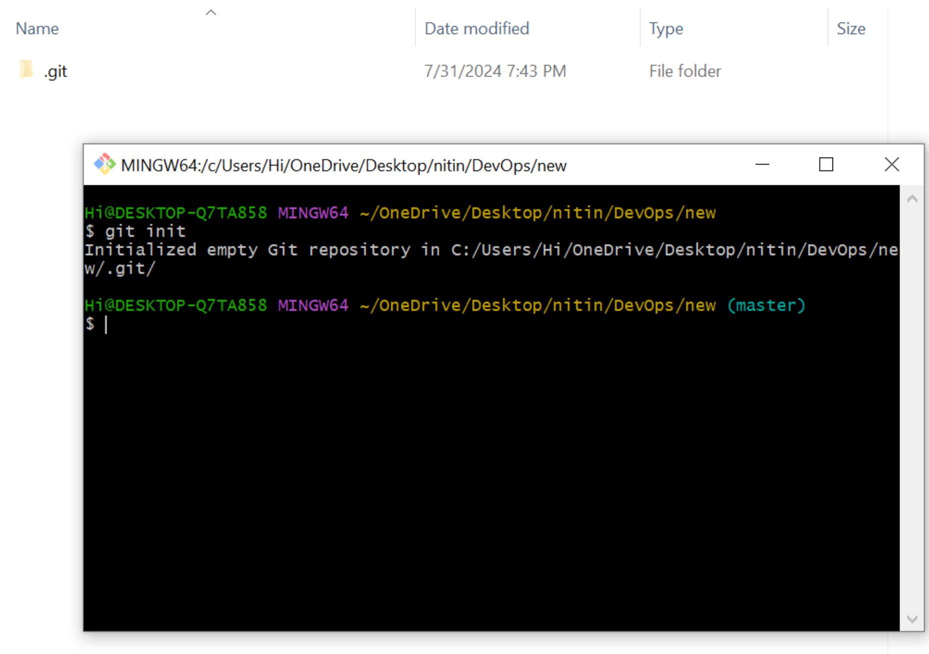
GIT Assignment

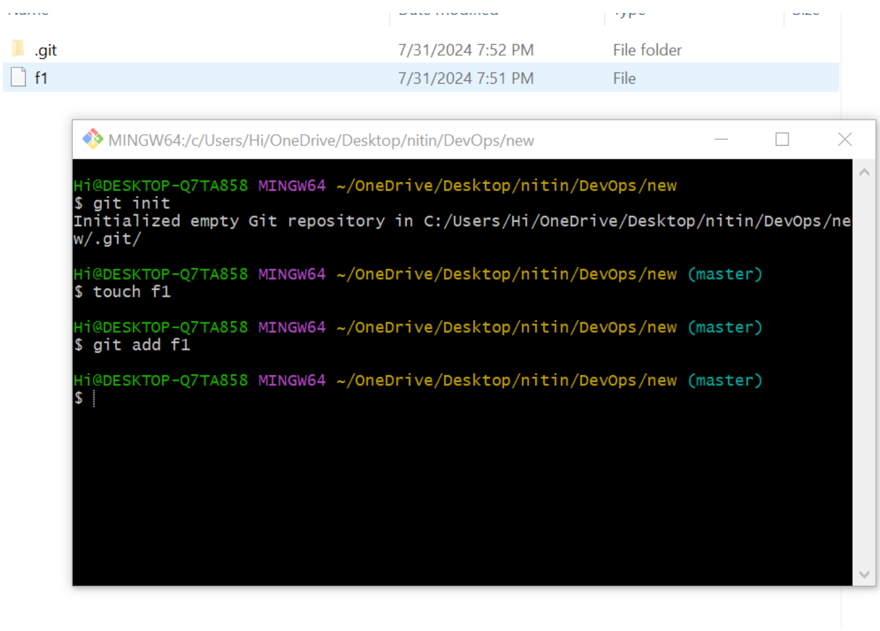


Name	Date modified	Type	Size
.git	7/31/2024 7:43 PM	File folder	

MINGW64:/c/Users/Hi/OneDrive/Desktop/nitin/DevOps/new

```
Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new
$ git init
Initialized empty Git repository in C:/Users/Hi/OneDrive/Desktop/nitin/DevOps/ne
w/.git/

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ |
```

Initating repository using **git init**

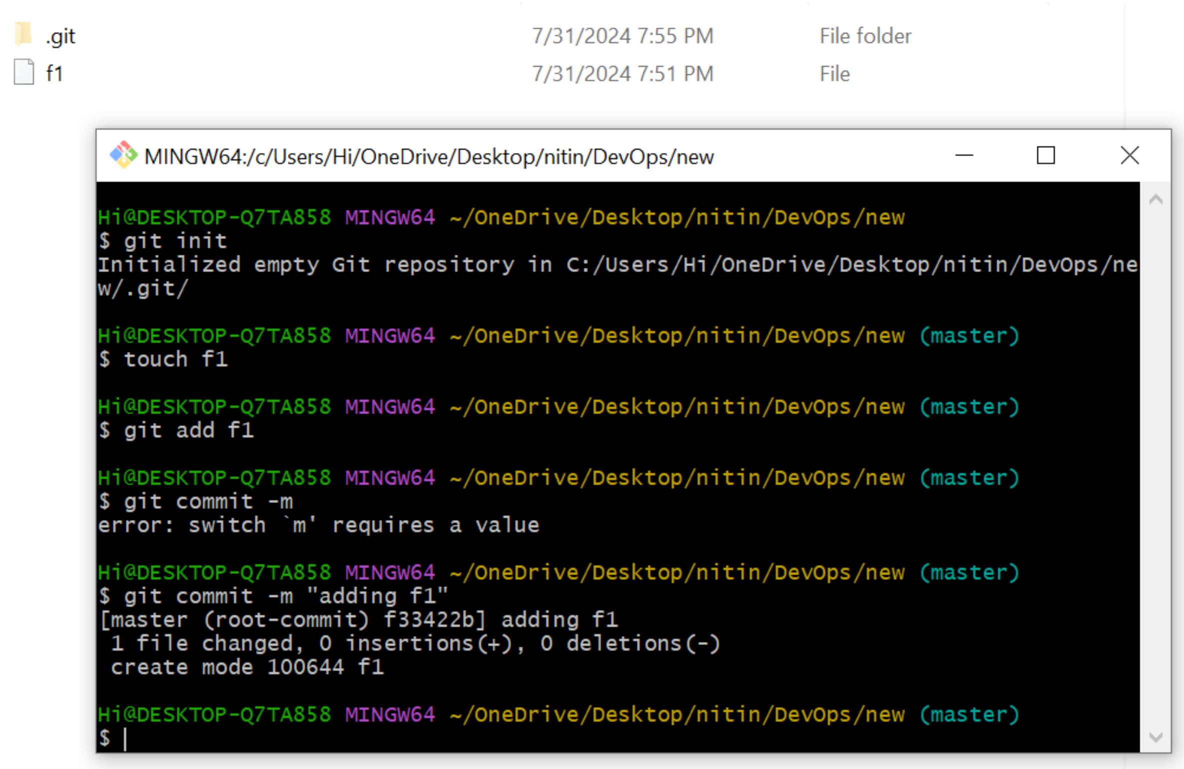


.git	7/31/2024 7:52 PM	File folder	
f1	7/31/2024 7:51 PM	File	

MINGW64:/c/Users/Hi/OneDrive/Desktop/nitin/DevOps/new

```
Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new
$ git init
Initialized empty Git repository in C:/Users/Hi/OneDrive/Desktop/nitin/DevOps/ne
w/.git/

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ touch f1

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ git add f1

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ |
```

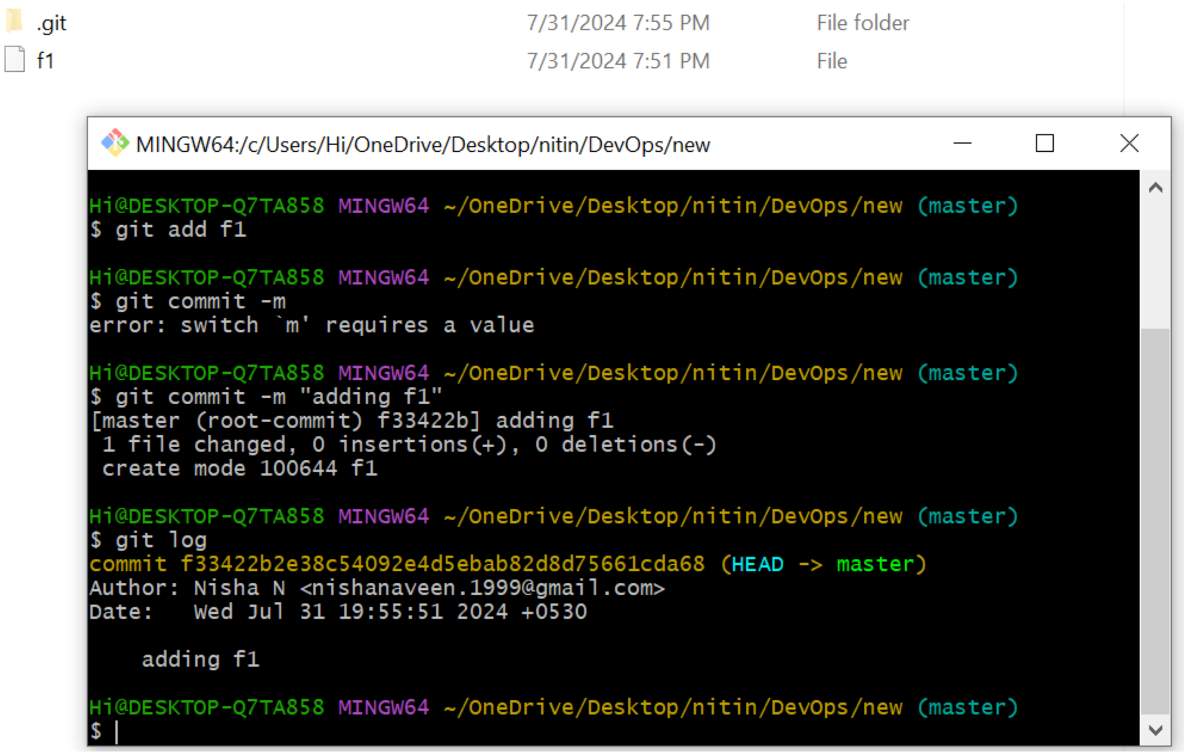
Preparing the file for changes using **git add**



.git	7/31/2024 7:55 PM	File folder
f1	7/31/2024 7:51 PM	File

```
Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new
$ git init
Initialized empty Git repository in C:/Users/Hi/OneDrive/Desktop/nitin/DevOps/ne
w/.git/

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ touch f1

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ git add f1

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ git commit -m
error: switch `m' requires a value

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ git commit -m "adding f1"
[master (root-commit) f33422b] adding f1
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 f1

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ |
```

