

If We Have Changes That Are Made In Specific Branch, And We Don't Committed It, Then We Can't Switch Branches.

```
Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (master)
$ echo "Jafar-Loka-Test-Master-Branch-01" >> file-03.txt

Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (master)
$ git switch bug-fixing
error: Your local changes to the following files would be overwritten by checkout:
      file-03.txt
Please commit your changes or stash them before you switch branches.
Aborting
```

To Save The Data Of Working Directory, In WIP Area:

- **Run Command:** *git stash push -m "Message Here"*

```
Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (master)
$ git stash push -m "Testing Stash-01"
warning: in the working copy of 'file-03.txt', LF will be replaced by CRLF the next time Git touches it
Saved working directory and index state On master: Testing Stash-01
```

If We Don't Add The *New File* To Staging Area, Then We Can't Stashing It:

- In This Situation We Can Use **-a** OR **--all**

```
Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (master)
$ touch file-04.txt

Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (master)
$ git status -s
?? file-04.txt

Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (master)
$ git stash push -m "Stashing-File-04"
No local changes to save
```

```
Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (master)
$ git stash push -m "Stashing-File-04"
No local changes to save

Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (master)
$ git stash push -a -m "Stashing-File-04"
warning: in the working copy of 'bin/file-02.txt', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'bin/test-01.txt', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'logs/test-01.txt', LF will be replaced by CRLF the next time Git touches it
Saved working directory and index state On master: Stashing-File-04
```

To View All Stashes We Can Use:

- *Run Command: git stash list*

```
Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (master)
$ git stash list
stash@{0}: On master: Stashing-File-04
stash@{1}: On master: Testing Stash-01
```

After Stashing, We Can Switch To Any Branch:

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

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

To Show The Changes In Specific Stash For Master Branch:

- *Run Command: git stash show stash-number-here*
 - *Ex: git stash show 1*

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

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

Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (master)
$ git stash show 1
file-03.txt | 1 +
1 file changed, 1 insertion(+)

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

To Apply Stash To The New Directory:

- Run Command: *git stash apply stash-number-here*
 - Ex: *git stash apply 1*

```
Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (master)
$ git stash apply 1
On branch master
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:   file-03.txt

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

Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (master)
$ ls
file-01.txt  file-02.txt  file-03.txt

Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (master)
$ git stash list
stash@{0}: On master: Stashing-File-04
stash@{1}: On master: Testing Stash-01

Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (master)
$ git stash apply 0
Already up to date.
On branch master
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:   file-03.txt

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        file-04.txt

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

Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (master)
$ ls
bin/  file-01.txt  file-02.txt  file-03.txt  file-04.txt  logs/

Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (master)
$ git status -s
 M file-03.txt
?? file-04.txt

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

To Drop Stash From List:

- *Run Command: `git stash drop stash-number`*
 - *Ex: `git stash drop 1`*

```
Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (master)
$ git stash list
stash@{0}: On master: Stashing-File-04
stash@{1}: On master: Testing Stash-01

Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (master)
$ git stash drop 1
Dropped refs/stash@{1} (4c634588aee6145adc1550edf872a85efc01de2)

Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (master)
$ git stash list
stash@{0}: On master: Stashing-File-04

Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (master)
$ git stash drop 0
Dropped refs/stash@{0} (08262e3f437b63748db85f9d80b97299a595573a)

Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (master)
$ git stash list

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

To Remove All Stashes:

- *Run Command: `git stash clear`*

Note (To Remember): Git All Commits That In bug-fixing Not In Main.

```
Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (main)
$ git log main..bug-fixing
commit 10b7bedbb0f7f498ed411d562afbb946f871c205 (bug-fixing)
Author: Jafar Loka <joyhess501@gmail.com>
Date: Thu Nov 28 10:05:20 2024 +0300

    Test Branches For Bug Fixing

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

Note (To Remember): To Show The Difference Between main branch and bug-fixing branch.

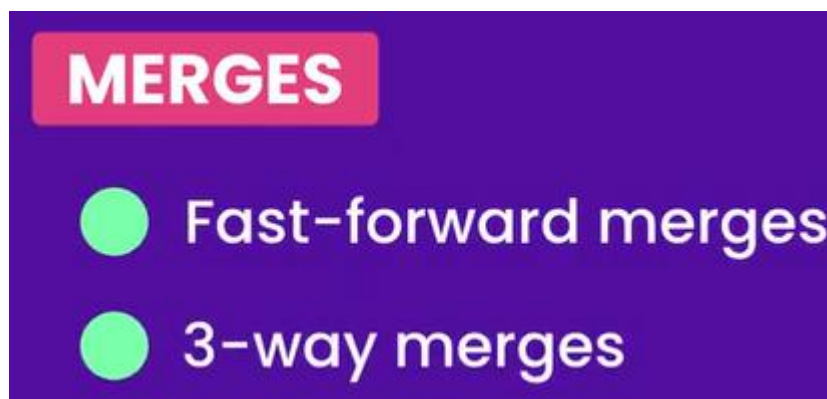
```
Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (main)
$ git diff main..bug-fixing
diff --git a/file-03.txt b/file-03.txt
index 557db03..f05f23c 100644
--- a/file-03.txt
+++ b/file-03.txt
@@ -1,2 @@
 Hello World
+Jafar Loka Test Branches-01
```

Note (To Remember): If We Are On the Main Branch, We Can Omit The First Part.

Note 2 (To Remember): Also We Can Use *--name-only* And *--name-status*.

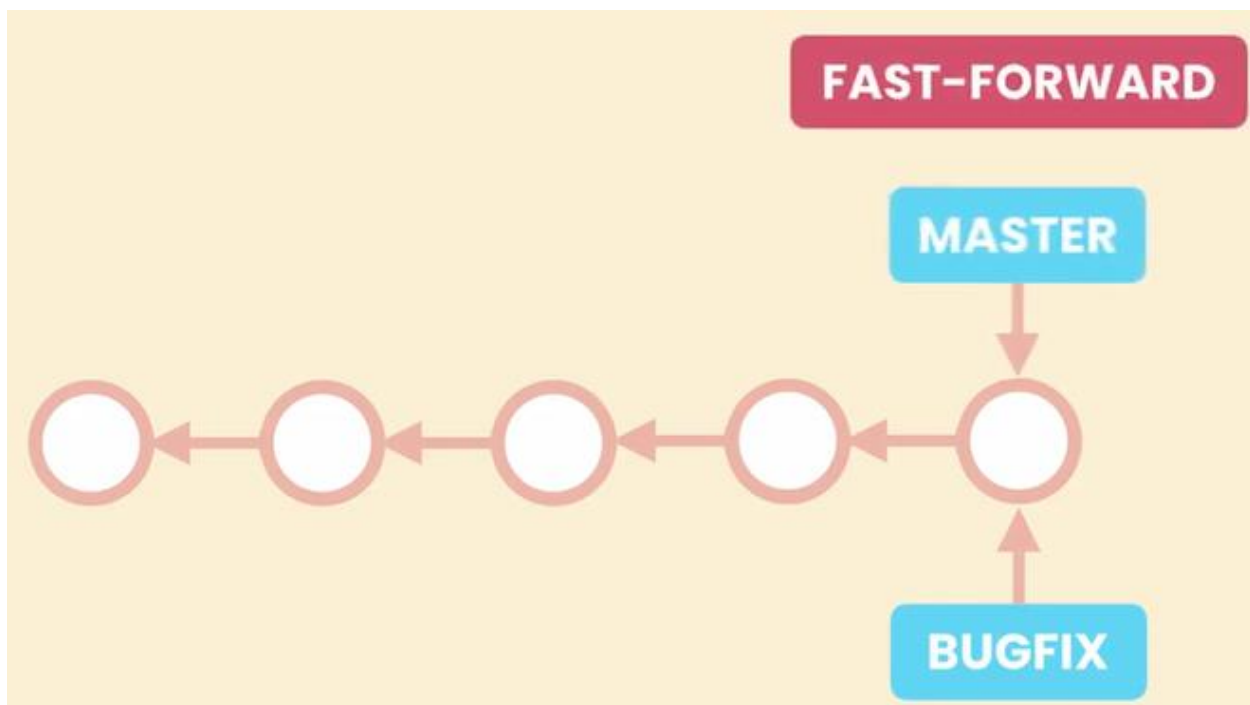
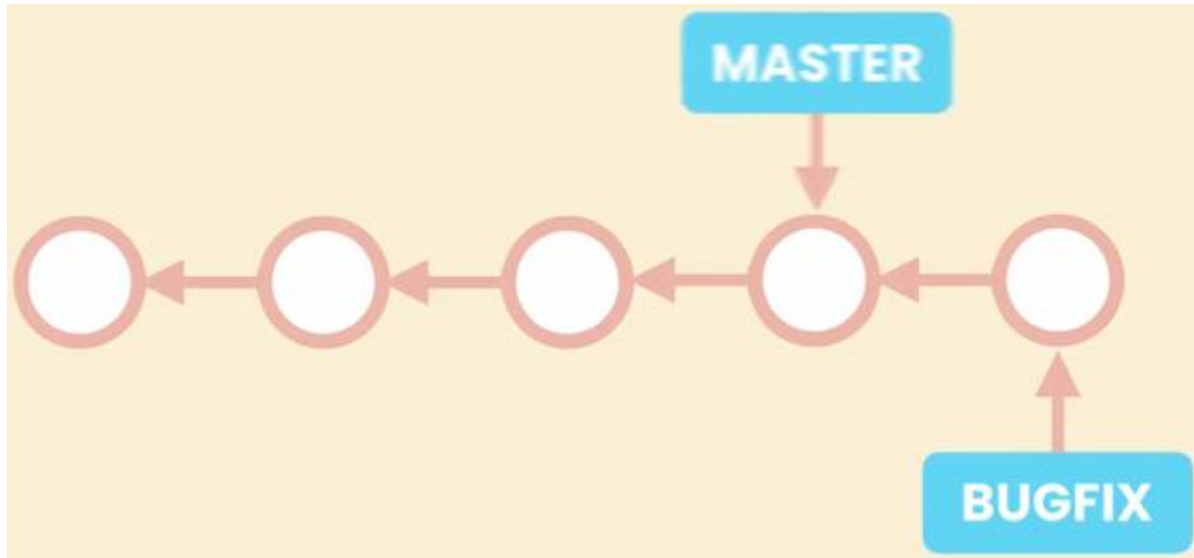
```
Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (main)
$ git diff bug-fixing
diff --git a/file-03.txt b/file-03.txt
index f05f23c..664087e 100644
--- a/file-03.txt
+++ b/file-03.txt
@@ -1,2 +1,2 @@
 Hello World
-Jafar Loka Test Branches-01
+Jafar-Loka-Test-Master-Branch-01
```

Stash Meaning: Save Something In Safe Place To Make The Changes Right. (Like To Switch To Another Branch Without Commit The Current Work).

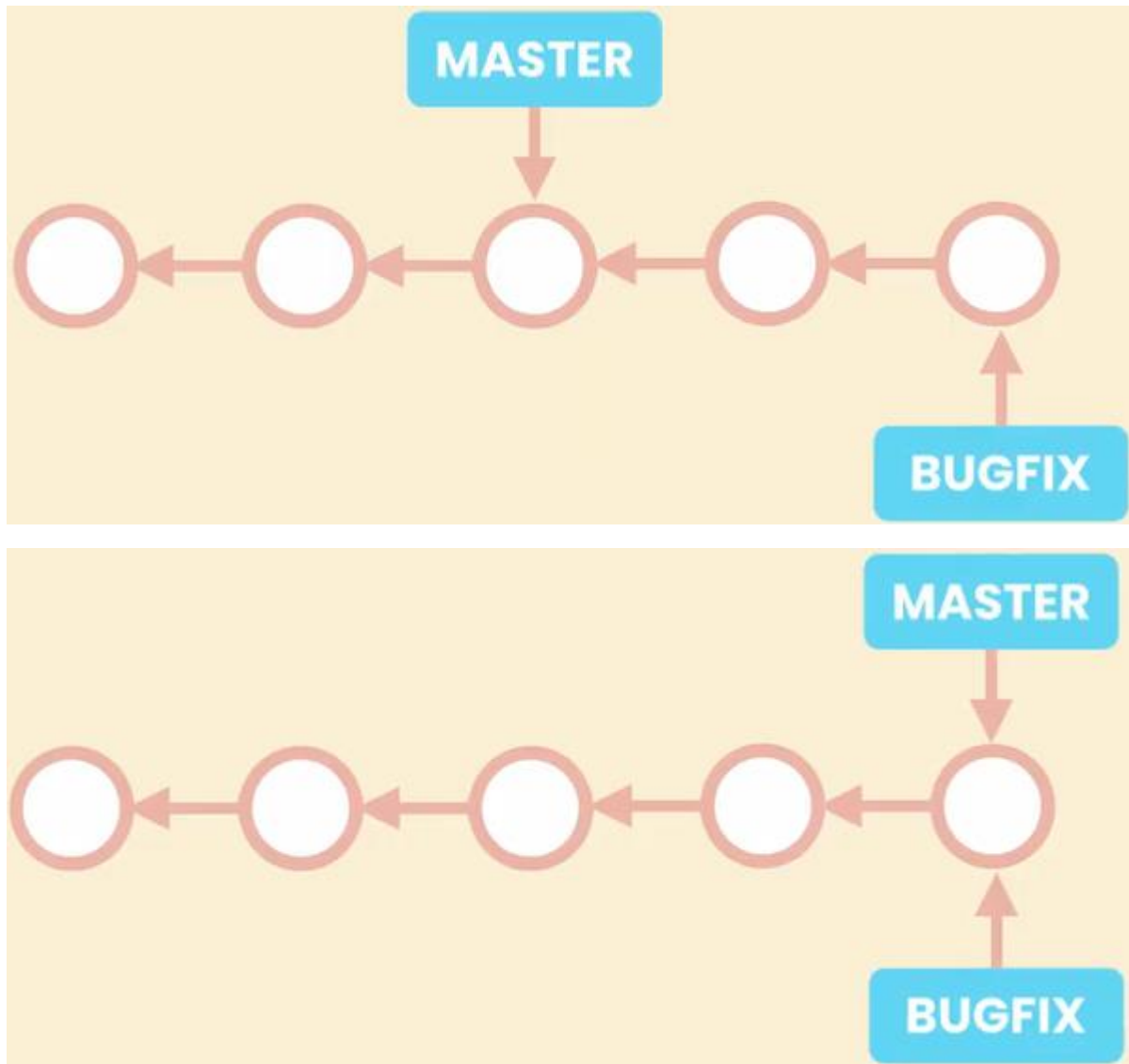


In This Way We Move The Pointer To The Branch Pointer.

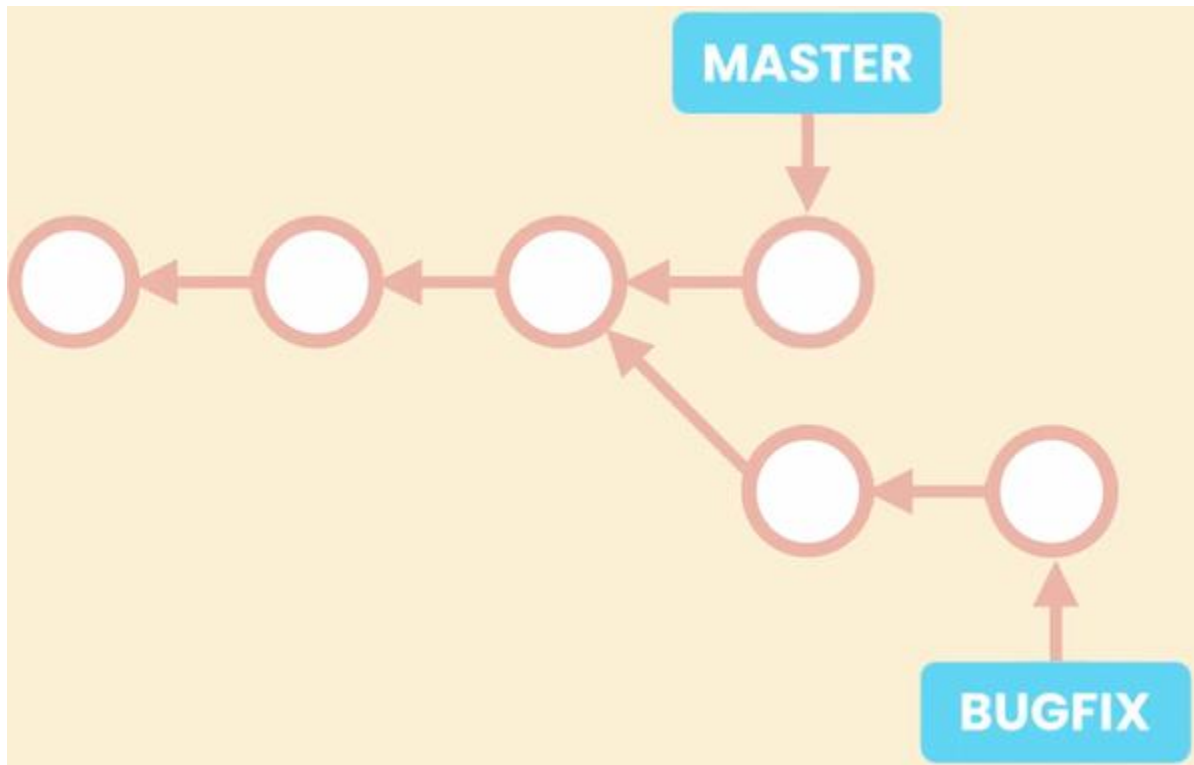
