EXPERIMENT-02

AIM: Explore Git and GitHub Commands

Objective

To explore and practice basic Git and GitHub commands for version control in a collaborative environment.

Software Requirements

- Git installed (https://git-scm.com/downloads)
- GitHub account (https://github.com/)
- Code editor (VS Code or any)

VERSION CONTROL SYSTEM

A **Version Control System (VCS)** is a tool that helps manage changes to files, particularly source code, over time. It tracks modifications, allows multiple people to collaborate on the same project, and ensures that a complete history of changes is maintained. VCS is essential for software development, but it can also be used for other types of documents or files.

GIT

Git is a distributed version control system (VCS) designed to manage and track changes in source code during software development. Created by Linus Torvalds in 2005, Git is widely used by developers to collaborate on projects, keep track of changes, and manage multiple versions of their codebase.

GITHUB

GitHub is a web-based platform and service that provides hosting for software development and version control using Git. It offers a collaborative environment where developers can manage and share their projects, track issues, and work together on code. GitHub is widely used in the software development community for both open-source and private projects.

Git Commands: Working With Local Repositories

1. git init

- The command git init is used to create an empty Git repository.
- After the git init command is used, a .git folder is created in the directory with some subdirectories. Once the repository is initialized, the process of creating other files begins.

  Command: git init

2. git status

- The git status command tells the current state of the repository.
- The command provides the current working branch. If the files are in the staging area, but not committed, it will be shown by the git status. Also, if there are no changes, it will show the message no changes to commit, working directory clean.

  Command: git status

  The git status command shows us details about:
  - **modified files** (files that are changed but not staged).
  - **untracked files** (files that Git is not tracking).
  - **staged files** (files that are staged and ready to be committed).

3. git config

- The git config command is used initially to configure the user.name and user.email. This specifies what email id and username will be used from a local repository.
- When git config is used with --global flag, it writes the settings to all repositories on the computer.

  Command: git config –global user.name “User-name”

  git config –global user.email “User-Email”

4. git add

- Add command is used after checking the status of the files, to add those files to the staging area.
- Before running the commit command, "git add" is used to add any new or modified files.

  Command: git add <file-name>

  git add . // to add all files

5. git commit

- The commit command makes sure that the changes are saved to the local repository.
- The command "git commit –m <message>" allows you to describe what has happened and help others understand.

  Command: git commit –m “Message”

6. git push

- The command git push is used to transfer the commits or pushing the content from the local repository to the remote repository.
- The command is used after a local repository has been modified, and the modifications are to be shared with the remote team members.

  Command: git push –u origin master

7. git branch

- The git branch command is used to determine what branch the local repository is on.
- The command enables adding and deleting a branch.

  Command: git branch <branch-name> // Create a new branch
  git branch –a //List al remote or local branches
  git branch –d // Delete a branch

8. git checkout

- The git checkout command is used to switch branches, whenever the work is to be started on a different branch.
- The command works on three separate entities: files, commits, and branches.

  Command: git checkout <branch-name> // Checkout an existing branch
  git checkout –b <new-branch> // Checkout and create a new branch with that name.

9. git merge

- The git merge command is used to integrate the branches together. The command combines the changes from one branch to another branch.
- It is used to merge the changes in the staging branch to the stable branch.

  Command: git merge <branch-name>

Git Commands: Working With Remote Repositories

1. git remote

- The git remote command is used to create, view, and delete connections to other repositories.
- The connections here are not like direct links into other repositories, but as bookmarks that serve as convenient names to be used as a reference.

Command: git remote add origin <repo-link>

git remote –v // List all currently configured remote repositories.

2. git clone

- The git clone command is used to create a local working copy of an existing remote repository.
- The command downloads the remote repository to the computer. It is equivalent to the Git init command when working with a remote repository.

Command: git clone <remote-URL>

3. git pull

- The git pull command is used to fetch and merge changes from the remote repository to the local repository.
- The command "git pull origin master" copies all the files from the master branch of the remote repository to the local repository.