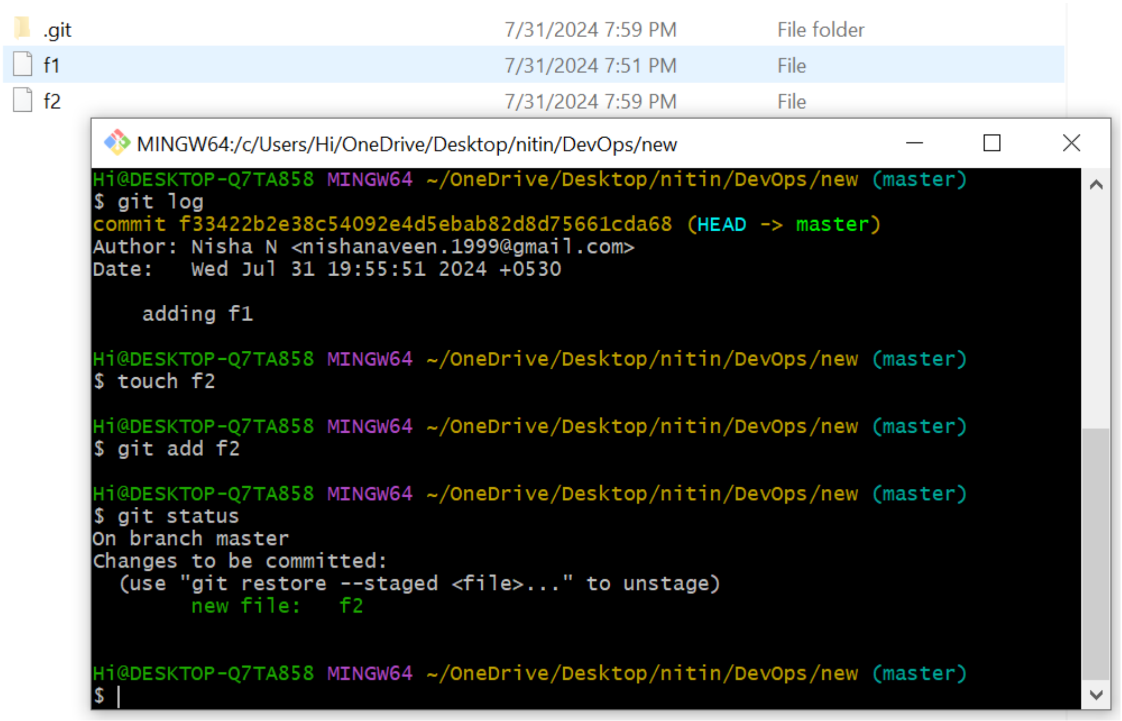
The changes saved is commited using **git commit –m**



.git	7/31/2024 7:55 PM	File folder
f1	7/31/2024 7:51 PM	File

```
Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ git add f1

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ git commit -m
error: switch `m' requires a value

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ git commit -m "adding f1"
[master (root-commit) f33422b] adding f1
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 f1

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ git log
commit f33422b2e38c54092e4d5ebab82d8d75661cda68 (HEAD -> master)
Author: Nisha N <nishanaveen.1999@gmail.com>
Date:   Wed Jul 31 19:55:51 2024 +0530

    adding f1

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ |
```

To check the history of the files and folder created we use **git log**

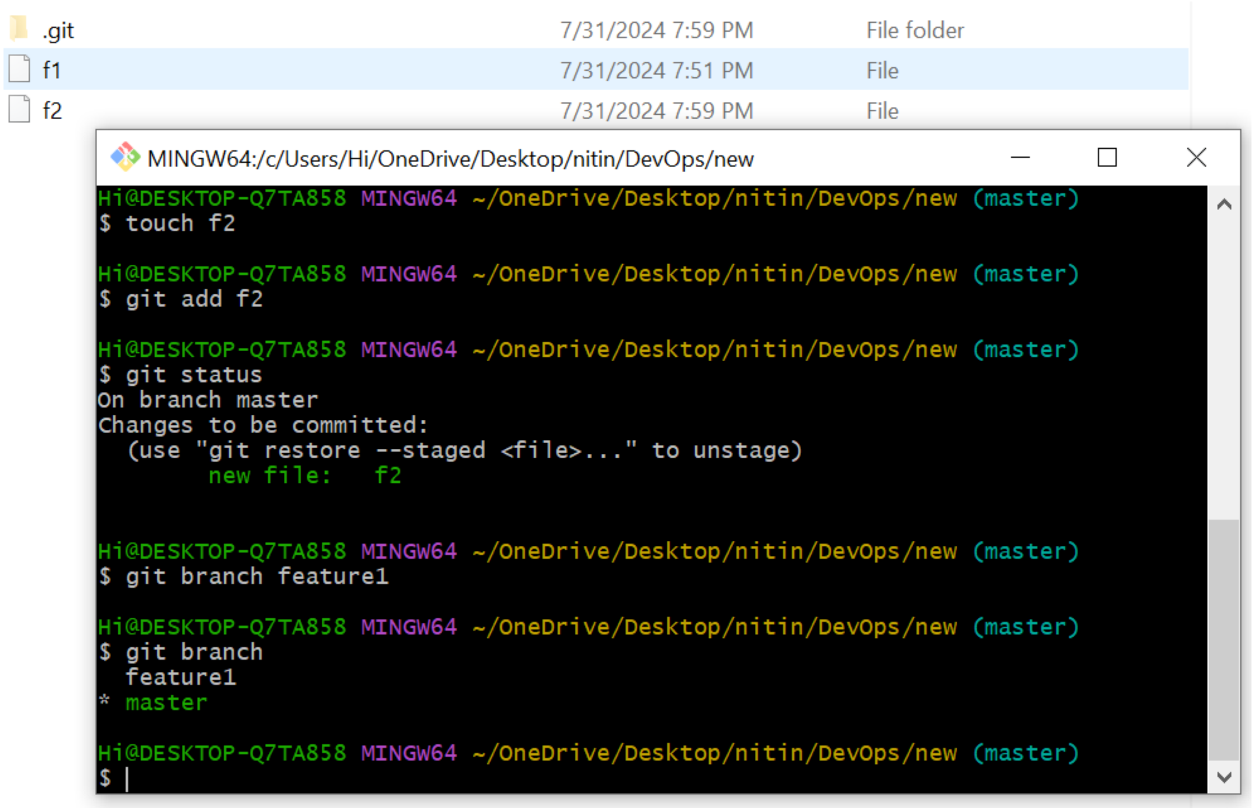


.git	7/31/2024 7:59 PM	File folder
f1	7/31/2024 7:51 PM	File
f2	7/31/2024 7:59 PM	File

MINGW64:/c/Users/Hi/OneDrive/Desktop/nitin/DevOps/new

```
Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ git log
commit f33422b2e38c54092e4d5ebab82d8d75661cda68 (HEAD -> master)
Author: Nisha N <nishanaveen.1999@gmail.com>
Date:   Wed Jul 31 19:55:51 2024 +0530

    adding f1

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ touch f2

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ git add f2

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   f2


Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ |
```

To check the status of the file whether it is staging area or working directory or repo we use **git status**

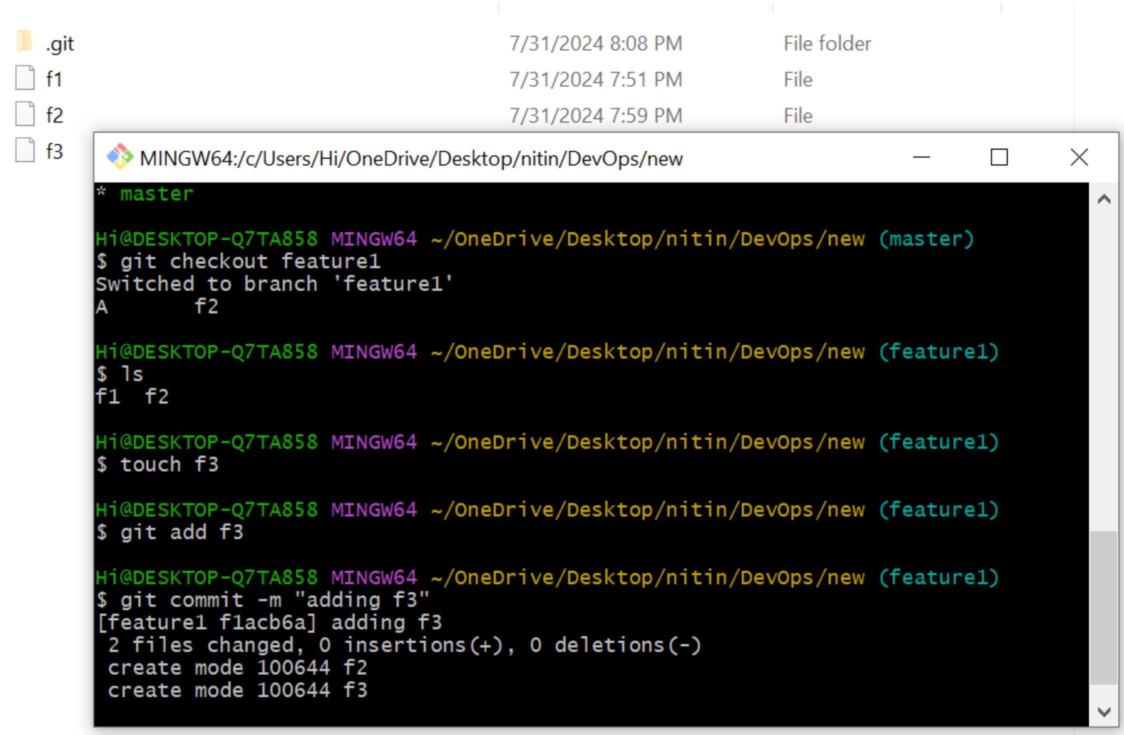


.git	7/31/2024 7:59 PM	File folder
f1	7/31/2024 7:51 PM	File
f2	7/31/2024 7:59 PM	File

MINGW64:/c/Users/Hi/OneDrive/Desktop/nitin/DevOps/new

```
Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ touch f2

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ git add f2

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   f2


Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ git branch feature1

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ git branch
  feature1
* master

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ |
```

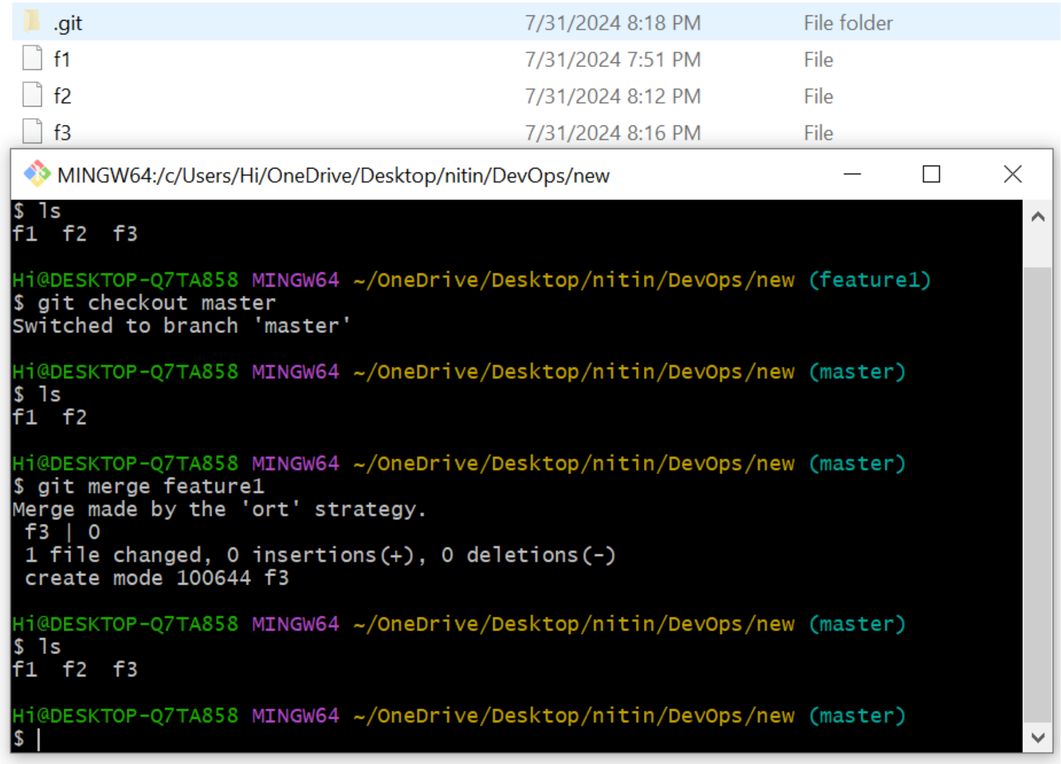
To create a branch we use **git branch branchname** which us used to work on the codes and by using **git branch** we can check the branches in the repository



.git	7/31/2024 8:08 PM	File folder
f1	7/31/2024 7:51 PM	File
f2	7/31/2024 7:59 PM	File
f3		

```
* master

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ git checkout feature1
Switched to branch 'feature1'
A       f2

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (feature1)
$ ls
f1  f2

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (feature1)
$ touch f3

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (feature1)
$ git add f3

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (feature1)
$ git commit -m "adding f3"
[feature1 f1acb6a] adding f3
 2 files changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 f2
 create mode 100644 f3
```

