1. Let us make few more commits to understand switching between versions.

```
PS D:\lab\Git> cd .\gitlab\

PS D:\lab\Git\gitlab> ls


    Directory: D:\lab\Git\gitlab


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----        2/23/2019  10:45 AM             21 hello.py

PS D:\lab\Git\gitlab> type .\hello.py
print ("Hello World")

PS D:\lab\Git\gitlab> notepad .\hello.py

PS D:\lab\Git\gitlab> type .\hello.py
print ("Hello World")
print ("Add first line of code")

PS D:\lab\Git\gitlab> git commit -a -m "second commit"
[master fd76073] second commit
 1 file changed, 2 insertions(+), 1 deletion(-)

PS D:\lab\Git\gitlab> notepad .\hello.py
PS D:\lab\Git\gitlab> type .\hello.py
print ("Hello World")
print ("Add first line of code")
print ("Add second line of code")

PS D:\lab\Git\gitlab> git commit -a -m "third commit"
[master 7d91a14] third commit
 1 file changed, 2 insertions(+), 1 deletion(-)

PS D:\lab\Git\gitlab> git log
commit 7d91a14fdf5207fad70e40cfae478a17ddecbc2b (HEAD -> master)
Author: Prakash <prakash@devops.com>
Date:   Mon Feb 25 02:53:13 2019 +0530

    third commit

commit fd7607337d37fc0c4d367311d3678374d569a9b9
Author: Prakash <prakash@devops.com>
Date:   Mon Feb 25 02:52:14 2019 +0530

    second commit
```

```
commit 6739347824b09657048bd57a051a31dc1cc2de41 (origin/master)
Author: Prakash <prakash@devops.com>
Date:   Sat Feb 23 11:06:05 2019 +0530

    first commit

PS D:\lab\Git\gitlab> git log --pretty=format:"%h %ad | %s%d
[%an]" --date=short
7d91a14 2019-02-25 | third commit (HEAD -> master) [Prakash]
fd76073 2019-02-25 | second commit [Prakash]
6739347 2019-02-23 | first commit (origin/master) [Prakash]
```

---

2. Switching between versions saved by git.

```
PS D:\lab\Git\gitlab> type .\hello.py
print ("Hello World")
print ("Add first line of code")
print ("Add second line of code")

PS D:\lab\Git\gitlab> git checkout 6739347
Note: checking out '6739347'.

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 performing another
checkout.

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

  git checkout -b <new-branch-name>

HEAD is now at 6739347 first commit

PS D:\lab\Git\gitlab> type .\hello.py
print ("Hello World")

PS D:\lab\Git\gitlab> git checkout 7d91a14
Previous HEAD position was 6739347 first commit
HEAD is now at 7d91a14 third commit

PS D:\lab\Git\gitlab> type .\hello.py
print ("Hello World")
print ("Add first line of code")
```

```
print ("Add second line of code")

PS D:\lab\Git\gitlab> git log --pretty=format:"%h %ad | %s%d
[%an]" --date=short
7d91a14 2019-02-25 | third commit (HEAD, master) [Prakash]
fd76073 2019-02-25 | second commit [Prakash]
6739347 2019-02-23 | first commit (origin/master) [Prakash]
```

