```
atweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (main)
git log --oneline --all --graph
df13941 (HEAD -> main, bug-fixing-02) Adding New File For Fast-Forward Merging
    076b09b Compare Changes For File-03
 * 048e6e4 (bug-fixing) Adding New Line For File-04
  10b7bed Test Branches For Bug Fixing
  9df8a5c Adding File-03 With Changes To Main Branch
 * 2d80b85 (refs/stash) On main: For Testing The VSCode With Branches
 * 0747d0d index on main: a6f4fa3 Restore File After Deleting
a6f4fa3 (tag: V1.1) Restore File After Deleting
 715de30 Deleting The New File: file-03.txt
 1b6514e Adding All Files
43a35dd Second Testing Of Files To Show Output
  5b550f Testing Short Status Command
 edc6933 Removing File From Index/Staging Area
 c047d34 Adding By Accident
014c4b9 Adding .gitignore To Project
 2df21af Trying Rename Files
 4702e82 After Deleting File2.txt
 c378284 Initial Commit
```

\*

#### If We Want To Create And Switch To Branch In One Command:

```
Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (main)
$ git switch -C bug-fixing-03
Switched to a new branch 'bug-fixing-03'

Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (bug-fixing-03)
$ |
```

\*

### If WE Want To Avoid Fast-Forward Merging:

- Run Command: git merge --no-ff branch-name-here
  - <u>Note</u>: In This Way The Editor Will Open To Make The Merge Commit.

```
Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (main)
$ git merge --no-ff bug-fixing-03
Merge made by the 'ort' strategy.
File-05.txt | 1 +
1 file changed, 1 insertion(+)

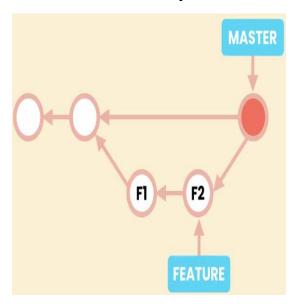
Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (main)
$ |
```

\*

```
atweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (main)
git log --oneline --all --graph
   6969780 (HEAD -> main) Merge branch 'bug-fixing-03' Into Main Branch.
 * 758dOb3 (bug-fixing-O3) Changes The File-O5.txt To Test Merging
 df13941 (bug-fixing-02) Adding New File For Fast-Forward Merging
   076b09b Compare Changes For File-03
 * 048e6e4 (bug-fixing) Adding New Line For File-04
   10b7bed Test Branches For Bug Fixing
   9df8a5c Adding File-03 With Changes To Main Branch
   2d80b85 (refs/stash) On main: For Testing The VSCode With Branches
  0747d0d index on main: a6f4fa3 Restore File After Deleting
 a6f4fa3 (tag: V1.1) Restore File After Deleting
 715de30 Deleting The New File: file-03.txt
 1b6514e Adding All Files
 43a35dd Second Testing Of Files To Show Output
 c5b550f Testing Short Status Command
 edc6933 Removing File From Index/Staging Area
 cO47d34 Adding By Accident
014c4b9 Adding .gitignore To Project
2df21af Trying Rename Files
4702e82 After Deleting File2.txt
 c378284 Initial Commit
atweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (main)
```

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Note: The Most Benefit Of --no-ff That We Can Return From Specific Commits To Another One If Any Problems Occur In The Current Version Of Project.



\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*



\*

<u>Note 1</u>: In This Way We Can Disable the Fast-Forward Merging For The Repository Only.



Note 2: And In This Way For All Repositories.

\*

To Show The Branches That Merged With Main-Branch:

```
Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (main)

$ git branch --merged
bug-fixing
bug-fixing-02
bug-fixing-03

* main
```

\*

#### To Delete Branch:

```
Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (main)
$ git branch -d bug-fixing
Deleted branch bug-fixing (was 048e6e4).
```

\*

To Get List Of Branches That Not Merged With Current Branch (*Here Is Main*):

Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (main)

\$ git branch --no-merged

\*

### Note (To Remember):

# **MERGES**

- Fast-forward (if branches have not diverged)
- 3-way (if branches have diverged)

\*

#### The Conflicts:

- 1st: The Same File Changes In The Two Branches.
- 2<sup>nd</sup>: The File Changed In One Branch, And Deleted On The Other.
- 3<sup>rd</sup>: The Same File Added In Two Branches.



