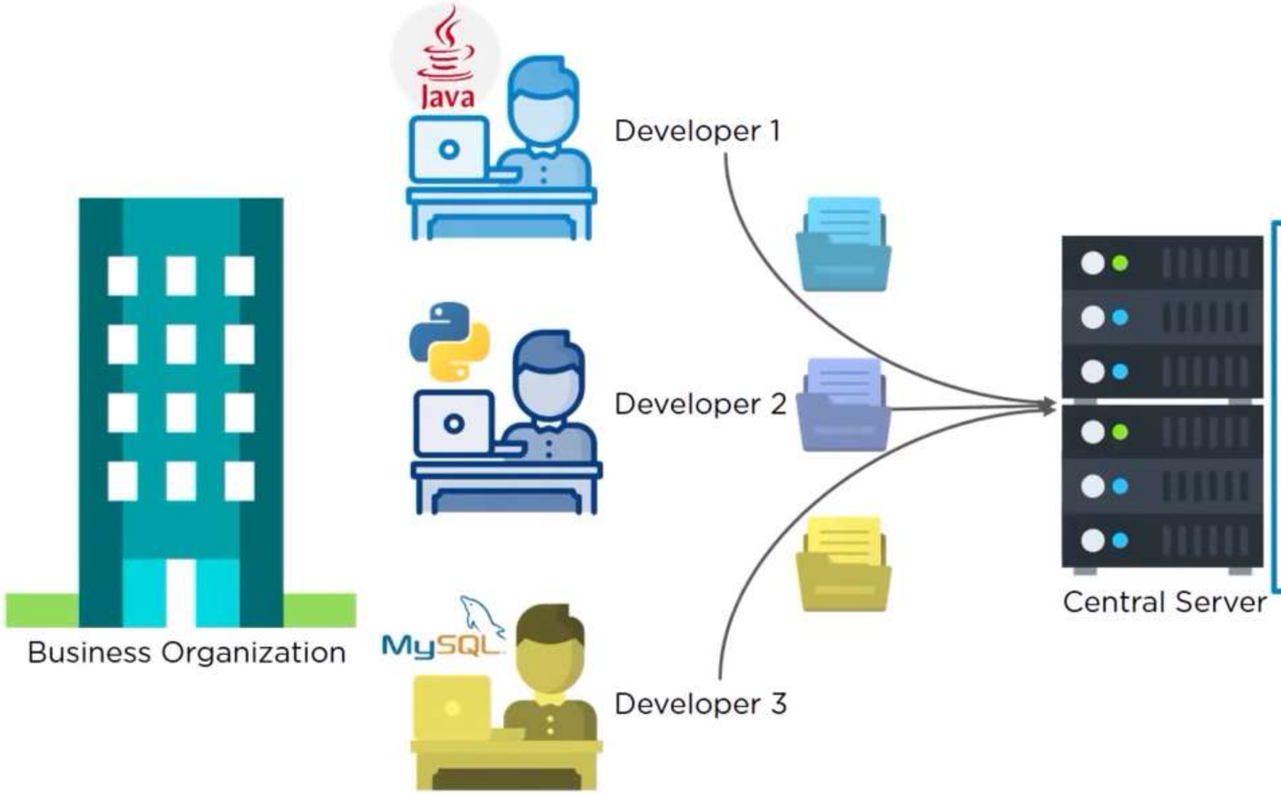
Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4
File View Play Navigate Favourites Help

Scenario before Git



- Developers used to submit their codes to the central server without having a copy of their own
- Any changes made to the source code were not known to the other developers
- There was no communication between any of the developers

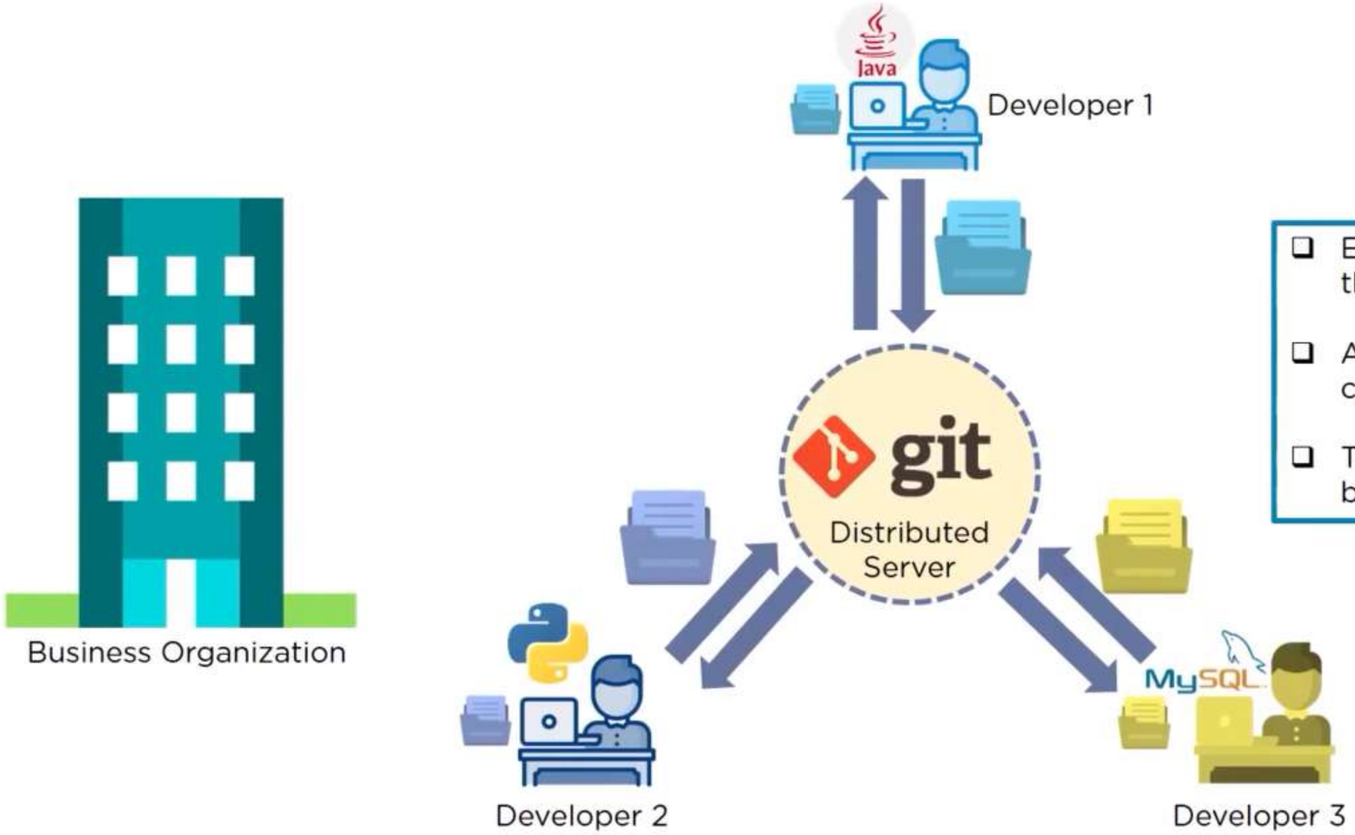
©Simplilearn. All rights reserved.

simplilearn

Playing [H/W]
00:01:03 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4
File View Play Navigate Favourites Help

Scenario after Git



- Every developer has an entire copy of the code on their local systems
- Any change made to the source code can be tracked by others
- There is regular communication between the developers

