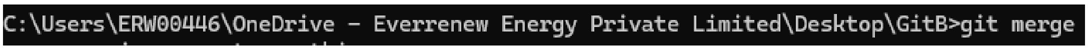
Cmd 4: git checkout – for switching to a new branch.



```
:\Users\ERW00446\OneDrive - Everrenew Energy Private Limited\Desktop\GitB>git checkout -b bleach
Switched to a new branch 'bleach'
```

Step 4: Repeat the above “Cmds and steps for creating another text file.

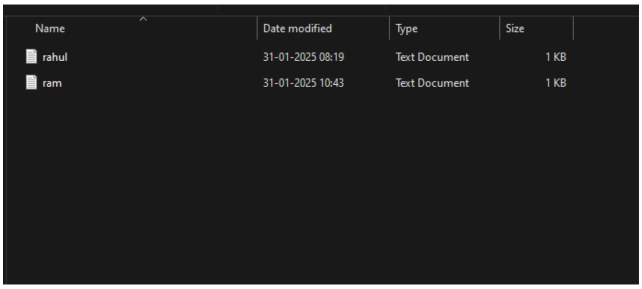
Step 5: Merge changes from rahul to krishna.



```
C:\Users\ERW00446\OneDrive - Everrenew Energy Private Limited\Desktop\GitB>git merge
```

Step 6: Once you are done with merging, you can delete the ram branch.

Step 7: Time to check the changes that has been made.



Name	Date modified	Type	Size
rahul	31-01-2025 08:19	Text Document	1 KB
ram	31-01-2025 10:43	Text Document	1 KB

OUTCOME

The outcome of creating a new branch, adding a feature, and merging it in Git is a structured and efficient workflow that enhances software development. It allows for **safe testing and experimentation** without disrupting the main codebase. Once merged, the new feature becomes part of the main branch, ensuring an **organized and controlled integration** of changes. This approach also improves **collaboration among developers**, reduces conflicts, and maintains a **clear version history**, making it easier to track modifications.