



FEBRUARY 22, 2023

# DevOps Classroomnotes 22/Feb/2023

## Git Branching

- Create a new folder and make it a repository
- add one commit and rename the default branch to main

```
Dell@DESKTOP-TM7SH71 MINGW64 /c/temp/gitpractice/branches_demo (master)
$ git branch
* master

Dell@DESKTOP-TM7SH71 MINGW64 /c/temp/gitpractice/branches_demo (master)
$ git log --oneline
649bed7 (HEAD -> master) Added basic folder structure

Dell@DESKTOP-TM7SH71 MINGW64 /c/temp/gitpractice/branches_demo (master)
$ git branch --help

Dell@DESKTOP-TM7SH71 MINGW64 /c/temp/gitpractice/branches_demo (master)
$ git branch -m main

Dell@DESKTOP-TM7SH71 MINGW64 /c/temp/gitpractice/branches_demo (main)
$ git branch
* main
```

- create one more commit on main

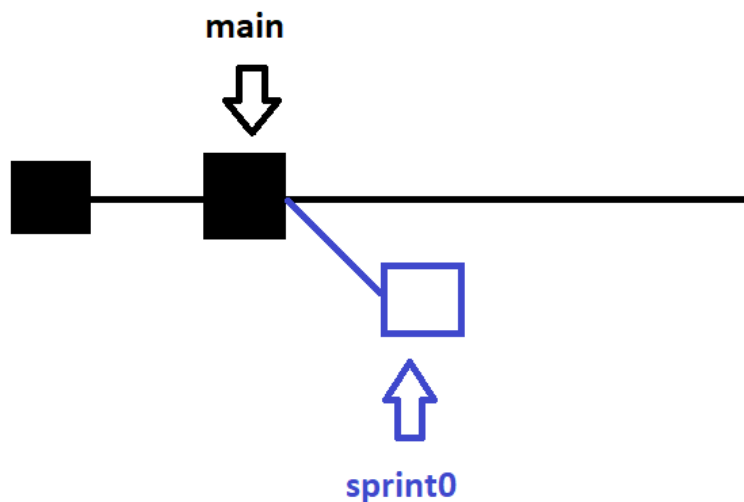
```
Dell@DESKTOP-TM7SH71 MINGW64 /c/temp/gitpractice/branches_demo/test (main)
$ git log --oneline
2d84516 (HEAD -> main) Added docs
649bed7 Added basic folder structure
```

- Lets create a branch called as sprint0

```
Dell@DESKTOP-TM7SH71 MINGW64 /c/temp/gitpractice/branches_demo/test (main)
$ git checkout -b sprint0
Switched to a new branch 'sprint0'

Dell@DESKTOP-TM7SH71 MINGW64 /c/temp/gitpractice/branches_demo/test (sprint0)
$ git branch
  main
* sprint0
```

- lets make one change in sprint0 in src/main.py



## Fast Forward Merge

- We want changes done in sprint0 to be merged to main
- The merge command is `git merge <source-branch>` i.e. you should be in target branch

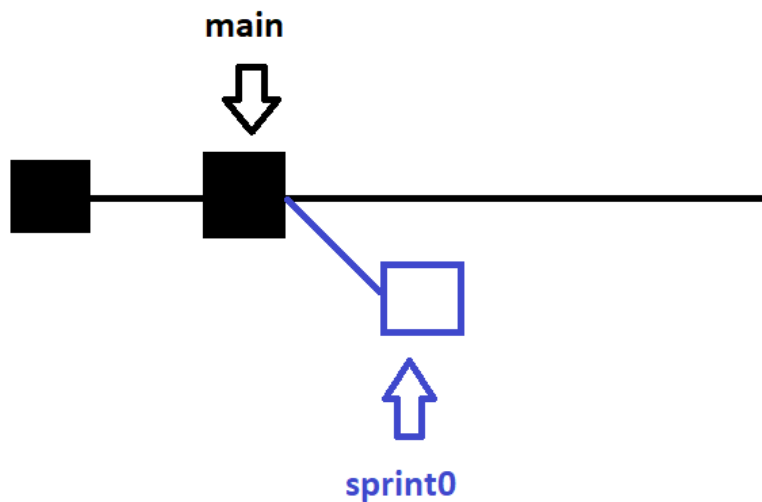
```

Dell@DESKTOP-TM7SH71 MINGW64 /c/temp/gitpractice/branches_demo (sprint0)
$ git log --oneline
36c0cb5 (HEAD -> sprint0) Added change
2d84516 (main) Added docs
649bed7 Added basic folder structure

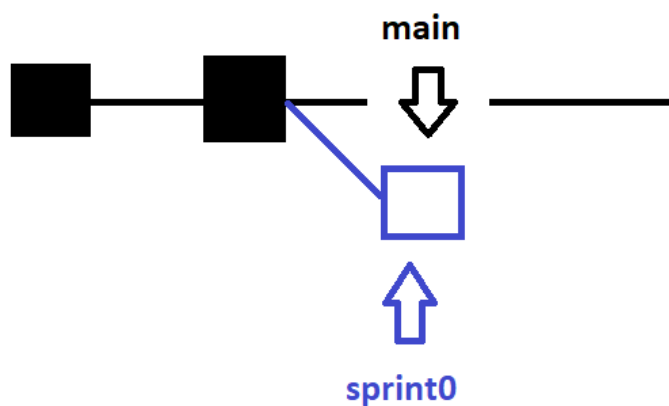
Dell@DESKTOP-TM7SH71 MINGW64 /c/temp/gitpractice/branches_demo (sprint0)
$ git checkout main
Switched to branch 'main'

Dell@DESKTOP-TM7SH71 MINGW64 /c/temp/gitpractice/branches_demo (main)
$ git merge sprint0
Updating 2d84516..36c0cb5
(Fast-forward
 src/main.py | 1 +
 1 file changed, 1 insertion(+)
```