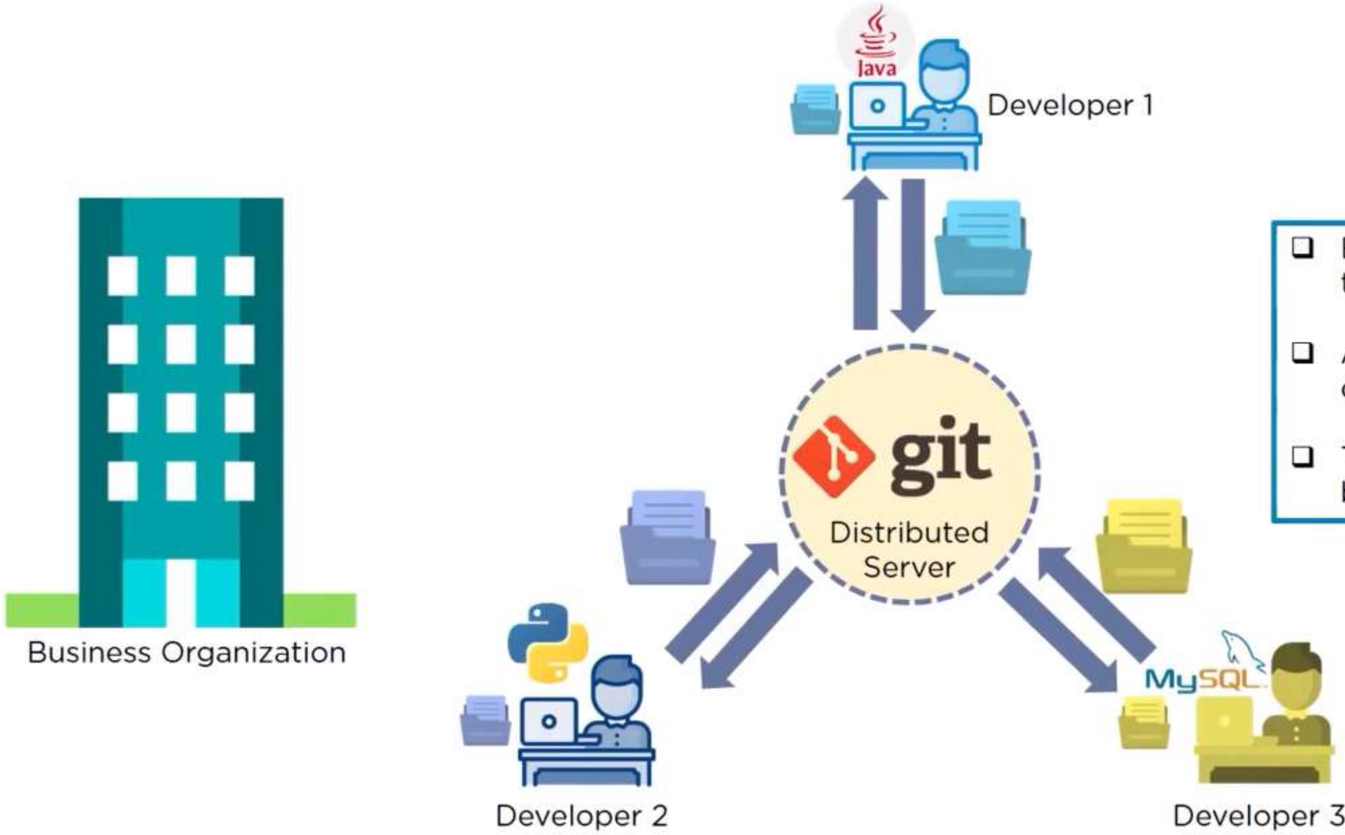
simplilearn

©Simplilearn. All rights reserved.

Playing [H/W] 00:01:40 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4
File View Play Navigate Favourites Help

Scenario after Git



- Every developer has an entire copy of the code on their local systems
- Any change made to the source code can be tracked by others
- There is regular communication between the developers

simplilearn

©Simplilearn. All rights reserved.

Playing [H/W] 00:01:44 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

What's in it for you?

- What is DevOps?
- Tools in DevOps
- Version Control System
- Centralized Vs Distributed Version Control System
- What is Git?

git

- Features of Git
- Workflow of Git
- Branch in Git
- Commands in Git
- Demo on Git

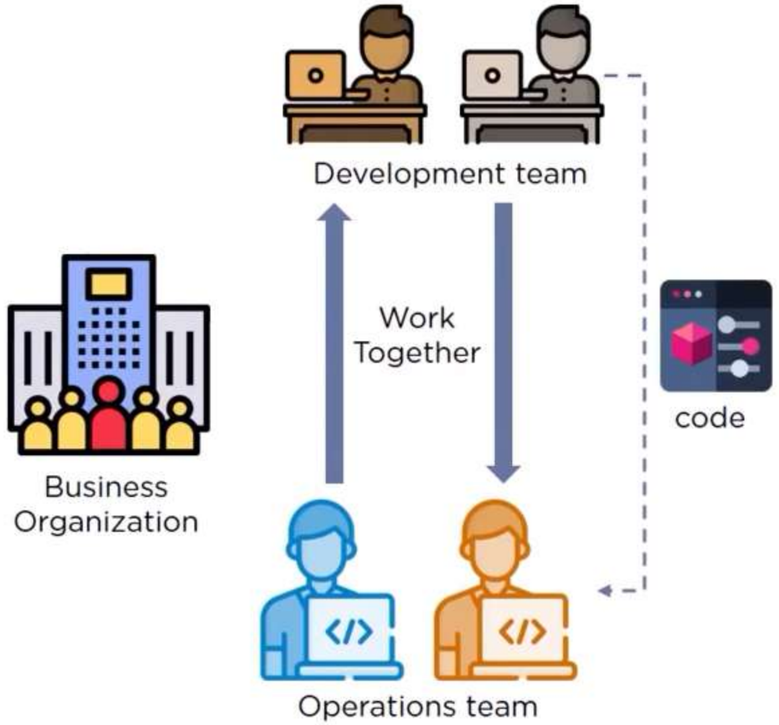
simplilearn

©Simplilearn. All rights reserved.

Playing [H/W] 00:02:28 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4
File View Play Navigate Favourites Help

What is DevOps?



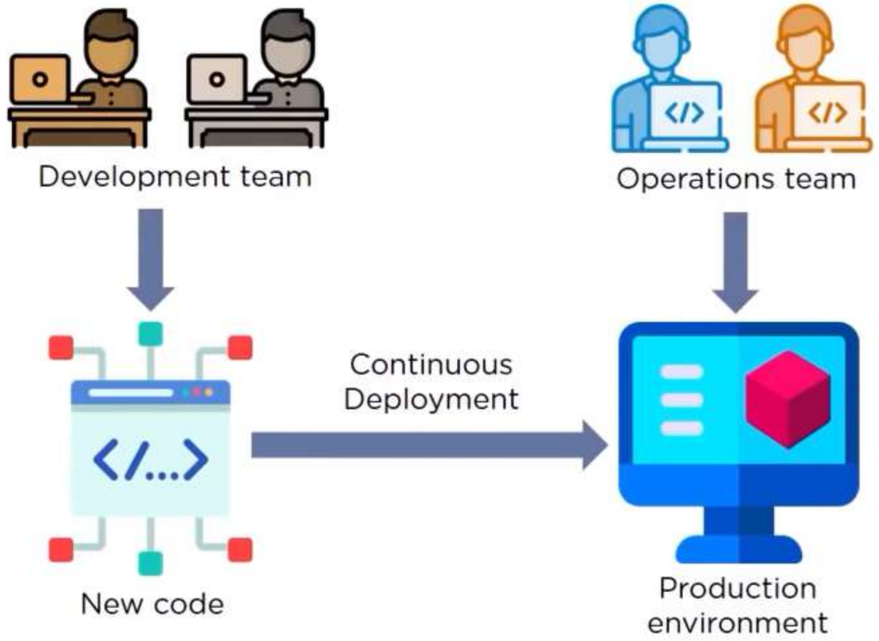
- DevOps is a culture that allows Development and Operations team to work together

©Simplilearn. All rights reserved.

Playing [H/W]
00:02:41 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4
File View Play Navigate Favourites Help

What is DevOps?



- DevOps is a culture that allows Development and Operations team to work together
- Developers continuously develop and test codes
- There is continuous integration taking place through out the lifecycle
- Operations team continuously deploy the codes to the production environment

