

## Section 7 : Cloning, exploring and modifying public repositories

### Cloning remote repository

The screenshot shows a GitHub repository page for 'leachim6/hello-world'. The 'Code' tab is selected. A context menu is open over a commit history entry, with the 'Download ZIP' option highlighted by a yellow circle.

```
+ ~ cd /Users/hemakshipandey/Downloads/hello-world-master
+ hello-world-master ls -la
total 88
drwxrwxr-x@ 75 hemakshipandey staff 2400 Aug 16 04:55 #
drwxrwxr-x@ 36 hemakshipandey staff 1152 Aug 16 04:55 .
drwx-----+ 9 hemakshipandey staff 288 Aug 17 13:27 ..
-rw-rw-r--@ 1 hemakshipandey staff 208 Aug 16 04:55 .editorconfig
drwxrwxr-x@ 4 hemakshipandey staff 128 Aug 16 04:55 .github
-rw-rw-r--@ 1 hemakshipandey staff 1296 Aug 16 04:55 .gitignore
drwxrwxr-x@ 96 hemakshipandey staff 3072 Aug 16 04:55 a
drwxrwxr-x@ 45 hemakshipandey staff 1440 Aug 16 04:55 b
drwxrwxr-x@ 56 hemakshipandey staff 1792 Aug 16 04:55 c
drwxrwxr-x@ 1 hemakshipandey staff 1065 Aug 16 04:55 contributing.md
drwxrwxr-x@ 38 hemakshipandey staff 1216 Aug 16 04:55 d
drwxrwxr-x@ 25 hemakshipandey staff 800 Aug 16 04:55 e
drwxrwxr-x@ 35 hemakshipandey staff 1120 Aug 16 04:55 f
drwxrwxr-x@ 31 hemakshipandey staff 992 Aug 16 04:55 g
drwxrwxr-x@ 26 hemakshipandey staff 832 Aug 16 04:55 h
drwxrwxr-x@ 22 hemakshipandey staff 704 Aug 16 04:55 i
drwxrwxr-x@ 18 hemakshipandey staff 576 Aug 16 04:55 j
drwxrwxr-x@ 12 hemakshipandey staff 384 Aug 16 04:55 k
drwxrwxr-x@ 33 hemakshipandey staff 1056 Aug 16 04:55 l
-rw-rw-r--@ 1 hemakshipandey staff 1087 Aug 16 04:55 license
drwxrwxr-x@ 42 hemakshipandey staff 1344 Aug 16 04:55 m
drwxrwxr-x@ 17 hemakshipandey staff 544 Aug 16 04:55 n
drwxrwxr-x@ 23 hemakshipandey staff 736 Aug 16 04:55 o
drwxrwxr-x@ 55 hemakshipandey staff 1760 Aug 16 04:55 p
drwxrwxr-x@ 13 hemakshipandey staff 416 Aug 16 04:55 q
drwxrwxr-x@ 36 hemakshipandey staff 1152 Aug 16 04:55 r
-rw-rw-r--@ 1 hemakshipandey staff 23234 Aug 16 04:55 readme.md
drwxrwxr-x@ 60 hemakshipandey staff 1920 Aug 16 04:55 s
drwxrwxr-x@ 27 hemakshipandey staff 864 Aug 16 04:55 t
drwxrwxr-x@ 10 hemakshipandey staff 320 Aug 16 04:55 u
-rw-rw-r--@ 1 hemakshipandey staff 1493 Aug 16 04:55 update_list.py
drwxrwxr-x@ 24 hemakshipandey staff 768 Aug 16 04:55 v
drwxrwxr-x@ 14 hemakshipandey staff 448 Aug 16 04:55 w
drwxrwxr-x@ 16 hemakshipandey staff 512 Aug 16 04:55 x
drwxrwxr-x@ 7 hemakshipandey staff 224 Aug 16 04:55 y
drwxrwxr-x@ 15 hemakshipandey staff 480 Aug 16 04:55 z
+ hello-world-master
```

**NOTE: If you choose Download repository option at GitHub, you will download only contents of the working directory.**

**Git repository (hidden .git folder) will not be downloaded in such case.**

```

+ ~ cd Desktop
+ Desktop git clone https://github.com/leachim6/hello-world.git
Cloning into 'hello-world'...
remote: Enumerating objects: 6398, done.
remote: Counting objects: 100% (754/754), done.
remote: Compressing objects: 100% (395/395), done.
remote: Total 6398 (delta 421), reused 615 (delta 319), pack-reused 5644
Receiving objects: 100% (6398/6398), 2.89 MiB / 13.51 MiB/s, done.
Resolving deltas: 100% (3191/3191), done.
warning: the following paths have collided (e.g. case-sensitive paths
on a case-insensitive filesystem) and only one from the same
colliding group is in the working tree:

'd/DOG'
'd/Dog'
+ Desktop ls
Relocated Items.nosync      first-project
Screenshot 2021-08-17 at 1.31.43 PM.png hello-world
Screenshot 2021-08-17 at 1.34.00 PM.png
+ Desktop cd hello-world
+ hello-world git:(master) x ls -la
total 88
drwxr-xr-x@ 75 hemakshipandey staff 2400 Aug 17 13:50 #
drwxr-xr-x@ 37 hemakshipandey staff 1184 Aug 17 13:50 .
drwxr-xr-x@ 9 hemakshipandey staff 288 Aug 17 13:50 ..
-rw-r--r-- 1 hemakshipandey staff 208 Aug 17 13:49 .editorconfig
drwxr-xr-x@ 12 hemakshipandey staff 384 Aug 17 13:50 .git
drwxr-xr-x@ 4 hemakshipandey staff 128 Aug 17 13:50 .github
-rw-r--r-- 1 hemakshipandey staff 1296 Aug 17 13:49 .gitignore
drwxr-xr-x@ 96 hemakshipandey staff 3072 Aug 17 13:50 a
drwxr-xr-x@ 45 hemakshipandey staff 1440 Aug 17 13:50 b
drwxr-xr-x@ 56 hemakshipandey staff 1792 Aug 17 13:50 c
-rw-r--r-- 1 hemakshipandey staff 1065 Aug 17 13:49 contributing.md
drwxr-xr-x@ 38 hemakshipandey staff 1216 Aug 17 13:50 d
drwxr-xr-x@ 25 hemakshipandey staff 800 Aug 17 13:50 e
drwxr-xr-x@ 35 hemakshipandey staff 1120 Aug 17 13:50 f
drwxr-xr-x@ 31 hemakshipandey staff 992 Aug 17 13:50 g
drwxr-xr-x@ 26 hemakshipandey staff 832 Aug 17 13:50 h
drwxr-xr-x@ 22 hemakshipandey staff 704 Aug 17 13:50 i
drwxr-xr-x@ 18 hemakshipandey staff 576 Aug 17 13:50 j

```

## Exploring contents of the cloned repository

```

+ hello-world git:(master) x git log
[1] + 1321 suspended git log
+ hello-world git:(master) x cat .git/HEAD
ref: refs/heads/master
+ hello-world git:(master) x cat .git/refs/heads/master
2a885c1cfe23f95d6155c8da8a62eda5221ed1d9
+ hello-world git:(master) x git cat-file -p 2a885c1
tree 2a377c17acd1df19cbc8a5f029b70b6e656816
parent 4cbf9ed3ddfa76ecff0246bc1c9368e797f3f3af
author MrBrain295 <66077254+MrBrain295@users.noreply.github.com> 1629069946 -0500
committer GitHub <noreply@github.com> 1629069946 +1200
gpgsig -----BEGIN PGP SIGNATURE-----
ws8cBAABCQAAQJhGaJ6CRBK7hj40v3rIwAALdYIACVYcR1chKsndnCr7AMubVMP
d091JghY7N7Qs6Gpn1VHpdtxLQ9zrMnxkjVgfbZcyIPB01nQriu1sdd4J2nJN0y
y789Z7p2rflCxiaDXHvJfjsJsiSocktTUhqgSh0htdUtDH84RfgPZD17bagJ4l2
B/FweyyJmmxs0X0XvMqdPShVLxp588toxrtgFws+WQjRtlD6skwJRcb9oRuW//CA
Abf2auqvElM6g85y+mIIkAsiuSdeH7JvnXefguHJTn7TiizEfwCs0eeKJS8pR0rL
aJHd+aNVqfo+rMwa3p1+DzZMxnDVewH7VCLSpQp0Y/pMYG0NEgOpPDL/Xp6gY90=
=t+k/
-----END PGP SIGNATURE-----

Add Mostawesomeprogramminglanguage (#1126)

Co-authored-by: Richie Bendall <richiebendall@gmail.com>
+ hello-world git:(master) x git cat-file -p 2a377c1
040000 tree 4d7d25d20f866a5abf0cabf4001984924d5ab2e0  #
100644 blob 38cb39efc115c1462f0f9dd63439ca61883068ef .editorconfig
040000 tree 71561216a51078ebf607161021e43e9374c278be .github
100644 blob 3913133eb088b09a9d90e7548859e90b51ca9e7 .gitignore
040000 tree d688a60145fcbb3ad1f2ce4af3b9fdb12a8bc75 a
040000 tree d0f5696ee1b8a6cc356349179535d735f721d62f b
040000 tree 3939183f8badee0d652b5762cleccc6430335d1 c
100644 blob ada91c2f47be2573d85ba05d004a902786b86ad contributing.md
040000 tree 93b864711e2f97889598f582ec40ad3cc39d563 d
040000 tree f9d419a79c4e64e0374a93a7523b8c6e3484ef3b e
040000 tree e15147284124a7b92fc6ba72c34a104c6acfef547 f
040000 tree 63c8ce31697ea9b52c1ded15c25f7d770edc2d66 g

```

```

040000 tree 71561216a51078ebf607161021e43e9374c278be .github
100644 blob 3913133eb088b09a9d90e75548859e90b51ca9e7 .gitignore
040000 tree cd68a60145fc33ad1f2ce4af3b9fdb12a8bc75 a
040000 tree d0f5696ee1b8adcc356349179535d735f721d62f b
040000 tree 3939183f8badee06d652b5762c1ecc6430335d1 c
100644 blob ada91c2f47be2573d85ba05d004aa902786b86ad contributing.md
040000 tree 93b8864711e2f97889598f582ec40ad3ccc39d563 d
040000 tree f9d419a79c4e64e0374a93d7523b8c6e3484ef3b e
040000 tree e15147284124a7b92fc6ba72c34a104c6acf547 f
040000 tree 63c8ce31697ea9b52c1ded15c25f7d770edc2d66 g
040000 tree 69200c60ed7ffbc90b1be6a9d6a7371ef24da445 h
040000 tree d2d35d9c0d0b2abf0b712e2793c609b0b25231e i
040000 tree 49b38c5584f56a1d115d2ee69d29eaa0f1f28d1f5 j
040000 tree f9ce5b63a943ac9dc9fd8a0ad9a5d71e2b443c5a k
040000 tree a23ca071f9a0ce0a98ec2aa83e4ae557df315b l
100644 blob 5d8f55742d01f91406efed297f61a6295d18153 license
040000 tree 1e1c7bd8ec813ca651e289a0cdc4f9efe83036bc m
040000 tree d8969c8e87e0ebb157643d8e9280d4b97af8abe7 n
040000 tree b2afe75fde8b981b2a4a324f8ff4bcd9e3934d6d o
040000 tree ed807e28067ac596a499b4562c57dced949041b2 p
040000 tree a39d77b23b070867d17486b5ead05fe7ca143930 q
040000 tree 7c1d54a94b6361b41768cc28190b171cb7a5ceb r
100644 blob c377facb4ed01fa0638e7b15d86174ba54f9d1a readme.md
040000 tree 74ea3bfe8fd2fffa4f2ad4fc22deb08ea9e420ff s
040000 tree f820169e04418532b85b7861b29f0b59ba7c0e1 t
040000 tree 2e16490dce6b272cff028b832685491d8c8169 u
100644 blob 25c02c86bc65f0f93c6c9d9c66fff3e39f2b5b02 update_list.py
040000 tree d0eb2c9d615da2e7b3d45b95bbce1905a010b2a v
040000 tree 203c9a285123e2bb91ff1075e2079041dcf40cba w
040000 tree b9abd145353ce1f5c5c43af782ed90260150cc8 x
040000 tree 87f0b608fb60f916d0fd46aec7b0f8e0a4f9e75 y
040000 tree 2a2f7953da8c6e4b108b76b6b7960c5945d61b4 z
+ hello-world git:(master) ✘ git cat-file -p cd688a
100644 blob ff495572c433563fd45605369e3eb3fa91c0a647 A++
100644 blob 50101979d046be0e60fd3997c3c3966f2172293f ABAP.abap
100644 blob 61be8719d2826911517bd1722d48acbe5668a443 ABC.abc
100644 blob e88e4c0712cc97f06401ebed1b526aa0cc01ecf8 ACIDIC.acidic
100644 blob ca5794514bcfa4eff4b4870e57a1dd97802b8f6 ACRONYM.acronym
100644 blob a32ef6da7c412da16aaeeddb61db063b36ac7d8 ACS.acs
100644 blob 35f8dc1aad2566fa395082f8843454088bd6c267 ALAGUF.alaguf

```

```

100644 blob fid138169a78f1a8b2da9df94ee75dd2a0d8545e AutoIt.au3
100644 blob db0ff23c397132a53cc0fa864bb12bc984f50e5c AutoLISP.lsp
100644 blob 5101979d046be0e60fd3997c3c3966f2172293f Avisynth.avs
100644 blob 61be8719d2826911517bd1722d48acbe5668a443 Awful.yuk
100644 blob 3f4f4961b1c00d6e55cd7bc8a23d98335621465e Axe.axe
100644 blob b5aa15b4a73487593e1ae624b962dd96167bf7 "A\313\270;""
+ hello-world git:(master) ✘ git cat-file -p 958e15
global start

section .text
start:
    mov        rax, 0x2000004
    mov        rdi, 1
    mov        rsi, msg
    mov        rdx, msg.len
    syscall

    mov        rax, 0x2000001
    mov        rdi, 0
    syscall

section .data
msg:   db      "Hello World", 10
.len:  equ    $ - msg
+ hello-world git:(master) ✘ git cat-file -p 61be87
#!/usr/bin/awful

:writeln s"Hello World"
+ hello-world git:(master) ✘ ls
#          contributing.md  g      k      n      r      u      x
a          d                  h      l      o      readme.md  update_list.py  y
b          e                  i      license  p      s      v      z
c          f                  j      m      q      t
+ hello-world git:(master) ✘ cat license
MIT License

Copyright (c) 2018 - 2021 Mike Donaghy [leachim6]

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal

```

```

.len: equ    $ - msg
+ hello-world git:(master) ✘ git cat-file -p 61be87
#!/usr/bin/awful

:writeln s"Hello World"
+ hello-world git:(master) ✘ ls
#          contributing.md  k      n      r      u      x
#          d                  h      l      o      p      s      t
#          e                  i      m      q      v      w
#          f                  j
+ hello-world git:(master) ✘ cat license
MIT License

Copyright (c) 2018 - 2021 Mike Donaghy [leachim6]

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
+ hello-world git:(master) ✘ cat update_list.py
#!/usr/bin/env python3

import os
import posixpath
from urllib.parse import quote
import re

```

```

SOFTWARE.
+ hello-world git:(master) ✘ cat update_list.py
#!/usr/bin/env python3

import os
import posixpath
from urllib.parse import quote
import re

def regexReplace(string, search, replacement):
    return re.compile(search).sub(replacement, string)

languageCount = 0
languagesText = ""

# List the available languages
for directory in sorted(os.listdir('.')):
    if not (directory == '.' or directory == '..' or directory[0] == '.' or os.path.isfile(directory)):
        for filename in sorted(os.listdir(directory), key=lambda s: s.lower()):
            if os.path.isfile(os.path.join(directory, filename)):
                language = (os.path.splitext(filename)[0]
                           .replace("-", "-")
                           .replace("/", "/")
                           .replace("\ ", "\\")

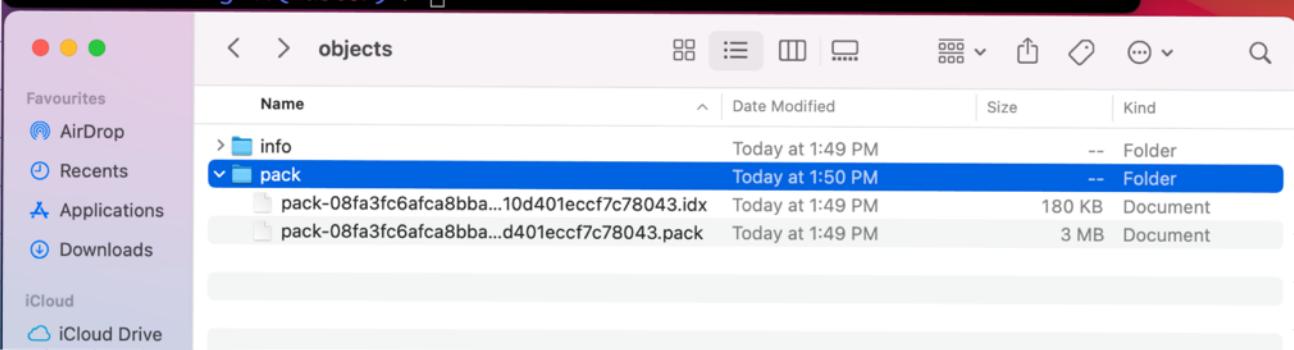
                           .replace(":", ":")
                           .replace("*", "*")
                           .replace("?", "?")
                           .replace(" ", "\ ")
                           .replace("<", "<")
                           .replace(">", ">")
                           .replace("!", "!"))
                languagesText += f'* [{language}]{(posixpath.join(quote(directory), quote(filename)))}\n'
                languageCount += 1

result = f'''<!--Languages start-->
## Languages ({languageCount} total)
{languagesText}<!--Languages end-->'''

```

# Unpacking Git Object

```
→ hello-world git:(master) ✘ open .  
→ hello-world git:(master) ✘
```



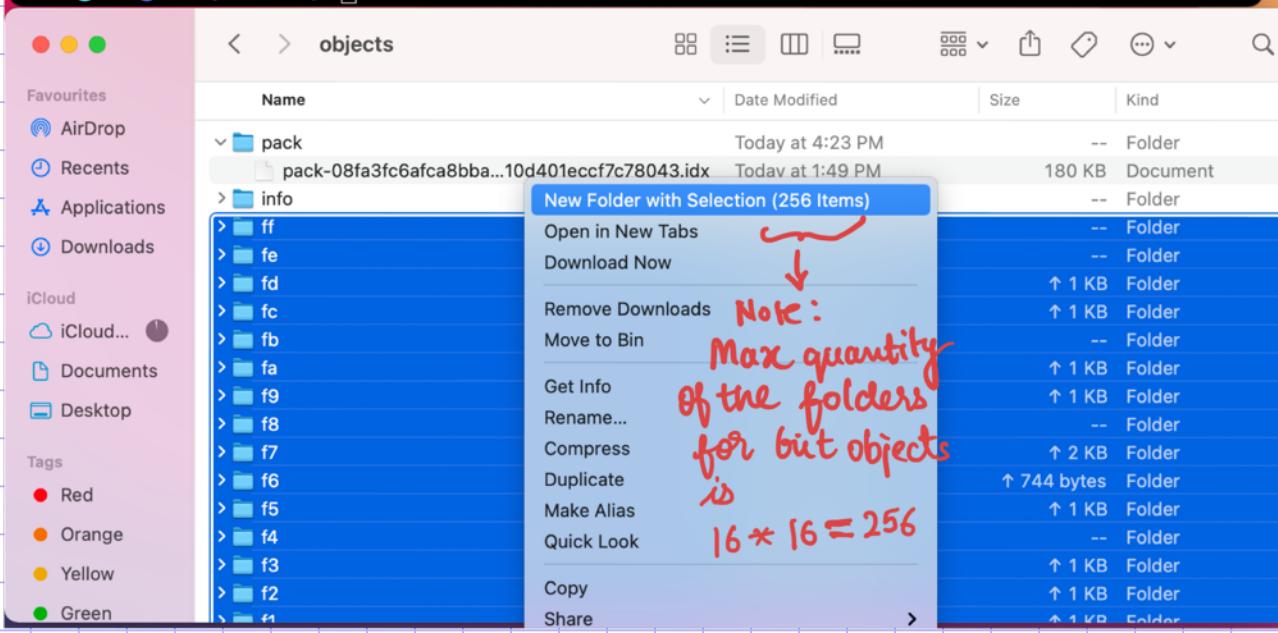
→ Git has the ability to merge together multiple objects into single files known as pack files.