- Fast forward merge happened which moves the main to the latest commit of sprint 0 branch.
- Before Merge

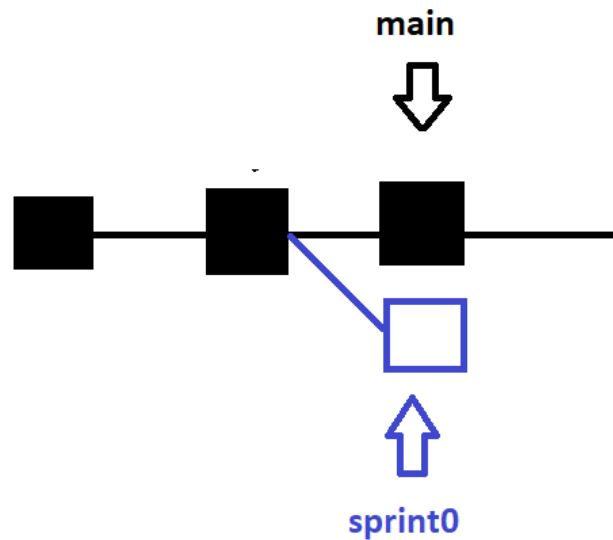


- After Merge



## Three Way Merge

- Lets create a commit on main branch and git graph should be as shown below



- Current view

```
Dell@DESKTOP-TM7SH71 MINGW64 /c/temp/gitpractice/branches_demo (main)
$ git log --oneline
053e156 (HEAD -> main) Added main file docs
2d84516 Added docs
649bed7 Added basic folder structure

Dell@DESKTOP-TM7SH71 MINGW64 /c/temp/gitpractice/branches_demo (main)
$ git checkout sprint0
Switched to branch 'sprint0'

Dell@DESKTOP-TM7SH71 MINGW64 /c/temp/gitpractice/branches_demo (sprint0)
$ git log --oneline
36c0cb5 (HEAD -> sprint0) Added change
2d84516 Added docs
649bed7 Added basic folder structure
```

- Lets merge the changes into main

```
Dell@DESKTOP-TM7SH71 MINGW64 /c/temp/gitpractice/branches_demo (main)
$ git merge sprint0
Auto-merging src/main.py
CONFLICT (content): Merge conflict in src/main.py
Automatic merge failed; fix conflicts and then commit the result.

Dell@DESKTOP-TM7SH71 MINGW64 /c/temp/gitpractice/branches_demo (main|MERGING)
#
```

- We get merge conflict as line 1 of src/main.py is conflicting
- now make the changes and add them to staging area and commit to create a merge commit

```
Dell@DESKTOP-TM7SH71 MINGW64 /c/temp/gitpractice/branches_demo (main|MERGING)
$ git status
On branch main
You have unmerged paths.
  (fix conflicts and run "git commit")
  (use "git merge --abort" to abort the merge)

Unmerged paths:
  (use "git add <file>..." to mark resolution)
    both modified:   src/main.py

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

Dell@DESKTOP-TM7SH71 MINGW64 /c/temp/gitpractice/branches_demo (main|MERGING)
$ git add src/main.py

Dell@DESKTOP-TM7SH71 MINGW64 /c/temp/gitpractice/branches_demo (main|MERGING)
$ git commit
[main 6054102] Merge branch 'sprint0'

Dell@DESKTOP-TM7SH71 MINGW64 /c/temp/gitpractice/branches_demo (main)
$ git log --oneline
6054102 (HEAD -> main) Merge branch 'sprint0'
```

- Merge commit is a special commit, which has two parents
- note: IF you are new to vi editor [Refer Here](#)

## Exercise

- Create a git repository
- create three files called as README.md in

- src
- test
- build
- create a commit
- create one more commit by editing README.md in src
- main will have two commits
- represent this in graph with HEAD and master in the graph
- Now create a branch sprint0 from master
- From sprint0 create two branches
  - sprint0\_poc
  - sprint0\_dev
- Update the same in graph
- Now checkout to sprint0\_poc and create a folder called as poc with Readme.me, update the graph
- Now checkout to sprint0\_dev and create a file src/main.py
- merge changes from sprint0\_poc tp sprint0\_dev
- Delete the sprint0\_dev

### Leave a Reply

Enter your comment here...

This site uses Akismet to reduce spam. [Learn how your comment data is processed.](#)



### About continuous learner

devops & cloud enthusiastic learner

[VIEW ALL POSTS](#)

◀ PREVIOUS POST

[Azure Classroomnotes 22/Feb/2023](#)

NEXT POST

[AWS Classroomnotes 22/Feb/2023](#)

POWERED BY [WORDPRESS.COM](#).