In This Way We Can Avoid The Problem Of Copying Files From Version To Another (Ex: If We Have 1000-Files And Directories).



Here We Have Linear Relation Between Branches.

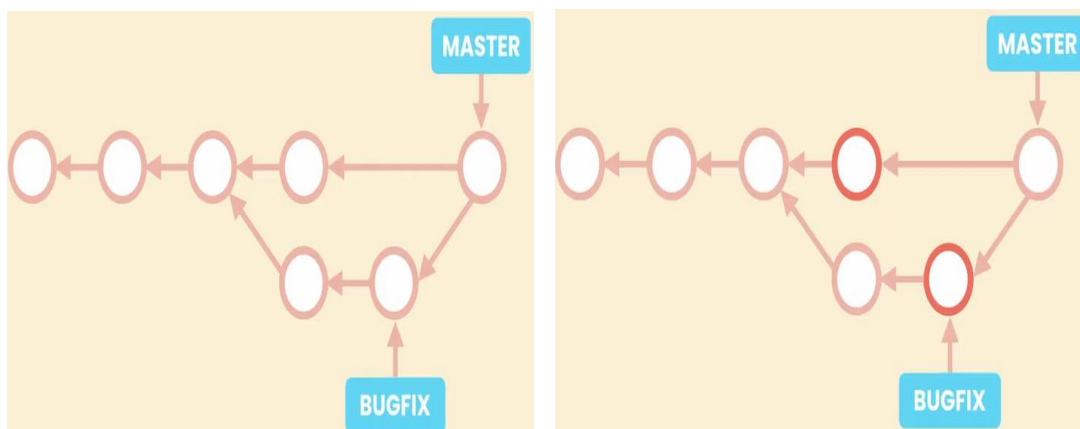


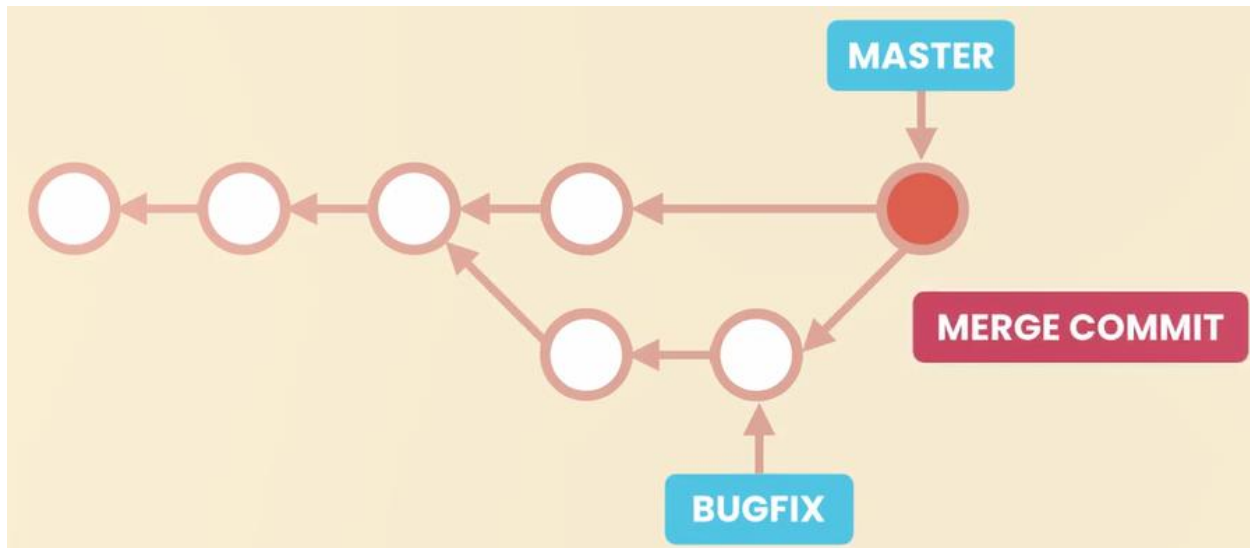
Here We Have Changes In Both Master And BUGFIX:



Here We Create New Commit That Combines That Changes In Two Branches.

Here In This Way We Compare The Changes Between Before & After Snapshots.





MERGES

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