→ These are, in essence, multiple objects stored with an efficient data compression scheme as a single compressed file.

## Action plan for unpacking .pack file

1. Move .pack file out of the pack folder.
2. Use "git unpack-objects" command.

```
→ hello-world git:(master) ✘ open .  
→ hello-world git:(master) ✘ cd .git  
→ .git git:(master) cat pack-08fa3fc6afca8bba42ab07510d401eccf7c78043.pack | git unpack-objects  
Unpacking objects: 100% (6398/6398), 2.89 MiB | 1.24 MiB/s, done.  
→ .git git:(master)
```

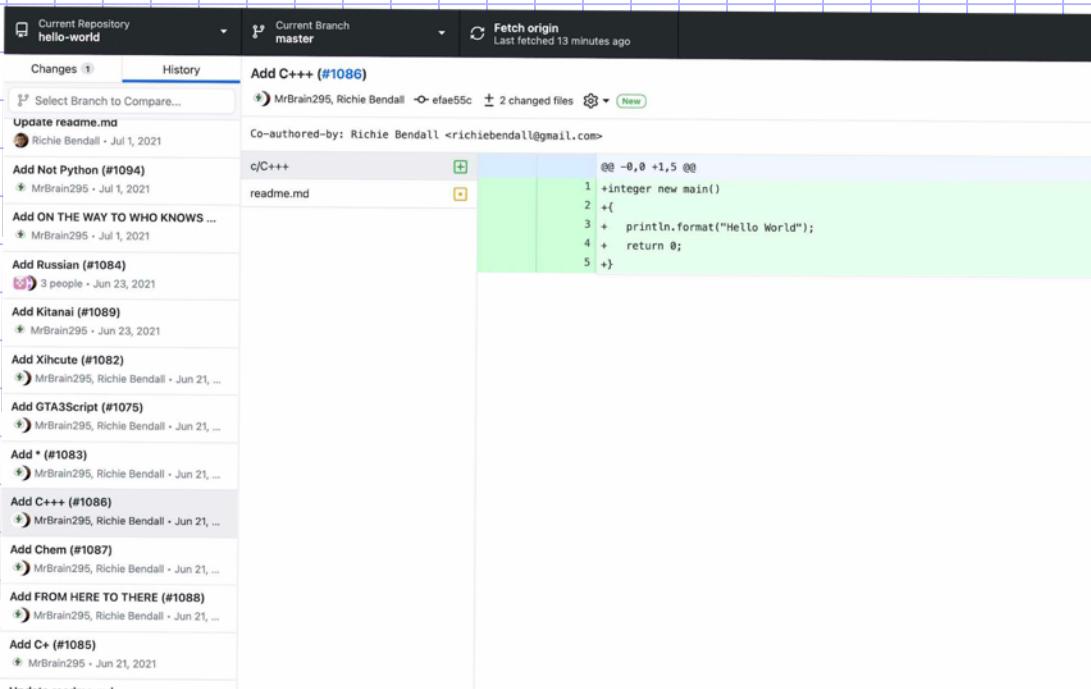
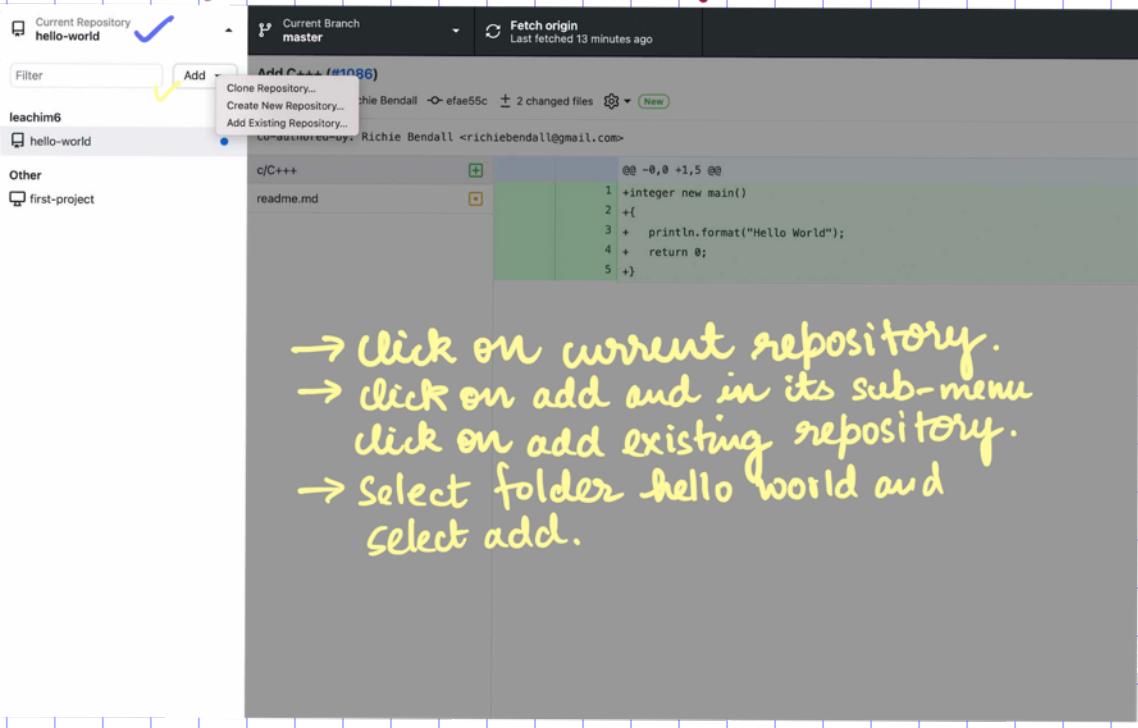


```

→ hello-world git:(master) ✘ open .
→ hello-world git:(master) ✘ cd .git
→ .git git:(master) cat pack-08fa3fc6afca8bba42ab07510d401eccf7c78043.pack | git unpack-objects
Unpacking objects: 100% (6398/6398), 2.89 MiB / 1.24 MiB/s, done.
→ .git git:(master) rm pack-08fa3fc6afca8bba42ab07510d401eccf7c78043.pack
override r--r-- hemakshipandey/staff for pack-08fa3fc6afca8bba42ab07510d401eccf7c78043.pack?
→ .git git:(master) ls
HEAD
config
description
hooks
index
info
logs
objects
pack-08fa3fc6afca8bba42ab07510d401eccf7c78043.pack
packed-refs
refs
→ file is removed → objects packed-refs refs
→ .git git:(master) ls
HEAD config description hooks index info logs objects packed-refs refs
→ .git git:(master) 

```

## Exploring cloned repository in GitHub Desktop



```
→ .git git:(master) clear
→ .git git:(master) cd ..
→ hello-world git:(master) x ls -l
total 72
drwxr-xr-x@ 75 hemakshipandey  staff  2400 Aug 17 13:50 #
drwxr-xr-x@ 96 hemakshipandey  staff  3072 Aug 17 13:51 a
drwxr-xr-x@ 45 hemakshipandey  staff  1440 Aug 17 13:51 b
drwxr-xr-x@ 56 hemakshipandey  staff  1792 Aug 17 13:51 c
-rw-r--r--  1 hemakshipandey  staff  1065 Aug 17 13:49 contributing.md
drwxr-xr-x@ 38 hemakshipandey  staff  1216 Aug 17 13:51 d
drwxr-xr-x@ 25 hemakshipandey  staff  800 Aug 17 13:51 e
drwxr-xr-x@ 35 hemakshipandey  staff  1120 Aug 17 13:51 f
drwxr-xr-x@ 31 hemakshipandey  staff  992 Aug 17 13:51 g
drwxr-xr-x@ 26 hemakshipandey  staff  832 Aug 17 13:51 h
drwxr-xr-x@ 22 hemakshipandey  staff  704 Aug 17 13:51 i
```

```
→ hello-world git:(master) x cat readme.md
# Hello World

Hello World in every computer language.

As I watch the collection expand, this project has blown up more than I ever thought possible.
Thanks to everyone who continues to contribute; new languages are created every day!

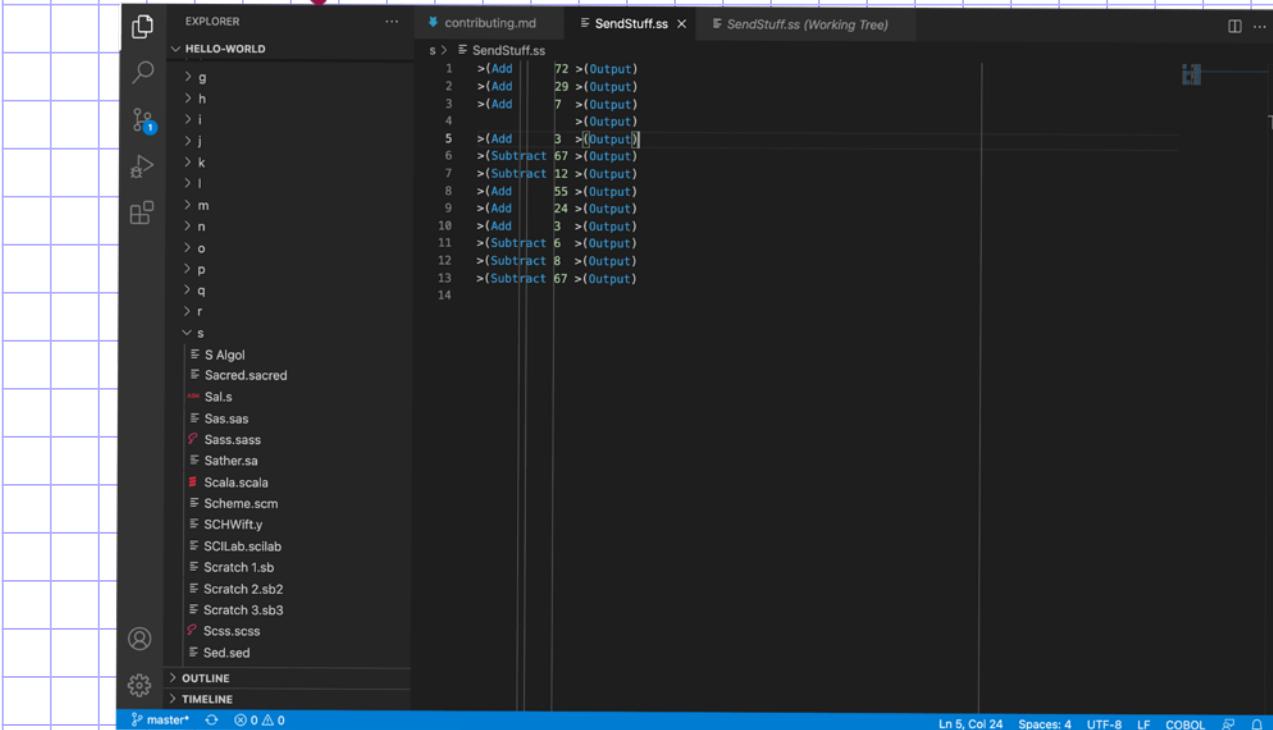
Make sure to see [contributing.md](/contributing.md) for instructions on contributing to the project!

<!--Languages start-->
## Languages (816 total)

* [*!@#$%^&*()_+](%23/%21%40%23%24%25%5E%26%E2%88%97%28%29_%2B)
* [*?51=](%23/?%2351=)
* [*():;#!](%23/%88%97:;%23?)
* [*[0(nop^)]](%23/0%28nop%5E%29)
* [*[0587]](%23/0587)
* [*[05AB1E]](%23/05AB1E)
* [*[0815]](%23/0815.0815)
* [*[1+]](%23/1%2B)
* [*[1C Enterprise]](%23/1C%20Enterprise)
* [*[256]](%23/256.256)
* [*[2B]](%23/2B.2b)
* [*[420]](%23/420.420)
```

## Installing text editor Visual Studio Code

# Exploring Visual Studio Code



Commit changes in the cloned repository

→ git branch new-feature

→ git branch

\* master  
new-feature  
(END)

```
→ hello-world git:(master) x git checkout new-feature
M d/DOG
Switched to branch 'new-feature'
→ hello-world git:(new-feature) x cat .git/HEAD
ref: refs/heads/new-feature
→ hello-world git:(new-feature) x ls
#               contributing.md g           k           n           r           u           x
a             d           h           l           o           readme.md   update_list.py y
b             e           i           license     p           s           v           z
c             f           j           m           q           t           w
→ hello-world git:(new-feature) x nano readme.md
→ hello-world git:(new-feature) x git status
On branch new-feature
Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:  d/DOG
    modified:  h/HEX.hex
    deleted:   p/Parser.p
    modified:  readme.md

no changes added to commit (use "git add" and/or "git commit -a")
→ hello-world git:(new-feature) x
```

```
h > HEX.hex
1 GBL;
2 Bug("1", "Hello World");
3 Scuttle("1");
4 Write;
5
```

```
1 GBL;
2+ Bug("1", "This is program in HEX language");
3 Scuttle("1");
4 Write;
5
```

```
① readme.md > # Hello World
1 # Hello World
2
3 Hello World in every computer language.
4
5 As I watch the collection expand, this project
has blown up more than I ever thought possible.
6 Thanks to everyone who continues to contribute;
new languages are created every day!
7
8 Make sure to see [contributing.md](/contributing-
-md) for instructions on contributing to the
-project!
```

```
1 # Hello World
2
3 Hello World in all programming languages.
4
5 As I watch the collection expand, this project
has blown up more than I ever thought possible.
6 Thanks to everyone who continues to contribute;
new languages are created every day!
7
8+ Make sure to see [contributing.md](/contributing-
+md) file for instructions on contributing to the
+project!
```

```
[3] + 3330 suspended git branch
→ hello-world git:(master) ✘ clear
→ hello-world git:(master) ✘ git checkout new-feature
M      d/DOG
Switched to branch 'new-feature'
→ hello-world git:(new-feature) ✘ cat .git/HEAD
ref: refs/heads/new-feature
→ hello-world git:(new-feature) ✘ ls
#          contributing.md  g          k          n          r          u          x
a          d          h          l          o          readme.md  update_list.py  y
b          e          i          license    p          s          v          z
c          f          j          m          q          t          w
→ hello-world git:(new-feature) ✘ nano readme.md
→ hello-world git:(new-feature) ✘ git status
On branch new-feature
Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:  d/DOG
    modified:  h/HEX.hex
    deleted:   p/Parser.p
    modified:  readme.md

no changes added to commit (use "git add" and/or "git commit -a")
→ hello-world git:(new-feature) ✘ git add .
→ hello-world git:(new-feature) ✘ git ls-files -s
100644 9f846d7a32bf4e6ed219f0218b16146e7c99915e 0      "#/!@#$%^&\342\210\227()_+"
100644 bcef582b1db00e7bba6b300fdb053e0598b47e0 0      "#();#?!"
100644 f4de0575014300b4a6da4fab0eb4b123d674ebfc 0      "#/(nop^)"
100644 2ce4c8e2298caecd36021c15f8236f8b11ad7eb 0      "#/0587
100644 55e3a44121ab7323f2e6a1fe19105b1c96647507 0      "#/05AB1E
100644 7414c30d49cf3c1b6040274c13e1803a7437ad86 0      "#/0815.0815
100644 483aa284f2fc71934f0170ace595c585efb5b16d 0      "#/1+
```

```
100644 de5f72d474cf2d855a873ef8eda4f6019d64ae93 0      z/Zetaplex.zeta
100644 20edd6d0a636ad32c9830a34ffd3fbcd22da5df 0      z/Zig.zig
100644 74cba2198699ff67dd2851ba019ca22d77db765 0      z/Zim.zim
100644 9b9d6be1fee77be758154204892e889176bf357 0      z/Zimbu.zu
100644 a5bba16f2af7bd098981dc119571b6573b6605c2 0      z/Zinc.zn
100644 b32c88ea239e4d335c1f6248795b8a45f504c78d 0      z/Zombie.zombie
100644 65feddb0265510578c0276d2221531ad705f8a77 0      z/Zonnon.znn
100644 6e55c61cdb96ce12192197bff3566754c5407779 0      z/zx.js
→ hello-world git:(new-feature) ✘ git status
On branch new-feature
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:  h/HEX.hex
    deleted:   p/Parser.p
    modified:  readme.md

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:  d/DOG

→ hello-world git:(new-feature) ✘ git commit -m "Modified Readme, adjusted HEX file and removed parser.p file"
[new-feature 2fde3bd] Modified Readme, adjusted HEX file and removed parser.p file
 3 files changed, 3 insertions(+), 7 deletions(-)
 delete mode 100644 p/Parser.p
→ hello-world git:(new-feature) ✘ git status
On branch new-feature
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:  d/DOG

no changes added to commit (use "git add" and/or "git commit -a")
→ hello-world git:(new-feature) ✘ █
```

→ github desktop

The screenshot shows the GitHub Desktop application interface. At the top, it displays the current repository as 'Current Repository: hello-world' and the current branch as 'Current Branch: new-feature'. There is also a button to 'Publish branch'.

The main area shows a diff view for a file named 'readme.md'. The changes are as follows:

```
@@ -1,11 +1,11 @@
1   1
2   2
3   3
4   4
5   5
6   6
7   7
8   8
9   9
10 10
11 11
```

The changes are:

- Hello World in **every computer language.**
- +Hello World in **all programming languages.**
- As I watch the collection expand, this project has blown up more than I ever thought possible.  
Thanks to everyone who continues to contribute; new languages are created every day!
- Make sure to see [\[contributing.md\]\(/contributing.md\)](#) for instructions on contributing to the project!
- +Make sure to see [\[contributing.md\]\(/contributing.md\)](#) file for instructions on contributing to the project!
- <!--Languages start-->
- ## Languages (816 total)

Below the diff, there is a summary section for the pull request:

Summary (required)  
Description  
R+  
⚠ You don't have write access to **hello-world**. Want to [create a fork?](#)

Commit to **new-feature**

→ git log

```
commit 2fde3bd52a32e658bfe34f1d6b68750c02d8040c (HEAD -> new-feature)
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>
Date: Tue Aug 17 19:09:51 2021 +0530

    Modified Readme, adjusted HEX file and removed parser.p file

commit 2a885c1cfe23f95d6155c8da8a62eda5221ed1d9 (origin/master, origin/HEAD, master)
Author: MrBrain295 <66077254+MrBrain295@users.noreply.github.com>
Date: Sun Aug 15 18:25:46 2021 -0500

    Add Mostawesomeprogramminglanguage (#1126)

    Co-authored-by: Richie Bendall <richiebendall@gmail.com>

commit 4cbf9ed3ddfa76ecff0246bc1c9368e797f3f3af
Author: Peter Harsfalvi <peter.harsfalvi@gmail.com>
Date: Mon Aug 16 01:22:51 2021 +0200

    Add Hungarian (#1127)

    Co-authored-by: Peter Harsfalvi <peter@qtc.hu>

commit 0c02a37ddd19c3e548e8d5a59f96e24be5127d59
Author: Henistein <57049813+Henistein@users.noreply.github.com>
Date: Sun Aug 1 04:31:40 2021 +0100

    Add Assembly ARM aarch64 (#1123)

    Co-authored-by: Richie Bendall <richiebendall@gmail.com>

commit bded46cd21decd0954f035f78a3d0c3f2dcddf08
Author: Austin Adams <me@austinjadams.com>
Date: Sat Jul 31 23:29:47 2021 -0400
:|
```

## Git diff command

- The git diff command displays the differences between files in two commits or between a commit and your current repository.
- You can see what text has been added to, removed from, and changed in a file. By default, the git diff command displays any uncommitted changes to your repository.

```
→ hello-world git:(master) ✘ git diff  
  
[5] + 4355 suspended git diff  
→ hello-world git:(master) ✘ git commit -m "changed update_list.py" -a  
[master 404da1c] changed update_list.py  
 1 file changed, 2 insertions(+), 1 deletion(-)  
→ hello-world git:(master) ✘ █
```

```
diff --git a/update_list.py b/update_list.py  
index 25c02c8..9fafffc1 100644  
--- a/update_list.py  
+++ b/update_list.py  
@@ -13,7 +13,7 @@ def regexReplace(string, search, replacement):  
languageCount = 0  
languagesText = ""  
  
-# List the available languages  
+# List the of available languages  
for directory in sorted(os.listdir('.')):  
    if not (directory == '.' or directory == '..' or directory[0] == '.' or os.path.isfile(directory)):  
        for filename in sorted(os.listdir(directory), key=lambda s: s.lower()):  
@@ -41,3 +41,4 @@ readmeContents = open('readme.md', 'r', encoding="utf-8").read()
```

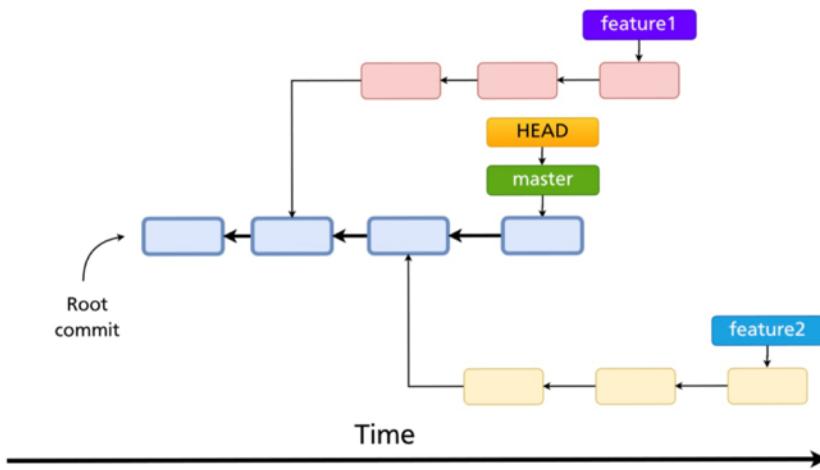
Q How we can "move" commits from the "new feature" feature branch to the "master" branch?