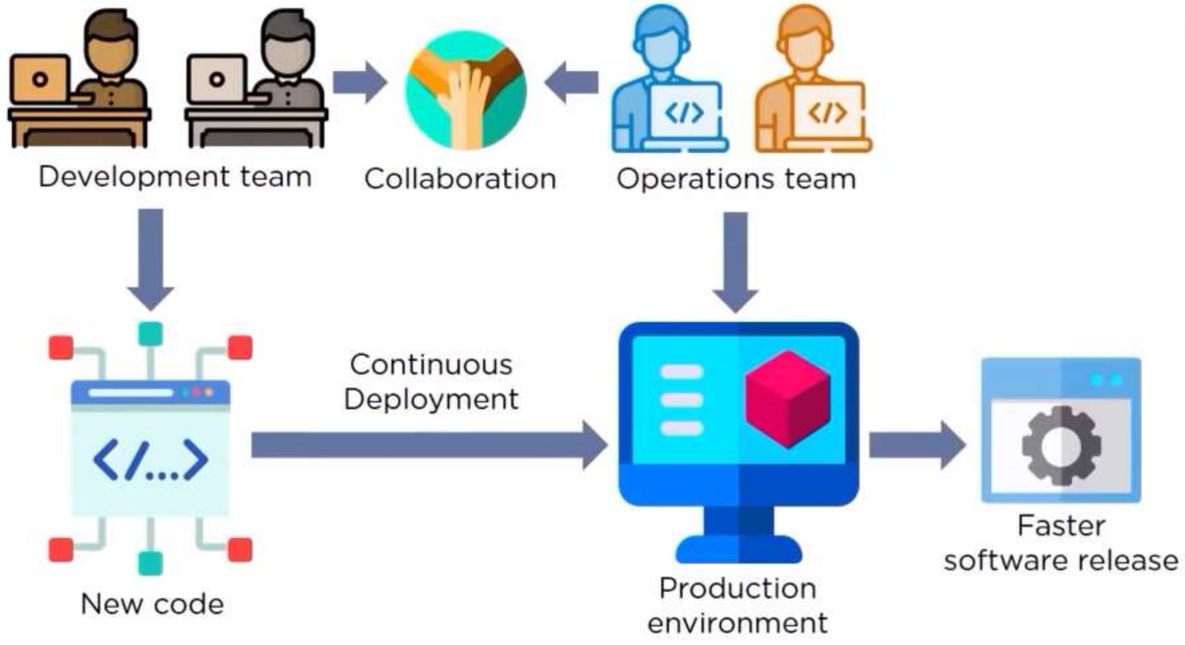
simplilearn

©Simplilearn. All rights reserved.

Playing [H/W] 00:03:13 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4
File View Play Navigate Favourites Help

What is DevOps?



- DevOps is a culture that allows Development and Operations team to work together
- Developers continuously develop and test codes
- There is continuous integration taking place through out the lifecycle
- Operations team continuously deploy the codes to the production environment
- Allows better collaboration, increased trust and faster software releases

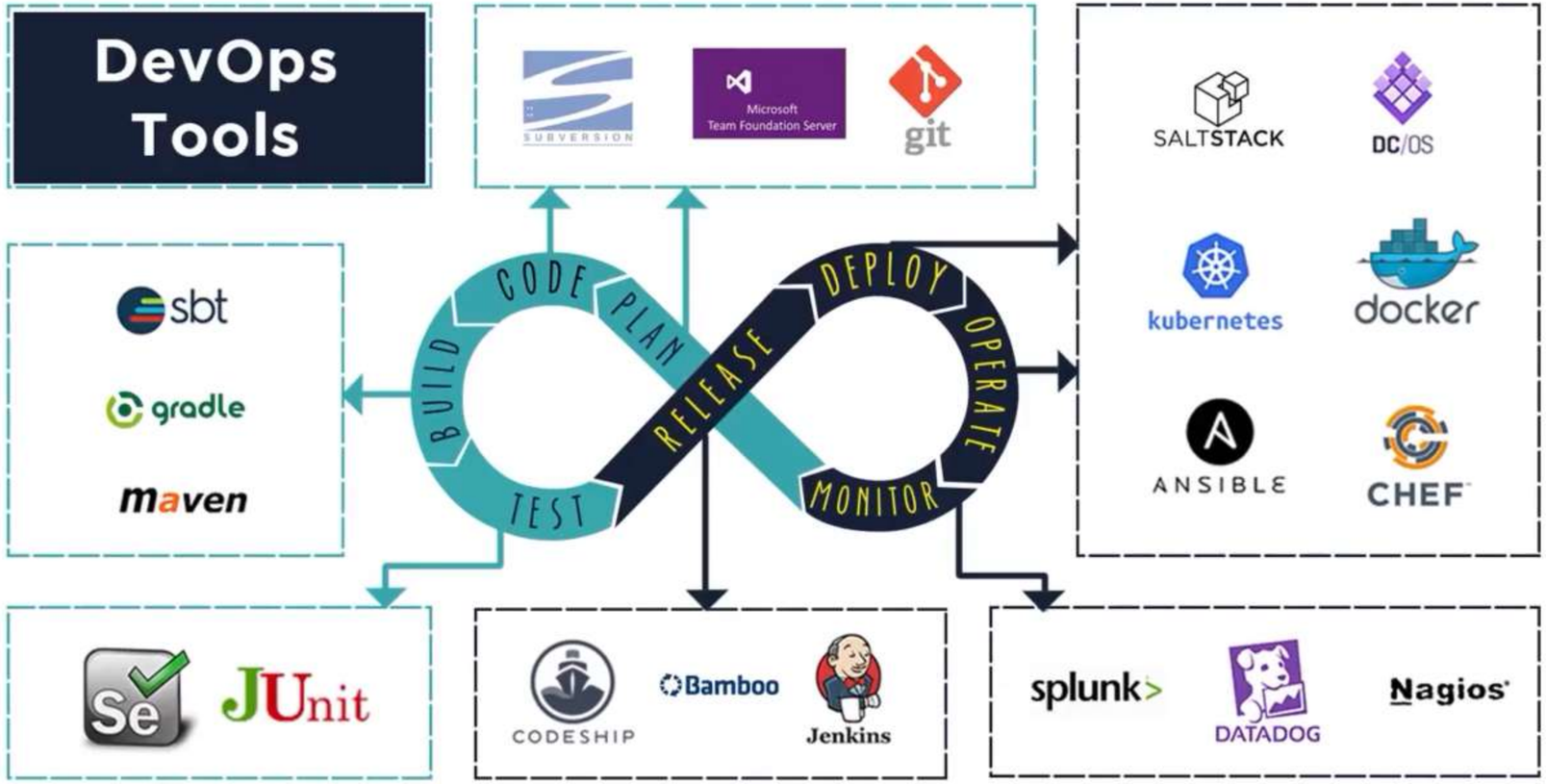
simplilearn

©Simplilearn. All rights reserved.

Playing [H/W] 00:03:17 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4
File View Play Navigate Favourites Help

Tools is DevOps



simplilearn

©Simplilearn. All rights reserved.

Playing [H/W] 00:03:52 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

Tools is DevOps

DevOps Tools

SUBVERSION | Microsoft Team Foundation Server | git

Subversion (SVN), TFS and Git are Source Code Management or Software Configuration Management tools (SCM) used for both planning and coding of the software

simplilearn

©Simplilearn. All rights reserved.

Playing [H/W] 00:04:25 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4
File View Play Navigate Favourites Help

Source Code Management Tools

SUBVERSION

TFS

git

- SVN is a centralized version control system
- It is distributed under the open source license
- It allows you to recover older versions of your data or examine how your data changes

simplilearn

©Simplilearn. All rights reserved.

Playing [H/W] 00:04:44 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4
File View Play Navigate Favourites Help

Source Code Management Tools

TFS

- TFS is a centralized source control tool like SVN
- It is distributed under the Trailware license
- TFS is an application lifecycle management solution that does issue tracking and document management

©Simplilearn. All rights reserved.