---

3. Tag the versions

```
PS D:\lab\Git\gitlab> git tag v1.0

PS D:\lab\Git\gitlab> git log --pretty=format:"%h %ad | %s%d
[%an]" --date=short
7d91a14 2019-02-25 | third commit (HEAD, tag: v1.0, master)
[Prakash]
fd76073 2019-02-25 | second commit [Prakash]
6739347 2019-02-23 | first commit (origin/master) [Prakash]
PS D:\lab\Git\gitlab> git checkout 6739347
Previous HEAD position was 7d91a14 third commit
HEAD is now at 6739347 first commit

PS D:\lab\Git\gitlab> type .\hello.py
print ("Hello World")

PS D:\lab\Git\gitlab> git checkout v1.0
Previous HEAD position was 6739347 first commit
HEAD is now at 7d91a14 third commit
PS D:\lab\Git\gitlab> type .\hello.py
print ("Hello World")
print ("Add first line of code")
print ("Add second line of code")

PS D:\lab\Git\gitlab> git checkout v1.0^
Previous HEAD position was 7d91a14 third commit
HEAD is now at fd76073 second commit
PS D:\lab\Git\gitlab> type hello.py
print ("Hello World")
print ("Add first line of code")

PS D:\lab\Git\gitlab> git checkout v1.0
Previous HEAD position was fd76073 second commit
HEAD is now at 7d91a14 third commit
PS D:\lab\Git\gitlab> type hello.py
print ("Hello World")
print ("Add first line of code")
print ("Add second line of code")
```

4. Reversing the staged changes.

```
PS D:\lab\Git\gitlab> notepad .\hello.py
PS D:\lab\Git\gitlab> type .\hello.py
print ("Hello World")
print ("Add first line of code")
print ("Add second line of code")
unwanted change

PS D:\lab\Git\gitlab> git status -s
 M hello.py

PS D:\lab\Git\gitlab> git status
HEAD detached at v1.0
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working
directory)

        modified:   hello.py

no changes added to commit (use "git add" and/or "git commit -
a")

PS D:\lab\Git\gitlab> git add .\hello.py

PS D:\lab\Git\gitlab> git status -s
M  hello.py

PS D:\lab\Git\gitlab> git status
HEAD detached at v1.0
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

        modified:   hello.py

PS D:\lab\Git\gitlab> git reset HEAD .\hello.py
Unstaged changes after reset:
M       hello.py

PS D:\lab\Git\gitlab> git status -s
 M hello.py

PS D:\lab\Git\gitlab> git status
HEAD detached at v1.0
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working
directory)
```

```
        modified:   hello.py

no changes added to commit (use "git add" and/or "git commit -
a")
```

---

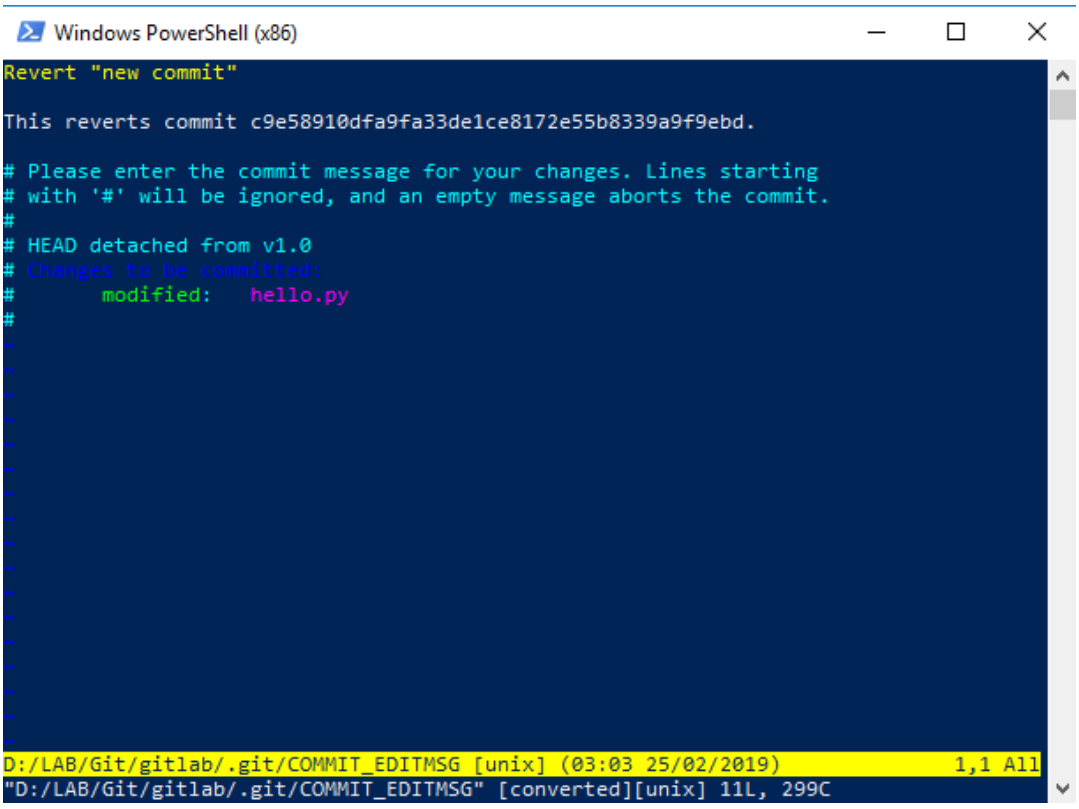
5. Reversing the committed changes.