Ans → Use Git branch merge

## Section 8 : Merging branches

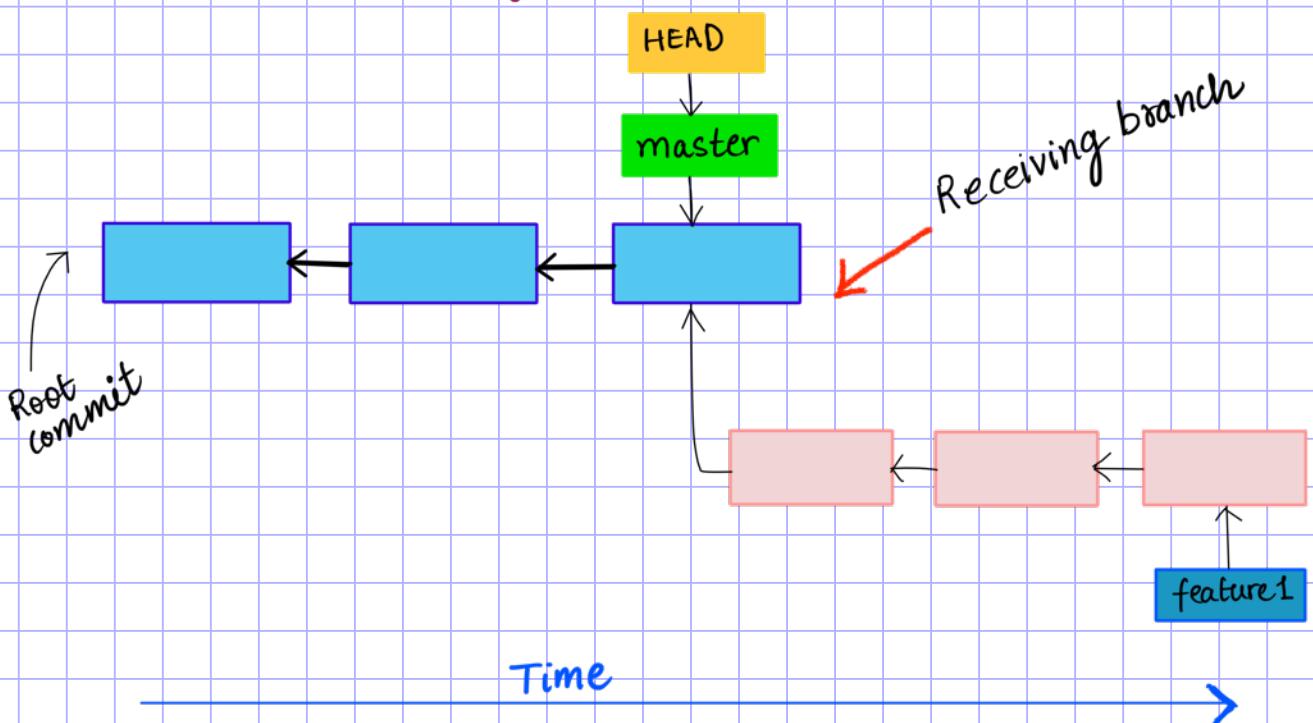
## Why branches merging is needed

## Merging branches



- new branches are created when you want to work on specific feature and you don't want to do that on main branch / master branch / release branch.
  - Merging takes your branch changes and implements them into the main branch.
  - Merging is a common practice for developers using version control system. Whether branches are created for testing, bug fixes or other reasons, merging commits changes to another location. To be more specific, merging takes the contents of a source branch and integrates them with a target branch. In this process only the target branch is changed. The source branch history remains the same.
  - There are two ways/ approaches used for merging branches
    - ① Fast-forward merge
    - ② 3-way merge

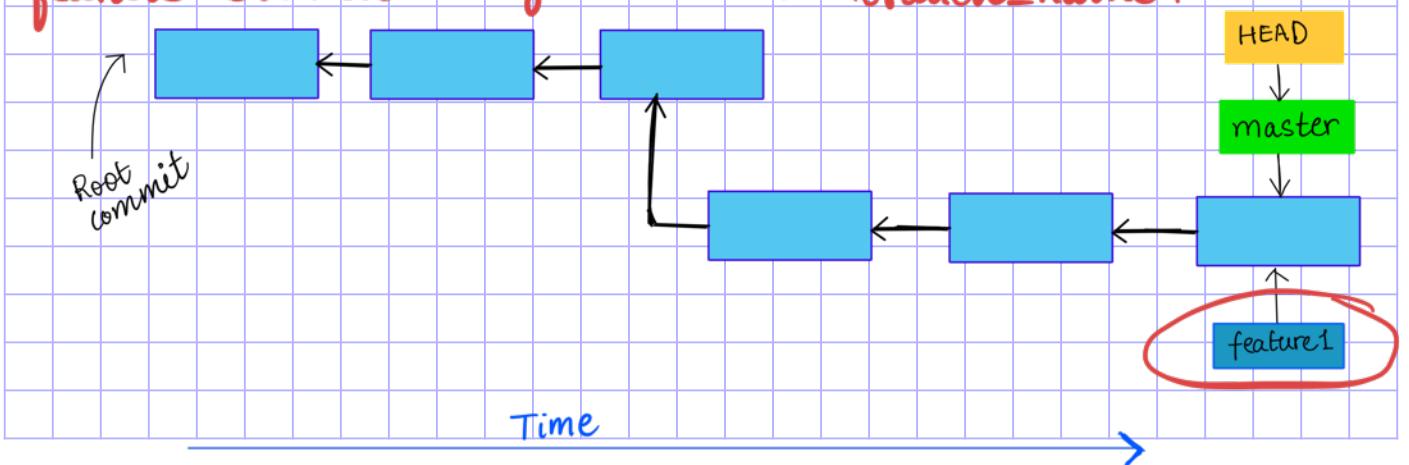
## Fast forward merge

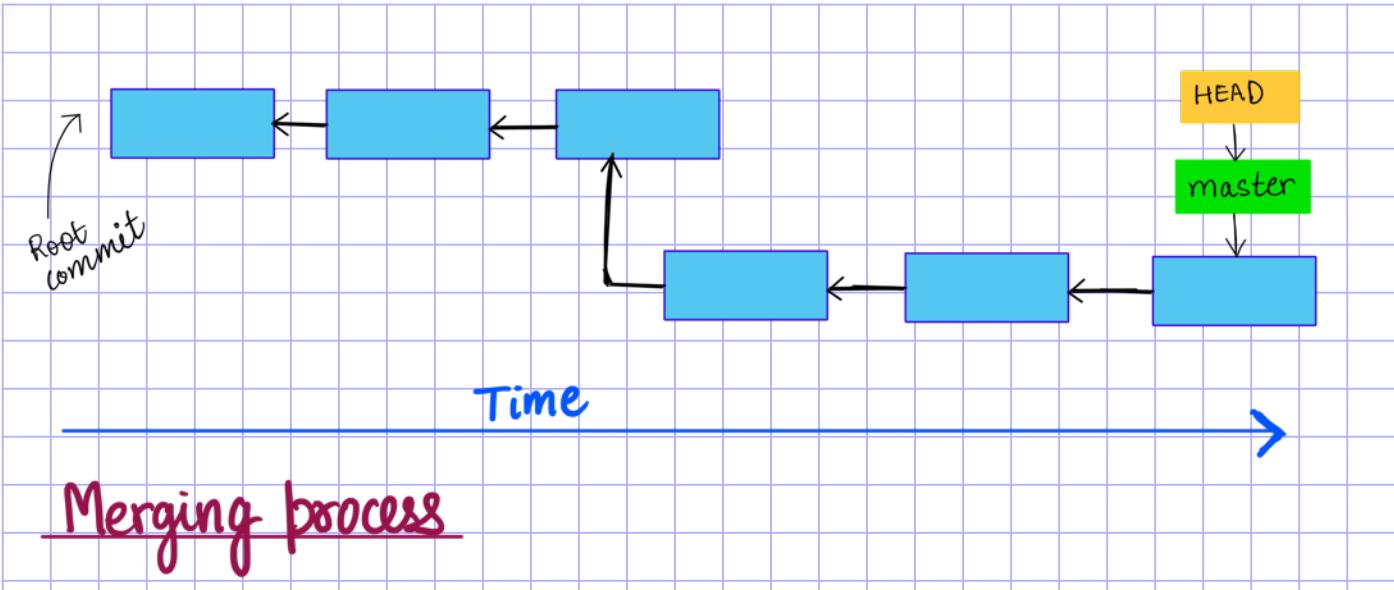


→ Suppose you want to merge changes made in feature-1 branch into master branch, in this case master branch is called receiving branch as it receives changes from other branch.

→ Fast forward merge is possible when there are no further commits in the receiving branch after the commit where feature branch was created.

→ When you merge feature-1 branch into master branch, it simply takes master branch pointer and moves it to last commit in feature-1 branch. You can safely remove feature-1 branch if you are not going to make any other changes in it. After merging you can delete feature branch → `git branch -d <branch_name>`.





## Merging process



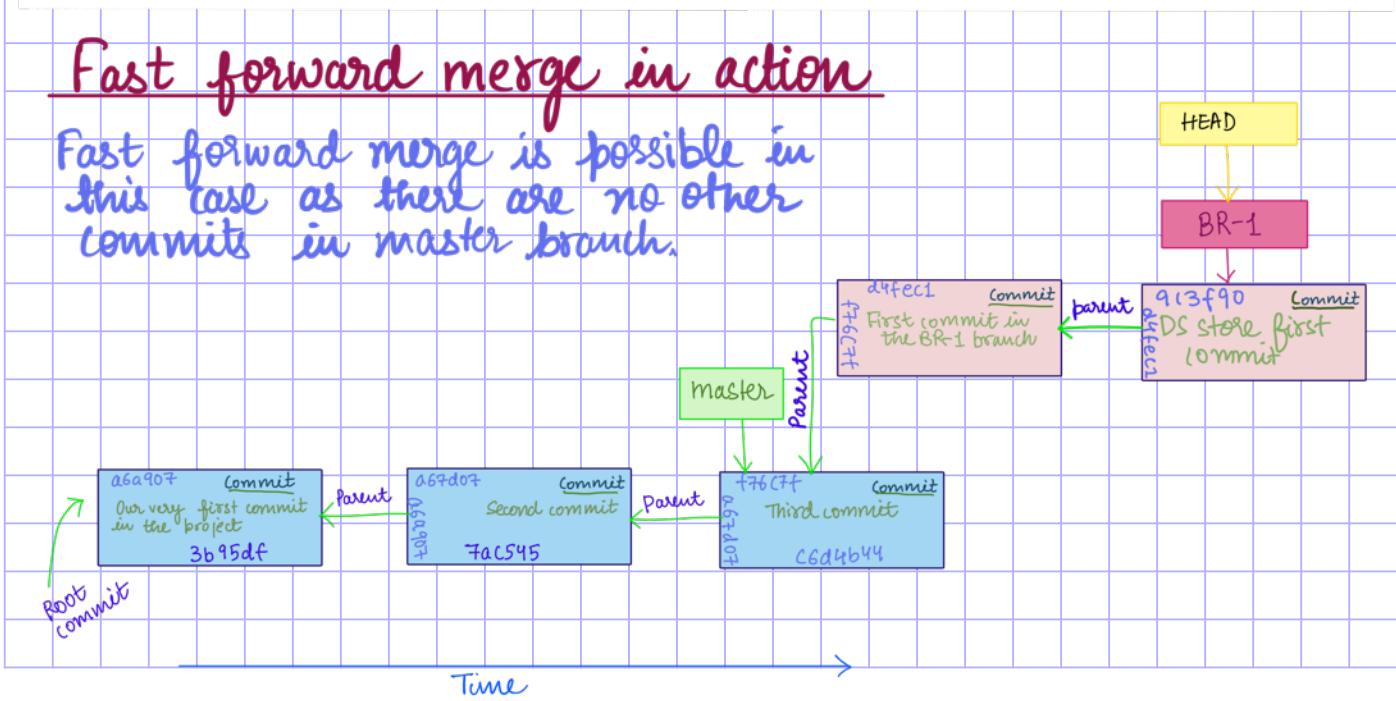
## Merging process

1. Create new feature branch from the main branch
2. Make changes in the new branch and commit them
3. Checkout main branch (receiving branch)
4. Merge feature branch to the current receiving branch

`git merge <feature-branch>`

## Fast forward merge in action

Fast forward merge is possible in this case as there are no other commits in master branch.



BR-1  
\* master  
(END)

```
→ ~ cd Desktop
→ Desktop cd first-project
→ first-project git:(master) ls -l
→ first-project git:(master) git branch

[1] + 1852 suspended git branch
→ first-project git:(master) git checkout BR-1
Switched to branch 'BR-1'
→ first-project git:(BR-1) ls -l
total 8
-rw-r--r-- 1 hemakshipandey staff 11 Aug 18 16:32 file4.txt
→ first-project git:(BR-1) git log

[2] + 1904 suspended git log
```

```
commit 913f90d254bc6a40fef1694bf228fd6d3a368f67 (HEAD -> BR-1)
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>
Date: Mon Aug 16 20:23:23 2021 +0530
```

ds store first commit

```
commit d4fec14d581e6bf5a6a7dd794e465ea7541f2cd2
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>
Date: Mon Aug 16 17:00:51 2021 +0530
```

First commit in the BR-1 branch

```
commit f76c7fcf9fdad7a694bb777abafe06980ce4c200 (master)
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>
Date: Sun Aug 15 13:43:30 2021 +0530
```

Third commit

```
commit a67d07dc0ea4af75344d161b22c4c1455093a8f1
Author: hemakshi1234 <mahe173fas@gmail.com>
Date: Fri Aug 13 14:44:21 2021 +0530
```

Second commit

```
→ first-project git:(BR-1) git checkout master
Switched to branch 'master'
→ first-project git:(master) git log

[3] + 1938 suspended git log
```

```
commit f76c7fcf9fd9aa7a694bb777abafe06980ce4c200 (HEAD -> master)
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>
Date: Sun Aug 15 13:43:30 2021 +0530

    Third commit

commit a67d07dc0ea4af75344d161b22c4c1455093a8f1
Author: hemakshi1234 <mahe173fas@gmail.com>
Date: Fri Aug 13 14:44:21 2021 +0530

    Second commit

commit d0fa6cf6f9bdef2c2ac54224f9eb5c09215814f
Author: hemakshi1234 <mahe173fas@gmail.com>
Date: Thu Aug 12 20:31:03 2021 +0530

    first commit

commit a6a90726f0c078e392724f40e1f295539c9c16b3
Author: hemakshi1234 <mahe173fas@gmail.com>
Date: Thu Aug 12 18:32:44 2021 +0530

    Our very first commit in the project
(CEND)
```

```
→ first-project git:(master) git merge BR-1
Updating f76c7fc..913f90d
Fast-forward
  .DS_Store | Bin 6148 -> 6148 bytes
  file4.txt | 1 +
  2 files changed, 1 insertion(+)
  create mode 100644 file4.txt
→ first-project git:(master) ls -l
total 8
-rw-r--r-- 1 hemakshipandey staff 11 Aug 18 16:38 file4.txt
```

What those  
two SHA1  
hashes  
mean?

before merging  
master branch was  
point to this commit  
After merging master  
branch is now pointing  
to this commit

```
→ first-project git:(master) git log
[4] + 1987 suspended git log
→ first-project git:(master) cat git/refs/heads/master
cat: git/refs/heads/master: No such file or directory
→ first-project git:(master) cat .git/refs/heads/master
913f90d254bc6a40fef1694bf228fd6d3a368f67
→ first-project git:(master)
```

```
commit 913f90d254bc6a40fef1694bf228fd6d3a368f67 (HEAD -> master, BR-1)
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>
Date: Mon Aug 16 20:23:23 2021 +0530
```

```
ds store first commit

commit d4fec14d581e6bf5a6a7dd794e465ea7541f2cd2
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>
Date: Mon Aug 16 17:00:51 2021 +0530

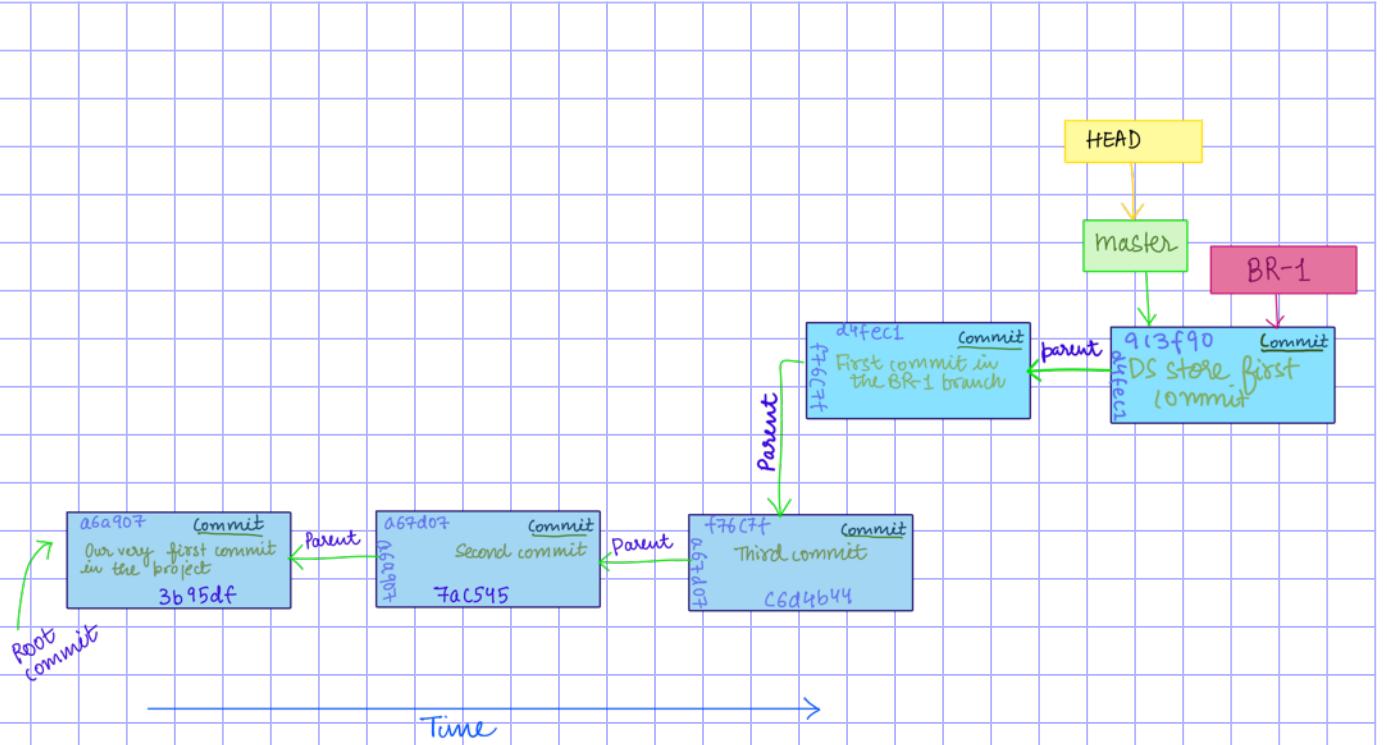
    First commit in the BR-1 branch

commit f76c7fcf9fd9aa7a694bb777abafe06980ce4c200
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>
Date: Sun Aug 15 13:43:30 2021 +0530

    Third commit

commit a67d07dc0ea4af75344d161b22c4c1455093a8f1
Author: hemakshi1234 <mahe173fas@gmail.com>
Date: Fri Aug 13 14:44:21 2021 +0530

    Second commit
```