Playing [H/W] 00:05:11 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

Source Code Management Tools

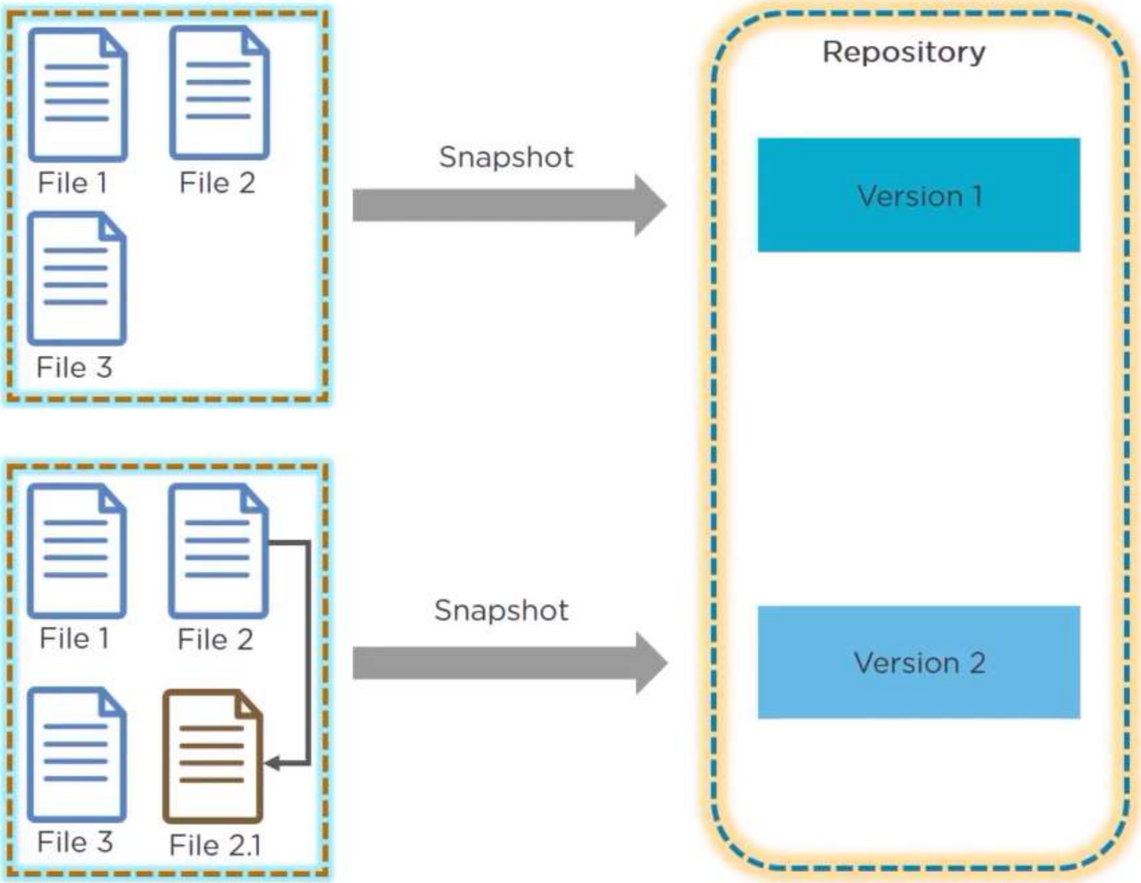
- Git is a distributed version control tool
- It is distributed under the GNU license
- Git is a used for maintaining historical and current versions of source code, web pages, etc

©Simplilearn. All rights reserved.

Paused [H/W] 00:06:15 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4
File View Play Navigate Favourites Help

Version Control System (VCS)



- The diagram shows there are 3 files in the local system
- A snapshot of these files are stored in the remote repository as Version 1
- Now, there has been some changes to file 2 and is updated to file 2.1. This change is stored as Version 2 in the repository
- VCS allows you to track the history of a collection of files
- Each version captures a snapshot of the files at a certain point in time and the VCS allows you to switch between these versions

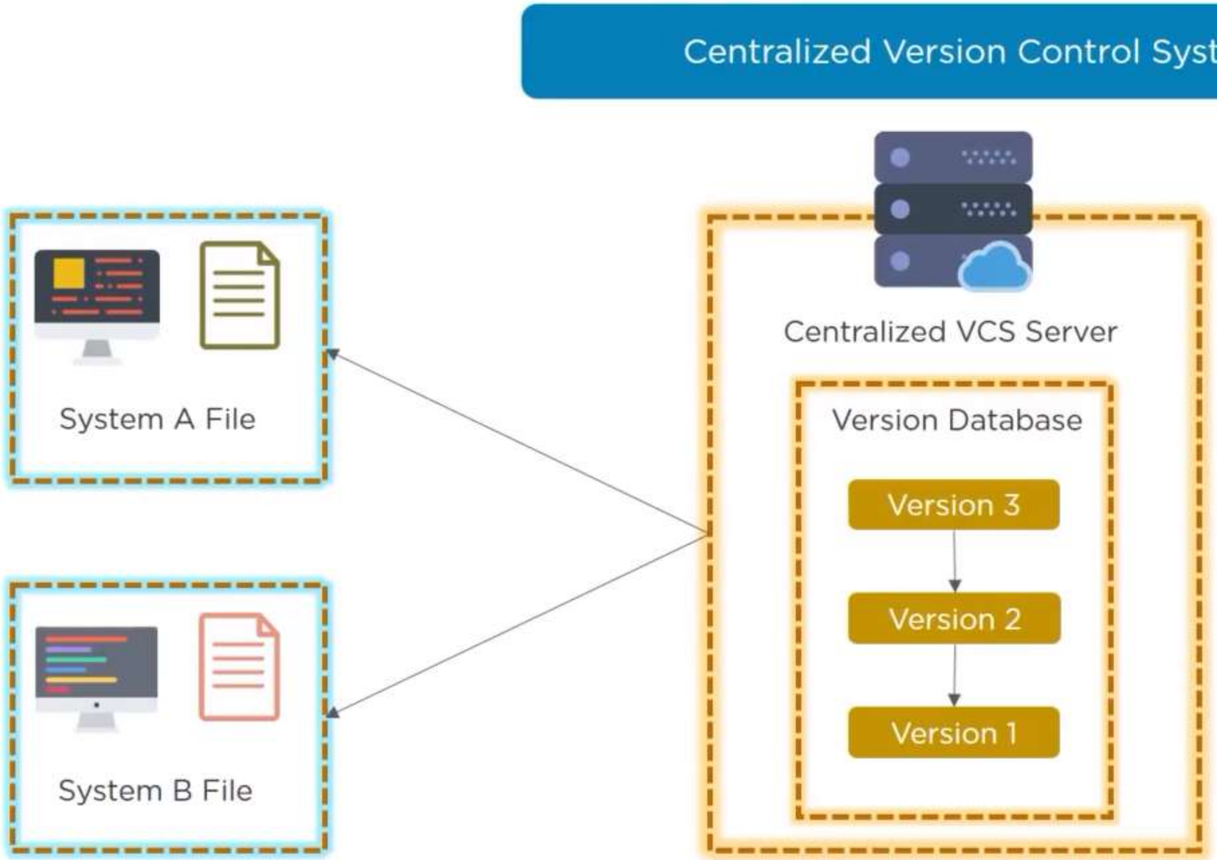
simplilearn

©Simplilearn. All rights reserved.

Playing [H/W] 00:07:16 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4
File View Play Navigate Favourites Help

Centralized Version Control System



- It uses a central server to store all the files
- Every operation is performed directly on the repository
- All the versions of the file are stored on the Central VCS server
- In case the central server crashes, entire data of the project will be lost. Hence, Distributed VCS was introduced

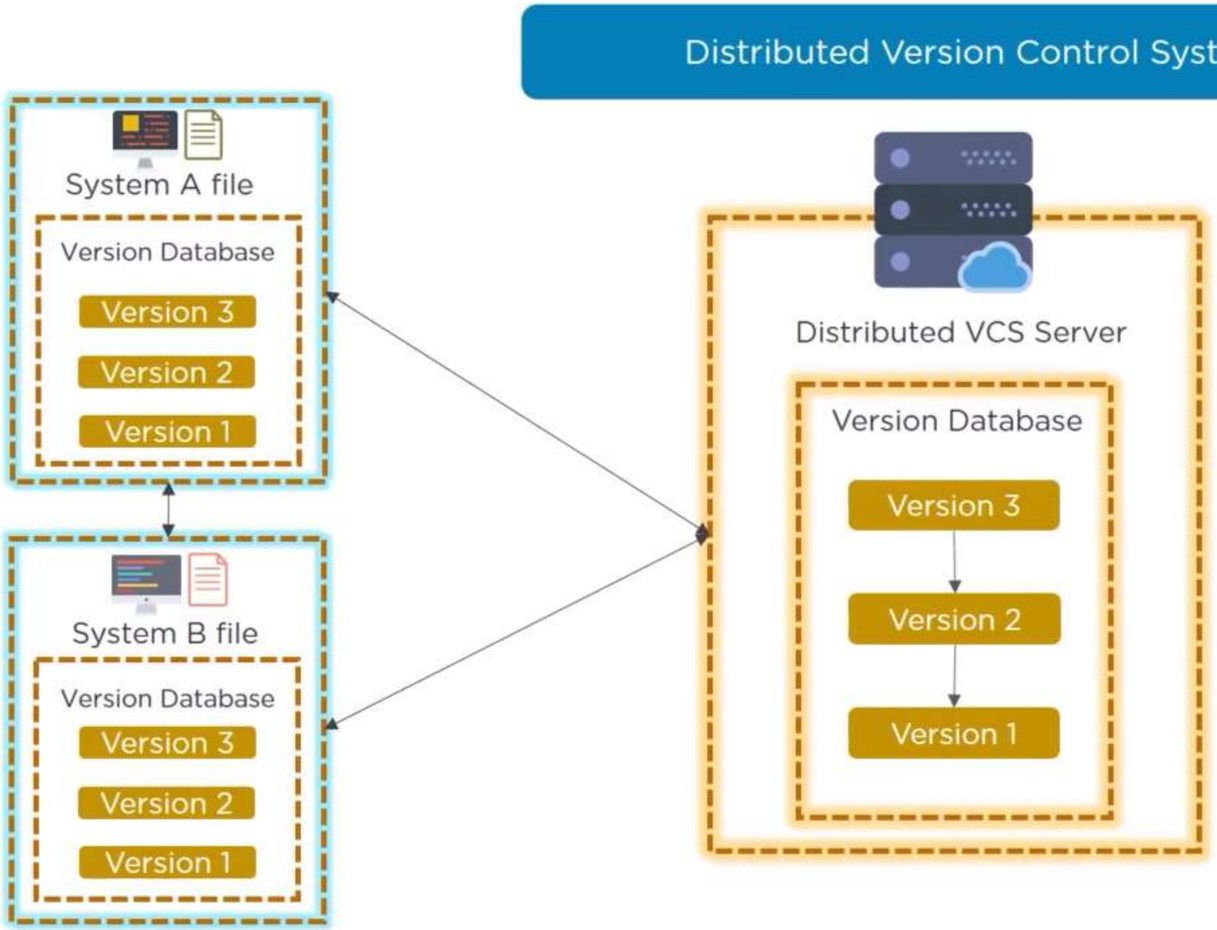
simplilearn

©Simplilearn. All rights reserved.