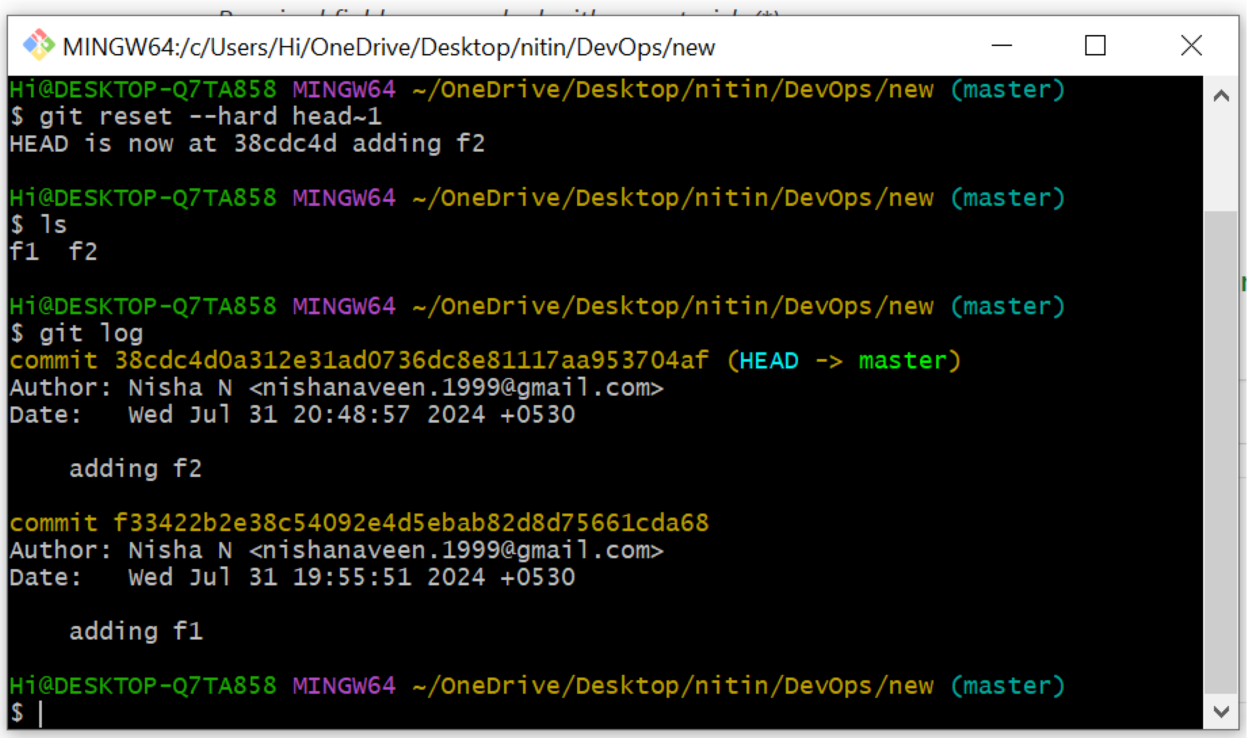
We can switch the branches using **git checkout branchname** and we have added a file f3 which shall be seen only in the feature



.git	7/31/2024 8:18 PM	File folder
f1	7/31/2024 7:51 PM	File
f2	7/31/2024 8:12 PM	File
f3	7/31/2024 8:16 PM	File

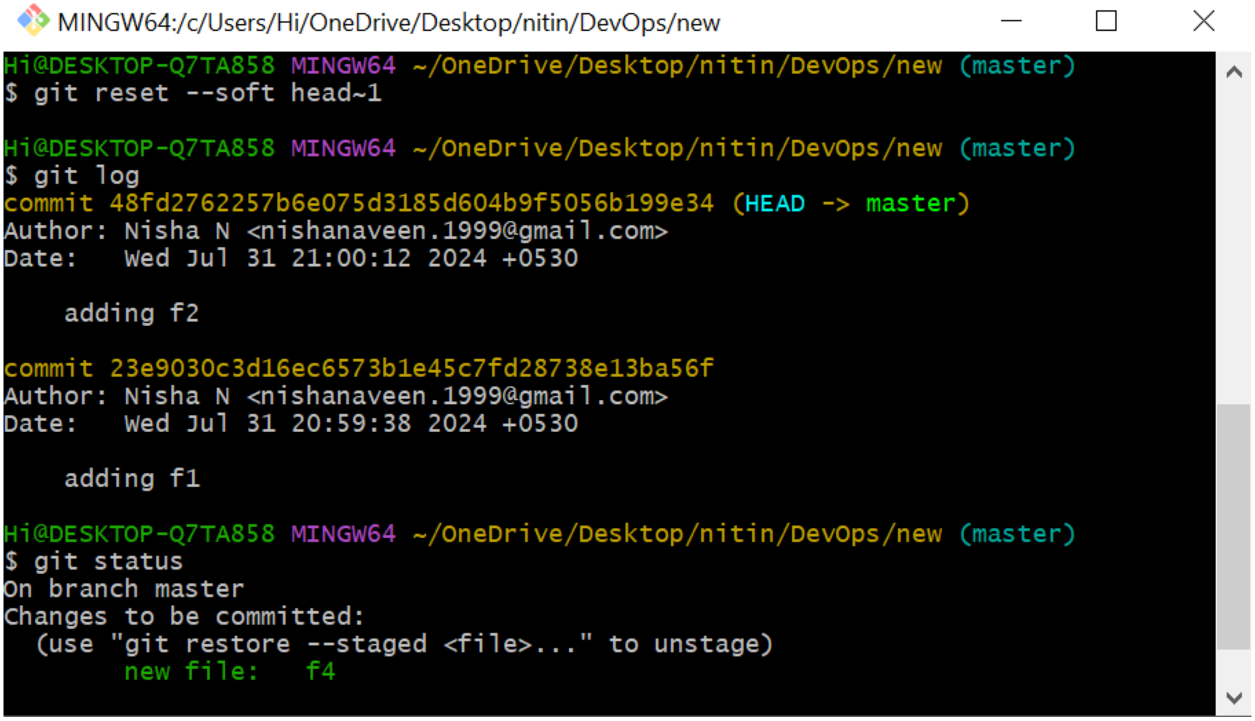
```
$ ls
f1  f2  f3

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (feature1)
$ git checkout master
Switched to branch 'master'

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ ls
f1  f2

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ git merge feature1
Merge made by the 'ort' strategy.
 f3 | 0
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 f3

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ ls
f1  f2  f3

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ |
```

The file, which is created in feature1 branch, can be merged into the master branch using **git merge branchname**



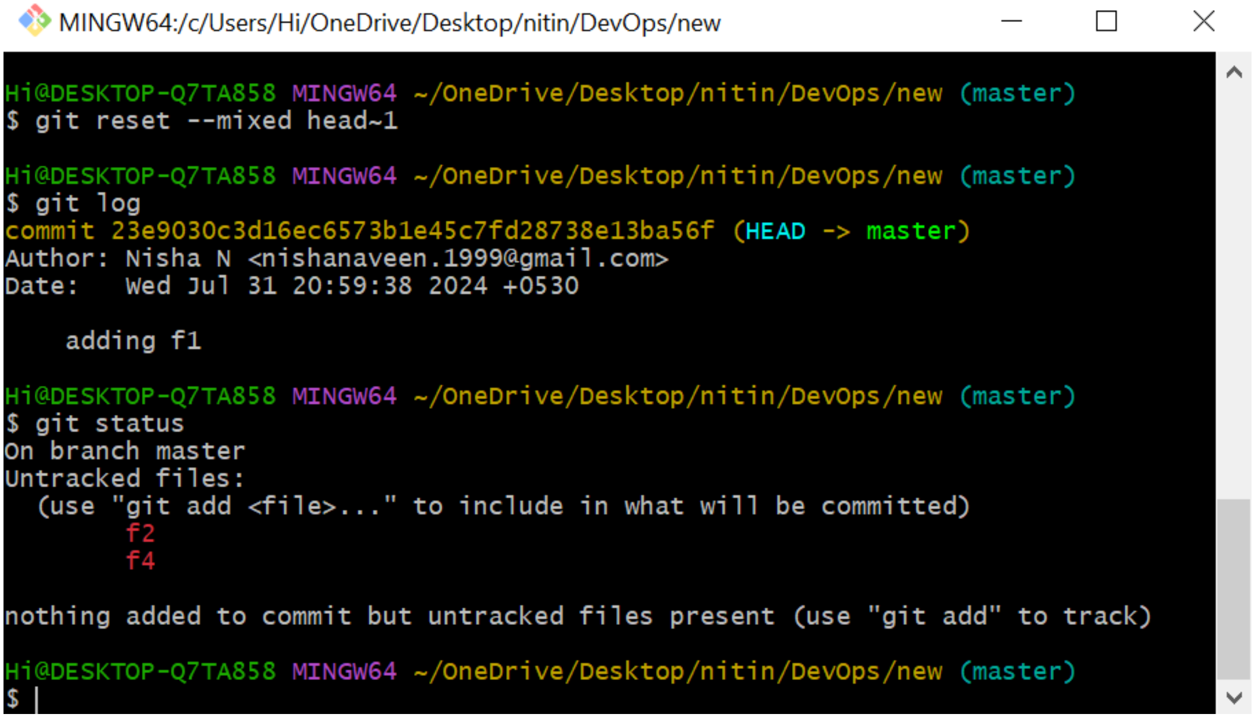
```
Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ git reset --hard head~1
HEAD is now at 38cdc4d adding f2

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ ls
f1  f2

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ git log
commit 38cdc4d0a312e31ad0736dc8e81117aa953704af (HEAD -> master)
Author: Nisha N <nishanaveen.1999@gmail.com>
Date:   Wed Jul 31 20:48:57 2024 +0530

    adding f2

commit f33422b2e38c54092e4d5ebab82d8d75661cda68
Author: Nisha N <nishanaveen.1999@gmail.com>
Date:   Wed Jul 31 19:55:51 2024 +0530

    adding f1

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ |
```

To remove the file we can use **git reset --hard head~n** , in the above case f3 file is removed



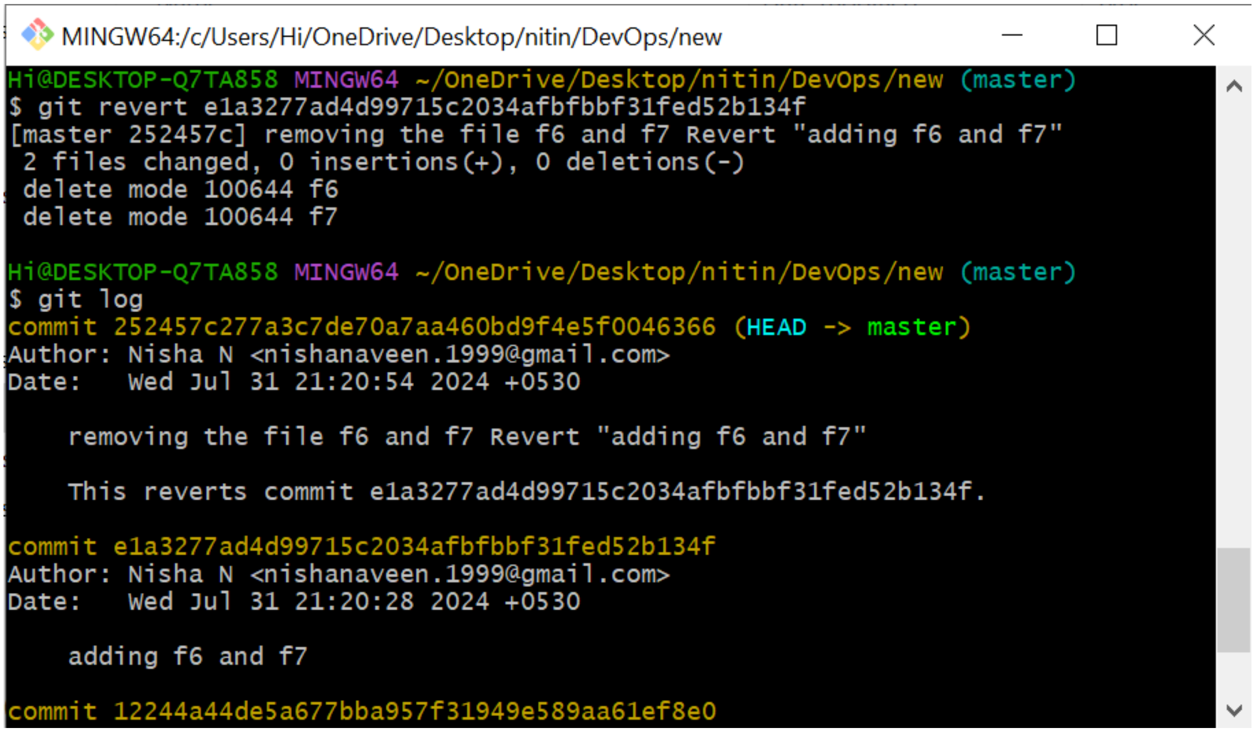
```
Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ git reset --soft head~1

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ git log
commit 48fd2762257b6e075d3185d604b9f5056b199e34 (HEAD -> master)
Author: Nisha N <nishanaveen.1999@gmail.com>
Date:   Wed Jul 31 21:00:12 2024 +0530

    adding f2

commit 23e9030c3d16ec6573b1e45c7fd28738e13ba56f
Author: Nisha N <nishanaveen.1999@gmail.com>
Date:   Wed Jul 31 20:59:38 2024 +0530

    adding f1

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   f4
```