Command: git pull <branch-name> <remote-URL>

4. git log - View commit history

Procedure

1. Configure Git (First Time Setup)

git config --global user.name "Your Name"
git config --global user.email "your-email@example.com"
git config –list

---

2. Initialize Local Repository

mkdir devops-lab
cd devops-lab
git init

---

3. Add Files and Commit Changes

```
echo "Hello DevOps" > readme.md
git status
git add readme.md
git commit -m "Initial commit: added readme"
```

---

4. View Commit History

```
git log
git log --oneline
```

---

5. Push to GitHub

1. Create a repository on GitHub.
2. Connect local repo to GitHub:

```
git remote add origin https://github.com/your-username/your-repo-name.git
git branch -M main
git push -u origin main
```

---

6. Clone Repository

```
git clone https://github.com/your-username/your-repo-name.git
```

---

7. Check Remote URL

```
git remote –v
```

---

8. Create and Manage Branches

```
git branch feature-1
git checkout feature-1
# or using
git switch feature-1
```

---

9. Update Files and Merge Branch

```
echo "Adding feature work" > readme.md
git add .
```

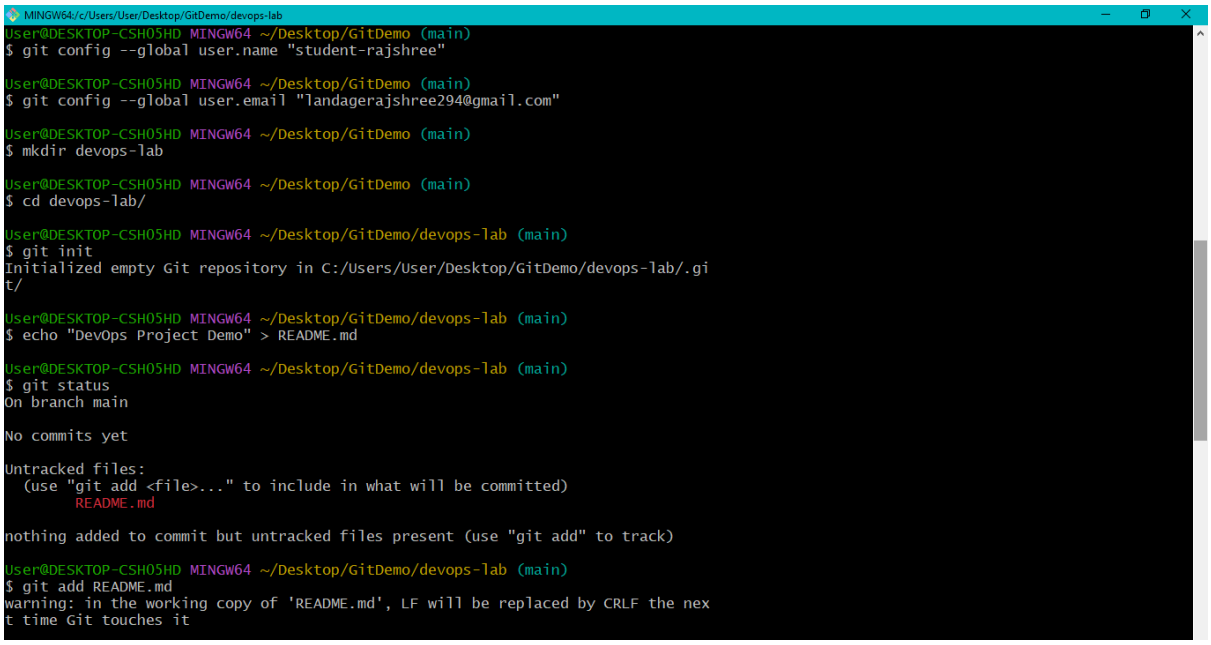
git commit -m "Updated readme in feature-1 branch"
git checkout main
git merge feature-1

