

Git

1. Suppose you had a file, called `first.md`, and you made a copy of this file, named it `second.md` and made some changes to it. Next, suppose you ran `diff -u first.md second.md`.

Here is the content of the original `first.md`

A

B

C

D

E

F

Here is the output of the diff command:

```
--- first.md      2021-10-15 13:11:12.737592900 +1100
+++ second.md     2021-10-15 13:14:02.548934200 +1100
@@ -1,6 +1,8 @@
  A
  B
+$
  C
-D
+#
+%
  E
  F
```

What is the content of `second.md`?

Answer: the content of `second.md` is:

A

B

\$

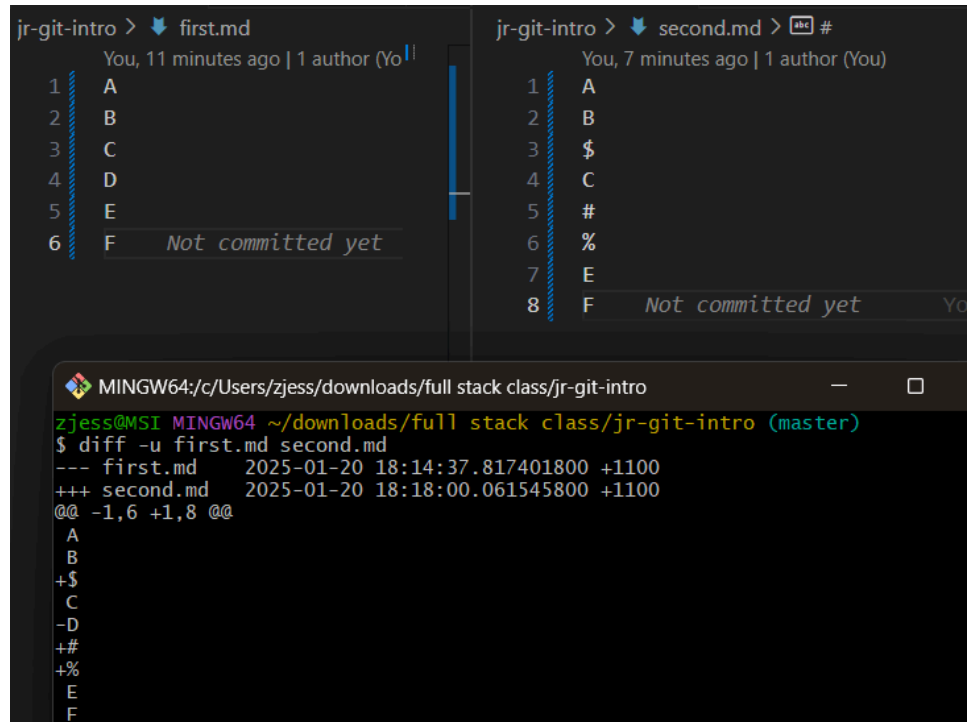
C

#

%

E

F



The screenshot shows a terminal window with two side-by-side diff views and a third window below them. The top-left window shows a diff for 'first.md' with lines 1-6: A, B, C, D, E, F. The top-right window shows a diff for 'second.md' with lines 1-8: A, B, \$, C, #, %, E, F. The bottom window shows the command `diff -u first.md second.md` and its output, which matches the content of the two diff windows above.

```
jr-git-intro > first.md
You, 11 minutes ago | 1 author (You)
1 A
2 B
3 C
4 D
5 E
6 F Not committed yet

jr-git-intro > second.md > #
You, 7 minutes ago | 1 author (You)
1 A
2 B
3 $
4 C
5 #
6 %
7 E
8 F Not committed yet

MINGW64/c/Users/zjess/downloads/full stack class/jr-git-intro
zjess@MSI MINGW64 ~/downloads/full stack class/jr-git-intro (master)
$ diff -u first.md second.md
--- first.md      2025-01-20 18:14:37.817401800 +1100
+++ second.md     2025-01-20 18:18:00.061545800 +1100
@@ -1,6 +1,8 @@
 A
 B
+$
 C
-D
+#
+%
 E
 F
```

2. (True or False) If you accidentally add a file to the staging area, you can remove it using `git reset`. For example, if you accidentally add `thrid.md`, but don't want it to be committed yet, run `git reset thrid.md` and the file will be removed from the staging area, **but it will still be in your working directory.**

Answer: True.

3. (True or False) The commands `git reset` and `git revert` can only be used to undo commits in the git repository.

Answer: False. `git reset` can be used on staging area as well.

4. (True or False) The commands `git checkout` can be used to roll back to a certain commit hash (check the documentation if you are unsure).

Answer: True. Use `git checkout <commit-hash>`

5. (True or False) We cannot commit changes in the working directory directly to the repo without adding it to the staging index first (read the documentation if you are unsure).

Answer: True. `git add` is a prerequisite of making commit.

6. (True or False) `git log -p` and `git log` will give you the same output.

Answer: False. `git log` simply lists the version history of current branch, while `git log -p` includes the additional diff information.

7. (True or False) `git log --oneline` and `git log --stat` will give you the same output.

Answer: False. `git log --oneline` gives a list of all the commits with corresponding messages, while `git log --stat` provides extra details such as commit timestamp and file change.

8. (True or False) It is recommended that in most cases we should use `git revert` rather than `git reset` to undo commits because `git revert` is safer.

Answer: True. `git reset` is more risky since it undo the changes by erasing all the commits in the version history after certain commit hash, however, `git revert` is safer since it keeps the original version history unchanged and roll back to a certain commit by creating a new commits, and user can refer to the previous commits in the timeline.