Playing [H/W] 00:08:06 / 02:30:20
Quick access
Desktop
Downloads
Documents
Pictures
PUBG MODELS
Family photos
Github tutorial vide
Images
Navbar
Screenshots
OneDrive
This PC
3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos
Adam textures
Blood
Captures
Face angels
fire
fortnight
fruits
Hip pop
thunders
Local Disk (C:)
New Volume (E:)
New Volume (F:)
Network
2 items 1 item selected
Type here to search
28°C Mostly cloudy ENG 14:20 21-11-2024

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4
File View Play Navigate Favourites Help

Distributed Version Control System



- Every programmer has a copy of all the versions of the code on their local system
- Distributed VCS moves from the client-server approach of Central VCS to peer-to-peer approach
- They can update their local repositories with new data from the central server and changes gets reflected to the main repository
- Git is one such Distributed VCS tool

©Simplilearn. All rights reserved.

simplilearn

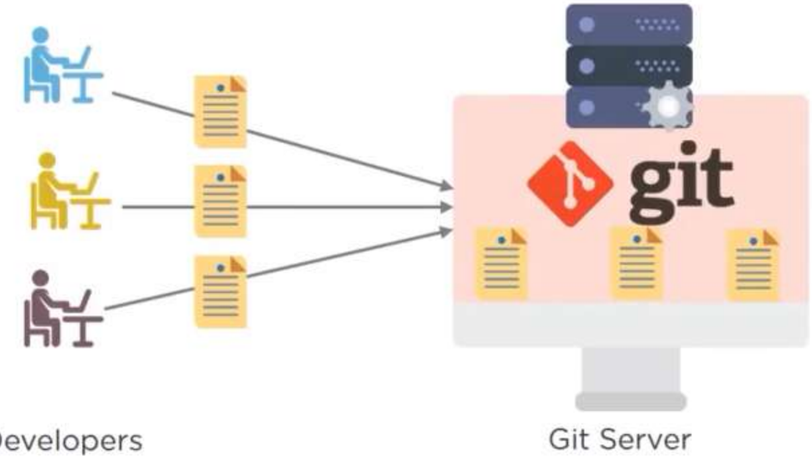
Playing [H/W] 00:09:07 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

What is Git?

Git is a Version Control System for tracking changes in computer files. It is generally used for Source Code Management in software development.



Git is used to track changes in the source code

Distributed version control tool used for source code management

Allows multiple developers to work together

simplilearn

©Simplilearn. All rights reserved.

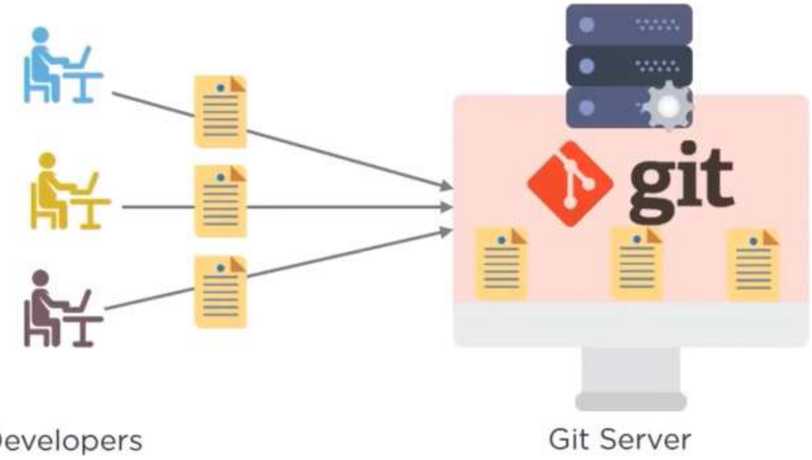
Playing [H/W] 00:10:12 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

What is Git?

Git is a Version Control System for tracking changes in computer files. It is generally used for Source Code Management in software development.



- Git is used to track changes in the source code
- Distributed version control tool used for source code management
- Allows multiple developers to work together
- Supports non-linear development because of its thousands of parallel branches

simplilearn

©Simplilearn. All rights reserved.

Playing [H/W] 00:10:26 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4
File View Play Navigate Favourites Help

Features of Git

git

1. Tracks History
2. Free and Open Source
3. Supports Non-Linear Development
4. Creates Backup
5. Scalable
6. Supports Collaboration
7. Branching is easier
8. Distributed Development

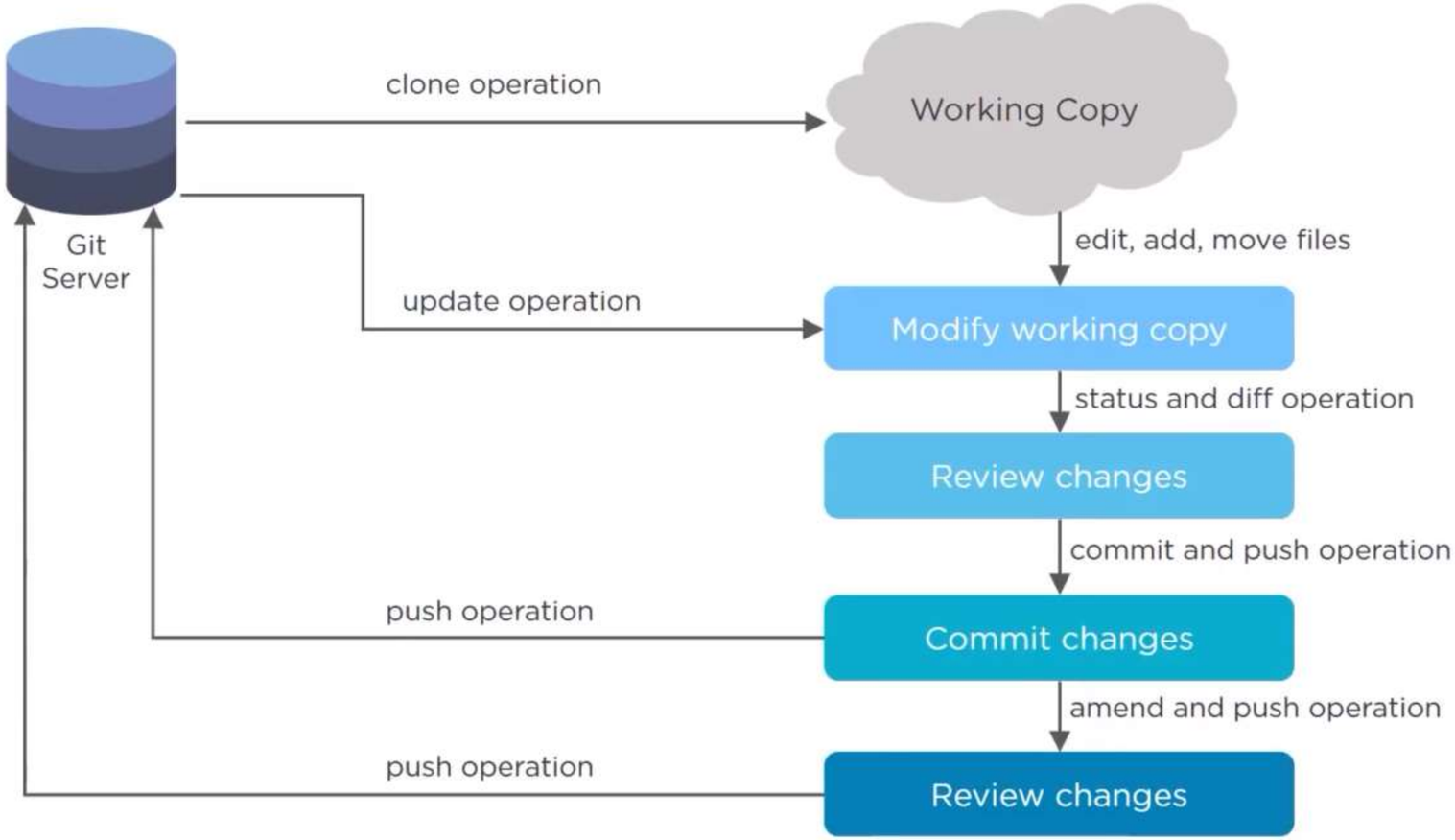
simplilearn

©Simplilearn. All rights reserved.

