

▼ \ No newline at end of file  
is the warning of diff because diff is a **line-based** text processor.

- Without the last newline, whenever the next line is appended to this file, the last line must be edited by adding newline to it, which is an unintended edit. This is a various problem.

▼ 1st, diff works with the whole line, hence, the last line will be marked as modified by the next commit.

▼ E.g. wrong lines in diffs between commits without the last newline:

▼ Final file

```
Test EOL@EOF
this line is in commit #1.
this line is in commit #2.
this line is in commit #3.
```

▼ commit #1

```
+Test EOL@EOF
+this line is in commit #1.
\ No newline at end of file
```

▼ commit #2

```
Test EOL@EOF
-this line is in commit #1.
\ No newline at end of file
+this line is in commit #1.
+this line is in commit #2
\ No newline at end of file
```

▼ commit #3

```
this line is in commit #1.
-this line is in commit #2.
\ No newline at end of file
+this line is in commit #2.
+this line is in commit #3
\ No newline at end of file
```

▼ 2nd, the last line of the last commit may be **blamed** for the next commit, which is misleading.

▼ Some git implementations do this misleading blame: GitHub, GitLab, VS Code

This line is not written by EVA\_vanghtc!!!



add script deploy ins\_gw

5c385468

EVA\_vanghtc committed 2 months ago

```
1897 ##### domdp_cheapest END #####
1898
1899 ##### ins_gw START #####
1900 # common ins_gw
1901 APP_INS_GW = 'ins_gw'
1902 USER_INS_GW = 'ins_gw'
1903 BACKUP_MODULE_LIST_INS_GW = ['build/li
1904 BUILD_COMMAND_LIST_INS_GW = [
1905     'export JAVA_HOME=/usr/lib/jvm/jav
1906     './gradlew clean build -x test']
1907 MODULE_LIST_PRO_INS_GW = [
```

```
1936
1937
1938 add script deploy ins_gw 3 months ago
1939
1940 This line is not written by EVA_vanghtc!!!
1941
1942
1943
1944
1945
1946
1947
COPY_FILE_DEV_LIST_DOMDP_CHEAPEST=[
MERGE_FILE_DEV_LIST_DOMDP_CHEAPEST=
##### domdp_cheapest END #####
##### ins_gw START #####
# common ins_gw
APP_INS_GW = 'ins_gw'
USER_INS_GW = 'ins_gw'
BACKUP_MODULE_LIST_INS_GW = ['build
BUILD_COMMAND_LIST_INS_GW = [
    'export JAVA_HOME=/usr/lib/jvm/
    './gradlew clean build -x test'
```

▼ Some git implementations tweak themselves to avoid this misleading blame: Ubuntu's git, JetBrains IDEs

```
043c4d78 (Do Thien Hoang Mar 18 2024 1937) MERGE_FILE_DEV_LIST_DOMDP_CHEAPEST=[]
043c4d78 (Do Thien Hoang Mar 18 2024 1938) ##### domdp_cheapest END #####
5c385468 (vanghtc Nov 1 2024 1939)
5c385468 (vanghtc Nov 1 2024 1940) ##### ins_gw START #####
5c385468 (vanghtc Nov 1 2024 1941) # common ins_gw
5c385468 (vanghtc Nov 1 2024 1942) APP_INS_GW = 'ins_gw'
5c385468 (vanghtc Nov 1 2024 1943) USER_INS_GW = 'ins_gw'
5c385468 (vanghtc Nov 1 2024 1944) BACKUP_MODULE_LIST_INS_GW = ['build/libs/ins_gw-0.0.1-SNAP
5c385468 (vanghtc Nov 1 2024 1945) BUILD_COMMAND_LIST_INS_GW = [
```

Correct!

18/03/2024 Hoang	# dev domdp_cheapest	
18/03/2024 Hoang	BUILD_PATH_DEV_DOMDP_CHEAPEST = '/opt/build/domdp_cheapest'	
18/03/2024 Hoang	COPY_FILE_DEV_LIST_DOMDP_CHEAPEST=[	
18/03/2024 Hoang	MERGE_FILE_DEV_LIST_DOMDP_CHEAPEST=[	
18/03/2024 Hoang	##### domdp_cheapest END #####	
01/11/2024 vanghtc	##### ins_gw START #####	<b>Correct!</b>
01/11/2024 vanghtc	# common ins_gw	
01/11/2024 vanghtc	APP_INS_GW = 'ins_gw'	
01/11/2024 vanghtc	USER_INS_GW = 'ins_gw'	
01/11/2024 vanghtc	BACKUP_MODULE_LIST_INS_GW = ['build/libs/ins_gw-0.0.1-SNAPSHOT.jar	
01/11/2024 vanghtc	BUILD_COMMAND_LIST_INS_GW = [	

- ▼ 3rd, **unintended conflict**: Some text editors and IDEs automatically add the last newline to editing files, making unintended changes which will conflict with a
- ▼ The conflict will be shown (for resolution) in a very obscure way, due to the difficulty of showing the newline itself.

```
6: Last line without newline
<<<<<< HEAD
7: Appended line by feature-testEOL-1
=====
>>>>>> feature-testEOL-2
```

- ▼ There are two options to fix this issue:
  - Force all IDEs to automatically add the last newline.
  - ▼ Stop Linux editors, like `vim`, `nano`, from fixing files missing newline at the end of file.
    - This option should be chosen for a project with lots of legacy codes missing the last newline.

#### ▼ Commands to refactor codes

- ▼ Add the missing last newline (ref: [unix.stackexchange](https://unix.stackexchange.com/questions/111111/adding-a-newline-to-the-end-of-a-file))

```
sed -i '$a\' file
```

- ▼ Remove last blank line, i.e. truly empty line (`^$ = \n` only)

```
sed -i '${/^$/d;}' file
```

- ▼ Find files missing the last newline:

```
find . -regex '.*\.(git|venv|idea)' -prune -o \
-type f -exec sh -c 'tail -c 1 "$1" | grep -q "." && echo "$1"' no-last-newline {} \;
```

- ▼ Find files with the last newline

```
find . -regex '.*\.(git|venv|idea)' -prune -o \
-type f -exec sh -c 'tail -c 1 "$1" | grep -q "$'"'"' && echo "$1"' with-last-newline {} \;
```