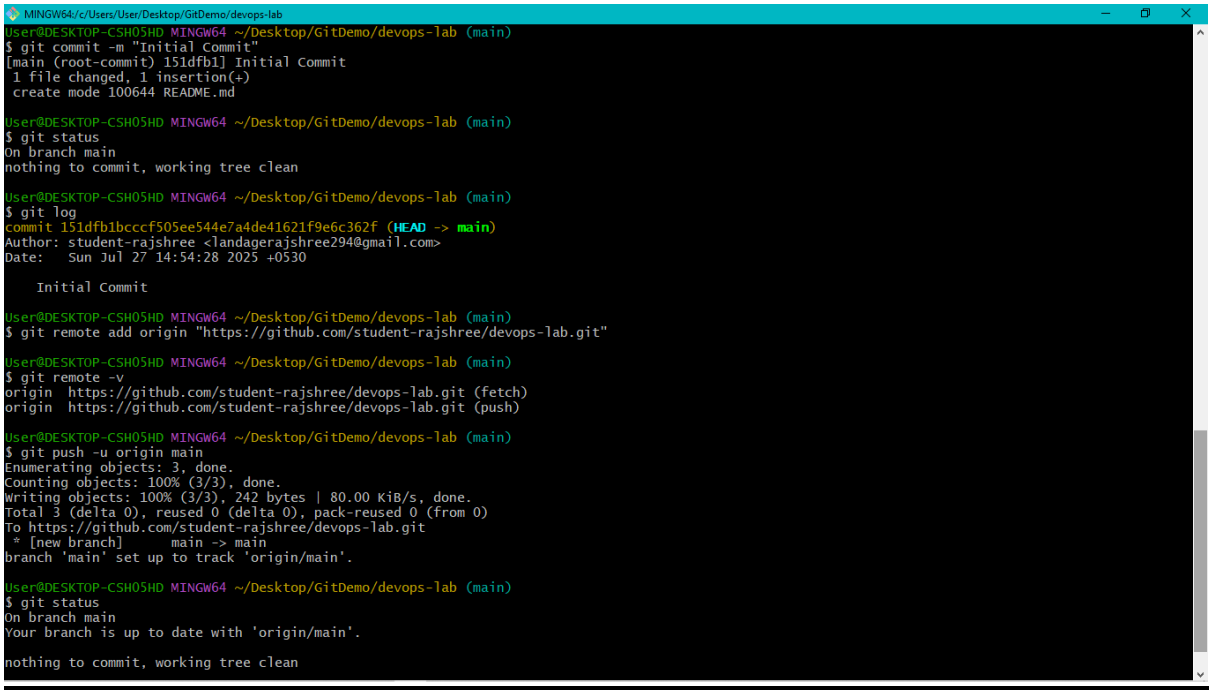
---

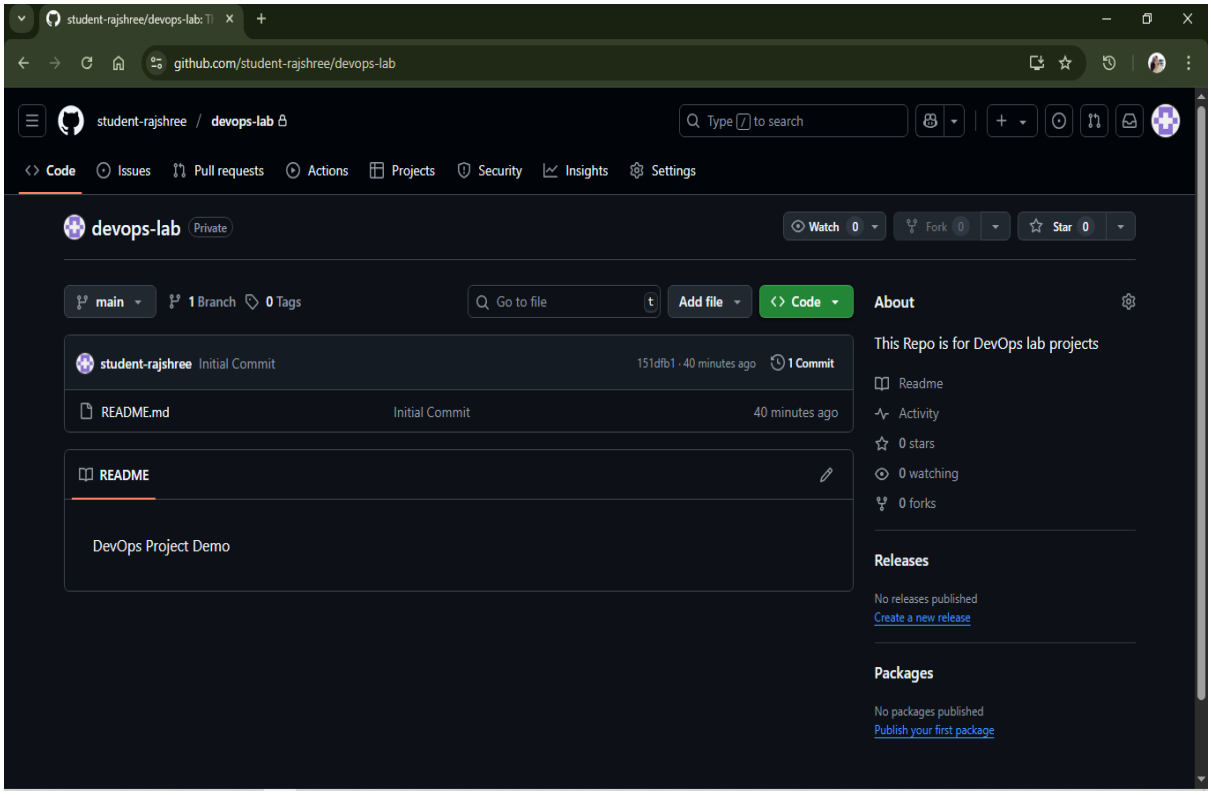
10. Pull Latest Changes

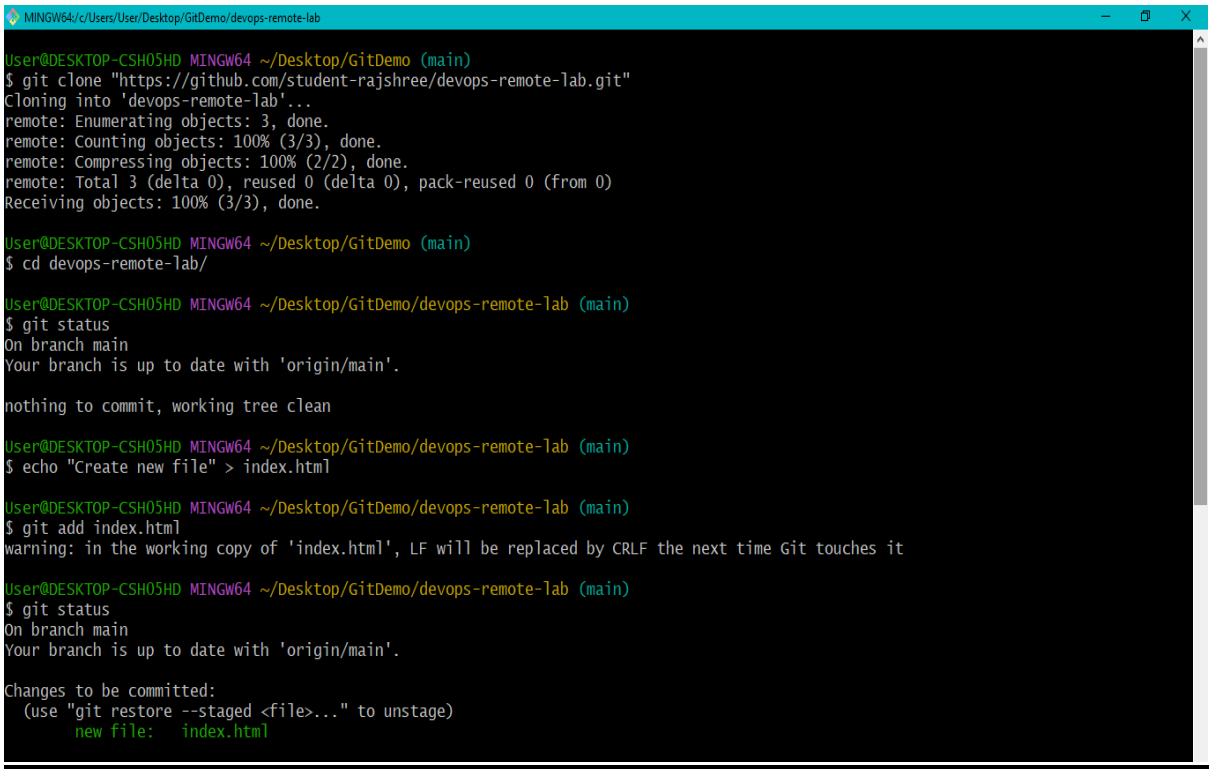
git pull origin main

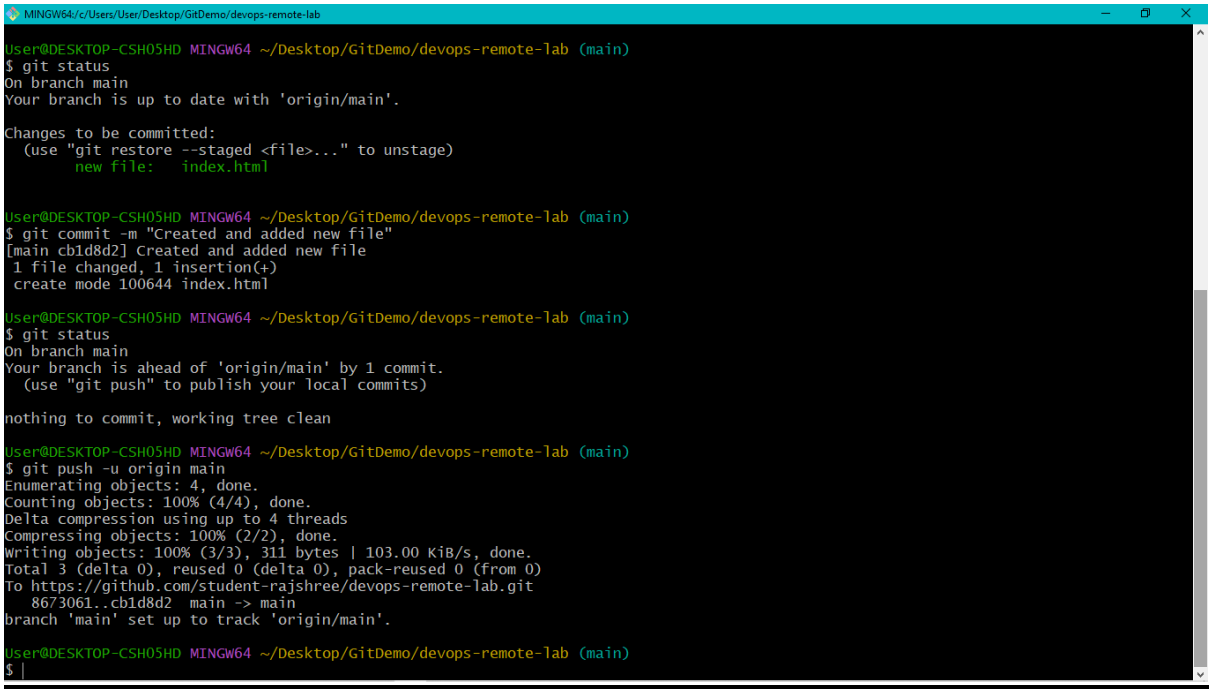
OUTPUT

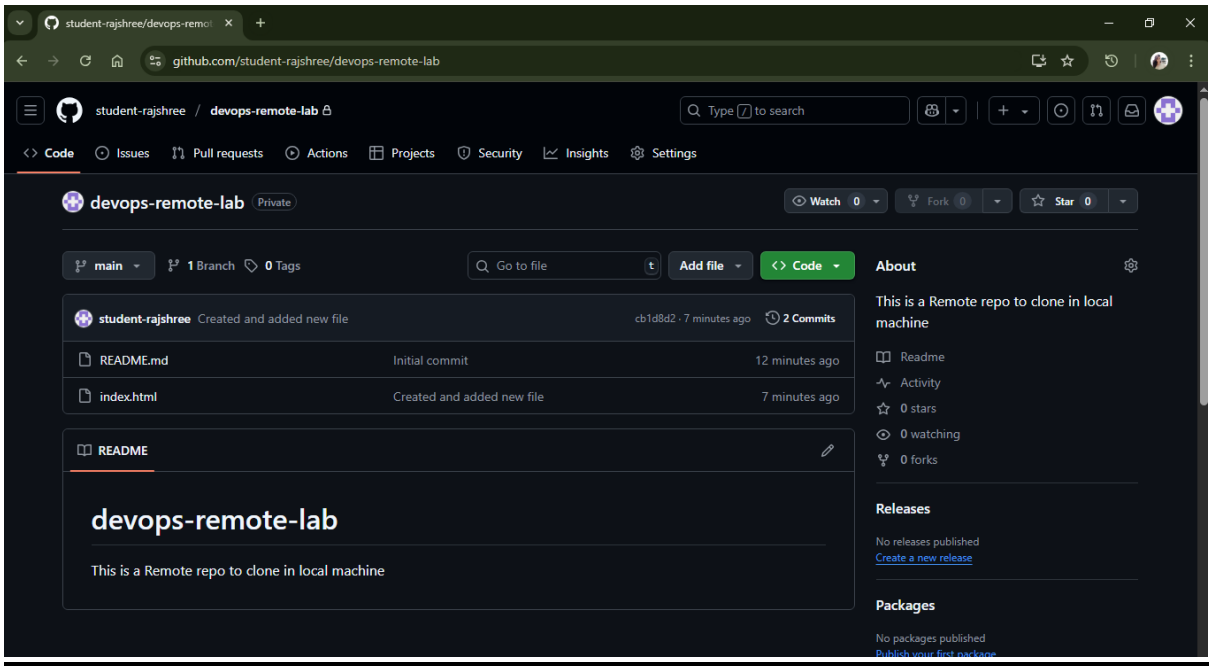





student-rajshree/devops-lab
github.com/student-rajshree/devops-lab
student-rajshree / devops-lab
Type / to search
Code
Issues
Pull requests
Actions
Projects
Security
Insights
Settings
devops-lab
Private
Watch 0
Fork 0
Star 0
main
1 Branch
0 Tags
Go to file
Add file
Code
About
This Repo is for DevOps lab projects
student-rajshree Initial Commit
151dfb1 · 40 minutes ago
1 Commit
README.md
Initial Commit
40 minutes ago
README
DevOps Project Demo
Readme
Activity
0 stars
0 watching
0 forks
Releases
No releases published
Create a new release
Packages
No packages published
Publish your first package


MINGW64:/c/Users/User/Desktop/GitDemo/devops-remote-lab
