

##Advance Git & GitHub for DevOps Engineers: Part-2 – Day 11

Task 1:

- Create a new branch and make some changes to it.
- Use git stash to save the changes without committing them.

```
P1324376@DESKTOP-1236IDC MINGW64 /d/90_Days_DevOps/MyTasks/Devops/Git (dev)
$ git branch
* dev
  main
  test

P1324376@DESKTOP-1236IDC MINGW64 /d/90_Days_DevOps/MyTasks/Devops/Git (dev)
$ touch file1.txt

P1324376@DESKTOP-1236IDC MINGW64 /d/90_Days_DevOps/MyTasks/Devops/Git (dev)
$ git add file1.txt

P1324376@DESKTOP-1236IDC MINGW64 /d/90_Days_DevOps/MyTasks/Devops/Git (dev)
$ git stash
Saved working directory and index state WIP on dev: 155ad26 Added Samplefile for Day 10 exercise.
```

- Switch to a different branch, make some changes and commit them.

```
P1324376@DESKTOP-1236IDC MINGW64 /d/90_Days_DevOps/MyTasks/Devops/Git (dev)
$ git branch
* dev
  main
  test

P1324376@DESKTOP-1236IDC MINGW64 /d/90_Days_DevOps/MyTasks/Devops/Git (dev)
$ touch file1.txt

P1324376@DESKTOP-1236IDC MINGW64 /d/90_Days_DevOps/MyTasks/Devops/Git (dev)
$ git add file1.txt

P1324376@DESKTOP-1236IDC MINGW64 /d/90_Days_DevOps/MyTasks/Devops/Git (dev)
$ git stash
Saved working directory and index state WIP on dev: 155ad26 Added Samplefile for Day 10 exercise.

P1324376@DESKTOP-1236IDC MINGW64 /d/90_Days_DevOps/MyTasks/Devops/Git (dev)
$ git switch main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.

P1324376@DESKTOP-1236IDC MINGW64 /d/90_Days_DevOps/MyTasks/Devops/Git (main)
$ vim Samplefile.txt

P1324376@DESKTOP-1236IDC MINGW64 /d/90_Days_DevOps/MyTasks/Devops/Git (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   Samplefile.txt

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

P1324376@DESKTOP-1236IDC MINGW64 /d/90_Days_DevOps/MyTasks/Devops/Git (main)
$ git add Samplefile.txt
warning: in the working copy of 'Git/Samplefile.txt', LF will be replaced by CRLF the next time Git touches it

P1324376@DESKTOP-1236IDC MINGW64 /d/90_Days_DevOps/MyTasks/Devops/Git (main)
$ git commit -m "Added new changes for #task1"
[main ea3719d] Added new changes for #task1
 1 file changed, 1 insertion(+)
```

- Use git stash pop to bring the changes back and apply them on top of the new commits.

```

P1324376@DESKTOP-1236IDC MINGW64 /d/90_Days_DevOps/MyTasks/Devops/Git (main)
$ git stash pop
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
(use "git push" to publish your local commits)

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   file1.txt

Dropped refs/stash@{0} (8e78ef9ca84bcedd667821bc4cfa324bda89908a)

P1324376@DESKTOP-1236IDC MINGW64 /d/90_Days_DevOps/MyTasks/Devops/Git (main)
$ git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
(use "git push" to publish your local commits)

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   file1.txt

P1324376@DESKTOP-1236IDC MINGW64 /d/90_Days_DevOps/MyTasks/Devops/Git (main)
$ git commit -m "Added stashed changes."
[main 3051f18] Added stashed changes.
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 Git/file1.txt

```

Task 2:

- In version01.txt of development branch add below lines after "This is the bug fix in development branch" that you added in Day10 and reverted to this commit.
- Line2>> After bug fixing, this is the new feature with minor alteration" Commit this with message "Added feature2.1 in development branch"
- Line3>> This is the advancement of previous feature Commit this with message "Added feature2.2 in development branch"
- Line4>> Feature 2 is completed and ready for release Commit this with message "Feature2 completed"

```
P1324376@DESKTOP-1236IDC MINGW64 /d/90_Days_DevOps/MyTasks/Devops/Git (main)
$ vim version01.txt

P1324376@DESKTOP-1236IDC MINGW64 /d/90_Days_DevOps/MyTasks/Devops/Git (main)
$ git add version01.txt

P1324376@DESKTOP-1236IDC MINGW64 /d/90_Days_DevOps/MyTasks/Devops/Git (main)
$ git commit -m "Added feature2.1 in development branch."
[main bd44513] Added feature2.1 in development branch.
1 file changed, 1 insertion(+)

P1324376@DESKTOP-1236IDC MINGW64 /d/90_Days_DevOps/MyTasks/Devops/Git (main)
$ vim version01.txt

P1324376@DESKTOP-1236IDC MINGW64 /d/90_Days_DevOps/MyTasks/Devops/Git (main)
$ git add version01.txt

P1324376@DESKTOP-1236IDC MINGW64 /d/90_Days_DevOps/MyTasks/Devops/Git (main)
$ git commit -m "Added feature2.2 in development branch."
[main bdef6b8] Added feature2.2 in development branch.
1 file changed, 1 insertion(+)

P1324376@DESKTOP-1236IDC MINGW64 /d/90_Days_DevOps/MyTasks/Devops/Git (main)
$ vim version01.txt

P1324376@DESKTOP-1236IDC MINGW64 /d/90_Days_DevOps/MyTasks/Devops/Git (main)
$ git add version01.txt

P1324376@DESKTOP-1236IDC MINGW64 /d/90_Days_DevOps/MyTasks/Devops/Git (main)
$ git commit -m "Feature2 completed."
[main 0f65f57] Feature2 completed.
1 file changed, 1 insertion(+)
```

- All these commits messages should be reflected in Production branch too which will come out from Master branch (Hint: try rebase).
 - git rebase main

```
P1324376@DESKTOP-1236IDC MINGW64 /d/90_Days_DevOps/MyTasks/Devops/Git (Production)
$ git rebase main
Current branch Production is up to date.

P1324376@DESKTOP-1236IDC MINGW64 /d/90_Days_DevOps/MyTasks/Devops/Git (Production)
$ git log --oneline
0f65f57 (HEAD -> Production, main) Feature2 completed.
bdef6b8 Added feature2.2 in development branch.
bd44513 Added feature2.1 in development branch.
16505bb Added feature2 in development branch.
9bfd8cd Added new feature.
61ed48d Initial commit
```

Task 3:

- In Production branch Cherry pick Commit "Added feature2.2 in development branch" and added below lines in it:
- Line to be added after Line3>> This is the advancement of previous feature
- Line4>>Added few more changes to make it more optimized.
Commit: Optimized the feature

```
P1324376@DESKTOP-1236IDC MINGW64 /d/90_Days_DevOps/MyTasks/Devops/Git (Production)
$ git cherry-pick bdef6b8a408a14b54894ed53f51cb63232042d80
Auto-merging Git/version01.txt
CONFLICT (content): Merge conflict in Git/version01.txt
error: could not apply bdef6b8... Added feature2.2 in development branch.
hint: After resolving the conflicts, mark them with
hint: "git add/rm <paths>", then run
hint: "git cherry-pick --continue".
hint: You can instead skip this commit with "git cherry-pick --skip".
hint: To abort and get back to the state before "git cherry-pick",
hint: run "git cherry-pick --abort".

P1324376@DESKTOP-1236IDC MINGW64 /d/90_Days_DevOps/MyTasks/Devops/Git (Production|CHERRY-PICKING)
$ vim version01.txt

P1324376@DESKTOP-1236IDC MINGW64 /d/90_Days_DevOps/MyTasks/Devops/Git (Production|CHERRY-PICKING)
$ git add version01.txt
```

```
P1324376@DESKTOP-1236IDC MINGW64 /d/90_Days_DevOps/MyTasks/Devops/Git (Production)
$ git commit -m "Optimized the feature."
[Production f9bdfc3] Optimized the feature.
1 file changed, 2 insertions(+), 2 deletions(-)

P1324376@DESKTOP-1236IDC MINGW64 /d/90_Days_DevOps/MyTasks/Devops/Git (Production)
$ git log --oneline
f9bdfc3 (HEAD -> Production) Optimized the feature.
f0097d8 Added feature2.2 in development branch.
0f65f57 (main) Feature2 completed.
bdef6b8 Added feature2.2 in development branch.
bd44513 Added feature2.1 in development branch.
16505bb Added feature2 in development branch.
9bfd8cd Added new feature.
61ed48d Initial commit
```