Playing [H/W] 00:11:37 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4
File View Play Navigate Favourites Help

Git Workflow



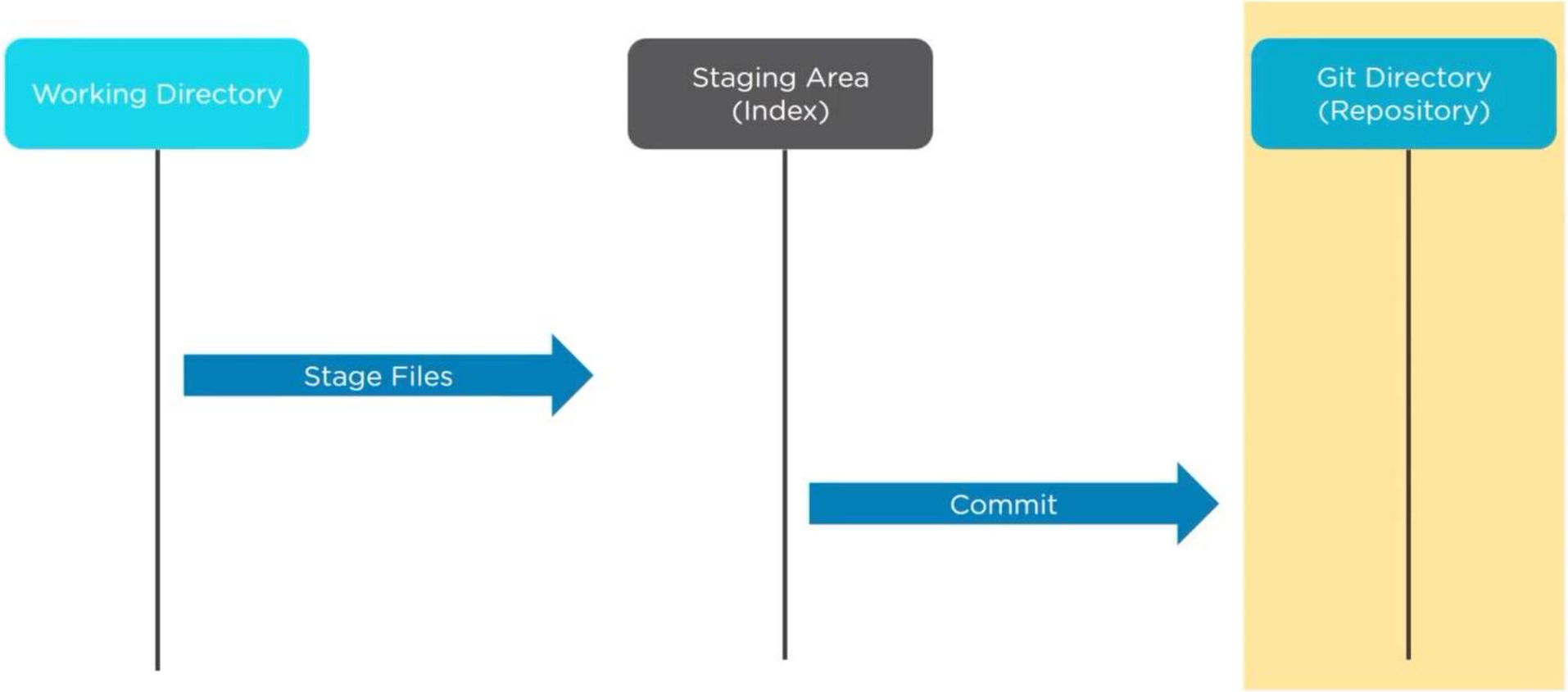
©Simplilearn. All rights reserved.

Playing [H/W] 00:11:49 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

Git Workflow – 3 States



Do a commit that stores the snapshots permanently to your Git directory

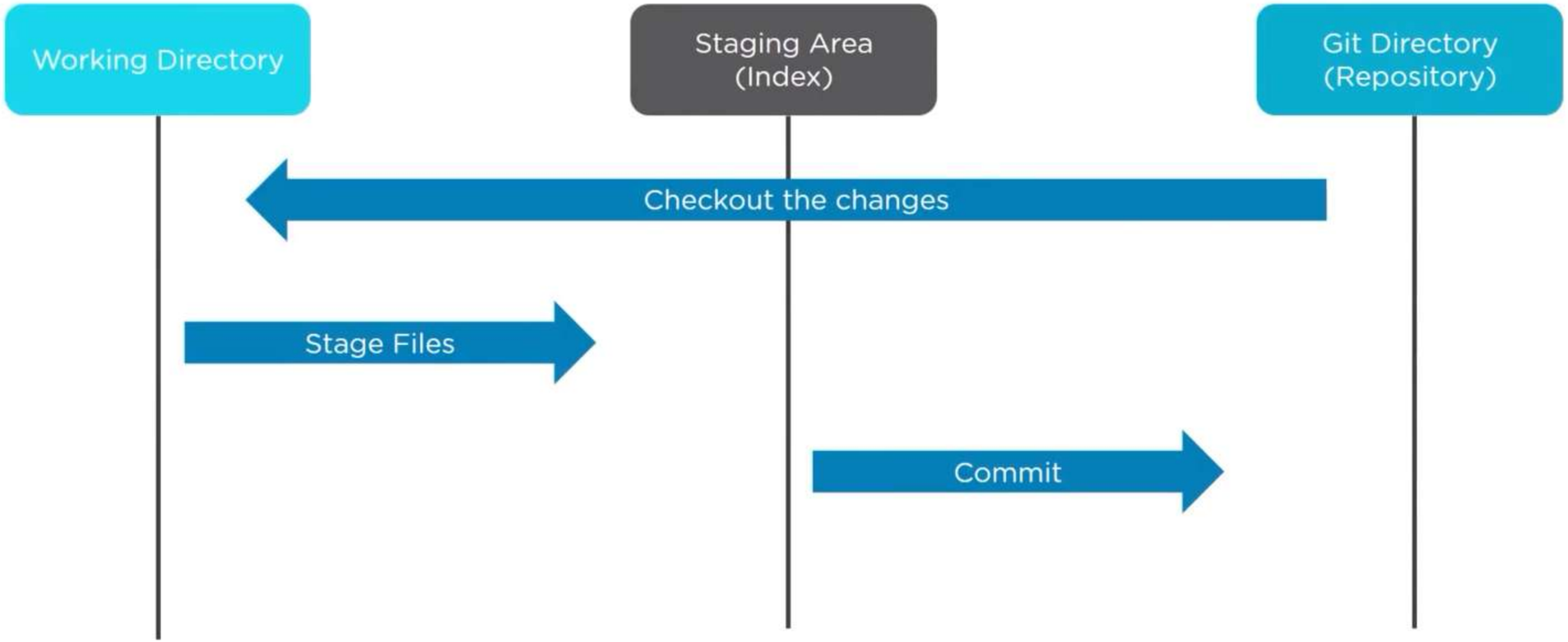
simplilearn

©Simplilearn. All rights reserved.