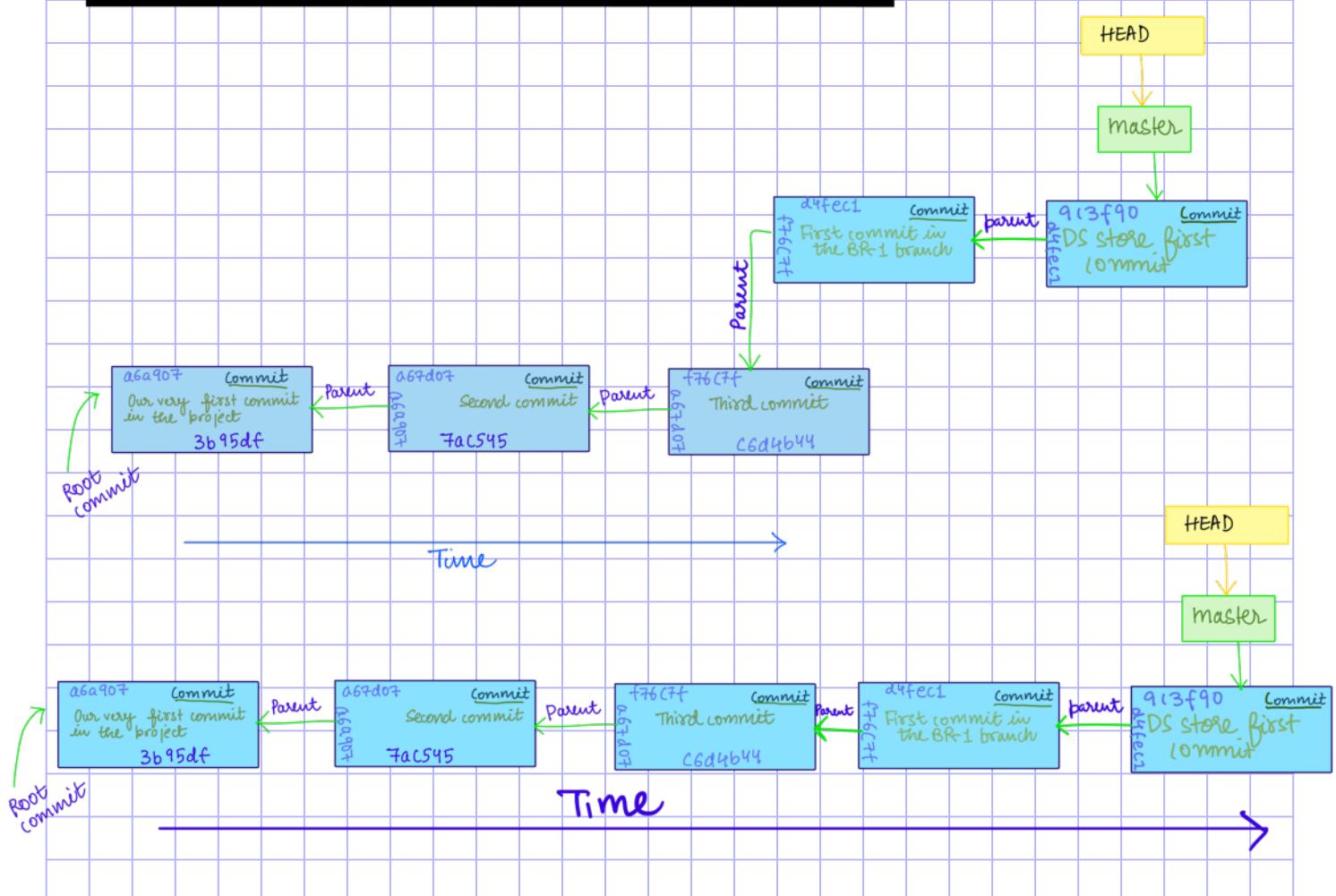


Now BR-1 branch can be safely deleted.

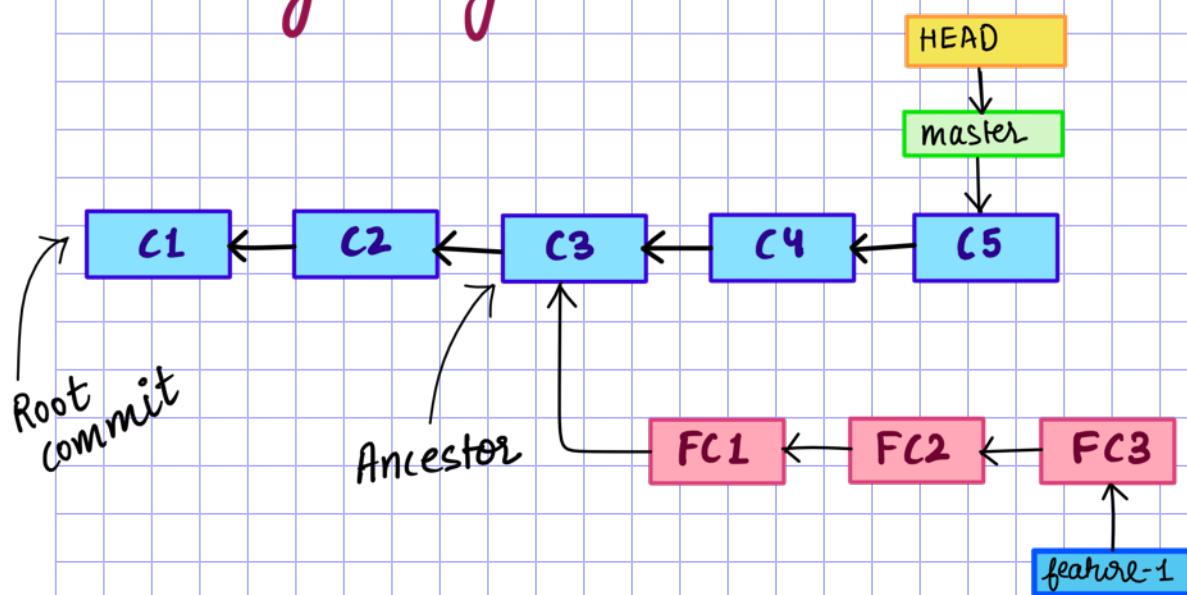
→ `first-project git:(master) git branch -d BR-1`

Deleted branch BR-1 (was 913f90d).

→ `first-project git:(master)`



## 3-Way merge



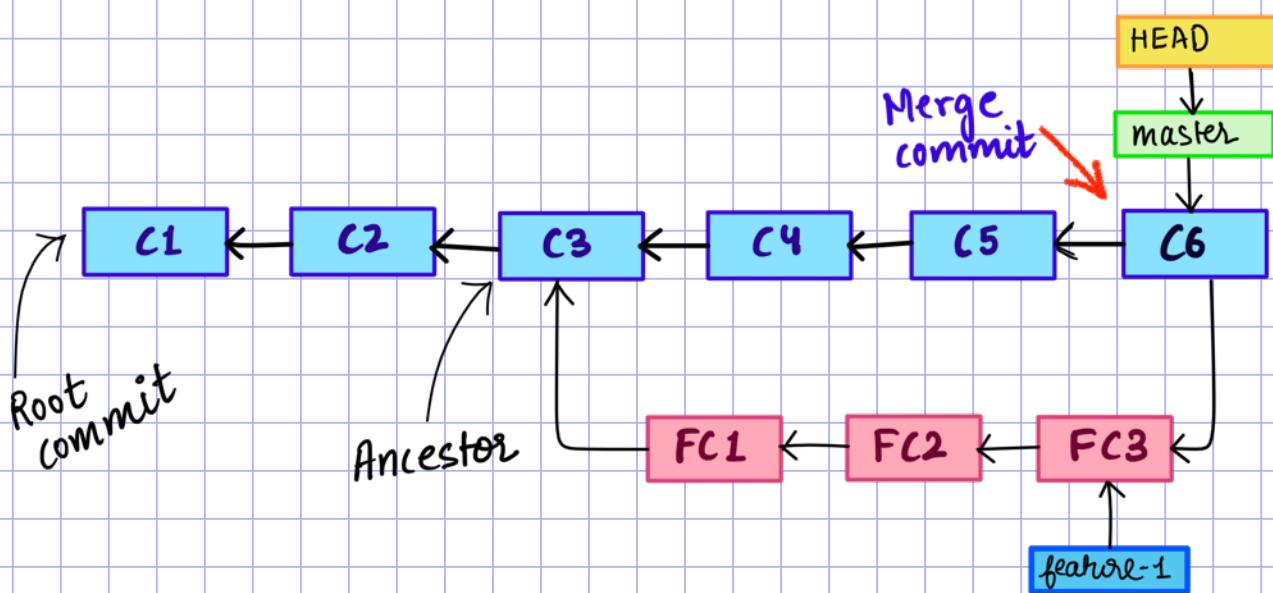
→ let's assume that there are two branches; Master branch and feature-1 branch and this feature-1 branch was created at the third commit of master branch that is, C3.

→ The C3 commit is the parent commit for the first commit → FC1 in feature-1 branch. And now we are not able to simply move master pointer to feature-1 pointer because in such case we will lose C4 and C5 commits; we will lose information.

→ That's why in such case, when you want to merge, feature-1 branch into a master branch, git applies 3-way merge.

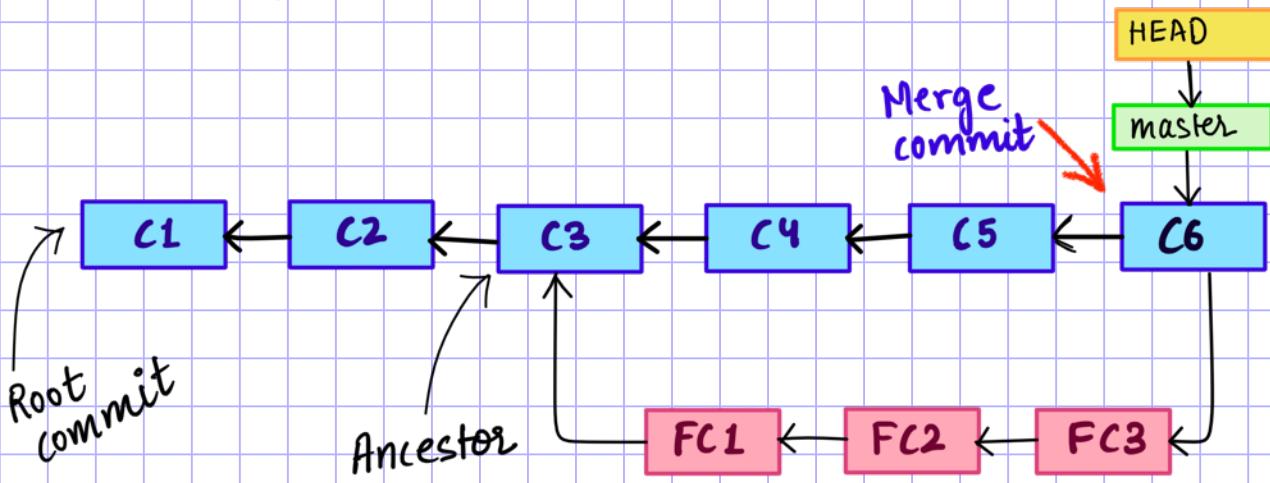
### How it works?

- In such case, it creates a brand new commit and this new commit is based on three other commits: C3, C5 and FC3.
- C3 commit is the first common commit for master and feature-1 branches and C3 is called ancestor (first nearest ancestor)
- git creates brand new merge commit after last commit (C5) of receiving branch.



→ Now C6 is the new merge commit. C6 will have another parent i.e last commit of feature-1 branch → FC3. Hence C6 have two parent commits: C5 and FC3.

→ And with such approach, git has merged two history lines. The merge commit C6 will have all changes that were made in feature-1 branch. All the three commit: FC1, FC2 and FC3 will be incorporated just in single merge commit. But feature-1 branch remain untouched. After creation of merge commit and after merging feature-1 branch into master branch, we can safely delete feature-1 branch.



## Plan :

1. Create new BR-2 branch.
2. Create commits in the BR-2 branch.
3. Switch back to master branch and commit there.
4. Merge BR-2 branch into master branch

## Performing 3-way merge

```
→ ~ cd Desktop/first-project
→ first-project git:(master) git branch
[1] + 2595 suspended git branch
→ first-project git:(master) git checkout -b BR-2
Switched to a new branch 'BR-2'
→ first-project git:(BR-2) ls
file4.txt
→ first-project git:(BR-2) git status
On branch BR-2
nothing to commit, working tree clean
→ first-project git:(BR-2) git log
```



```
commit 8d14fbba602757ee88330cd322816c28f2b8e3b6 (HEAD → BR-2)
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>
Date: Thu Aug 19 13:51:48 2021 +0530
```

Three new files were created in the BR-2 branch

```
commit 913f90d254bc6a40fef1694bf228fd6d3a368f67 (master)
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>
Date: Mon Aug 16 20:23:23 2021 +0530
```

ds store first commit

```
commit d4fec14d581e6bf5a6a7dd794e465ea7541f2cd2
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>
Date: Mon Aug 16 17:00:51 2021 +0530
```

First commit in the BR-1 branch

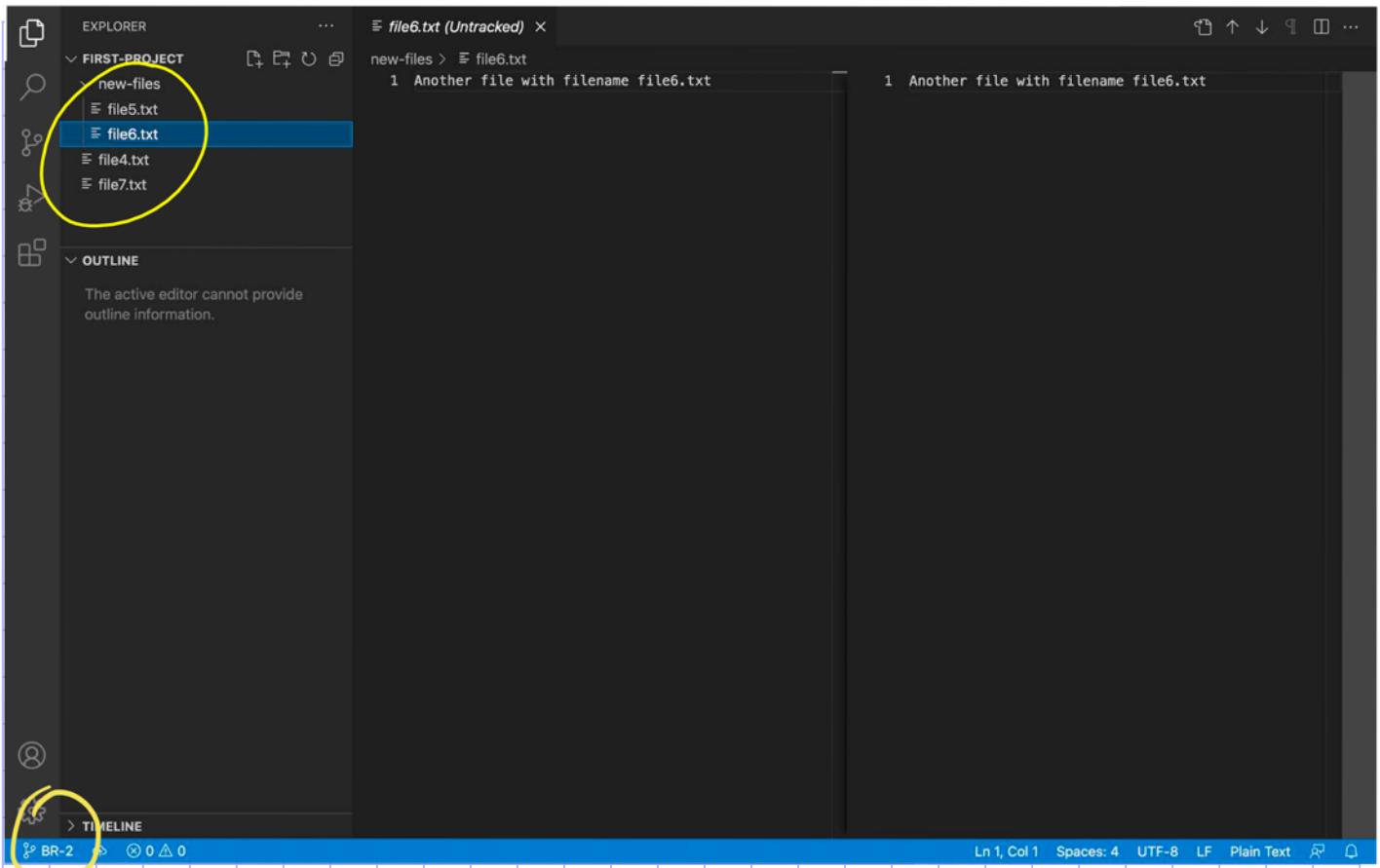
```
commit f76c7fcf9fd8a7a694bb777abafe06980ce4c200
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>
Date: Sun Aug 15 13:43:30 2021 +0530
```

Third commit

```
commit a67d07dc0ea4af75344d161b22c4c1455093a8f1
Author: hemakshi1234 <mahe173fas@gmail.com>
Date: Fri Aug 13 14:44:21 2021 +0530
```

Second commit

```
commit d0fa6fcfc6f9bdef2c2ac54224f9eb5c09215814f
Author: hemakshi1234 <mahe173fas@gmail.com>
Date: Thu Aug 12 20:31:03 2021 +0530
:|
```



```
→ first-project git:(BR-2) ✘ git status
On branch BR-2
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:   new-files/file5.txt

no changes added to commit (use "git add" and/or "git commit -a")
→ first-project git:(BR-2) ✘ git commit -a -m "Changed file5.txt"
[BR-2 351390c] Changed file5.txt
 1 file changed, 4 insertions(+), 1 deletion(-)
```

The screenshot shows a GitHub repository interface. At the top, there are dropdown menus for 'Current Repository' (set to 'first-project') and 'Current Branch' (set to 'BR-2'). There is also a button to 'Publish repository'.

The main area shows a list of commits on the left and a detailed commit view on the right.

**Commits List:**

- Changes
- History (selected)
- Select Branch to Compare...