To move the file to staging area we can use **git reset --soft head~n** , in the above case f4 file is moved to staging area



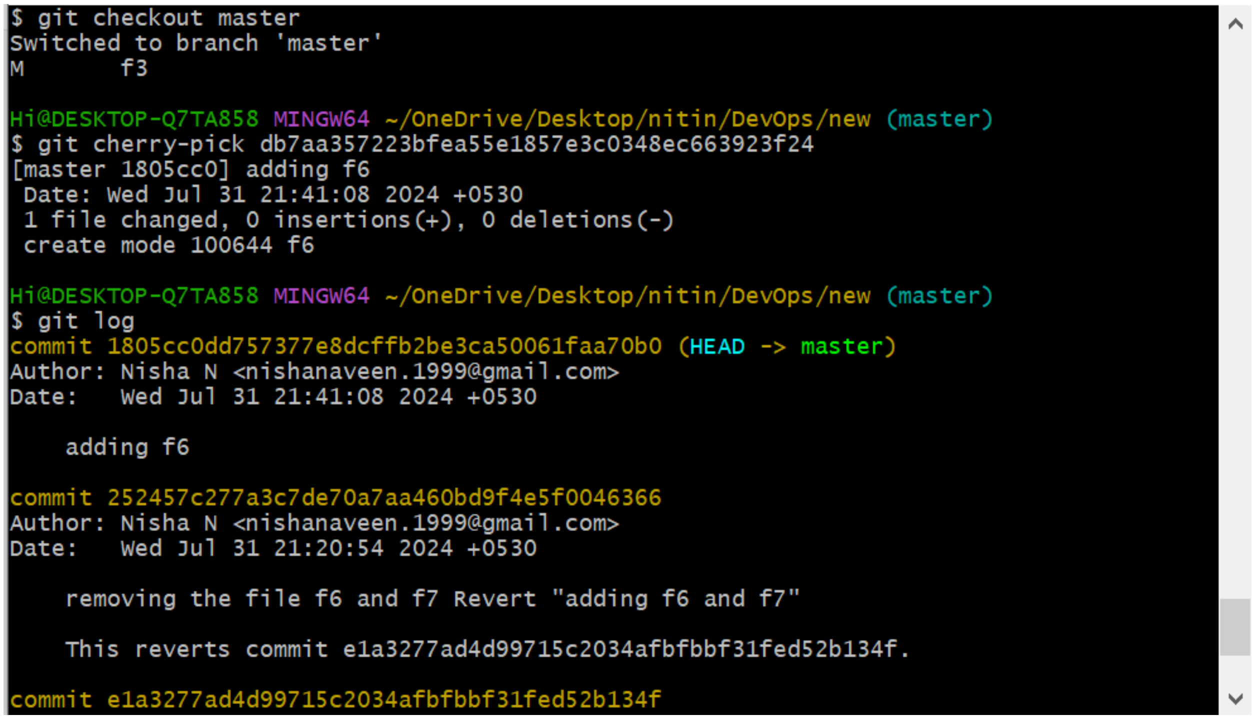
```
Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ git reset --mixed head~1

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ git log
commit 23e9030c3d16ec6573b1e45c7fd28738e13ba56f (HEAD -> master)
Author: Nisha N <nishanaveen.1999@gmail.com>
Date:   Wed Jul 31 20:59:38 2024 +0530

    adding f1

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        f2
        f4

nothing added to commit but untracked files present (use "git add" to track)

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ |
```

To move the file to working directory we can use **git reset --mixed head~n** , in the above case f2 and f4 file is moved to working space



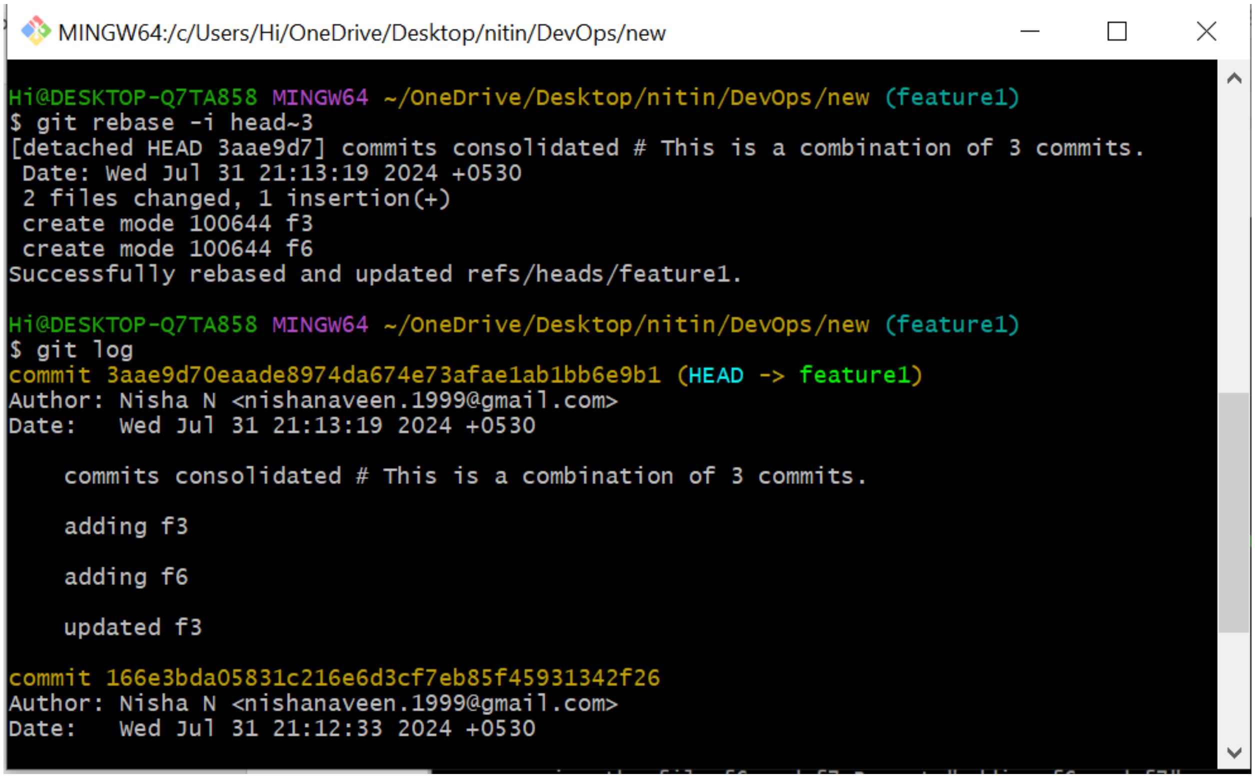
```
Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ git revert e1a3277ad4d99715c2034afbfbbf31fed52b134f
[master 252457c] removing the file f6 and f7 Revert "adding f6 and f7"
 2 files changed, 0 insertions(+), 0 deletions(-)
 delete mode 100644 f6
 delete mode 100644 f7

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ git log
commit 252457c277a3c7de70a7aa460bd9f4e5f0046366 (HEAD -> master)
Author: Nisha N <nishanaveen.1999@gmail.com>
Date:   Wed Jul 31 21:20:54 2024 +0530

    removing the file f6 and f7 Revert "adding f6 and f7"

    This reverts commit e1a3277ad4d99715c2034afbfbbf31fed52b134f.

commit e1a3277ad4d99715c2034afbfbbf31fed52b134f
Author: Nisha N <nishanaveen.1999@gmail.com>
Date:   Wed Jul 31 21:20:28 2024 +0530

    adding f6 and f7

commit 12244a44de5a677bba957f31949e589aa61ef8e0
```

To revert the the files we can use **git revert commit id** to go back to the previous commit.



```
$ git checkout master
Switched to branch 'master'
M       f3

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ git cherry-pick db7aa357223bfea55e1857e3c0348ec663923f24
[master 1805cc0] adding f6
 Date: Wed Jul 31 21:41:08 2024 +0530
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 f6

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ git log
commit 1805cc0dd757377e8dcffb2be3ca50061faa70b0 (HEAD -> master)
Author: Nisha N <nishanaveen.1999@gmail.com>
Date:   Wed Jul 31 21:41:08 2024 +0530

    adding f6

commit 252457c277a3c7de70a7aa460bd9f4e5f0046366
Author: Nisha N <nishanaveen.1999@gmail.com>
Date:   Wed Jul 31 21:20:54 2024 +0530

    removing the file f6 and f7 Revert "adding f6 and f7"

    This reverts commit e1a3277ad4d99715c2034afbfbbf31fed52b134f.

commit e1a3277ad4d99715c2034afbfbbf31fed52b134f
```

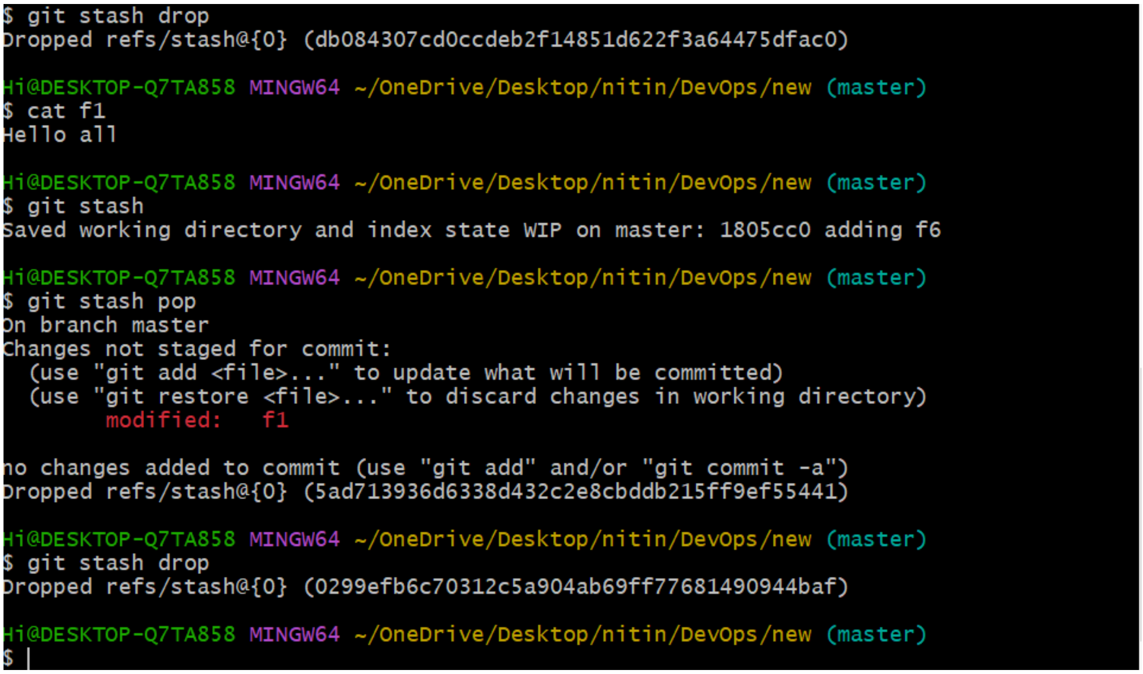
Copying a commit i.e a file from one branch to another using **git cherry-pick commit id**