Playing [H/W] 00:12:41 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4
File View Play Navigate Favourites Help

Git Workflow – 3 States



Checkout any existing version, make changes, stage them and commit

simplilearn

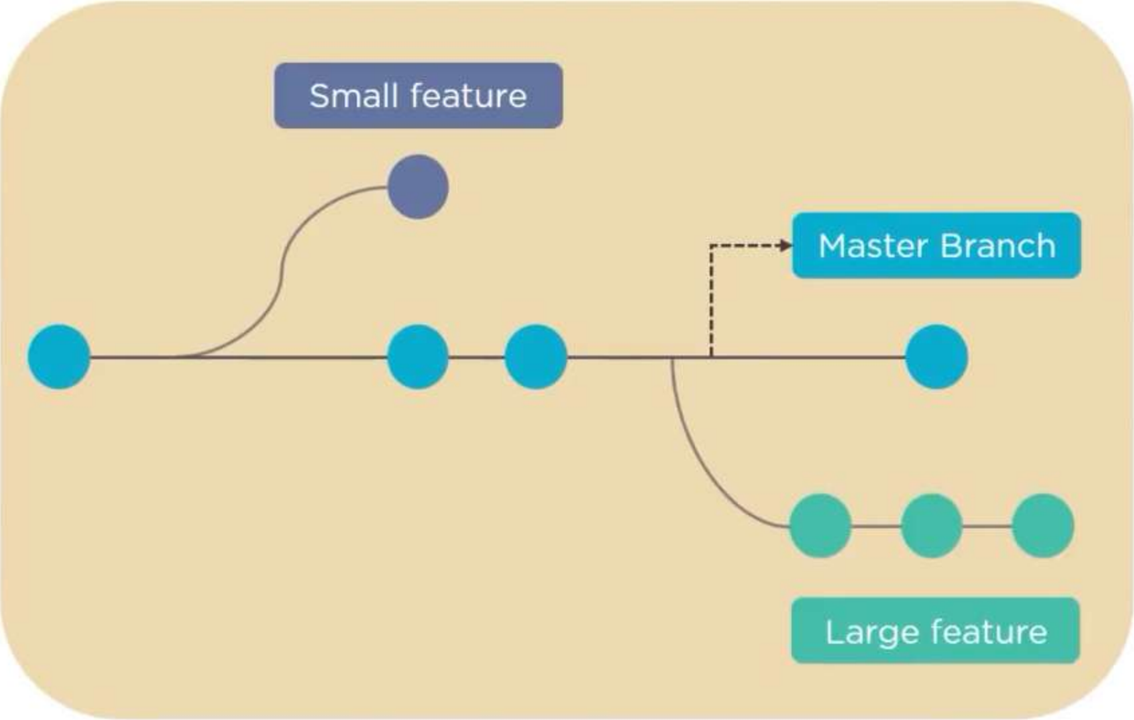
©Simplilearn. All rights reserved.

Playing [H/W] 00:12:51 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4
File View Play Navigate Favourites Help

Branch in Git

- Branch in Git is used to keep your changes until they are ready
- You can do your work on a branch while the main branch (master) remains stable. After you are done with your work, you can merge it to the main branch



- The diagram shows there is a Master branch
- There are 2 more branches Small feature and Large feature working separately
- Once the work is complete for the two separate branches, you can merge it to the master branch

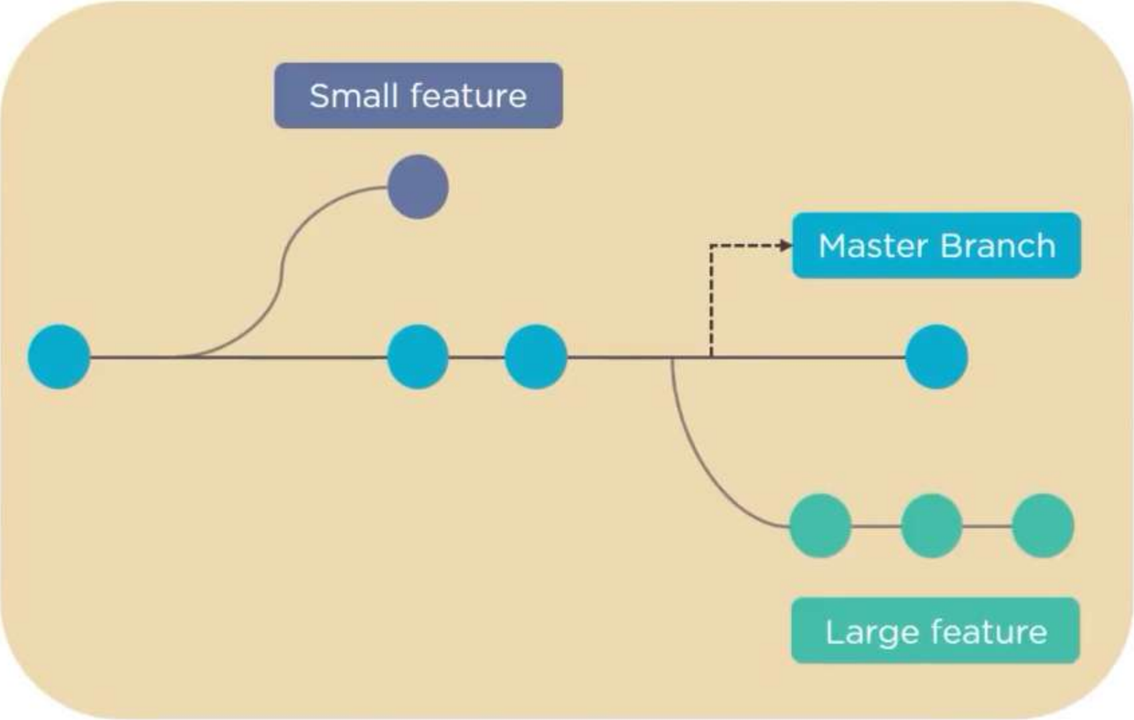
©Simplilearn. All rights reserved.

14:07
Playing [H/W]
00:14:10 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4
File View Play Navigate Favourites Help

Branch in Git

- Branch in Git is used to keep your changes until they are ready
- You can do your work on a branch while the main branch (master) remains stable. After you are done with your work, you can merge it to the main branch



- The diagram shows there is a Master branch
- There are 2 more branches Small feature and Large feature working separately
- Once the work is complete for the two separate branches, you can merge it to the master branch

simplilearn

©Simplilearn. All rights reserved.

Playing [H/W] 00:14:10 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4
File View Play Navigate Favourites Help

Commands in Git

simplilearn

©Simplilearn. All rights reserved.

Playing [H/W] 00:14:16 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

What's in it for you?

- Download and install Git
- Git bash interface
- Basic Git commands
- Create a local repository
- Connect to remote repository
- Push the file to GitHub

simplilearn

©Simplilearn. All rights reserved.

Playing [H/W] 00:15:18 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

MINGW64:/c/Users/Simplilearn

```
SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~ (master)
$ git --version
git version 2.19.1.windows.1

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~ (master)
$ |
```

Playing [H/W]

00:16:46 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

MINGW64:/c/Users/Simplilearn

```
SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~ (master)
$ git --version
git version 2.19.1.windows.1

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~ (master)
$ git help config

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~ (master)
$ |
```

Playing [H/W] 00:17:19 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

MINGW64:/c/Users/Simplilearn/test

```
SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~ (master)
$ git --version
git version 2.19.1.windows.1

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~ (master)
$ git help config

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~ (master)
$ git config --help

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~ (master)
$ mkdir test

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~ (master)
$ cd test

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ git init
Initialized empty Git repository in C:/Users/Simplilearn/test/.git/

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$
```

Playing [H/W] 00:18:04 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

MINGW64:/c/Users/Simplilearn/test

```
SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~ (master)
$ git --version
git version 2.19.1.windows.1

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~ (master)
$ git help config

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~ (master)
$ git config --help

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~ (master)
$ mkdir test

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~ (master)
$ cd test

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ git init
Initialized empty Git repository in C:/Users/Simplilearn/test/.git/

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$
```

Playing [H/W] 00:17:53 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

MINGW64:/c/Users/Simplilearn/test

```
SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~ (master)
$ git --version
git version 2.19.1.windows.1

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~ (master)
$ git help config

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~ (master)
$ git config --help

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~ (master)
$ mkdir test

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~ (master)
$ cd test

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ git init
Initialized empty Git repository in C:/Users/Simplilearn/test/.git/

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ git status
On branch master

No commits yet

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

        demo.txt

nothing added to commit but untracked files present (use "git add" to track)

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ git add demo.txt

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ git commit -m "comitting a text file"
[master (root-commit) ba65497] comitting a text file
 1 file changed, 1 insertion(+)
 create mode 100644 demo.txt

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$
```