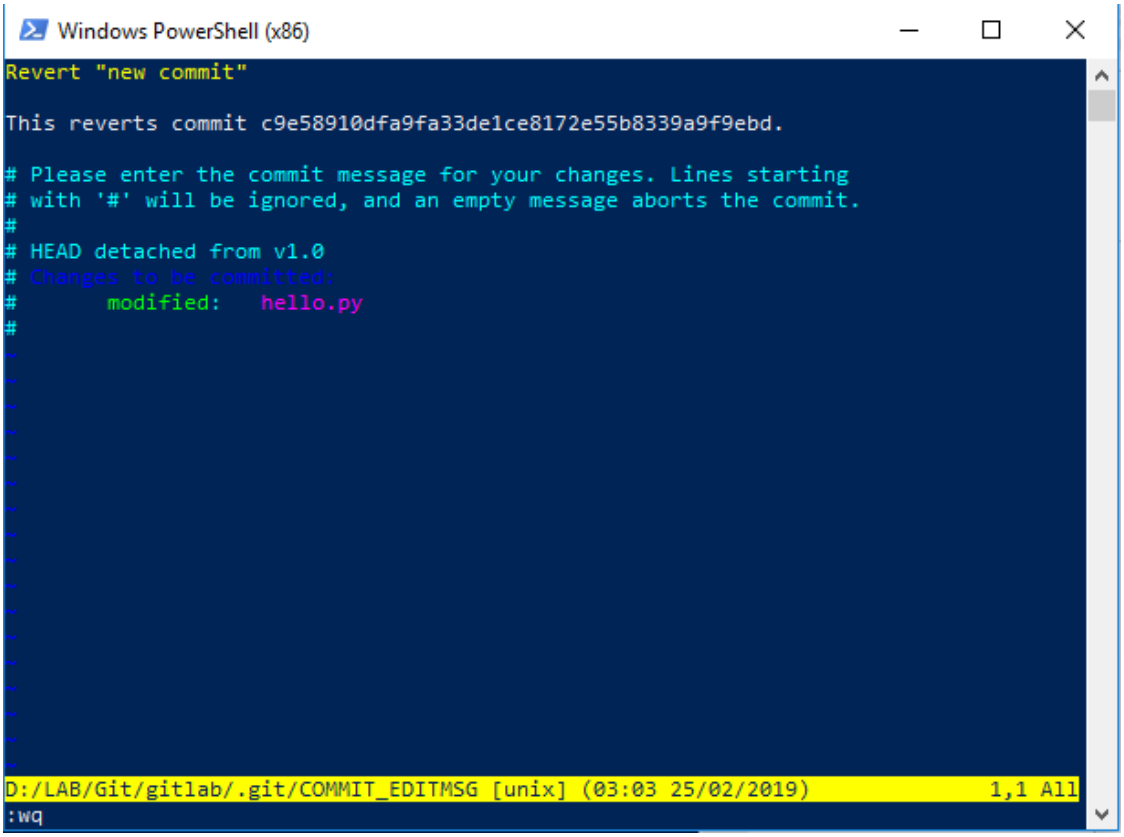
```
PS D:\lab\Git\gitlab> git commit -a -m "new commit"
[detached HEAD c9e5891] new commit
 1 file changed, 2 insertions(+), 1 deletion(-)

PS D:\lab\Git\gitlab> git status -s
PS D:\lab\Git\gitlab>

PS D:\lab\Git\gitlab> git log --pretty=format:"%h %ad | %s%d
[%an]" --date=short
c9e5891 2019-02-25 | new commit (HEAD) [Prakash]
7d91a14 2019-02-25 | third commit (tag: v1.0, master) [Prakash]
fd76073 2019-02-25 | second commit [Prakash]
6739347 2019-02-23 | first commit (origin/master) [Prakash]

PS D:\lab\Git\gitlab> git revert HEAD
```






```
PS D:\lab\Git\gitlab> git revert HEAD
[detached HEAD e04ae2f] Revert "new commit"
 1 file changed, 1 insertion(+), 2 deletions(-)

PS D:\lab\Git\gitlab> git log --pretty=format:"%h %ad | %s%d
[%an]" --date=short
e04ae2f 2019-02-25 | Revert "new commit" (HEAD) [Prakash]
c9e5891 2019-02-25 | new commit [Prakash]
7d91a14 2019-02-25 | third commit (tag: v1.0, master) [Prakash]
fd76073 2019-02-25 | second commit [Prakash]
6739347 2019-02-23 | first commit (origin/master) [Prakash]

PS D:\lab\Git\gitlab> git reset --hard v1.0
HEAD is now at 7d91a14 third commit
PS D:\lab\Git\gitlab> git log --pretty=format:"%h %ad | %s%d
[%an]" --date=short
7d91a14 2019-02-25 | third commit (HEAD, tag: v1.0, master)
[Prakash]
fd76073 2019-02-25 | second commit [Prakash]
6739347 2019-02-23 | first commit (origin/master) [Prakash]
```