MINGW64:/c/Users/Hi/OneDrive/Desktop/nitin/DevOps/new

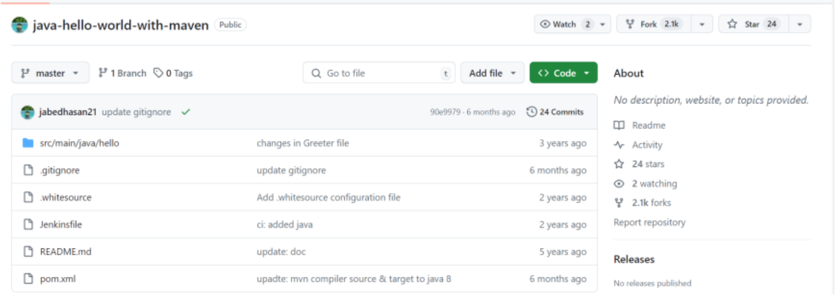
```
Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (feature1)
$ git rebase -i head~3
[detached HEAD 3aae9d7] commits consolidated # This is a combination of 3 commits.
 Date: Wed Jul 31 21:13:19 2024 +0530
 2 files changed, 1 insertion(+)
 create mode 100644 f3
 create mode 100644 f6
Successfully rebased and updated refs/heads/feature1.

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (feature1)
$ git log
commit 3aae9d70eaade8974da674e73afae1ab1bb6e9b1 (HEAD -> feature1)
Author: Nisha N <nishanaveen.1999@gmail.com>
Date:   Wed Jul 31 21:13:19 2024 +0530

    commits consolidated # This is a combination of 3 commits.

    adding f3

    adding f6

    updated f3

commit 166e3bda05831c216e6d3cf7eb85f45931342f26
Author: Nisha N <nishanaveen.1999@gmail.com>
Date:   Wed Jul 31 21:12:33 2024 +0530
```

We can consolidate the number of commits into one using **git rebase-i head~n** by selecting the commits in editor mode and squashing them.



```
$ git stash drop
Dropped refs/stash@{0} (db084307cd0ccdeb2f14851d622f3a64475dfac0)

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ cat f1
Hello all

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ git stash
Saved working directory and index state WIP on master: 1805cc0 adding f6

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ git stash pop
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   f1

no changes added to commit (use "git add" and/or "git commit -a")
Dropped refs/stash@{0} (5ad713936d6338d432c2e8cbddb215ff9ef55441)

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ git stash drop
Dropped refs/stash@{0} (0299efb6c70312c5a904ab69ff77681490944baf)

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ |
```

We can store the content of the file by using **git stash** without committing , **git stash pop** will give the information of the file and **git stash drop** will the content from the file.



jabedhasan21 update gitignore ✓ 90e9979 · 6 months ago 24 Commits

src/main/java/hello	changes in Greeter file	3 years ago
.gitignore	update gitignore	6 months ago
.whitesource	Add .whitesource configuration file	2 years ago
Jenkinsfile	ci: added java	2 years ago
README.md	update: doc	5 years ago
pom.xml	upadte: mvn compiler source & target to java 8	6 months ago

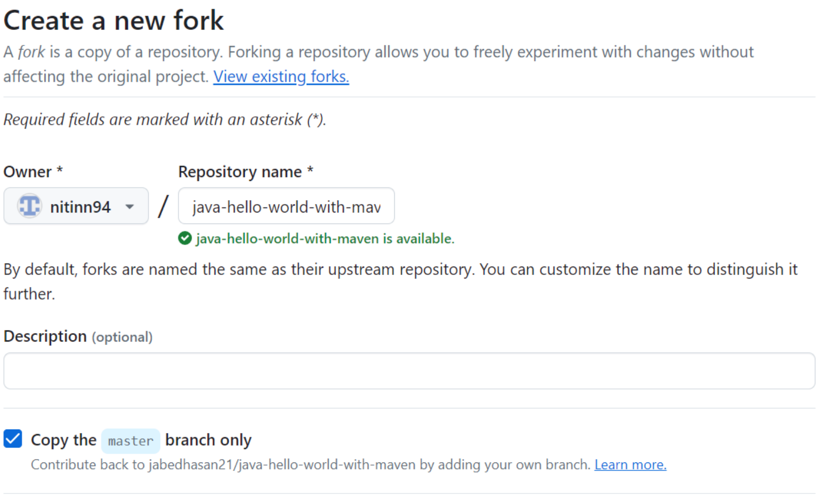
About

No description, website, or topics provided.

Readme
Activity
24 stars
2 watching
2.1k forks
Report repository

Releases

No releases published



Create a new fork

A *fork* is a copy of a repository. Forking a repository allows you to freely experiment with changes without affecting the original project. View existing forks.

Required fields are marked with an asterisk (*).

Owner * / Repository name *

nitinn94 / java-hello-world-with-mav

✓ java-hello-world-with-maven is available.

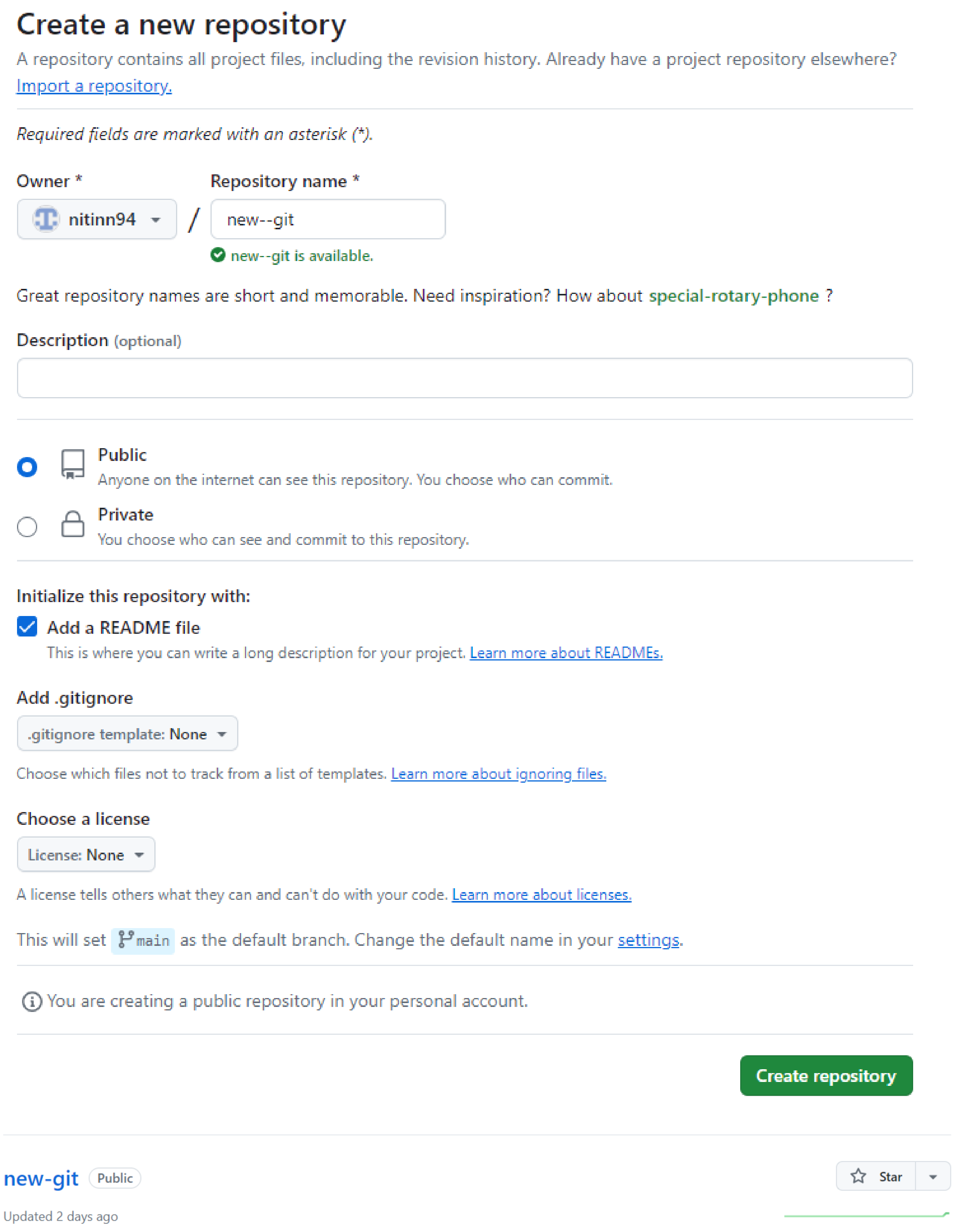
By default, forks are named the same as their upstream repository. You can customize the name to distinguish it further.

Description (optional)

☑ Copy the `master` branch only

Contribute back to jabedhasan21/java-hello-world-with-maven by adding your own branch. Learn more.

In the above we have used **git fork** to have a copy of the repository to work on a project



Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? Import a repository.

Required fields are marked with an asterisk (*).

Owner * nitinn94 / Repository name * new--git

✓ new--git is available.

Great repository names are short and memorable. Need inspiration? How about special-rotary-phone ?

Description (optional)

● Public
Anyone on the internet can see this repository. You choose who can commit.

○ Private
You choose who can see and commit to this repository.

Initialize this repository with:

☑ Add a README file
This is where you can write a long description for your project. Learn more about READMEs.

Add .gitignore

.gitignore template: None

Choose which files not to track from a list of templates. Learn more about ignoring files.

Choose a license

License: None

A license tells others what they can and can't do with your code. Learn more about licenses.

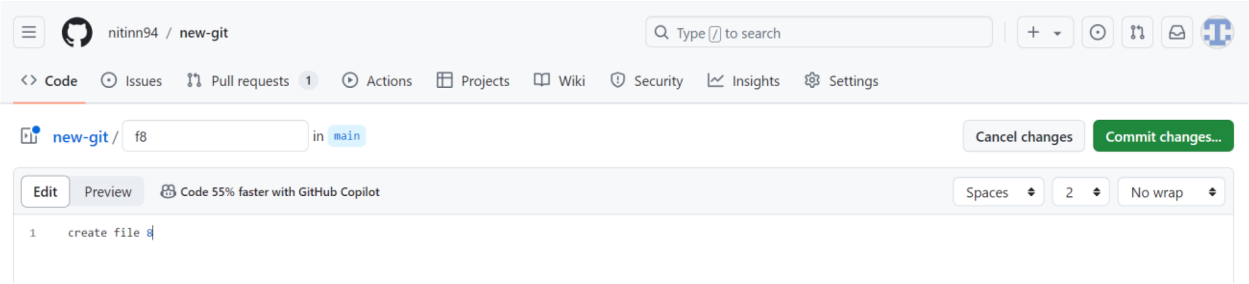
This will set `main` as the default branch. Change the default name in your settings.