```
Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (main)
$ git merge bug-fixing
Auto-merging file-03.txt
CONFLICT (content): Merge conflict in file-03.txt
Automatic merge failed; fix conflicts and then commit the result.

Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (main|MERGING)
$ git status -s
UU file-03.txt
A file-04.txt

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

Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (main|MERGING)
$ git status -s
M file-03.txt
A file-04.txt

Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (main|MERGING)
$ git commit -m "Compare Changes For File-03"
[main 076b09b] Compare Changes For File-03

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

Here We Create New Branch And Do Only One Change Linear Of Main Branch.

```
Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (bug-fixing-02)
$ echo "Hello World From Jafar Loka-01" >> File-05.txt

Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (bug-fixing-02)
$ git status -s
?? File-05.txt

Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (bug-fixing-02)
$ git add .
warning: in the working copy of 'File-05.txt', LF will be replaced by CRLF the next time Git touches it

Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (bug-fixing-02)
$ git commit -m "Adding New File For Fast-Forward Merging"
[bug-fixing-02 df13941] Adding New File For Fast-Forward Merging
1 file changed, 1 insertion(+)
create mode 100644 File-05.txt

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

Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (main)
$ git log --oneline --all --graph
* df13941 (bug-fixing-02) Adding New File For Fast-Forward Merging
* 076b09b (HEAD -> main) 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
* c047d34 Adding By Accident
* 014c4b9 Adding .gitignore To Project
* 2df21af Trying Rename Files
* 4702e82 After Deleting File2.txt
* c378284 Initial Commit

Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Github-And-GitLab/Tutorial-02-Code-With-Mosh (main)
$ git merge bug-fixing-02
Updating 076b09b..df13941
Fast-forward
```