User@DESKTOP-CSHO5HD MINGW64 ~/Desktop/GitDemo (main)
$ git clone "https://github.com/student-rajshree/devops-remote-lab.git"
Cloning into 'devops-remote-lab'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (3/3), done.
User@DESKTOP-CSHO5HD MINGW64 ~/Desktop/GitDemo (main)
$ cd devops-remote-lab/
User@DESKTOP-CSHO5HD MINGW64 ~/Desktop/GitDemo/devops-remote-lab (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.
nothing to commit, working tree clean
User@DESKTOP-CSHO5HD MINGW64 ~/Desktop/GitDemo/devops-remote-lab (main)
$ echo "Create new file" > index.html
User@DESKTOP-CSHO5HD MINGW64 ~/Desktop/GitDemo/devops-remote-lab (main)
$ git add index.html
warning: in the working copy of 'index.html', LF will be replaced by CRLF the next time Git touches it
User@DESKTOP-CSHO5HD MINGW64 ~/Desktop/GitDemo/devops-remote-lab (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.
Changes to be committed:
(use "git restore --staged <file>..." to unstage)
new file: index.html


MINGW64:/c/Users/User/Desktop/GitDemo/devops-remote-lab
User@DESKTOP-CSH05HD MINGW64 ~/Desktop/GitDemo/devops-remote-lab (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.
Changes to be committed:
(use "git restore --staged <file>..." to unstage)
new file: index.html
User@DESKTOP-CSH05HD MINGW64 ~/Desktop/GitDemo/devops-remote-lab (main)
$ git commit -m "Created and added new file"
[main cb1d8d2] Created and added new file
1 file changed, 1 insertion(+)
create mode 100644 index.html
User@DESKTOP-CSH05HD MINGW64 ~/Desktop/GitDemo/devops-remote-lab (main)
$ git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
(use "git push" to publish your local commits)
nothing to commit, working tree clean
User@DESKTOP-CSH05HD MINGW64 ~/Desktop/GitDemo/devops-remote-lab (main)
$ git push -u origin main
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 311 bytes | 103.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/student-rajshree/devops-remote-lab.git
8673061..cb1d8d2 main -> main
branch 'main' set up to track 'origin/main'.
User@DESKTOP-CSH05HD MINGW64 ~/Desktop/GitDemo/devops-remote-lab (main)
$


github.com/student-rajshree/devops-remote-lab
student-rajshree / devops-remote-lab
Type / to search
Code
Issues
Pull requests
Actions
Projects
Security
Insights
Settings
devops-remote-lab
Private
Watch 0
Fork 0
Star 0
main
1 Branch
0 Tags
Go to file
Add file
Code
student-rajshree Created and added new file
cb1d8d2 · 7 minutes ago
2 Commits
README.md
Initial commit
12 minutes ago
index.html
Created and added new file
7 minutes ago
README
devops-remote-lab
This is a Remote repo to clone in local machine
About
This is a Remote repo to clone in local machine
Readme
Activity
0 stars
0 watching
0 forks
Releases
No releases published
Create a new release
Packages
No packages published