

Q)

▼ Q1 : Git Task Instructions

Branch Setup

1. Create two branches:

- `develop`
- `alpha`

A)

```
[root@localhost wd]# git branch develop
[root@localhost wd]# git branch alpha
[root@localhost wd]# git branch
  alpha
  develop
* master
[root@localhost wd]#
```

Q)

4. View the log in one-line format and take a screenshot:

```
git log --oneline
```

A)

```
root@localhost:~/wd

[root@localhost wd]# git log --oneline
dbb8f00 (HEAD -> master) second-commit
5d3bc29 first-commit
c78ae78 (develop, alpha) initial commit
[root@localhost wd]#
```

Q)

5. Use `git reset --hard` to move back to the first commit and take a screenshot:

```
git reset --hard HEAD~1
```

A)

```
root@localhost:~/wd

[root@localhost wd]# git reset --hard HEAD~1
HEAD is now at 5d3bc29 first-commit
[root@localhost wd]# git log --oneline
5d3bc29 (HEAD -> master) first-commit
c78ae78 (develop, alpha) initial commit
[root@localhost wd]# ls
file1  initial
[root@localhost wd]#
```

Q)

6. Use `git reflog` to find the previous state (second commit), reset back to it, and take screenshots from `reflog` and after resetting:

```
git reflog
git reset --hard <commit-hash-of-second-commit>
```

A)

```
root@localhost:~/wd

[root@localhost wd]# git reflog
dbb8f00 (HEAD -> master) HEAD@{0}: reset: moving to dbb8f00
5d3bc29 HEAD@{1}: reset: moving to HEAD~1
dbb8f00 (HEAD -> master) HEAD@{2}: commit: second-commit
5d3bc29 HEAD@{3}: commit: first-commit
c78ae78 (develop, alpha) HEAD@{4}: commit (initial): initial com
[root@localhost wd]# ls
file1 file2 initial
[root@localhost wd]# git log --oneline
dbb8f00 (HEAD -> master) second-commit
5d3bc29 first-commit
c78ae78 (develop, alpha) initial commit
[root@localhost wd]#
```

Q)

3. Switch again to `alpha` and visualize the log graph:

```
git checkout alpha
git log --oneline --decorate --all --graph
```

A)

```
[root@localhost wd]# git checkout alpha
git log --oneline --decorate --all --graph
Switched to branch 'alpha'
* 7b8a632 (develop) fourth-commit
| * 50dc90d (HEAD -> alpha) third-commit
|/
| * dbb8f00 (master) second-commit
| * 5d3bc29 first-commit
|/
* c78ae78 initial commit
[root@localhost wd]#
```

Q)

Part 3: Rebase Operation

1. Rebase `develop` onto `alpha`:

```
git rebase develop
```

A)

```
root@localhost:~/wd

[root@localhost wd]# git log --oneline --decorate --all --graph
* 7b8a632 (develop) fourth-commit
| * 50dc90d (HEAD -> alpha) third-commit
|/
| * dbb8f00 (master) second-commit
| * 5d3bc29 first-commit
|/
* c78ae78 initial commit
[root@localhost wd]# git branch
* alpha
  develop
  master
[root@localhost wd]# git rebase develop
Successfully rebased and updated refs/heads/alpha.
[root@localhost wd]# git log --oneline --decorate --all --graph
* 99056ca (HEAD -> alpha) third-commit
* 7b8a632 (develop) fourth-commit
| * dbb8f00 (master) second-commit
| * 5d3bc29 first-commit
|/
* c78ae78 initial commit
[root@localhost wd]#
```