Recent commits:

- Changed file5.txt (Hemakshi Pandey · 10m)
- Three new files were created in the B... (Hemakshi Pandey · 23m)
- ds store first commit (Hemakshi Pandey · 2d)
- First commit in the BR-1 branch (Hemakshi Pandey · 4d)
- Third commit (Hemakshi Pandey · 5d)
- Second commit (hemakshi1234 · 5d)
- first commit (hemakshi1234 · 6d)
- Our very first commit in the project (hemakshi1234 · 6d)

**Commit Detail View:**

Changed file5.txt

Hemakshi Pandey · 351390c ± 1 changed file [New](#)

new-files/file5.txt	1	@@ -1 +1,4 @@
	1	-It is a new file in the new-files folder
	2	+It is a new file in the new-files folder
	3	+
	4	+File5.txt was just changed

```
→ first-project git:(BR-2) git checkout master
Switched to branch 'master'
→ first-project git:(master) git log
```

```
git (less)          man (less)          2x1          2x2 + 
commit 913f90d254bc6a40fef1694bf228fd6d3a368f67 (HEAD -> master)
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>
Date:  Mon Aug 16 20:23:23 2021 +0530

  ds store first commit

commit d4fec14d581e6bf5a6a7dd794e465ea7541f2cd2
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>
Date:  Mon Aug 16 17:00:51 2021 +0530

  First commit in the BR-1 branch

commit f76c7fcf9fdaa7a694bb777abafe06980ce4c200
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>
Date:  Sun Aug 15 13:43:30 2021 +0530

  Third commit

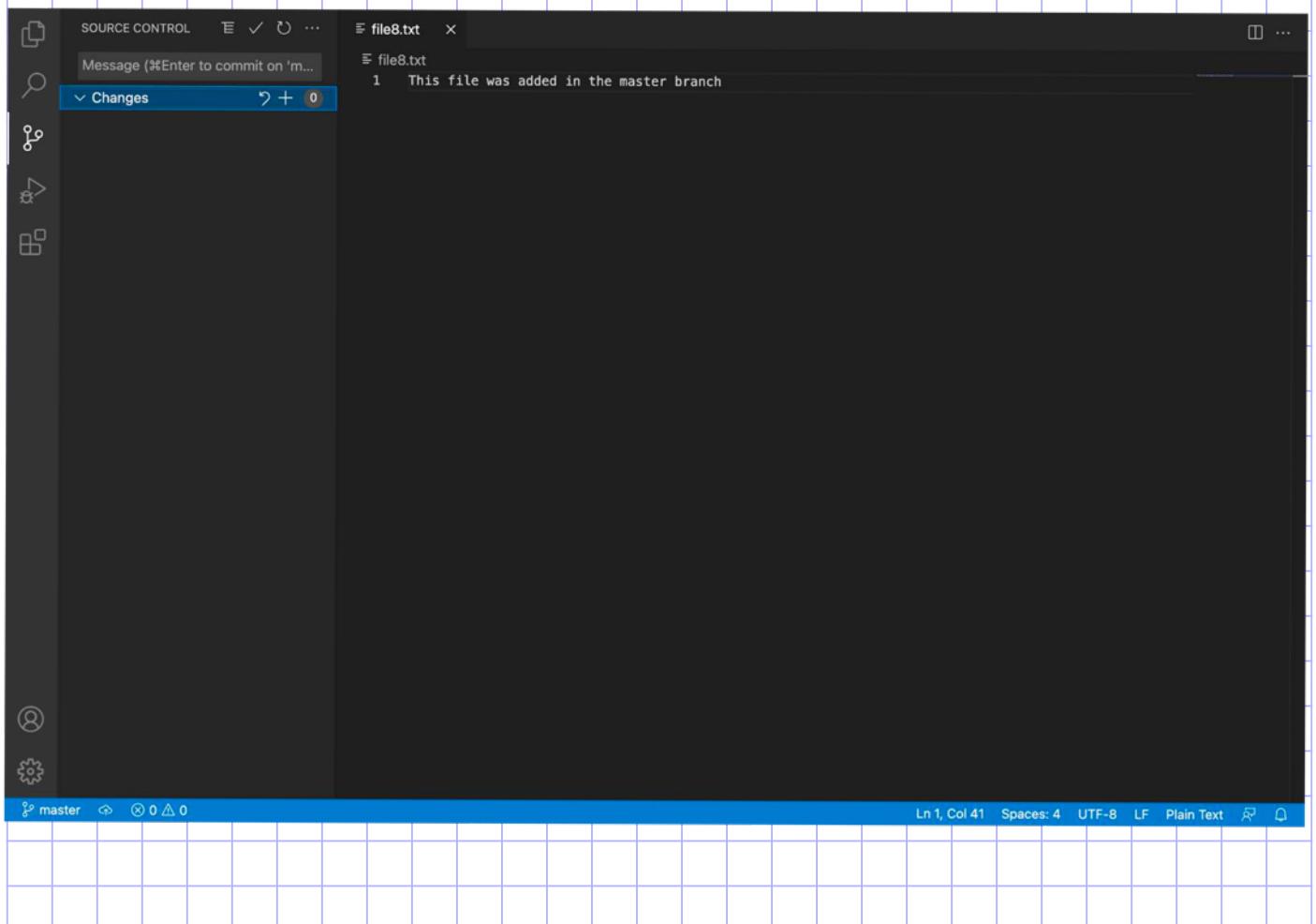
commit a67d07dc0ea4af75344d161b22c4c1455093a8f1
Author: hemakshi1234 <mahe173fas@gmail.com>
Date:  Fri Aug 13 14:44:21 2021 +0530

  Second commit

commit d0fa6cfc6f9bdef2c2ac54224f9eb5c09215814f
Author: hemakshi1234 <mahe173fas@gmail.com>
Date:  Thu Aug 12 20:31:03 2021 +0530

  first commit

commit a6a90726f0c078e392724f40e1f295539c9c16b3
Author: hemakshi1234 <mahe173fas@gmail.com>
:|
```



```
→ first-project git:(master) git merge BR-2
```

Merge made by the 'recursive' strategy.

```
file7.txt          | 1 +
new-files/file5.txt | 4 +++
new-files/file6.txt | 1 +
3 files changed, 6 insertions(+)
create mode 100644 file7.txt
create mode 100644 new-files/file5.txt
create mode 100644 new-files/file6.txt
→ first-project git:(master) █
```

## Observing Git repository after 3-way merge

```
→ first-project git:(master) ls -l → contents of working directory
total 24
-rw-r--r-- 1 hemakshipandey staff 11 Aug 18 16:38 file4.txt
-rw-r--r-- 1 hemakshipandey staff 54 Aug 19 14:26 file7.txt
-rw-r--r-- 1 hemakshipandey staff 40 Aug 19 14:25 file8.txt
drwxr-xr-x@ 4 hemakshipandey staff 128 Aug 19 14:26 new-files
→ first-project git:(master) git log
```

[4] + 3860 suspended git log  
→ first-project git:(master) git cat-file -p c416191 → contents of last merge commit object  
tree a0140744e12a06773b7eee1d76557c195d2562f9  
parent 4dcda286b5b281fc9624be30d9fbef50ee020de36  
parent 351390c64cbe3534b7ac26827ebca71581417a1c  
author Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com> 1629363398 +0530  
committer Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com> 1629363398 +0530

Merge branch 'BR-2' into master branch → commit message  
→ first-project git:(master) █

```
git (less)          201 man (less)      202
commit c416191ffc7e15d2d60b38ae273d75105402f3c5 (HEAD -> master)
Merge: 4dcda286 351390c
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>
Date: Thu Aug 19 14:26:38 2021 +0530
}
} merge commit

Merge branch 'BR-2' into master branch

commit 4dcda286b5b281fc9624be30d9fbef50ee020de36
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>
Date: Thu Aug 19 14:22:40 2021 +0530
}

Create file8.txt
commit 351390c64cbe3534b7ac26827ebca71581417a1c (BR-2)
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>
Date: Thu Aug 19 14:05:08 2021 +0530
}
} Feature branch that was
} merge wasn't changed
Changed file5.txt

commit 8d14fbba602757ee88330cd322816c28f2b8e3b6
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>
Date: Thu Aug 19 13:51:48 2021 +0530
}

Three new files were created in the BR-2 branch

commit 913f90d254bc6a40fef1694bf228fd6d3a368f67
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>
Date: Mon Aug 16 20:23:23 2021 +0530
}

ds store first commit

commit d4fec14d581e6bf5a6a7dd794e465ea7541f2cd2
:█
```

```
→ first-project git:(master) git cat-file -p 4dcd286 → content of parent commit objects
tree 458757f526e75c91f847649ef2df1bbe569549d8
parent 913f90d254bc6a40fef1694bf228fd6d3a368f67
author Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com> 1629363160 +0530
committer Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com> 1629363160 +0530
Create file8.txt → last commit in master branch
→ first-project git:(master) git cat-file -p 351390 → content of parent commit objects
tree 2ff52f7b1c0af0fe2b254a5cff98c3764387410
parent 8d14fbba602757ee88330cd322816c28f2b8e3b6
author Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com> 1629362108 +0530
committer Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com> 1629362108 +0530
Changed file5.txt → last commit in BR-2 branch
→ first-project git:(master)
```

```
→ first-project git:(master) git branch
```

```
[5] + 3980 suspended git branch
```

```
→ first-project git:(master) git branch -d BR-2
```

```
Deleted branch BR-2 (was 351390c). → deleting BR-2 branch
```

```
→ first-project git:(master)
```

BR-2

\* master

(END)

```
→ first-project git:(master) git log
```

```
[6] + 4018 suspended git log
```

```
→ first-project git:(master) git cat-file -p c41619
```

```
tree a0140744e12a06773b7eea1d76557c195d2562f9
```

```
parent 4dcd286b5b281fc9624be30d9fbef50ee020de36
```

```
parent 351390c64cbe3534b7ac26827ebca71581417a1c
```

```
author Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com> 1629363398 +0530
```

```
committer Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com> 1629363398 +0530
```

```
Merge branch 'BR-2' into master branch
```

```
→ first-project git:(master)
```

git (less)

⌘1

man (less)

⌘2

```
commit c416191ffc7e15d2d60b38ae273d75105402f3c5 (HEAD -> master)
Merge: 4dcd286 351390c
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>
Date: Thu Aug 19 14:26:38 2021 +0530
```

```
Merge branch 'BR-2' into master branch
```

```
commit 4dcd286b5b281fc9624be30d9fbef50ee020de36
```

```
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>
```

```
Date: Thu Aug 19 14:22:40 2021 +0530
```

```
Create file8.txt
```

```
commit 351390c64cbe3534b7ac26827ebca71581417a1c
```

```
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>
```

```
Date: Thu Aug 19 14:05:08 2021 +0530
```

```
Changed file5.txt
```

```
commit 8d14fbba602757ee88330cd322816c28f2b8e3b6
```

```
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>
```

```
Date: Thu Aug 19 13:51:48 2021 +0530
```

```
Three new files were created in the BR-2 branch
```

```
commit 913f90d254bc6a40fef1694bf228fd6d3a368f67
```

```
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>
```

```
Date: Mon Aug 16 20:23:23 2021 +0530
```

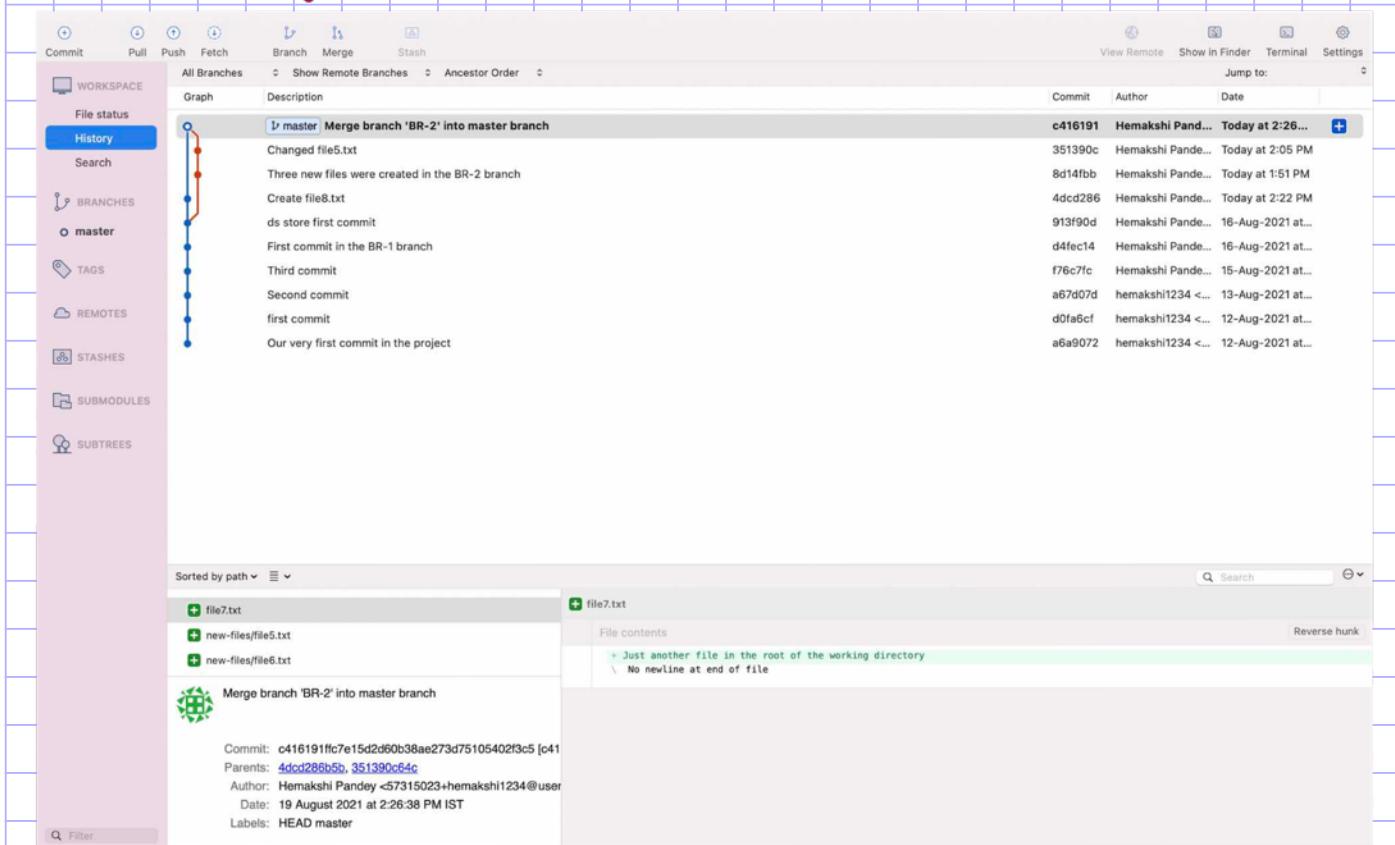
```
ds store first commit
```

```
commit d4fec14d581e6bf5a6a7dd794e465ea7541f2cd2
```

