

## Module-2: GIT Assignment - 2

Do the following tasks:

- Create a git working directory with feature1.txt and feature2.txt in the master branch
- Create 3 branches develop, feature1 and feature2
- In develop branch create develop.txt, do not stage or commit it
- Stash this file, and checkout to feature1 branch
- Create new.txt file in feature1 branch, stage and commit this file
- Checkout to develop, unstash this file and commit

Solution:

### Create a git working directory with feature1.txt and feature2.txt in the master branch

- 1) Initialize your Git repo ----- git init
- 2) Create a new directory and navigate into it----- mkdir <name>    cd <name>

```
git $mkdir test
git $cd test
git $|
```

- 3) Create the feature1.txt and feature2.txt files ----- touch feature1.txt  
touch feature2.txt

```
git $touch feature1.txt feature2.txt
git $ll
total 0
-rw-r--r-- 1 bhats 197609 0 Jan 20 22:26 feature1.txt
-rw-r--r-- 1 bhats 197609 0 Jan 20 22:26 feature2.txt
git $
```

### Create 3 branches develop, feature1 and feature2

- 1) cd path/to/your/repository
- 2) Create and switch to develop branch-----git checkout -b develop
- 3) Create and switch to feature1 branch-----git checkout -b feature1
- 4) Create and switch to feature2 branch-----git checkout -b feature2

```
git $pwd
/c/Users/bhats/.git
git $git checkout -b develop
Switched to a new branch 'develop'
git $git checkout -b feature1
Switched to a new branch 'feature1'
git $git checkout -b feature2
Switched to a new branch 'feature2'
git $
```

### In develop branch create develop.txt, do not stage or commit it

- 1) Switch to the develop branch----- git checkout develop
- 2) Create the develop.txt file----- touch develop.txt
- 3) At this point, the develop.txt file has been created in the develop branch, but it has not been staged or committed.

```
git $git checkout develop
Switched to branch 'develop'
git $touch develop.txt
git $
```

### Stash this file, and checkout to feature1 branch

- 1) Stash the develop.txt file----- git stash
- 2) Switch to the feature1 branch----- git checkout feature1

```
git $git stash
No local changes to save
git $git checkout feature1
Switched to branch 'feature1'
git $
```

### Create new.txt file in feature1 branch, stage and commit this file

- 1) Create the new.txt file ----- touch new.txt
- 2) Stage the new.txt file ----- git add new.txt
- 3) Commit the staged file ----- git commit -m "Add new.txt file to feature1 branch"

```
git $touch new.txt
git $git add new.txt
git $git commit -m "Add new.txt file to feature1 branch"
[feature1 d257aed] Add new.txt file to feature1 branch
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 new.txt
git $
```

### Checkout to develop, unstash this file and commit

- 1) Switch to the develop branch ----- git checkout develop
- 2) List the stash entries ----- git stash list
- 3) Apply the stashed changes----- git stash apply
- 4) Stage the develop.txt file ----- git add develop.txt
- 5) Commit the changes ----- git commit -m "Add develop.txt file after unstashing"

```
git $git checkout develop
Switched to branch 'develop'
git $git stash list
git $git stash apply
No stash entries found.
git $git add develop.txt
git $git commit -m "Add develop.txt file after unstashing"
[develop ee32422] Add develop.txt file after unstashing
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 develop.txt
git $
```