\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#### To Abort Merge:

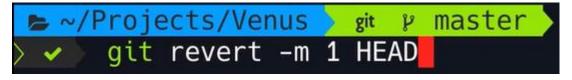
```
c ~/Projects/Venus git p master merge
did not merge --abort
c ~/Projects/Venus git p master
did not merge --abort
```

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*



\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

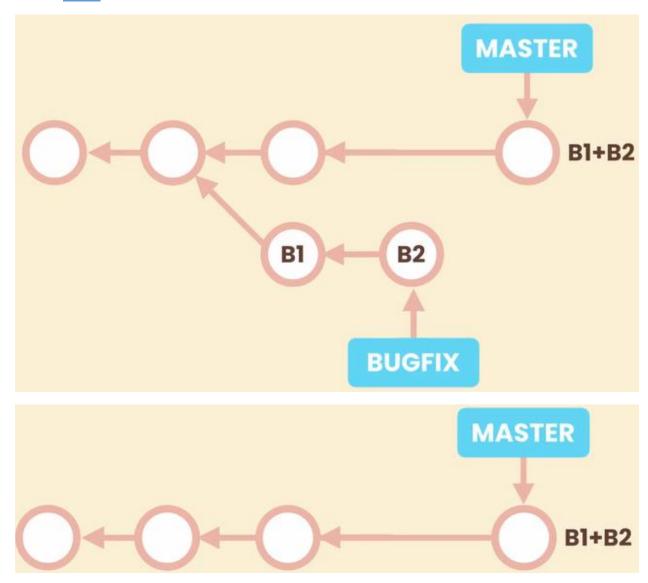
This Will Revert The Merge, And Go To First Parent (That is Main-Branch):



\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# That is Called Squash Merging:

- Combine The 2-Features Of Bug Fix, And Make Commit.
- Then Move The Pointer Of Master To That Commit.
- Then We Can Delete The BugFix Branch, And Get Simple Linear History.
- Note: Here We Must Use It For Small Features.



\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#### Here We Don't Commit Yet:

- Run Command: git merge --squash branch-name-here
  - o Note: We Must Be On The Main Branch.
- The New Branch Will Not Be In The List Of Merged Branches.

```
Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (main)
$ git merge --squash bugfix/photo-upload
Updating 6969780..e958bc4
Fast-forward
Squash commit -- not updating HEAD
file-01.txt | 1 +
file-02.txt | 1 +
file-03.txt | 4 ----
3 files changed, 2 insertions(+), 4 deletions(-)
```

```
Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (main)
$ git status -s
M file-01.txt
M file-02.txt
M file-03.txt
```

```
Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (main)

$ git branch --merged
bug-fixing-02
bug-fixing-03

* main
```

```
Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (main)
$ git branch --no-merged
bugfix/photo-upload
```

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

## Git Not See The Squash Merge As *Real Merge*:

- Note: We Must Use –D
- Note: The Benefit Of This To Get Simple Linear History.

```
Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (main)
$ git branch -d bugfix/photo-upload
error: the branch 'bugfix/photo-upload' is not fully merged
hint: If you are sure you want to delete it, run 'git branch -D bugfix/photo-upload'
hint: Disable this message with "git config advice.forceDeleteBranch false"

Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (main)
$ git branch -D bugfix/photo-upload
Deleted branch bugfix/photo-upload (was e958bc4).
```

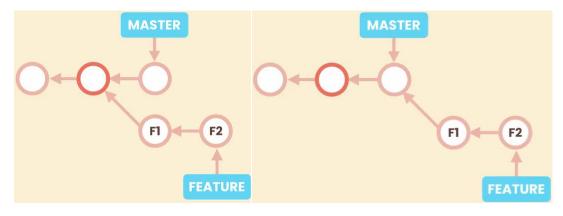
```
Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (main)
$ cat file-01.txt
hello
world
Jafar Loka
Jafar Loka Here
Jafar Loka -02
bugfix
```

\*

```
atweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (main)
git log --oneline --all --graph
 ba352bf (HEAD -> main) Fix The Bug On The Photo Upload Page.
  6969780 Merge branch 'bug-fixing-03' Into Main Branch.
 * 758d0b3 (bug-fixing-03) Changes The File-05.txt To Test Merging
 df13941 (bug-fixing-02) Adding New File For Fast-Forward Merging
  076b09b Compare Changes For File-03
 * 048e6e4 Adding New Line For File-04
  10b7bed Test Branches For Bug Fixing
  9df8a5c Adding File-03 With Changes To Main Branch
  2d80b85 (refs/stash) On main: For Testing The VSCode With Branches
  0747d0d index on main: a6f4fa3 Restore File After Deleting
 a6f4fa3 (tag: V1.1) Restore File After Deleting
 715de30 Deleting The New File: file-03.txt
 1b6514e Adding All Files
 43a35dd Second Testing Of Files To Show Output
 c5b550f Testing Short Status Command
 edc6933 Removing File From Index/Staging Area
 c047d34 Adding By Accident
014c4b9 Adding .gitignore To Project
 2df21af Trying Rename Files
4702e82 After Deleting File2.txt
 c378284 Initial Commit
```

\*

# Rebasing (Change The Base Of Features, So; We End With Linear History):



\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*