

```
PS C:\Users\sujib\Downloads> cd .\day-four-undo-exercise-file\
```

```
PS C:\Users\sujib\Downloads\day-four-undo-exercise-file> git log
```

```
commit 1d9c17234ba8f440b261d530f47e089e253f370b (HEAD -> main, origin/main, origin/HEAD)
```

```
Author: example-git-student <example-git-student@protonmail.com>
```

```
Date: Fri Oct 7 14:51:35 2022 -0700
```

fourth commit

```
commit f6f4273dddf1310270cb765d7fa48f996c9f134f
```

```
Author: example-git-student <example-git-student@protonmail.com>
```

```
Date: Fri Oct 7 14:51:10 2022 -0700
```

third commit

```
commit 461e423600e7338e3a0d8cfb5e5300b8f4a9b4a5
```

```
Author: example-git-student <example-git-student@protonmail.com>
```

```
Date: Fri Oct 7 14:50:54 2022 -0700
```

second commit

```
commit 5f2f5159999d83e2bec6f16c72cedc24eecf72d7
```

```
Author: example-git-student <example-git-student@protonmail.com>
```

```
Date: Fri Oct 7 14:50:40 2022 -0700
```

first commit

```
PS C:\Users\sujib\Downloads\day-four-undo-exercise-file> git log --oneline
```

```
1d9c172 (HEAD -> main, origin/main, origin/HEAD) fourth commit
```

```
f6f4273 third commit
```

```
461e423 second commit
```

```
5f2f515 first commit
```

```
PS C:\Users\sujib\Downloads\day-four-undo-exercise-file> git checkout 5f2f515
```

Note: switching to '5f2f515'.

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.

```
GNU nano 2.7.5                                     File: .\myfile.txt
FIRST LINE AND FIRST COMMIT
```

```
PS C:\Users\sujib\Downloads\day-four-undo-exercise-file> git branch -a
```

```
* (HEAD detached at 5f2f515)
```

```
main
```

```
remotes/origin/HEAD -> origin/main
```

```
remotes/origin/main
```

```
PS C:\Users\sujib\Downloads\day-four-undo-exercise-file> git branch main
```

```
fatal: a branch named 'main' already exists
```

```
PS C:\Users\sujib\Downloads\day-four-undo-exercise-file> git switch main
```

Previous HEAD position was 5f2f515 first commit

Switched to branch 'main'

Your branch is up to date with 'origin/main'.

```
PS C:\Users\sujib\Downloads\day-four-undo-exercise-file> git log --oneline
```

```
1d9c172 (HEAD -> main, origin/main, origin/HEAD) fourth commit
```

```
f6f4273 third commit
```

```
461e423 second commit
```

```
5f2f515 first commit
```

```
PS C:\Users\sujib\Downloads\day-four-undo-exercise-file> git checkout 461e423
```

Note: switching to '461e423'.

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:

```
GNU nano 2.7.5                                File: .\myfile.txt
GNU nano 2.7.5                                File: .\myfile.txt
GNU nano 2.7.5                                File: .\myfile.txt
FIRST LINE AND FIRST COMMIT
SECOND LINE AND SECOND COMMIT
new branch text
GNU nano 2.7.5                                File: .\myfile.txt
FIRST LINE AND FIRST COMMIT
SECOND LINE AND SECOND COMMIT
new branch text
PS C:\Users\sujib\Downloads\day-four-undo-exercise-file> git switch new_branch

M    myfile.txt

Switched to branch 'new_branch'

PS C:\Users\sujib\Downloads\day-four-undo-exercise-file> git add .\myfile.txt

PS C:\Users\sujib\Downloads\day-four-undo-exercise-file> git commit -m "new branch text"

[new_branch 450cd42] new branch text

1 file changed, 3 insertions(+), 1 deletion(-)

PS C:\Users\sujib\Downloads\day-four-undo-exercise-file> git log --oneline

450cd42 (HEAD -> new_branch) new branch text
461e423 second commit
5f2f515 first commit

PS C:\Users\sujib\Downloads\day-four-undo-exercise-file>
```