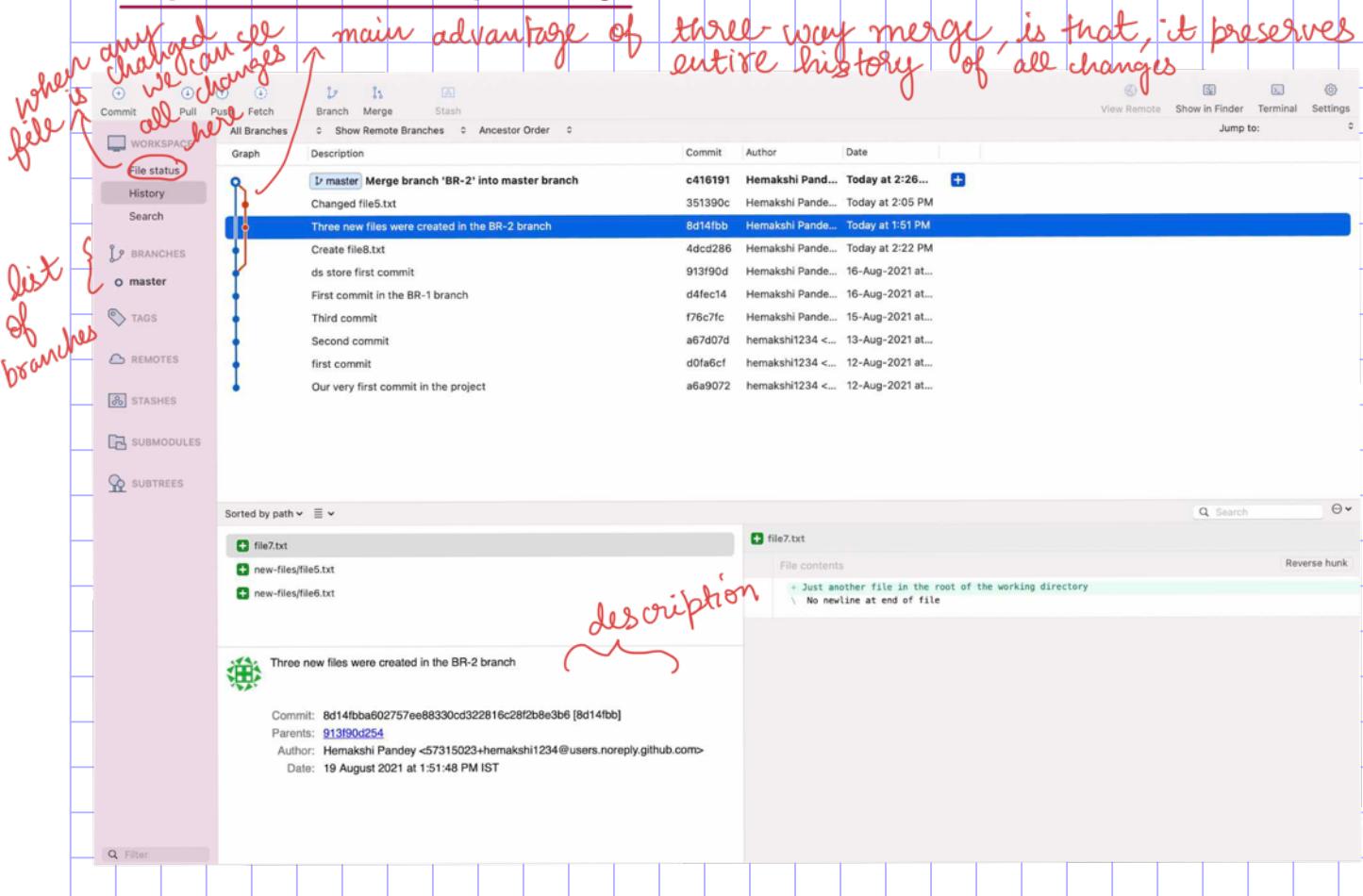
```
:|
```

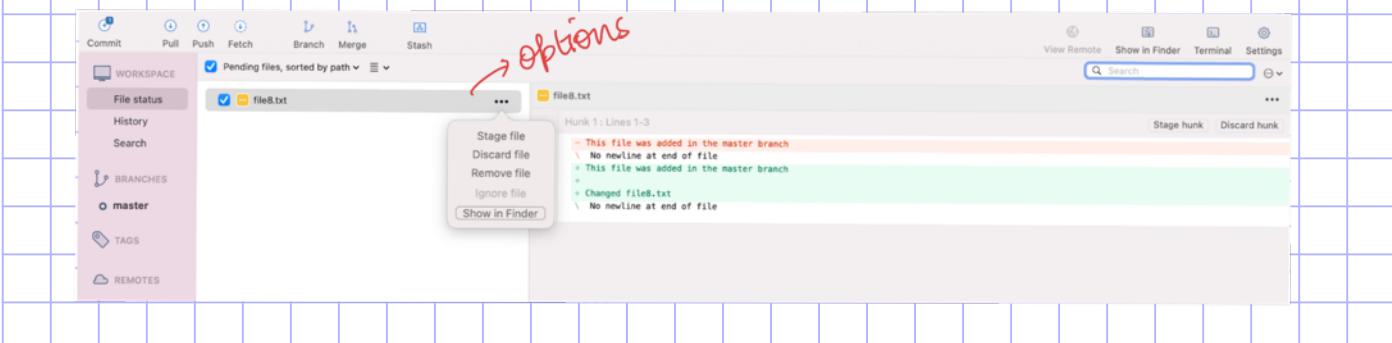
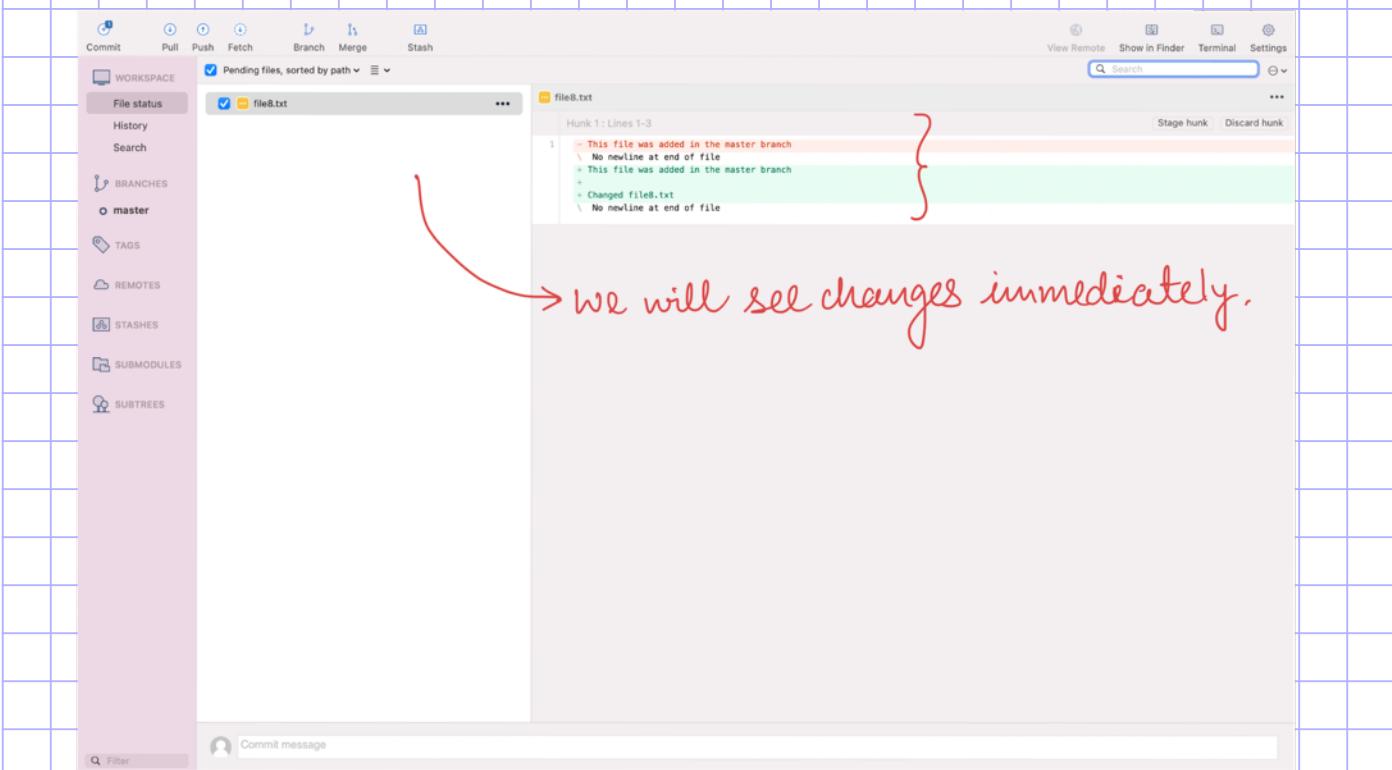
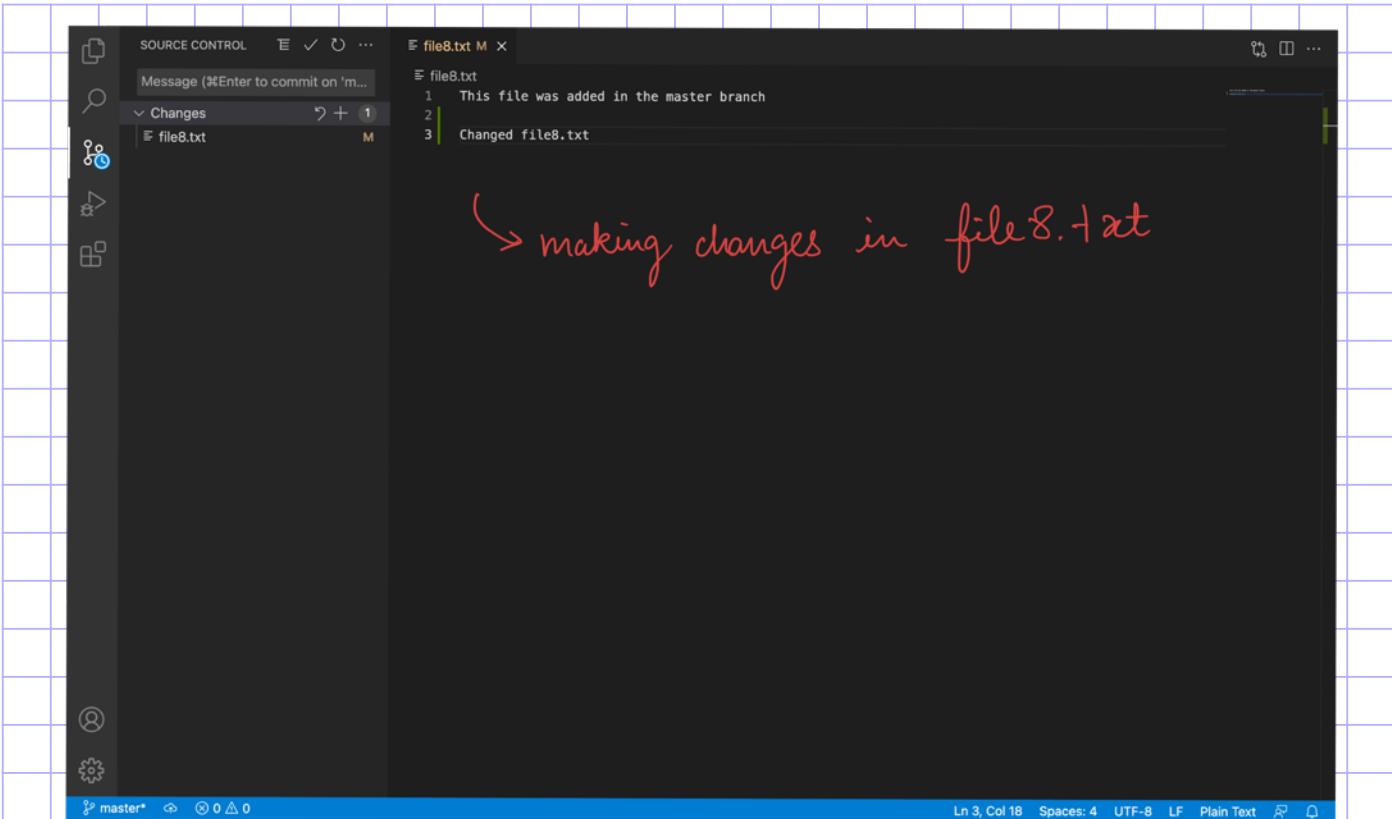
pointer to BR-2  
branch has  
disappeared

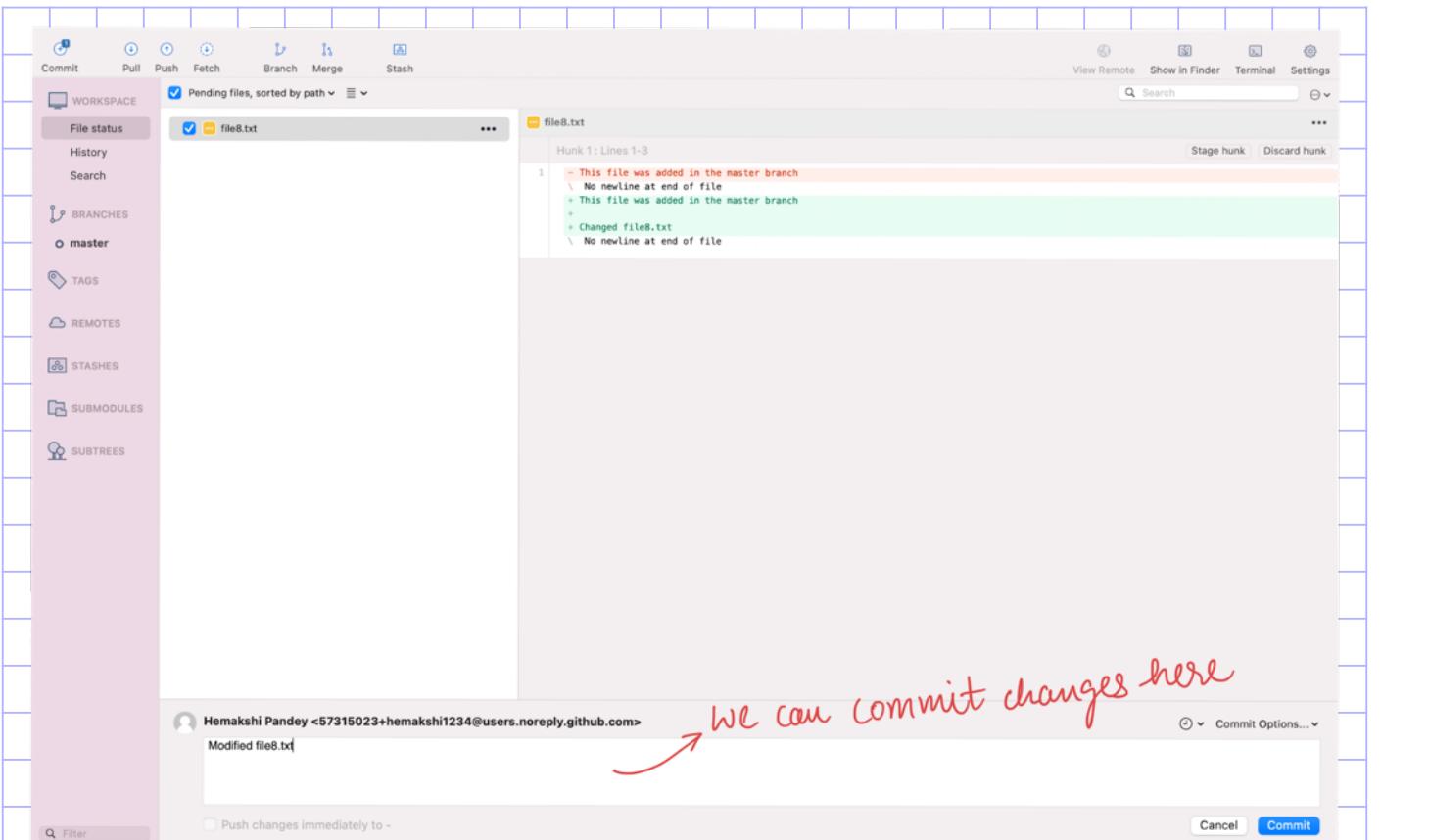
# Installing SourceTree



## SourceTree in action

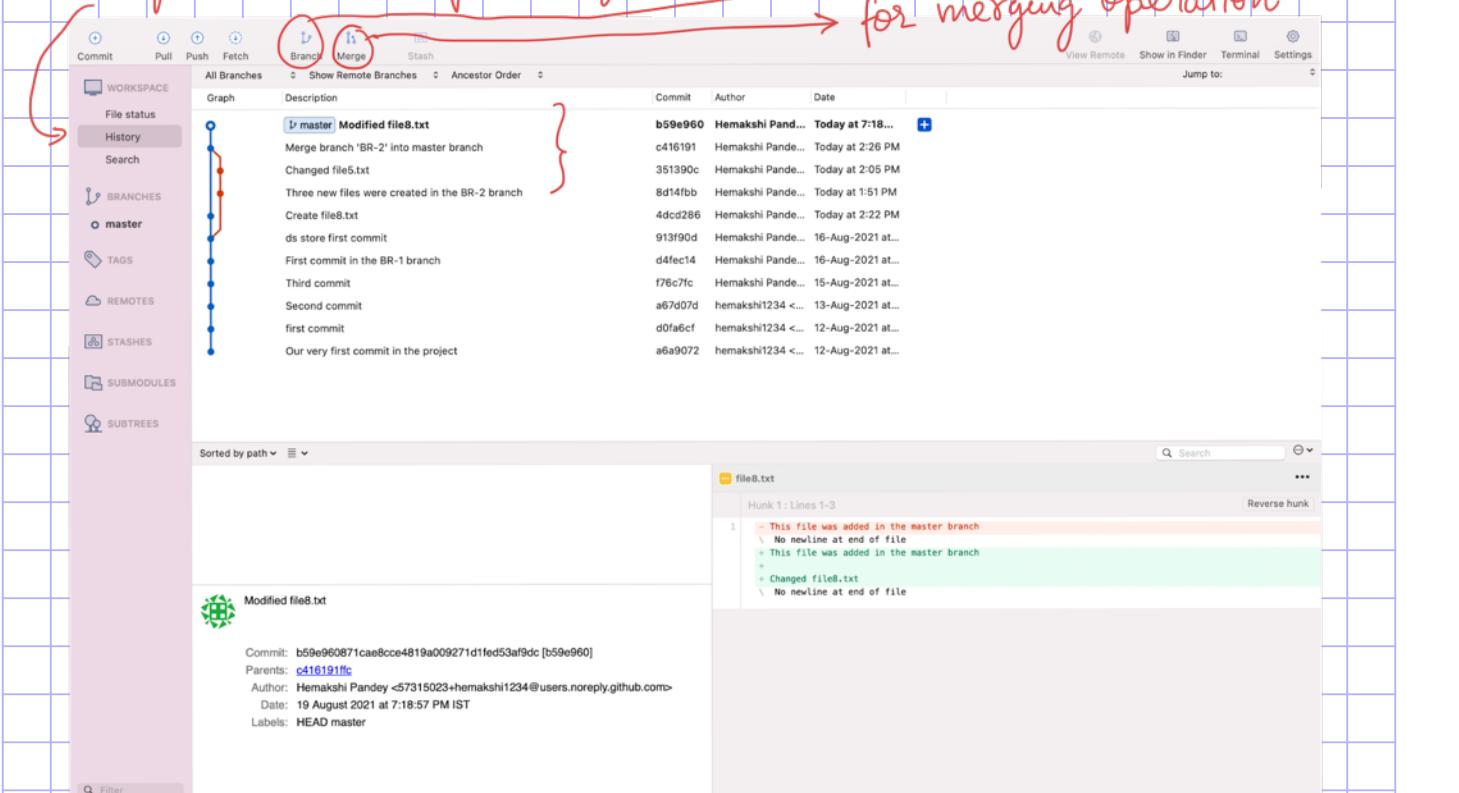






After committing changes

used for creation of new branch  
for merging operation



→ Sourcetree is full featured graphical user interface for git

detached head state

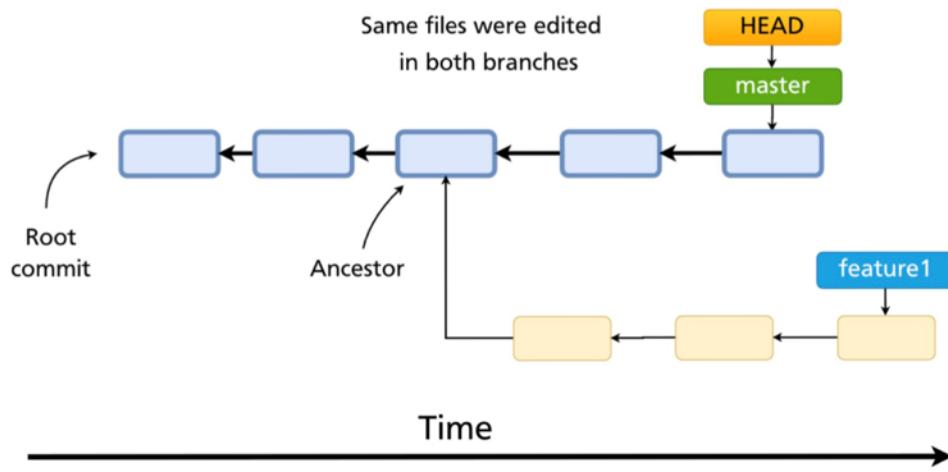
→ By double clicking on any commit , you can change working copy and checkout specific commit.

## what is merge conflict



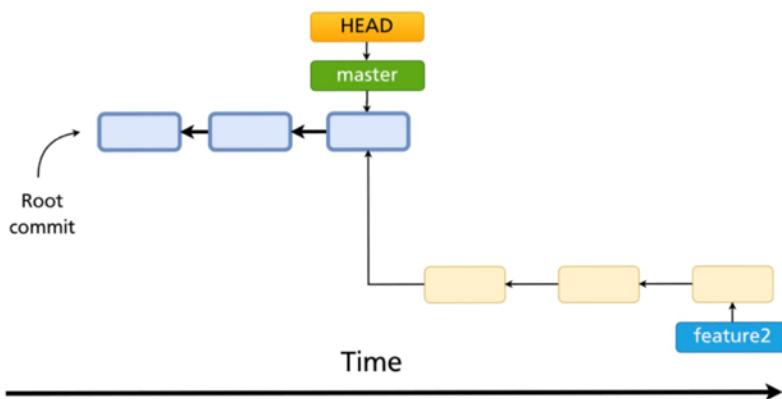
**GIT** COMPLETE GUIDE

## Merge conflicts



- Merge conflicts may appear when you have edited some files in different branches.
- merge conflicts are only possible when there are additional commits made in receiving branch after you have created a feature branch.

Merge conflicts don't appear in fast forward merge



- It is not possible to get merge conflicts in fast forward merge, when there are not other commit after the last commit in master branch. Thus conflicts may appear only during 3-way merge process.
- During the process of merging, if git finds out that there are changes made in the same files in master branch, it will stop merging right before creation of merge commit and it will ask you to resolve conflicts and you need to resolve conflicts manually and you need to decide which changes you want to keep in merge commit; changes from master branch, or changes from feature branch or some other changes. You can edit the files that have conflicts in any way you want. After editing the files that have conflicts you need to stage them and commit and after that git will create merge commit automatically.

Conflicts generally arise when two people have changed the same lines in a file, or if one developer deleted a file while another developer was modifying it. In these cases, Git cannot automatically determine what is correct. Conflicts only affect the developer conducting the merge, the rest of the team is unaware of the conflict. Git will mark the file as being conflicted and halt the merging process. It is then the developers' responsibility to resolve the conflict.

