

Module 2: Git Assignment - 5 - Solution - 2

1. Create a Git Flow work flow architecture on Git.

Solution:-

```
$ sudo su
# apt update
```

```
ubuntu@ip-172-31-85-241:~$ sudo su
root@ip-172-31-85-241:/home/ubuntu# apt update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [11
9 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease [
108 kB]
Get:4 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
```

```
# apt upgrade
# apt install git
# which git
# git --version
```

```
root@ip-172-31-85-241:/home/ubuntu# apt upgrade
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
#
# An OpenSSL vulnerability has recently been fixed with USN-6188-1 & 6119-1:
# CVE-2023-2650: possible DoS translating ASN.1 object identifiers.
# Ensure you have updated the package to its latest version.
#
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
root@ip-172-31-85-241:/home/ubuntu# apt install git
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
git is already the newest version (1:2.34.1-1ubuntu1.9).
git set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
root@ip-172-31-85-241:/home/ubuntu# which git
/usr/bin/git
root@ip-172-31-85-241:/home/ubuntu# git --version
git version 2.34.1
```

```
# git config --global user.name "Amit Tiwari"
# git config --global user.email "redhat.amitiwari@gmail.com"
# git config --list
# mkdir gitdir
# cd gitdir
```

```
root@ip-172-31-85-241:/home/ubuntu# git config --global user.name "Amit Tiwari"
root@ip-172-31-85-241:/home/ubuntu# git config --global user.email "redhat.amitiwari@gmail.com"
root@ip-172-31-85-241:/home/ubuntu# git config --list
user.name=Amit Tiwari
user.email=redhat.amitiwari@gmail.com
root@ip-172-31-85-241:/home/ubuntu# mkdir gitdir
root@ip-172-31-85-241:/home/ubuntu# cd gitdir
root@ip-172-31-85-241:/home/ubuntu/gitdir#
```

```
# mkdir workflow
# cd workflow
```

```
root@ip-172-31-85-241:/home/ubuntu/gitdir# mkdir workflow
root@ip-172-31-85-241:/home/ubuntu/gitdir# cd workflow
```

```
# apt install git-flow
```

```
root@ip-172-31-85-241:/home/ubuntu/gitdir/workflow# apt install git-flow
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  git-flow
0 upgraded, 1 newly installed, 0 to remove and 82 not upgraded.
Need to get 38.6 kB of archives.
After this operation, 329 kB of additional disk space will be used.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 git-flow all 1.12.3-3 [38.6 kB]
Fetched 38.6 kB in 0s (871 kB/s)
Selecting previously unselected package git-flow.
(Reading database ... 64295 files and directories currently installed.)
Preparing to unpack .../git-flow 1.12.3-3 all.deb ...
```

```
# git flow init
```

```
root@ip-172-31-85-241:/home/ubuntu/gitdir/workflow# git flow init
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /home/ubuntu/gitdir/workflow/.git/
No branches exist yet. Base branches must be created now.
Branch name for production releases: [master]
Branch name for "next release" development: [develop]

How to name your supporting branch prefixes?
Feature branches? [feature/]
Bugfix branches? [bugfix/]
Release branches? [release/]
Hotfix branches? [hotfix/]
Support branches? [support/]
Version tag prefix? []
Hooks and filters directory? [/home/ubuntu/gitdir/workflow/.git/hooks]
```

```
# git branch
```

```
root@ip-172-31-85-241:/home/ubuntu/gitdir/workflow# git branch
* develop
master
```

2. Create all the required branches.

Solution:-

```
# git branch -a
# git flow feature start feature1
# git branch -a
```

```
root@ip-172-31-85-241:/home/ubuntu/gitdir/workflow# git branch -a
* develop
  master
root@ip-172-31-85-241:/home/ubuntu/gitdir/workflow# git flow feature start feature1
Switched to a new branch 'feature/feature1'

Summary of actions:
- A new branch 'feature/feature1' was created, based on 'develop'
- You are now on branch 'feature/feature1'

Now, start committing on your feature. When done, use:

    git flow feature finish feature1

root@ip-172-31-85-241:/home/ubuntu/gitdir/workflow# git branch -a
  develop
* feature/feature1
  master
```

- Starting from the feature branch, push the branch to the master, following the architecture.