ⓘ You are creating a public repository in your personal account.

new-git Public

Updated 2 days ago

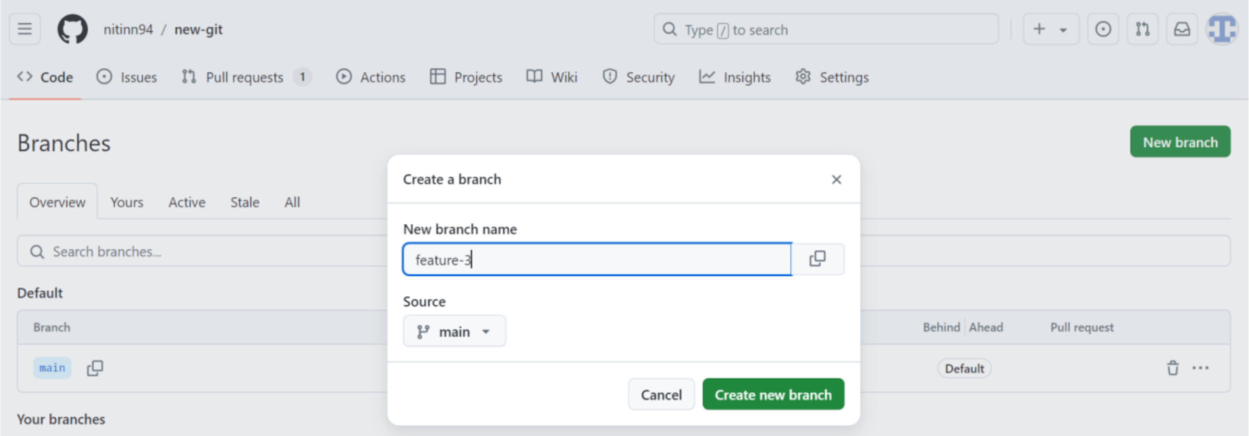
Creating a new repository in the **Git hub**



new-git / f8 in main

```
create file 8
```

Creating a file in the githib i.e file f8

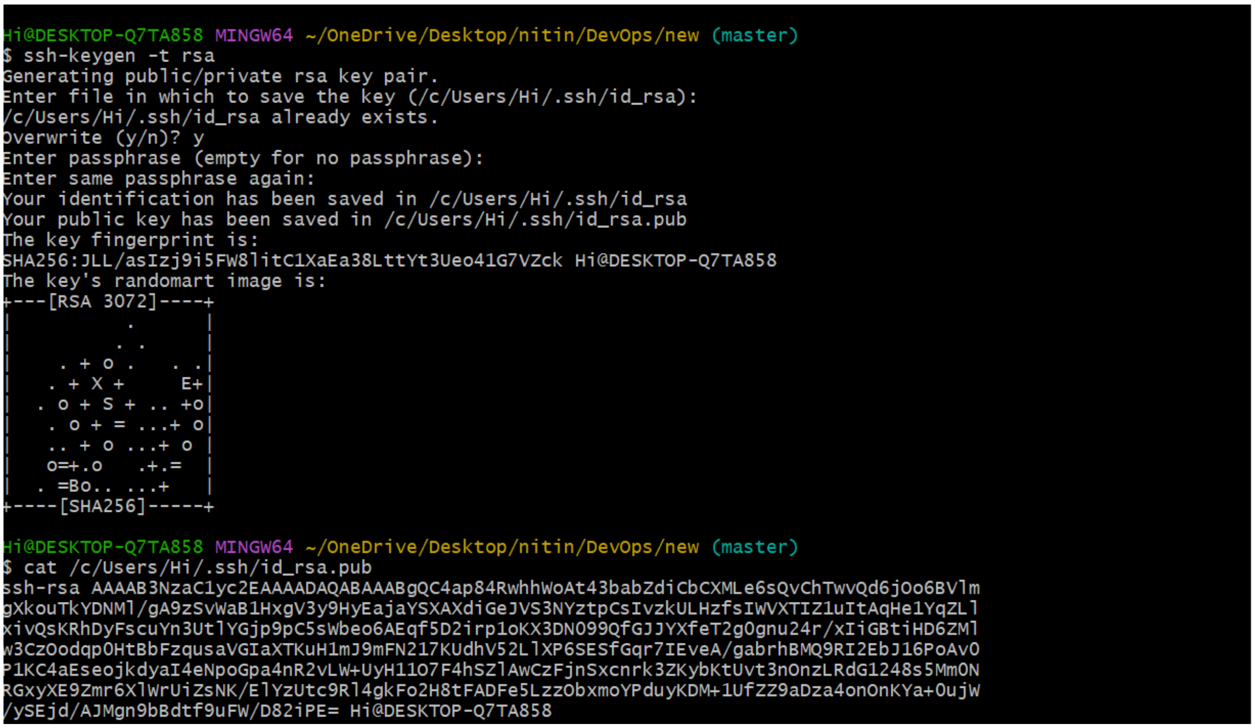


Create a branch

New branch name: feature-3

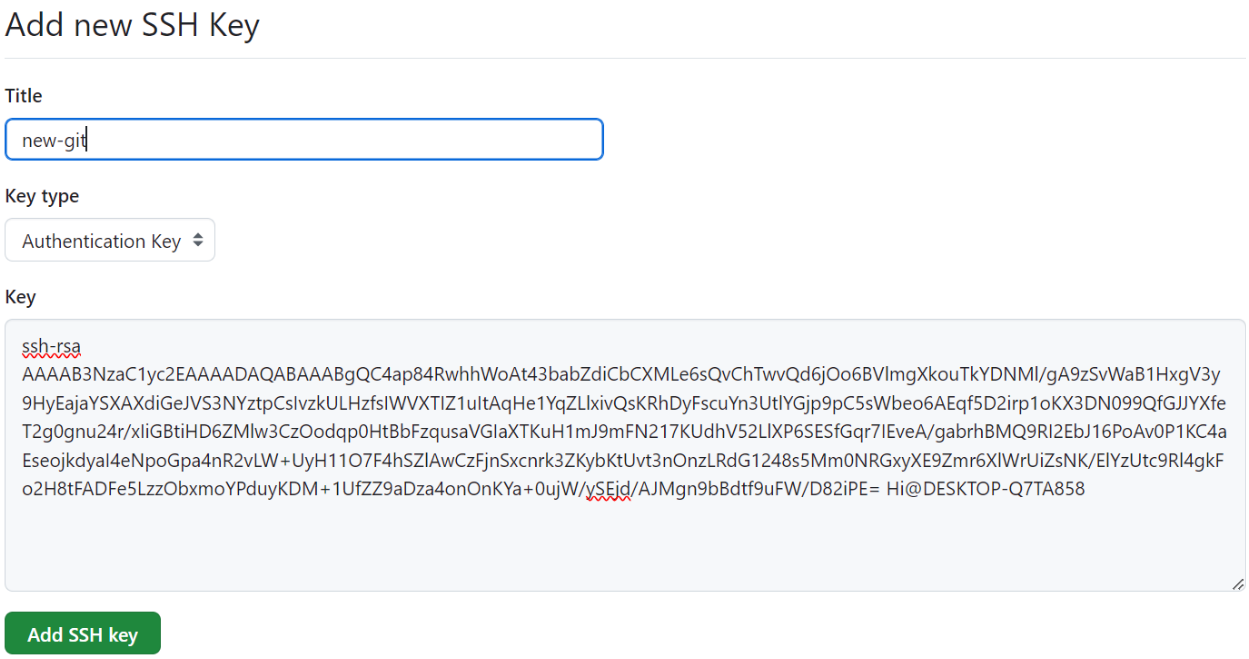
Source: main

Creating a branch in githib i.e feature-3



```
Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/c/Users/Hi/.ssh/id_rsa):
/c/Users/Hi/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /c/Users/Hi/.ssh/id_rsa
Your public key has been saved in /c/Users/Hi/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:JLL/asIzj9i5FW8litC1XaEa38LttYt3Ueo41G7VZck Hi@DESKTOP-Q7TA858
The key's randomart image is:
+---[RSA 3072]----+
|          .      |
|         . .     |
|    . + o .   . .|
|   . + X +     E+|
|  . o + S + .. +o|
|   . o + = ...+ o|
|   .. + o ...+ o |
|   o=+.o   .+.=  |
|  . =Bo.. ...+   |
+----[SHA256]-----+

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new (master)
$ cat /c/Users/Hi/.ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQC4ap84RwhhWoAt43babZdiCbCXMLe6sQvChTwvQd6jOo6BVlm
gXkouTkYDNMl/gA9zSvWaB1HxgV3y9HyEajaYSXAXdiGeJVS3NYztpCsIvzkULHzfsIWVXTIZ1uItAqHe1YqZLl
xivQsKRhDyFscuYn3UtlYGjp9pC5sWbeo6AEqf5D2irp1oKX3DN099QfGJJYXfeT2g0gnu24r/xIiGBtiHD6ZMl
w3CzOodqp0HtBbFzqusaVGIaXTKuH1mJ9mFN217KUdhV52LlXP6SESfGqr7IEveA/gabrhBMQ9RI2EbJ16PoAvO
P1KC4aEseojkdyaI4eNpoGpa4nR2vLW+UyH1107F4hSZlAwCzFjnSxcnrk3ZKybKtUvt3nOnzLRdG1248s5MmON
RGxyXE9Zmr6XlWrUiZsNK/ElYzUtc9Rl4gkFo2H8tFADFe5LzzObxmoYPduyKDM+1UfZZ9aDza4onOnKYa+OujW
/ySEjd/AJMgn9bBdtf9uFW/D82iPE= Hi@DESKTOP-Q7TA858
```

To establish a connection between local repository and central repository we use **ssh-keygen –t rsa** to generate public key.



Add new SSH Key

Title

new-git

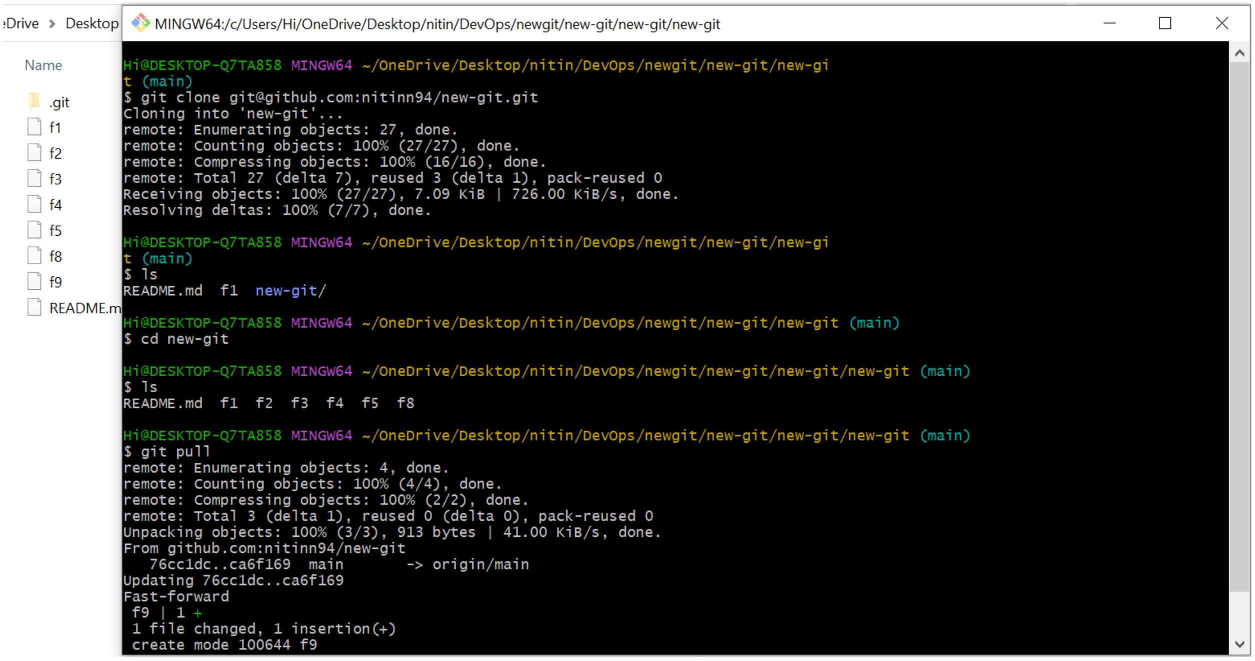
Key type

Authentication Key

Key

```
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABgQC4ap84RwhhWoAt43babZdiCbCXMLe6sQvChTwvQd6jOo6BVlmgXkouTkYDNMl/gA9zSvWaB1HxgV3y
9HyEajaYSXAXdiGeJVS3NYztpCslvzkULHzfsIWVXTIZ1ultAqHe1YqZLlxivQsKRhDyFscuYn3UtlYGjp9pC5sWbeo6AEqf5D2irp1oKX3DN099QfGJJYXfe
T2g0gnu24r/xliGBtiHD6ZMlw3CzOodqp0HtBbFzqusaVGlaXTKuH1mJ9mFN217KUdhV52LlXP6SESfGqr7lEveA/gabrhBMQ9Rl2EbJ16PoAv0P1KC4a
Eseojkdyal4eNpoGpa4nR2vLW+UyH11O7F4hSZlAwCzFjnSxcnrk3ZKybKtUvt3nOnzLRdG1248s5Mm0NRGxyXE9Zmr6XlWrUiZsNK/ElYzUtc9Rl4gkF
o2H8tFADFe5LzzObxmoYPduyKDM+1UfZZ9aDza4onOnKYa+0ujW/ySEjd/AJMgn9bBdtf9uFW/D82iPE= Hi@DESKTOP-Q7TA858
```