## Creating merge conflict

Plan:

1. Create new BR-3 branch
2. Change some files in the BR-3 branch and commit changes.
3. Return to master branch and change some files in the same places.
4. Try to merge BR-3 branch into master branch.
5. Resolve conflicts manually (edit files in any editor)
6. Stage files with resolved conflicts.
7. Commit changes.
8. Git will create merge commit.

```

→ ~ cd Desktop/first-project
→ first-project git:(master) git branch
[1] + 952 suspended git branch
→ first-project git:(master) git branch BR-3
→ first-project git:(master) git checkout BR-3
Switched to branch 'BR-3'
→ first-project git:(BR-3) ls
file4.txt file7.txt file8.txt new-files
→ first-project git:(BR-3)
→ first-project git:(master) git log

```

\* master  
(END)

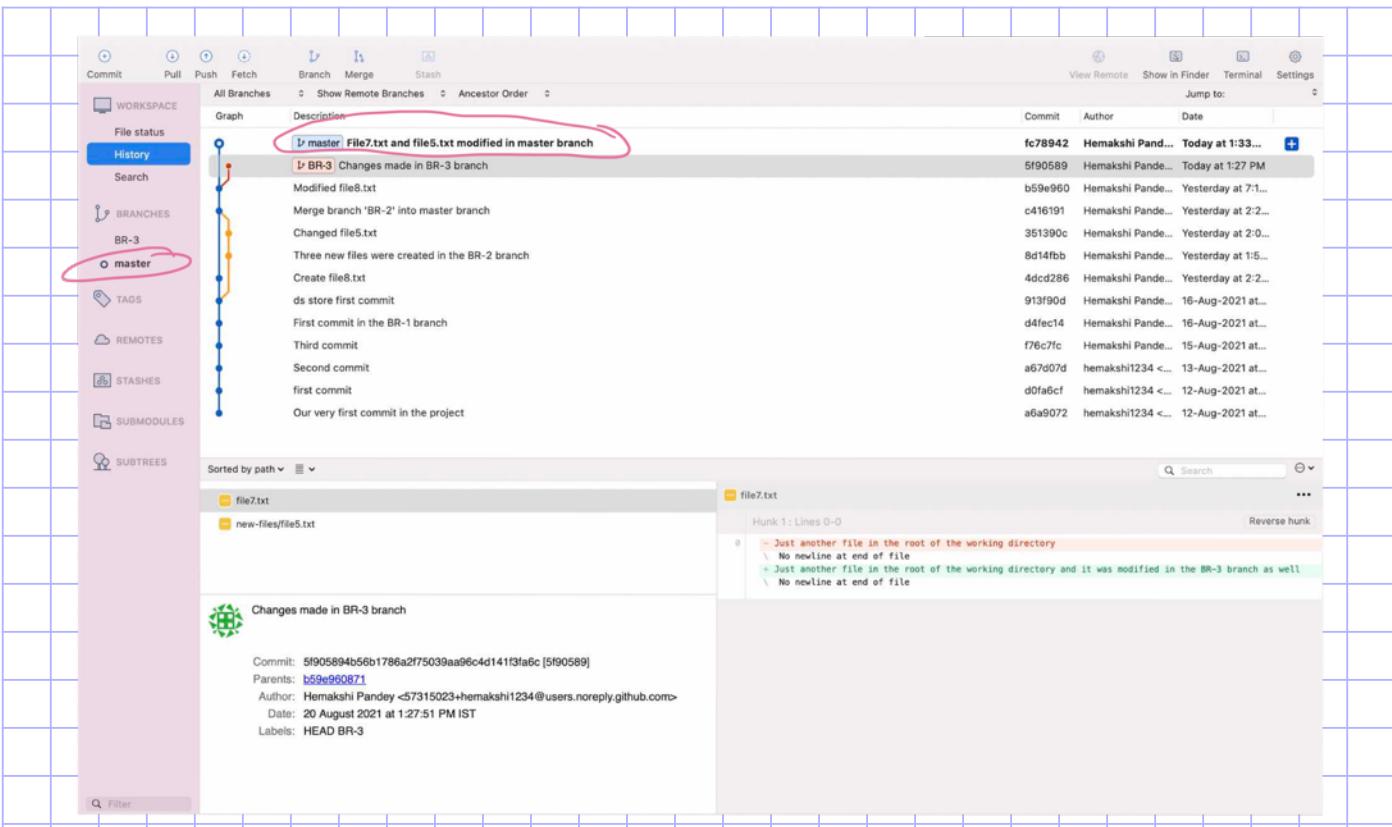
creating new branch  
BR-3

→ Editing file5.txt and file7.txt in Vs code and committing changes in BR-3 branch,

### Source Tree

Commit	Author	Date
b59e960871	Hemakshi Pandey	Today at 1:27 PM
c416191	Hemakshi Pandey	Yesterday at 2:2...
351390c	Hemakshi Pandey	Yesterday at 2:0...
8d14fb1	Hemakshi Pandey	Yesterday at 1:5...
4dcfd286	Hemakshi Pandey	Yesterday at 2:2...
913f90d	Hemakshi Pandey	16-Aug-2021 at...
d4fec14	Hemakshi Pandey	16-Aug-2021 at...
f76c7fc	Hemakshi Pandey	15-Aug-2021 at...
a67d07d	hemakshi1234 <...>	13-Aug-2021 at...
d0fa6cf	hemakshi1234 <...>	12-Aug-2021 at...
a6a9072	hemakshi1234 <...>	12-Aug-2021 at...

Editing file5.txt and file7.txt in Vs code and committing changes in master branch



```

commit fc7894249357720a67776fe6fc7ba66226c2c1f8 (HEAD -> master)
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>
Date:   Fri Aug 20 13:33:12 2021 +0530

    File7.txt and file5.txt modified in master branch

commit b59e960871cae8cce4819a009271d1fed53af9dc
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>
Date:   Thu Aug 19 19:18:57 2021 +0530

    Modified file8.txt

commit c416191ffc7e15d2d60b38ae273d75105402f3c5
Merge: 4dcfd86 351390c
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>
Date:   Thu Aug 19 14:26:38 2021 +0530

    Merge branch 'BR-2' into master branch

commit 4dcfd86b5b281fc9624be30d9fbef50ee020de36
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>
Date:   Thu Aug 19 14:22:40 2021 +0530

    Create file8.txt

commit 351390c64cbe3534b7ac26827ebca71581417a1c
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>
Date:   Thu Aug 19 14:05:08 2021 +0530

    Changed file5.txt

commit 8d14fbba602757ee88330cd322816c28f2b8e3b6
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>
:|
```

```

→ first-project git:(master) git checkout BR-3
Switched to branch 'BR-3'
→ first-project git:(BR-3) git log|
```

```

commit 5f905894b56b1786a2f75039aa96c4d141f3fa6c (HEAD -> BR-3)
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>
Date:   Fri Aug 20 13:27:51 2021 +0530

    Changes made in BR-3 branch

commit b59e960871cae8cce4819a009271d1fed53af9dc
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>
Date:   Thu Aug 19 19:18:57 2021 +0530

    Modified file8.txt

commit c416191ffc7e15d2d60b38ae273d75105402f3c5
Merge: 4dcfd286 351390c
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>
Date:   Thu Aug 19 14:26:38 2021 +0530

    Merge branch 'BR-2' into master branch

commit 4dcfd286b5b281fc9624be30d9fbef50ee020de36
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>
Date:   Thu Aug 19 14:22:40 2021 +0530

    Create file8.txt

commit 351390c64cbe3534b7ac26827ebca71581417a1c
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>
Date:   Thu Aug 19 14:05:08 2021 +0530

    Changed file5.txt

commit 8d14fbba602757ee88330cd322816c28f2b8e3b6
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>
:1

```

```

→ first-project git:(BR-3) git checkout master
Switched to branch 'master'
→ first-project git:(master) git merge BR-3
Auto-merging new-files/file5.txt
CONFLICT (content): Merge conflict in new-files/file5.txt
Auto-merging file7.txt
CONFLICT (content): Merge conflict in file7.txt
Automatic merge failed; fix conflicts and then commit the result.
→ first-project git:(master) ✘

```

*changes in the project*

*merge was unsuccessful*

*are not yet committed*

```

→ first-project git:(master) ✘ git status
On branch master
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:  file7.txt
    both modified:  new-files/file5.txt

```

*we see which files have conflicts*

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

```
→ first-project git:(master) ✘
```

SOURCE CONTROL

Merge branch 'BR-3'

Merge Changes

Changes

file8.txt

file5.txt !

file7.txt !

file7.txt

Accept Current Change | Accept Incoming Change | Accept Both Changes | Compare Changes

<<<< HEAD (Current Change)

Line replaced in master branch

=====

Just another file in the root of the working directory and it was modified in the BR-3 branch as well

>>>> BR-3 (Incoming Change)

Ln 6, Col 1 Spaces: 4 UTF-8 Plain Text

SOURCE CONTROL

Merge branch 'BR-3'

Merge Changes

new-files > file5.txt

Changes

file8.txt

file5.txt !

file7.txt !

file5.txt

It is a new file in the new-files folder

=====

Accept Current Change | Accept Incoming Change | Accept Both Changes | Compare Changes

<<<< HEAD (Current Change)

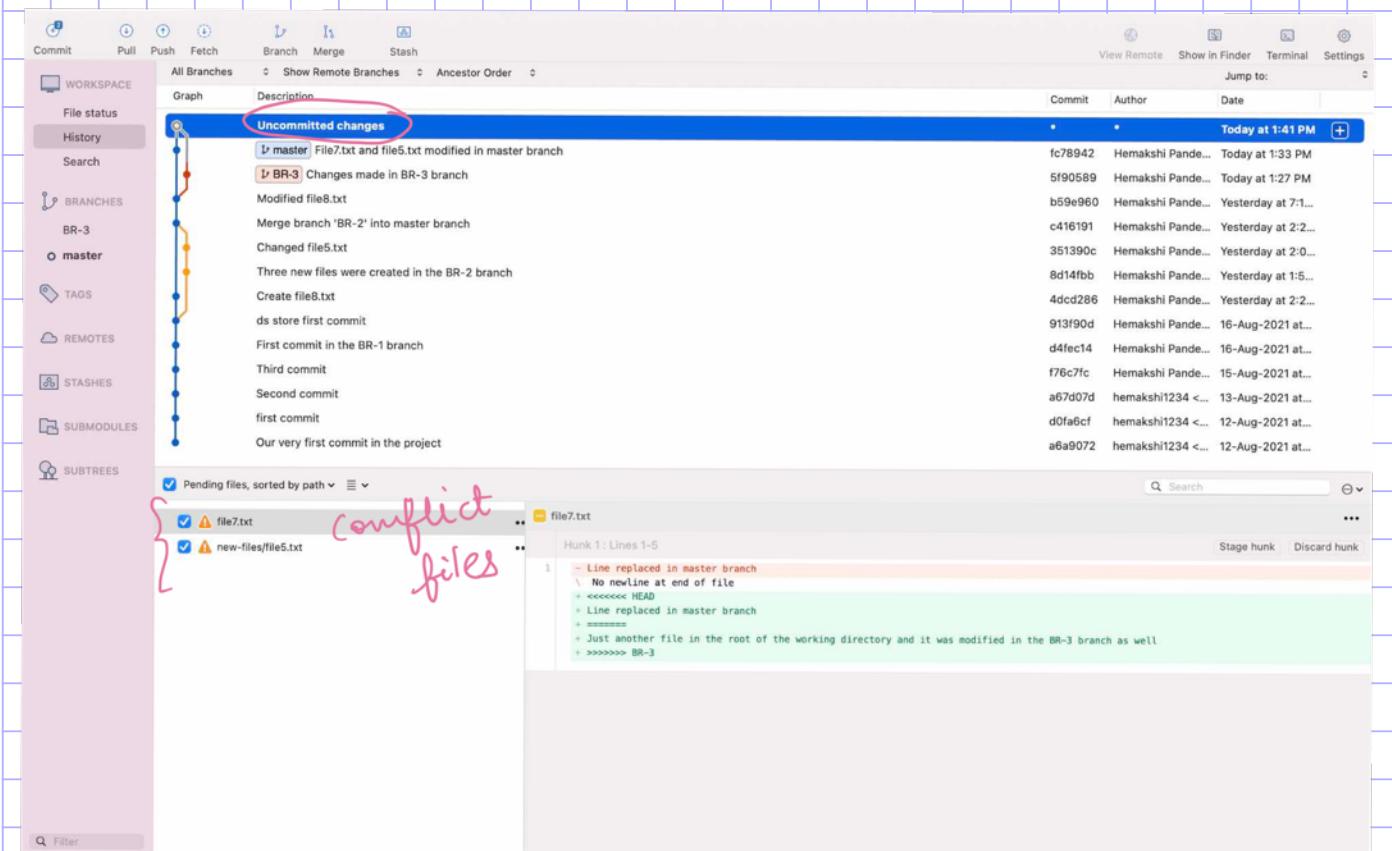
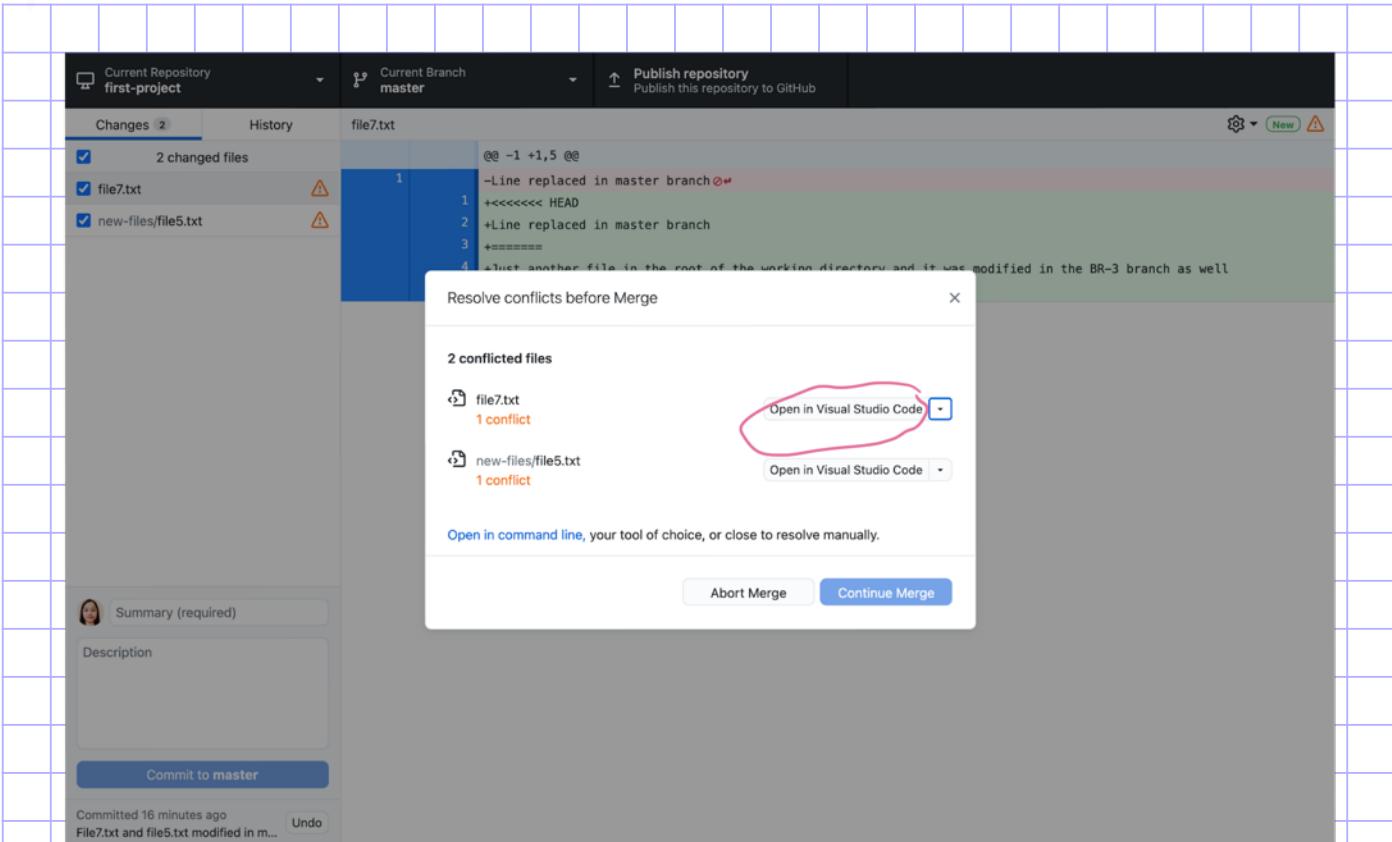
File5.txt was just changed. This line was modified again in master branch

=====

File5.txt was just changed. This line was changed in BR-3 branch

>>>> BR-3 (Incoming Change)

Ln 4, Col 13 Spaces: 4 UTF-8 LF Plain Text



# Observing conflicts in staging area and working directory

listing files in staging area

```
→ first-project git:(master) ✘ git ls-files -s
100644 356f49ac9ff5360d59fd309dec47a501b10f7032 0 .DS_Store
100644 b7aec520dec0a7516c18eb4c68b64ae1eb9b5a5e 0 file4.txt
100644 2107281b79ed03e0147ea53fb62eb5f9a47aba2a 1 file7.txt
100644 5749b350c8358f5e9b384e52b804d883280f4d53 2 file7.txt
100644 8783f8d5f77bc79e3bd2095620b1cf4c1fd316ca 3 file7.txt
100644 a529592a90242ad5002712bd5ba3b7ae97496666 0 file8.txt
100644 6b4407fa2265f9d47da3662dc353d077fec5f260 1 new-files/file5.txt
100644 197cacc5205db2d6d3e099fd23cef7c0a58e1201 2 new-files/file5.txt
100644 46ea30340cc914b727a5eee3633a0fed3c5311ce 3 new-files/file5.txt
100644 cf9ca3f4f20ab449121b916d6387328a789df46f 0 new-files/file6.txt
→ first-project git:(master) ✘
```

There are 3 versions of file7.txt and file5.txt and have different numbers

why we have here 3 different versions of same file

```
(1) → first-project git:(master) ✘ git cat-file -p 210728
Just another file in the root of the working directory
(2) → first-project git:(master) ✘ git cat-file -p 5749b
Line replaced in master branch
(3) → first-project git:(master) ✘ git cat-file -p 8783f8
Just another file in the root of the working directory and it was modified in the BR-3 branch as well
→ first-project git:(master) ✘
```

Initial common version of file7.txt  
file7.txt changed in master branch  
file7.txt changed in BR-3 branch

Based on those three different version of the same file, git shows us merge conflicts and it shows us merge conflicts, actually in the files, in working directory

```
→ first-project git:(master) ✘ cat file7.txt
<<<<< HEAD
Line replaced in master branch
=====
Just another file in the root of the working directory and it was modified in the BR-3 branch as well
>>>>> BR-3
→ first-project git:(master) ✘
```

we see the version of the file where we need to resolve conflicts.

```
→ first-project git:(master) ✘ git cat-file -p 6b4407
It is a new file in the new-files folder
File5.txt was just changed%
→ first-project git:(master) ✘ git cat-file -p 197cac
It is a new file in the new-files folder
version from master branch
File5.txt was just changed. This line was modified again in master branch%
→ first-project git:(master) ✘ git cat-file -p 46ea30
It is a new file in the new-files folder
version from feature branch
File5.txt was just changed. This line was changed in BR-3 branch%
→ first-project git:(master) ✘
```

Three different versions of same file5.txt

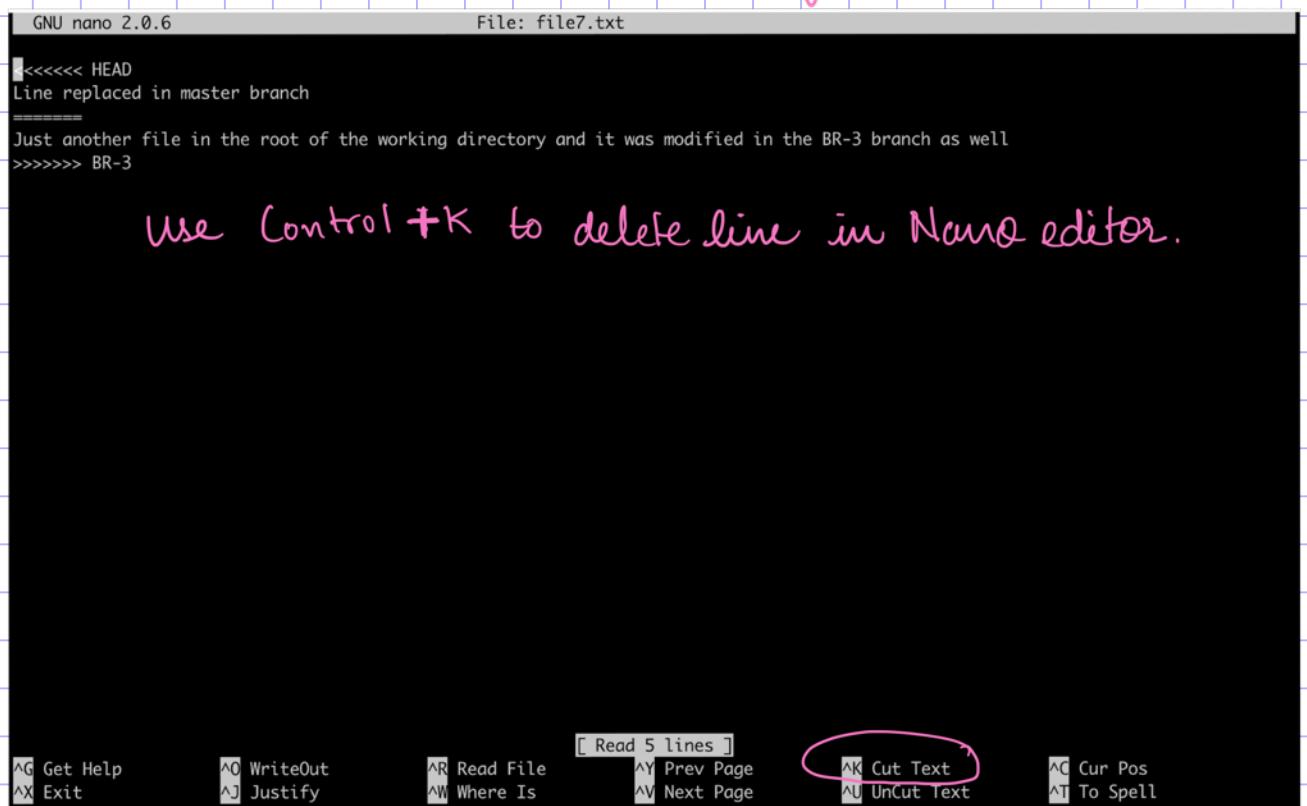
These version are now located in staging area