Solution:-

```
# git checkout feature/feature1
# touch feature.html
# ls
# git add .
# git commit -m "feature complete!" feature.html
# git branch -a
# git reflog
```

```
root@ip-172-31-85-241:/home/ubuntu/gitdir/workflow# git checkout feature/feature1
Already on 'feature/feature1'
root@ip-172-31-85-241:/home/ubuntu/gitdir/workflow# touch feature.html
root@ip-172-31-85-241:/home/ubuntu/gitdir/workflow# ls
feature.html
root@ip-172-31-85-241:/home/ubuntu/gitdir/workflow# git add .
root@ip-172-31-85-241:/home/ubuntu/gitdir/workflow# git commit -m "feature complete!" feature.html
[feature/feature1 a729e61] feature complete!
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 feature.html
root@ip-172-31-85-241:/home/ubuntu/gitdir/workflow# git branch -a
  develop
* feature/feature1
  master
root@ip-172-31-85-241:/home/ubuntu/gitdir/workflow# git reflog
a729e61 (HEAD -> feature/feature1) HEAD@{0}: commit: feature complete!
527e276 (master, develop) HEAD@{1}: checkout: moving from feature/feature1 to feature/feature1
527e276 (master, develop) HEAD@{2}: checkout: moving from develop to feature/feature1
527e276 (master, develop) HEAD@{3}: checkout: moving from master to develop
527e276 (master, develop) HEAD@{4}: commit (initial): Initial commit
```

```
# git flow feature finish feature1
# git checkout master
```

```
root@ip-172-31-85-241:/home/ubuntu/gitdir/workflow# git flow feature finish feature1
Switched to branch 'develop'
Updating 527e276..a729e61
Fast-forward
 feature.html | 0
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 feature.html
Deleted branch feature/feature1 (was a729e61).

Summary of actions:
- The feature branch 'feature/feature1' was merged into 'develop'
- Feature branch 'feature/feature1' has been locally deleted
- You are now on branch 'develop'

root@ip-172-31-85-241:/home/ubuntu/gitdir/workflow# git checkout master
Switched to branch 'master'
```


4. Push a urgent.txt on master using hotfix

Solution:-

```
# git branch -a
# git flow hotfix start '0.1.1'
# touch urgent.txt
# git add . && git commit -m "A git flow hotfix from start to finish."
urgent.txt
```

```
root@ip-172-31-85-241:/home/ubuntu/gitdir/workflow# git branch -a
develop
* master
root@ip-172-31-85-241:/home/ubuntu/gitdir/workflow# git flow hotfix start '0.1.1'
Switched to a new branch 'hotfix/0.1.1'

Summary of actions:
- A new branch 'hotfix/0.1.1' was created, based on 'master'
- You are now on branch 'hotfix/0.1.1'

Follow-up actions:
- Start committing your hot fixes
- Bump the version number now!
- When done, run:

    git flow hotfix finish '0.1.1'

root@ip-172-31-85-241:/home/ubuntu/gitdir/workflow# touch urgent.txt
root@ip-172-31-85-241:/home/ubuntu/gitdir/workflow# git add . && git commit -m "
A git flow hotfix from start to finish." urgent.txt
[hotfix/0.1.1 a223109] A git flow hotfix from start to finish.
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 urgent.txt
```

```
# git branch -a
# git checkout master
# git flow hotfix finish '0.1.1'
```

```
root@ip-172-31-85-241:/home/ubuntu/gitdir/workflow# git branch -a
develop
* hotfix/0.1.1
master
root@ip-172-31-85-241:/home/ubuntu/gitdir/workflow# git checkout master
Switched to branch 'master'
root@ip-172-31-85-241:/home/ubuntu/gitdir/workflow# git flow hotfix finish '0.1.1'
Already on 'master'
Merge made by the 'ort' strategy.
urgent.txt | 0
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 urgent.txt
Switched to branch 'develop'
Merge made by the 'ort' strategy.
urgent.txt | 0
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 urgent.txt
Deleted branch hotfix/0.1.1 (was a223109).

Summary of actions:
- Hotfix branch 'hotfix/0.1.1' has been merged into 'master'
- The hotfix was tagged '0.1.1'
- Hotfix tag '0.1.1' has been back-merged into 'develop'
- Hotfix branch 'hotfix/0.1.1' has been locally deleted
- You are now on branch 'develop'
```