6. Alternate way of amending the commit.

```
PS D:\lab\Git\gitlab> notepad .\hello.py

PS D:\lab\Git\gitlab> type .\hello.py
print ("Hello World")
print ("Add first line of code")
print ("Add second line of code")
Unwanted change

PS D:\lab\Git\gitlab> git commit -a -m "Making a change"
[detached HEAD 1f1db10] Making a change
 1 file changed, 2 insertions(+), 1 deletion(-)

PS D:\lab\Git\gitlab> git log --pretty=format:"%h %ad | %s%d
[%an]" --date=short
1f1db10 2019-02-25 | Making a change (HEAD) [Prakash]
7d91a14 2019-02-25 | third commit (tag: v1.0, master) [Prakash]
fd76073 2019-02-25 | second commit [Prakash]
6739347 2019-02-23 | first commit (origin/master) [Prakash]
PS D:\lab\Git\gitlab>

PS D:\lab\Git\gitlab> notepad .\hello.py
PS D:\lab\Git\gitlab> type .\hello.py
print ("Hello World")
print ("Add first line of code")
print ("Add second line of code")
print ("Making required change")

PS D:\lab\Git\gitlab> git commit -a --amend -m "Right Change"
[detached HEAD 4c3faab] Right Change
 Date: Mon Feb 25 03:22:37 2019 +0530
 1 file changed, 2 insertions(+), 1 deletion(-)

PS D:\lab\Git\gitlab> git log --pretty=format:"%h %ad | %s%d
[%an]" --date=short
4c3faab 2019-02-25 | Right Change (HEAD) [Prakash]
7d91a14 2019-02-25 | third commit (tag: v1.0, master) [Prakash]
fd76073 2019-02-25 | second commit [Prakash]
6739347 2019-02-23 | first commit (origin/master) [Prakash]
```