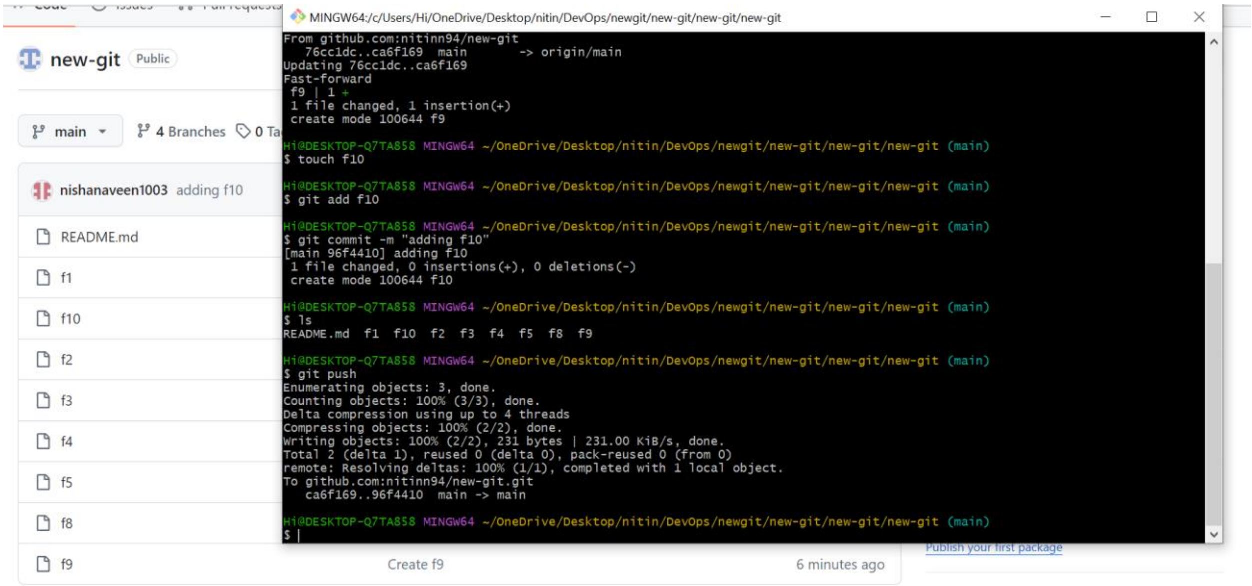
To add the public key into the git hub to access the local repository and central repository.



MINGW64:/c/Users/Hi/OneDrive/Desktop/nitin/DevOps/newgit/new-git/new-git/new-git

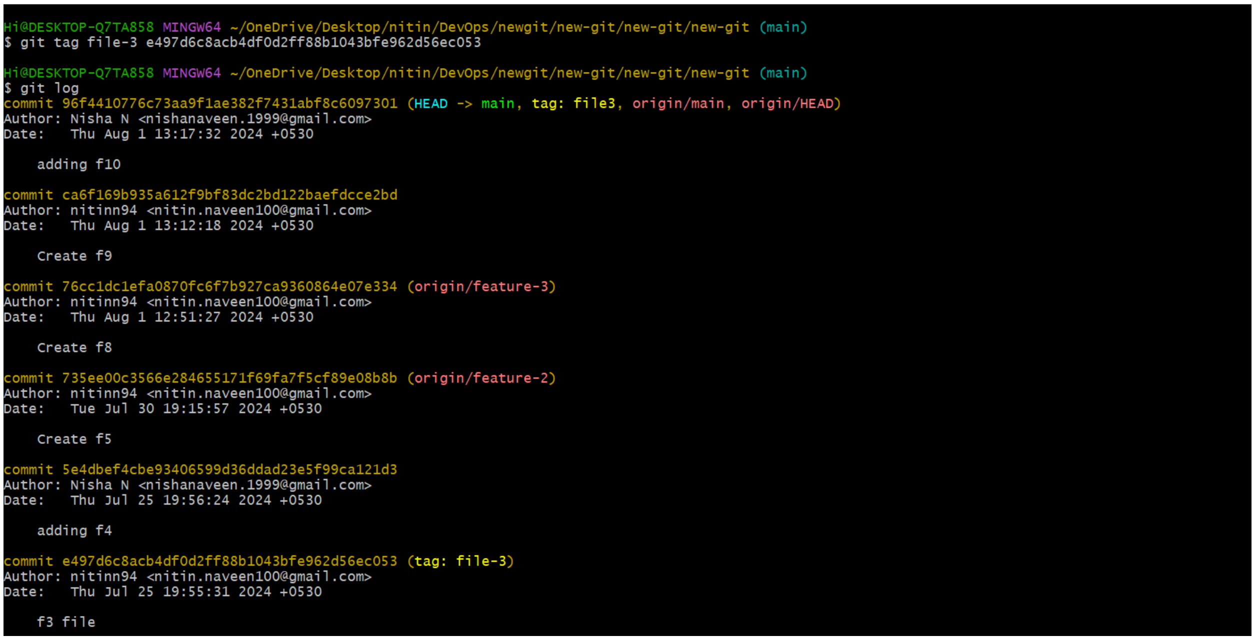
```
Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/newgit/new-git/new-gi
t (main)
$ git clone git@github.com:nitinn94/new-git.git
Cloning into 'new-git'...
remote: Enumerating objects: 27, done.
remote: Counting objects: 100% (27/27), done.
remote: Compressing objects: 100% (16/16), done.
remote: Total 27 (delta 7), reused 3 (delta 1), pack-reused 0
Receiving objects: 100% (27/27), 7.09 KiB | 726.00 KiB/s, done.
Resolving deltas: 100% (7/7), done.

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/newgit/new-git/new-gi
t (main)
$ ls
README.md  f1  new-git/

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/newgit/new-git/new-git (main)
$ cd new-git

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/newgit/new-git/new-git/new-git (main)
$ ls
README.md  f1  f2  f3  f4  f5  f8

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/newgit/new-git/new-git/new-git (main)
$ git pull
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 913 bytes | 41.00 KiB/s, done.
From github.com:nitinn94/new-git
   76cc1dc..ca6f169  main       -> origin/main
Updating 76cc1dc..ca6f169
Fast-forward
 f9 | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 f9
```

Using **git clone** and ssh url we have can access local and remote repository . We have added a file f9 in remote repository and have used git pull to pull the file into the local repository


new-git
Public
main
4 Branches
0 Ta
nishanaveen1003 adding f10
README.md
f1
f10
f2
f3
f4
f5
f8
f9
Create f9
6 minutes ago
Publish your first package
MINGW64:/c/Users/Hi/OneDrive/Desktop/nitin/DevOps/newgit/new-git/new-git/new-git

```
From github.com:nitinn94/new-git
   76cc1dc..ca6f169  main       -> origin/main
Updating 76cc1dc..ca6f169
Fast-forward
 f9 | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 f9

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/newgit/new-git/new-git/new-git (main)
$ touch f10

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/newgit/new-git/new-git/new-git (main)
$ git add f10

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/newgit/new-git/new-git/new-git (main)
$ git commit -m "adding f10"
[main 96f4410] adding f10
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 f10

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/newgit/new-git/new-git/new-git (main)
$ ls
README.md  f1  f10  f2  f3  f4  f5  f8  f9

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/newgit/new-git/new-git/new-git (main)
$ git push
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 231 bytes | 231.00 KiB/s, done.
Total 2 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To github.com:nitinn94/new-git.git
   ca6f169..96f4410  main -> main

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/newgit/new-git/new-git/new-git (main)
$ |
```

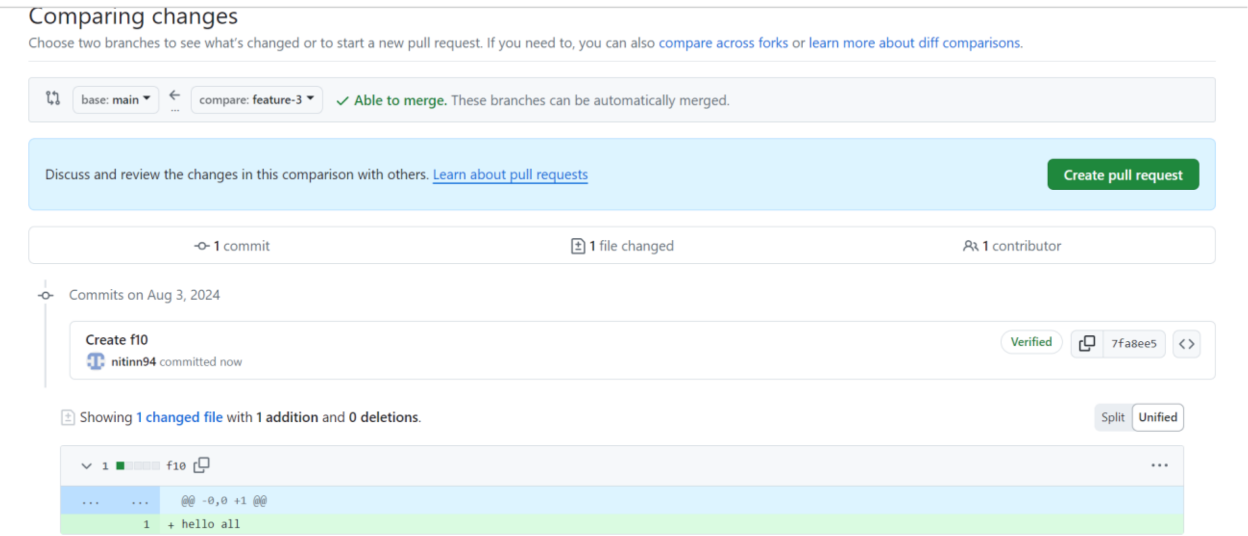
Using **git push** I have pushed the file from the local repository to remote repository. i.e f10 is moved to the remote repository



```
Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/newgit/new-git/new-git/new-git (main)
$ git tag file-3 e497d6c8acb4df0d2ff88b1043bfe962d56ec053

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/newgit/new-git/new-git/new-git (main)
$ git log
commit 96f4410776c73aa9f1ae382f7431abf8c6097301 (HEAD -> main, tag: file3, origin/main, origin/HEAD)
Author: Nisha N <nishanaveen.1999@gmail.com>
Date:   Thu Aug 1 13:17:32 2024 +0530

    adding f10

commit ca6f169b935a612f9bf83dc2bd122baefdcce2bd
Author: nitinn94 <nitin.naveen100@gmail.com>
Date:   Thu Aug 1 13:12:18 2024 +0530

    Create f9

commit 76cc1dc1efa0870fc6f7b927ca9360864e07e334 (origin/feature-3)
Author: nitinn94 <nitin.naveen100@gmail.com>
Date:   Thu Aug 1 12:51:27 2024 +0530

    Create f8

commit 735ee00c3566e284655171f69fa7f5cf89e08b8b (origin/feature-2)
Author: nitinn94 <nitin.naveen100@gmail.com>
Date:   Tue Jul 30 19:15:57 2024 +0530

    Create f5

commit 5e4dbef4cbe93406599d36ddad23e5f99ca121d3
Author: Nisha N <nishanaveen.1999@gmail.com>
Date:   Thu Jul 25 19:56:24 2024 +0530

    adding f4

commit e497d6c8acb4df0d2ff88b1043bfe962d56ec053 (tag: file-3)
Author: nitinn94 <nitin.naveen100@gmail.com>
Date:   Thu Jul 25 19:55:31 2024 +0530

    f3 file
```

Using **git tag tagname commit id** I have assigned a name to the f3 as file-3.