That's why all application like VSCode, GitHub desktop, SourceTree recognise all those conflicts because they are located in staging area and files that have conflicts are located now in working directory.

```
→ first-project git:(master) ✘ cat new-files/file5.txt  
It is a new file in the new-files folder  
  
→ contents of  
file5.txt  
in working  
directory  
  
<<<<< HEAD  
File5.txt was just changed. This line was modified again in master branch  
=====  
File5.txt was just changed. This line was changed in BR-3 branch  
>>>>> BR-3  
→ first-project git:(master) ✘
```

## Resolving merge conflict in Terminal

```
→ first-project git:(master) ✘ cat file7.txt  
→ contents of file7.txt  
=====  
Line replaced in master branch  
=====  
Just another file in the root of the working directory and it was modified in the BR-3 branch as well  
=====  
>>>>> BR-3  
→ first-project git:(master) ✘ nano file7.txt  
→ using nano editor to  
modify the file.
```



```
GNU nano 2.0.6          File: file7.txt          Modified  
Line replaced in master branch  
  
File Name to Write: file7.txt  
^G Get Help      ^T To Files      M-M Mac Format      M-P Prepend  
^C Cancel        M-D DOS Format    M-A Append       M-B Backup File
```

Now we will not see any conflicts anymore,

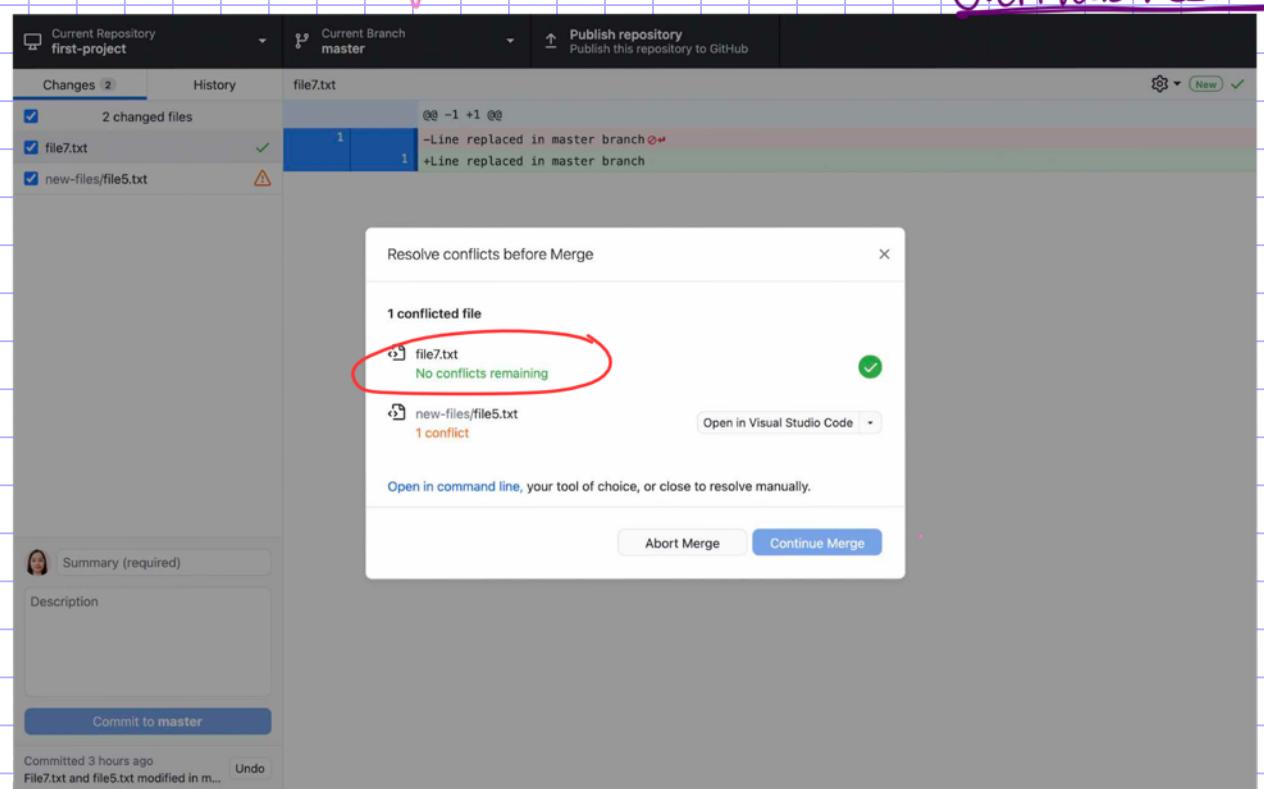
→ first-project git:(master) ✘ cat file7.txt

Line replaced in master branch

→ first-project git:(master) ✘ █

because we have just resolve conflict in this file manually.

GitHub Desktop



```

→ first-project git:(master) ✘ git status
On branch master
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:  file7.txt
    both modified:  new-files/file5.txt
}

no changes added to commit (use "git add" and/or "git commit -a")
→ first-project git:(master) ✘ git add file7.txt
→ first-project git:(master) ✘ git status
On branch master
You have unmerged paths.
  (fix conflicts and run "git commit")
  (use "git merge --abort" to abort the merge)

Changes to be committed:
  modified:  file7.txt

Unmerged paths:
  (use "git add <file>..." to mark resolution)
    both modified:  new-files/file5.txt
}

→ first-project git:(master) ✘

```

Since we have already fixed conflicts in file7.txt

## Resolving conflict in VS Code

Using nano editor for resolving conflicts is not too convenient. Hence sophisticated applications like VS Code are used for conflict resolution.

```

→ first-project git:(master) ✘ git ls-files -s → contents of staging area
100644 356f49ac9ff5360d59fd309dec47a501b10f7032 0
100644 b7aec520dec0a7516c18eb4c68b64ae1eb9b5a5e 0
100644 1e4937a843f913aa04c9a4943c5802cf90f406bc 0
100644 a529592a90242ad5002712bd5ba3b7ae97496666 0
100644 6b4407fa2265f9d47da3662dc353d077fec5f260 1
100644 197cacc5205db2d6d3e099fd23cef7c0a58e1201 2
100644 46ea30340cc914b727a5eee3633a0fed3c5311ce 3
100644 cf9ca3f4f20ab449121b916d6387328a789df46f 0
→ first-project git:(master) ✘ git cat-file -p 1e4937
Line replaced in master branch
→ first-project git:(master) ✘ cat file7.txt
Line replaced in master branch
→ first-project git:(master) ✘

```

.DS\_Store  
file4.txt  
file7.txt → now we don't see three versions of file7.txt  
file8.txt  
new-files/file5.txt  
new-files/file5.txt  
new-files/file5.txt  
new-files/file6.txt

> because we have resolved conflicts in file7.txt and all other versions of file are simply removed

but we still see three different versions of file5.txt and that's because at this moment, conflict in this file is not resolved.

## VSCode

We see here

new-files > file5.txt

```
1 It is a new file in the new-files folder
2
3
4 <<<< HEAD (Current Change)
5 File5.txt was just changed. This line was modified again in master branch
6 =====
7 File5.txt was just changed. This line was changed in BR-3 branch
8 >>>> BR-3 (Incoming Change)
```

SOURCE CONTR... ✓ ⌂ ...

Merge branch 'BR-3'

✓ Merge Changes 1

file5.txt new-files + !

Staged Changes 1

file7.txt M 0

Changes 0

Accept Current Change Accept Incoming Change Accept Both Changes Compare Changes

<<<< HEAD (Current Change)

File5.txt was just changed. This line was modified again in master branch

=====

File5.txt was just changed. This line was changed in BR-3 branch

>>>> BR-3 (Incoming Change)

Ln 7, Col 22 Spaces: 4 UTF-8 LF Plain Text ⌂ ⌂

we need to resolve conflict in this file first and after that stage it

click on this to stage changes

new-files > file5.txt

```
1 It is a new file in the new-files folder
2
3
4 File5.txt was just changed. This line was changed in BR-3 branch
5
```

SOURCE CONTR... ✓ ⌂ ...

Merge branch 'BR-3'

✓ Merge Changes 1

file5.txt new-files + !

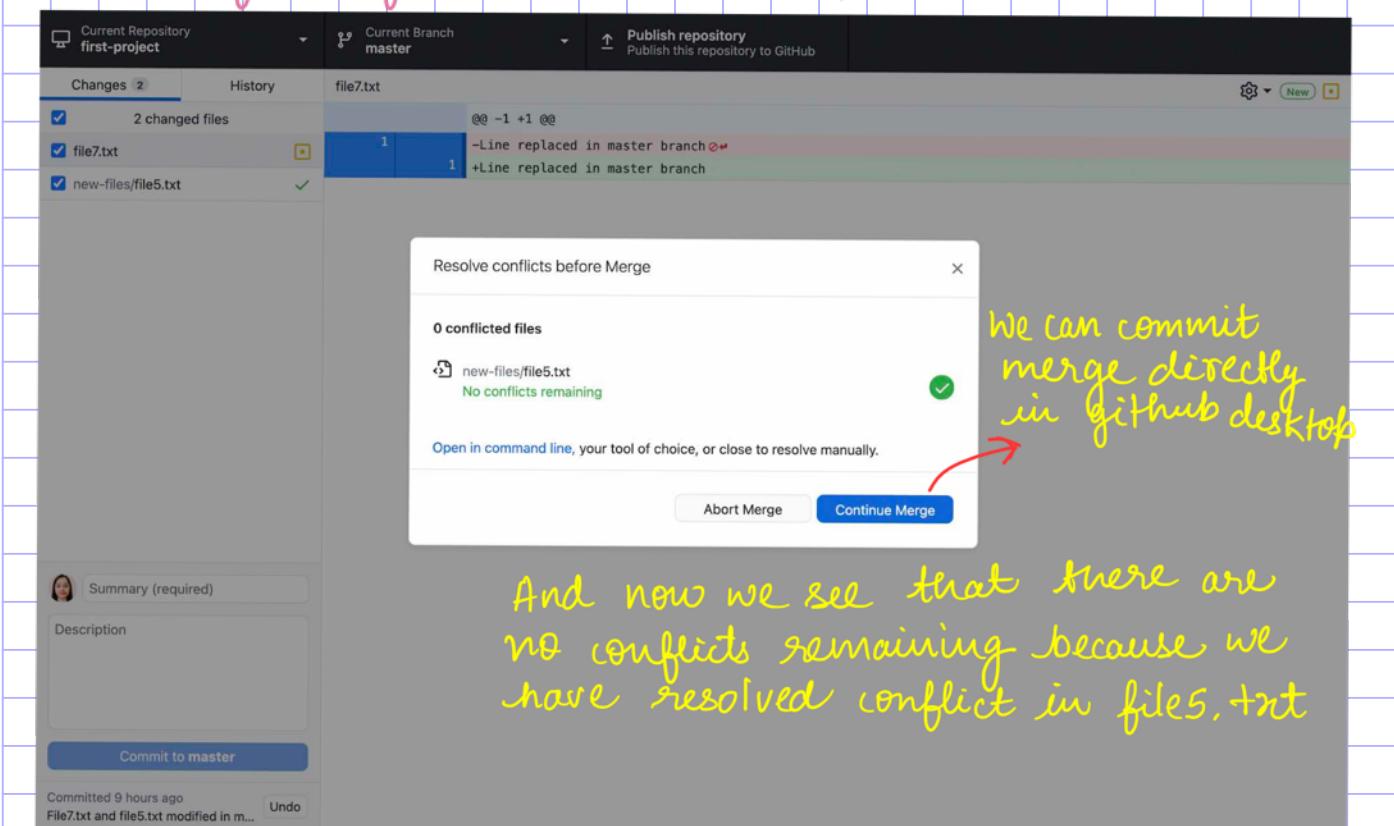
Staged Changes 1

file7.txt M 0

Changes 0

Ln 5, Col 1 Spaces: 4 UTF-8 LF Plain Text ⌂ ⌂

# Seeing changes in GitHub Desktop



And now we see that there are no conflicts remaining because we have resolved conflict in file5.txt

## After staging changes in VSCode.

```
→ first-project git:(master) ✘ git ls-files -s → contents of staging area
100644 356f49ac9ff5360d59fd309dec47a501b10f7032 0
100644 b7aec520dec0a7516c18eb4c68b64ae1eb9b5a5e 0
100644 1e4937a843f913aa04c9a4943c5802cf90f406bc 0
100644 a529592a90242ad5002712bd5ba3b7ae97496666 0
100644 8202d6030d75bd641c0d2531221606061c84ca21 0
100644 cf9ca3f4f20ab449121b916d6387328a789df46f 0
→ first-project git:(master) ✘ git status
On branch master
All conflicts fixed but you are still merging.
  (use "git commit" to conclude merge)
Changes to be committed:
  modified:   file7.txt
  modified:   new-files/file5.txt } →
→ first-project git:(master) ✘
```

now there is single version of file5.txt

## After entering git commit command

```
Merge branch 'BR-3'  
# Conflicts:  
#     file7.txt  
#     new-files/file5.txt  
#  
# It looks like you may be committing a merge.  
# If this is not correct, please run  
#     git update-ref -d MERGE_HEAD  
# and try again.  
  
# Please enter the commit message for your changes. Lines starting  
# with '#' will be ignored, and an empty message aborts the commit.  
#  
# On branch master  
# All conflicts fixed but you are still merging.  
#  
# Changes to be committed:  
#     modified:   file7.txt  
#     modified:   new-files/file5.txt  
#  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~/Desktop/first-project/.git/COMMIT_EDITMSG" 22L, 498B
```

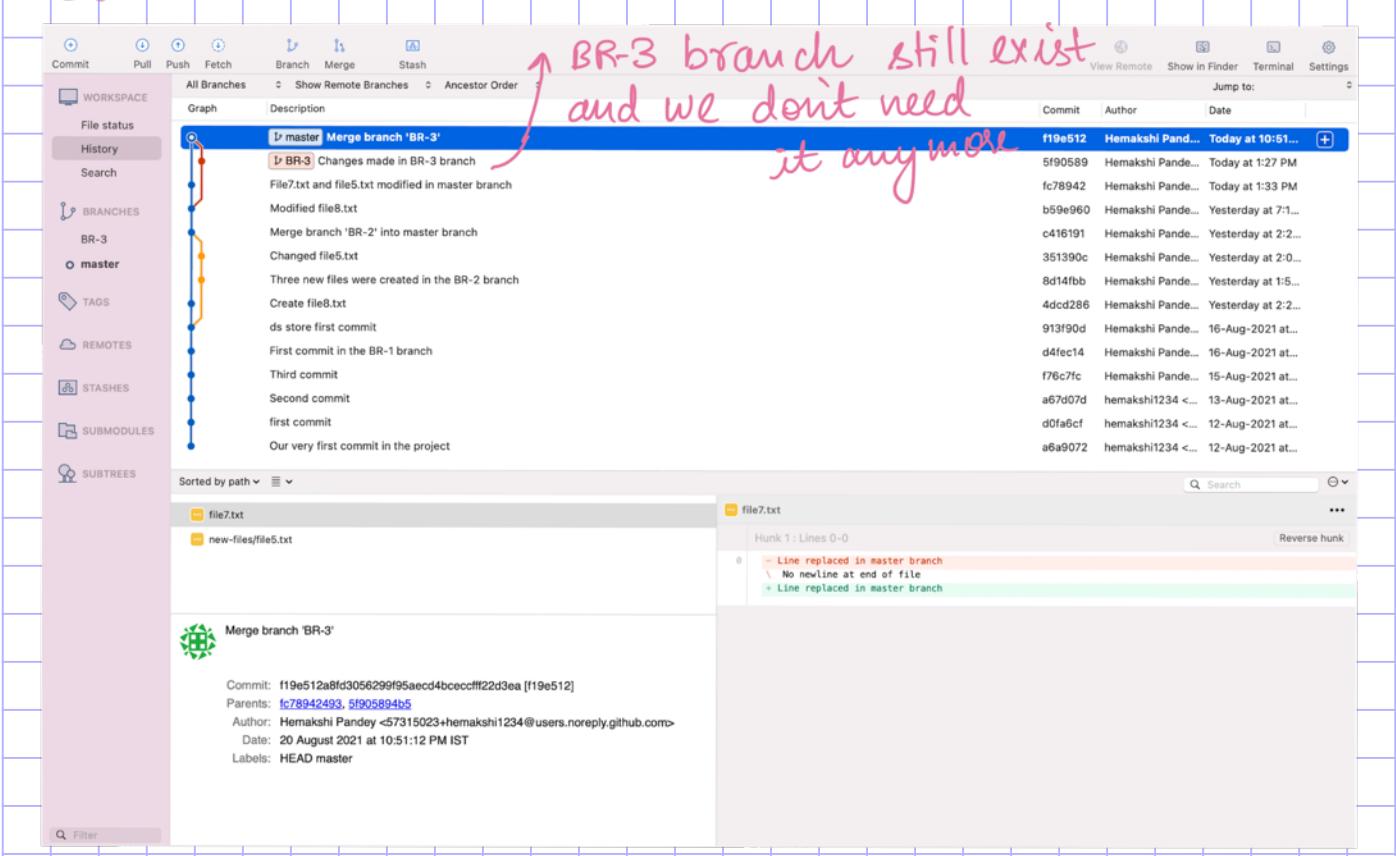
```
→ ~ cd Desktop/first-project  
→ first-project git:(master) ✘ cat .git/MERGE_HEAD  
5f905894b56b1786a2f75039aa96c4d141f3fa6c  
→ first-project git:(master) ✘ git cat-file -p 5f9058  
tree 2d2ab7a532a52392aa3b35557be315cea4af51f6  
parent b59e960871cae8cce4819a009271d1fed53af9dc  
author Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com> 1629446271 +0530  
committer Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com> 1629446271 +0530  
Changes made in BR-3 branch } last commit in BR-3 branch  
→ first-project git:(master) ✘ cat .git/HEAD  
ref: refs/heads/master  
→ first-project git:(master) ✘ cat .git/refs/heads/master  
fc7894249357720a67776fe6fc7ba66226c2c1f8  
→ first-project git:(master) ✘ git log| → SHA1 hash of last commit in master branch
```

It means that during merging process, git creates another HEAD file called MERGE\_HEAD and it contains pointer to last commit in feature branch that you are merging into receiving branch

```
git (vi)  
commit fc7894249357720a67776fe6fc7ba66226c2c1f8 (HEAD -> master)  
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>  
Date: Fri Aug 20 13:33:12 2021 +0530  
  
File7.txt and file5.txt modified in master branch  
  
commit b59e960871cae8cce4819a009271d1fed53af9dc  
Author: Hemakshi Pandey <57315023+hemakshi1234@users.noreply.github.com>  
Date: Thu Aug 19 19:18:57 2021 +0530  
  
Modified file8.txt  
  
commit c416191ffc7e15d2d60b38ae273d75105402f3c5
```



# SourceTree



```
→ first-project git:(master) git branch -d BR-3
Deleted branch BR-3 (was 5f90589). → BR-3 is deleted
→ first-project git:(master)
```