Pull request



Comparing changes

Choose two branches to see what's changed or to start a new pull request. If you need to, you can also compare across forks or learn more about diff comparisons.

base: main ← compare: feature-3 ✓ Able to merge. These branches can be automatically merged.

Discuss and review the changes in this comparison with others. Learn about pull requests

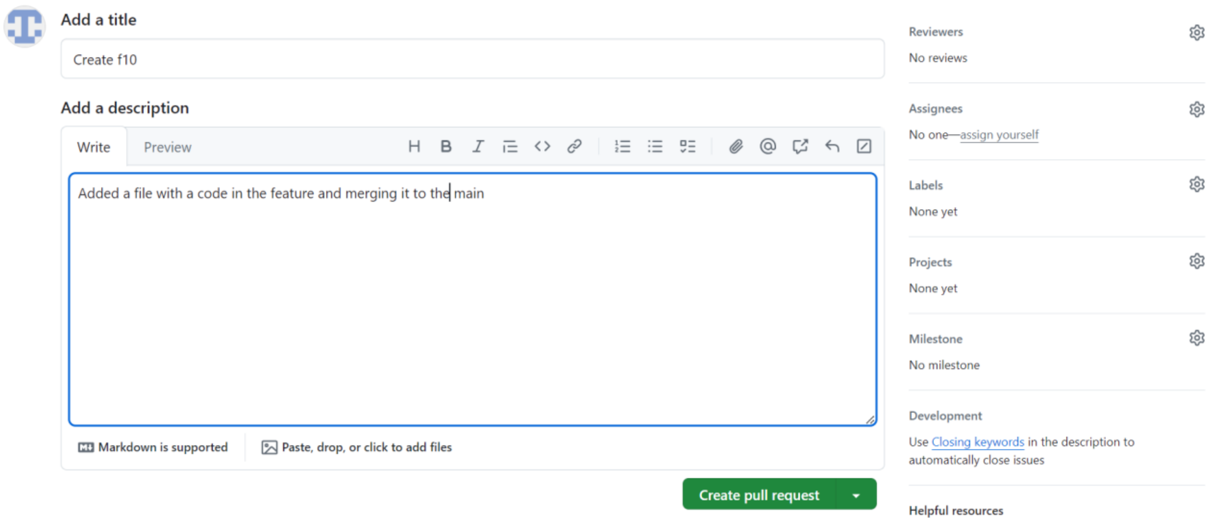
Commits on Aug 3, 2024

Create f10
nitinn94 committed now

Showing 1 changed file with 1 addition and 0 deletions.

```
1 f10
...  ...  @@ -0,0 +1 @@
       1  + hello all
```

Created a file 10 in the feature 3 and will create a pull request to merge it to the main branch



Add a title

Create f10

Add a description

Added a file with a code in the feature and merging it to the main

Reviewers
No reviews

Assignees
No one—assign yourself

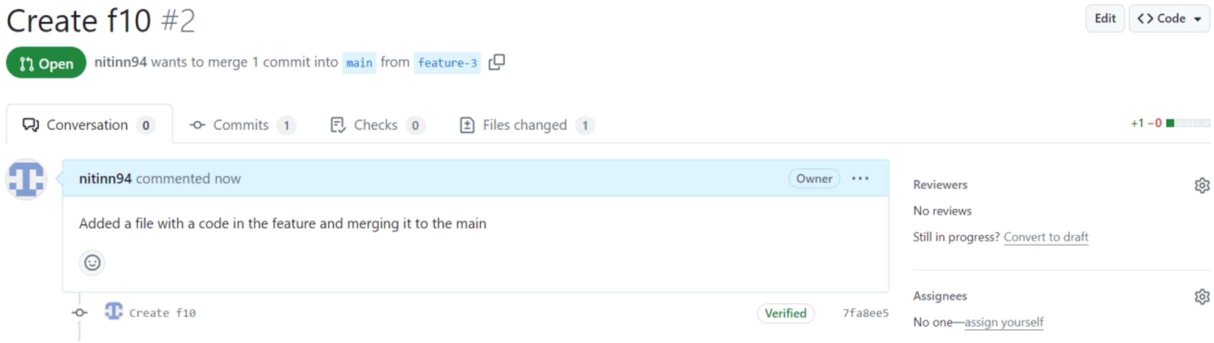
Labels
None yet

Projects
None yet

Milestone
No milestone

Development
Use Closing keywords in the description to automatically close issues

Helpful resources



Create f10 #2

Open nitinn94 wants to merge 1 commit into `main` from `feature-3`

nitinn94 commented now

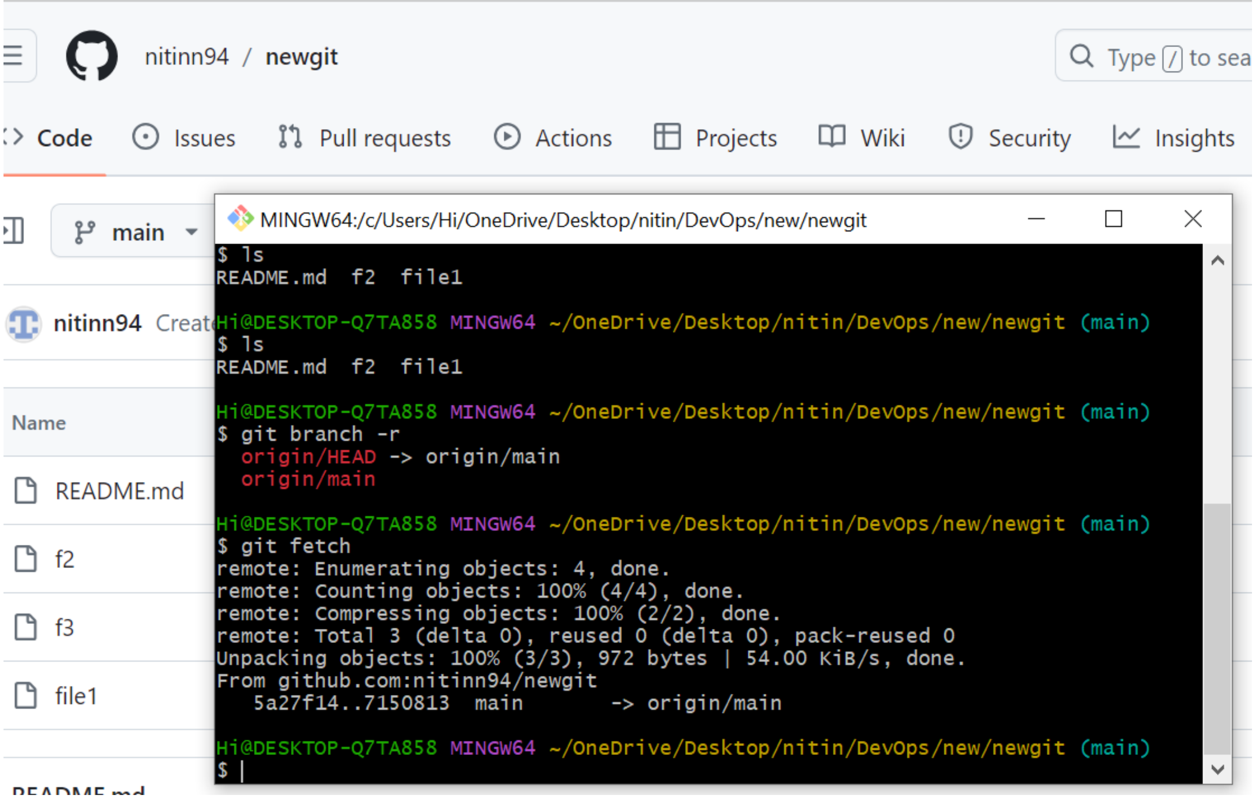
Added a file with a code in the feature and merging it to the main

Create f10

Reviewers
No reviews
Still in progress? Convert to draft

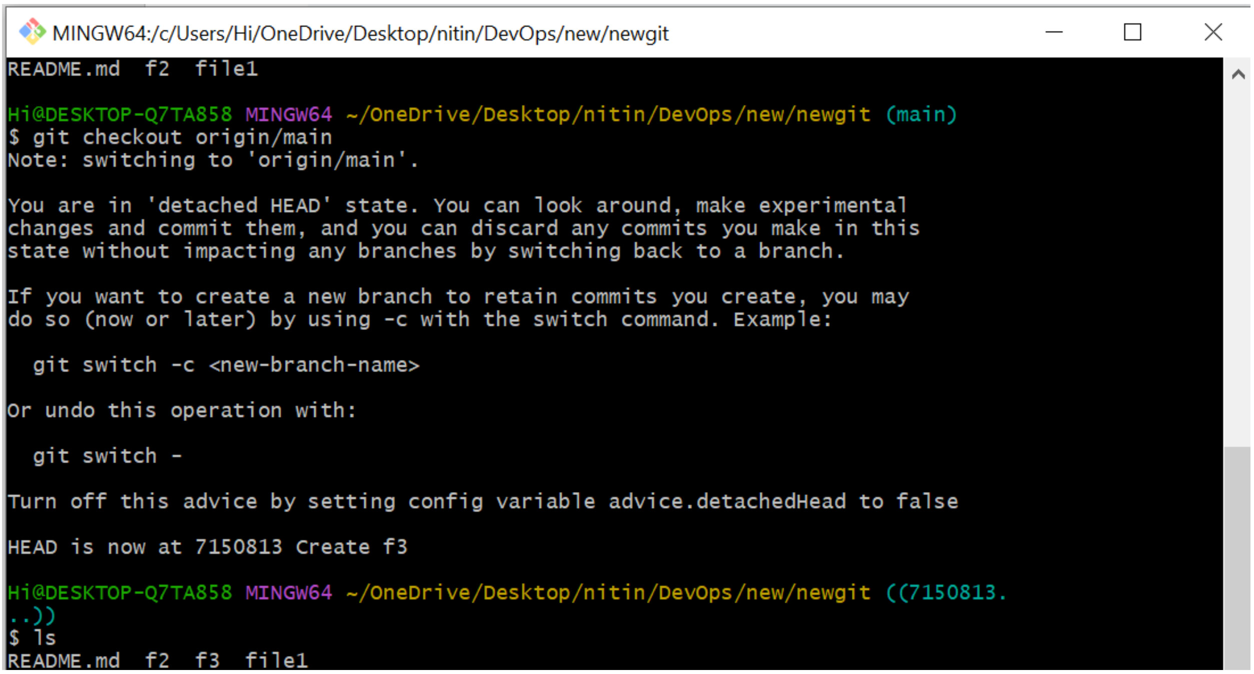
Assignees
No one—assign yourself

Created a pull request which has to approved by the organization.



```
$ ls
README.md  f2  file1

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new/newgit (main)
$ ls
README.md  f2  file1

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new/newgit (main)
$ git branch -r
  origin/HEAD -> origin/main
  origin/main

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new/newgit (main)
$ git fetch
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 972 bytes | 54.00 KiB/s, done.
From github.com:nitinn94/newgit
   5a27f14..7150813  main       -> origin/main

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new/newgit (main)
$ |
```

We have created file f3 in the central repository; using **Git fetch** we are trying to copy the changes into the specified branch.


MINGW64:/c/Users/Hi/OneDrive/Desktop/nitin/DevOps/new/newgit

```
README.md  f2  file1

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new/newgit (main)
$ git checkout origin/main
Note: switching to 'origin/main'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by switching back to a branch.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -c with the switch command. Example:

  git switch -c <new-branch-name>

Or undo this operation with:

  git switch -

Turn off this advice by setting config variable advice.detachedHead to false

HEAD is now at 7150813 Create f3

Hi@DESKTOP-Q7TA858 MINGW64 ~/OneDrive/Desktop/nitin/DevOps/new/newgit ((7150813.
..))
$ ls
README.md  f2  f3  file1
```

The files is in the **origin/main** which can be merged if required using **git merge origin/main.**