Playing [H/W] 00:19:01 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

MINGW64:/c/Users/Simplilearn/test

```
SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~ (master)
$ git --version
git version 2.19.1.windows.1

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~ (master)
$ git help config

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~ (master)
$ git config --help

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~ (master)
$ mkdir test

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~ (master)
$ cd test

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ git init
Initialized empty Git repository in C:/Users/Simplilearn/test/.git/

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ git status
On branch master

No commits yet

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

        demo.txt

nothing added to commit but untracked files present (use "git add" to track)

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ git add demo.txt

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ git commit -m
```

5:28 AM
11/8/2018

Playing [H/W]

00:18:51 / 02:30:20

Quick access
Desktop
Downloads
Documents
Pictures
PUBG MODELS
Family photos
Github tutorial vi
Images
Navbar
Screenshots
OneDrive
This PC
3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos
Adam textures
Blood
Captures
Face angels
fire
fortnight
fruits
Hip pop
thunders
Local Disk (C:)
New Volume (E:)
New Volume (F:)
Network

2 items 1 item selecte

Type here to search

28°C Mostly cloudy ENG 14:38 21-11-2024

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

MINGW64:/c/Users/Simplilearn/test

```
SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~ (master)
$ git --version
git version 2.19.1.windows.1

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~ (master)
$ git help config

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~ (master)
$ git config --help

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~ (master)
$ mkdir test

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~ (master)
$ cd test

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ git init
Initialized empty Git repository in C:/Users/Simplilearn/test/.git/

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ git status
On branch master

No commits yet

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

        demo.txt

nothing added to commit but untracked files present (use "git add" to track)

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ git add demo.txt

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ git commit -m "comitting a text file"
[master (root-commit) ba65497] comitting a text file
 1 file changed, 1 insertion(+)
 create mode 100644 demo.txt

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ clear
```

Playing [H/W] 00:19:04 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

MINGW64:/c/Users/Simplilearn/test

```
SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ git config --global user.username simplilearn-github

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ git remote add origin https://github.com/simplilearn-github/test_demo.git

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ git push origin master
```

5:31 AM
11/8/2018

Playing [H/W]

00:20:30 / 02:30:20

Quick access
Desktop
Downloads
Documents
Pictures
PUBG MODELS
Family photos
Github tutorial vi
Images
Navbar
Screenshots
OneDrive
This PC
3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos
Adam textures
Blood
Captures
Face angels
fire
fortnight
fruits
Hip pop
thunders
Local Disk (C:)
New Volume (E:)
New Volume (F:)
Network

2 items 1 item selecte

Type here to search

28°C Mostly cloudy ENG 14:42 21-11-2024

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

MINGW64:/c/Users/Simplilearn/test

```
SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ git config --global user.username simplilearn-github

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ git remote add origin https://github.com/simplilearn-github/test_demo.git

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ git push origin master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 238 bytes | 79.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
remote:
remote: Create a pull request for 'master' on GitHub by visiting:
remote:      https://github.com/simplilearn-github/test_demo/pull/new/master
remote:
To https://github.com/simplilearn-github/test_demo.git
 * [new branch]      master -> master

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$
```

5:31 AM
11/8/2018

Playing [H/W]

00:20:33 / 02:30:20

28°C Mostly cloudy ENG 14:42 21-11-2024

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

Demo on Git

Let's implement some commands on Git using a Git Shell

1. *Check the version of Git*

```
SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~
$ git --version
git version 2.18.0.windows.1
```

2. *Set up some global config variables* ------> If you are working with other developers, you need to know who is checking the code in and out and making the changes

```
SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~
$ git config --global user.name "Sandeep.D"
```

```
SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~
$ git config --global user.email "sandeep.d@simplilearn.net"
```

```
SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~
$ git config --list
```

simplilearn

©Simplilearn. All rights reserved.

Playing [H/W] 00:21:07 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

```
SSPL-LP-DNS-YT01+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~ (master)
$ git --version
git version 2.18.0.windows.1

SSPL-LP-DNS-YT01+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~ (master)
$ git config --global user.name "Sandeep"

SSPL-LP-DNS-YT01+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~ (master)
$ git config --global user.email 'sandeep.d@simplilearn.net'

SSPL-LP-DNS-YT01+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~ (master)
$ git config --list
core.symlinks=false
core.autocrlf=true
core.fscache=true
color.diff=auto
color.status=auto
color.branch=auto
color.interactive=true
help.format=html
rebase.autosquash=true
http.sslcainfo=C:/Program Files/Git/mingw64/ssl/certs/ca-bundle.crt
http.sslbackend=openssl
diff.astextplain.textconv=astextplain
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
credential.helper=manager
user.username=simplilearn-github
user.name=Sandeep
user.email=sandeep.d@simplilearn.net
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
color.ui=true
suer.name=Sandeep.D
core.repositoryformatversion=0
core.filemode=false
core.bare=false
core.logallrefupdates=true
core.symlinks=false
core.ignorecase=true

SSPL-LP-DNS-YT01+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~ (master)
$
```

01:56 -21:08

©Simplilearn. All rights reserved.

simplilearn

Playing [H/W] 00:22:13 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4
File View Play Navigate Favourites Help

Demo on Git

Let's implement some of the commands on Git using a Git Shell

3. *Need Help?*

```
SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~
$ git help config

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~
$ git config --help
```

-------> This will lead to the Git help page on the browser

```
SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~
$ git help add
```

git-add(1) Manual Page

NAME

git-add - Add file contents to the index

SYNOPSIS

git add [--verbose | -v] [--dry-run | -n] [--force | -f] [--interactive | -i] [--patch | -p] [--edit | -e] [--[no-]all | --[no-]ignore-removal | [--update | -u]] [--intent-to-add | -N] [--refresh] [--ignore-errors] [--ignore-missing] [--renormalize] [--chmod=(+|-)x] [--] [<pathspec>...]

git-config(1) Manual Page

NAME

git-config - Get and set repository or global options

SYNOPSIS

git config [<file-option>] [--type=<type>] [--show-origin] [-z|--null] name [value [value_regex]]
git config [<file-option>] [--type=<type>] --add name value
git config [<file-option>] [--type=<type>] --replace-all name value [value_regex]
git config [<file-option>] [--type=<type>] [--show-origin] [-z|--null] --get name [value_regex]
git config [<file-option>] [--type=<type>] [--show-origin] [-z|--null] --get-all name [value_regex]
git config [<file-option>] [--type=<type>] [--show-origin] [-z|--null] [--name-only] --get-regexp name_regex [value_regex]
git config [<file-option>] [--type=<type>] [-z|--null] --get-urlmatch name URL

©Simplilearn. All rights reserved.

simplilearn

Playing [H/W] 00:22:22 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

Demo on Git

Let's implement some of the commands on Git using a Git Shell

4. *Create a "test" repository in the local system*

```
SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~
$ mkdir test
```

5. *Move to the test repository*

```
SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~
$ cd test
```

6. *Create a new git instance for a project*

```
SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test
$ git init
Initialized empty Git repository in C:/Users/Simplilearn/test/.git/
```

©Simplilearn. All rights reserved.

simplilearn

Playing [H/W] 00:22:57 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

Demo on Git

Let's implement some of the commands on Git using a Git Shell

7. *Create a text file called info.txt in the test folder, write something and save it*

info - Notepad

File Edit Format View Help

```
Name - Sam
Registration no.: 12479
```

8. *Check the status of the repository*

```
SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ git status
On branch master

No commits yet

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

        info.txt

nothing added to commit but untracked files present (use "git add" to track)
```

©Simplilearn. All rights reserved.

simplilearn

Playing [H/W] 00:23:39 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4
File View Play Navigate Favourites Help

Demo on Git

Let's implement some of the commands on Git using a Git Shell

9. *Add the file you created to make a commit*

```
SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ git add info.txt
```

10. *Commit those changes to the repository's history with a short message*

```
SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ git commit -m "commiting a text file"
[master (root-commit) 1c0673b] commiting a text file
 1 file changed, 2 insertions(+)
 create mode 100644 info.txt
```

©Simplilearn. All rights reserved.

simplilearn
simplilearn

Playing [H/W]
00:24:36 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

Demo on Git

Let's implement some of the commands on Git using a Git Shell

11. *Make some changes to the file and save*

info - Notepad

```
Name - Sam
Registration no.: 12479
Address - #303, East Main Street, Stamford
```

12. *View the difference in file now and how it was at your last commit*

```
SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ git diff
diff --git a/info.txt b/info.txt
index 0dd82a6..706570c 100644
--- a/info.txt
+++ b/info.txt
@@ -1,2 +1,6 @@
 Name - Sam
 Registration no.: 12479
+Address - #303, East Main Street, Stamford
```

©Simplilearn. All rights reserved.

Playing [H/W] 00:25:31 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

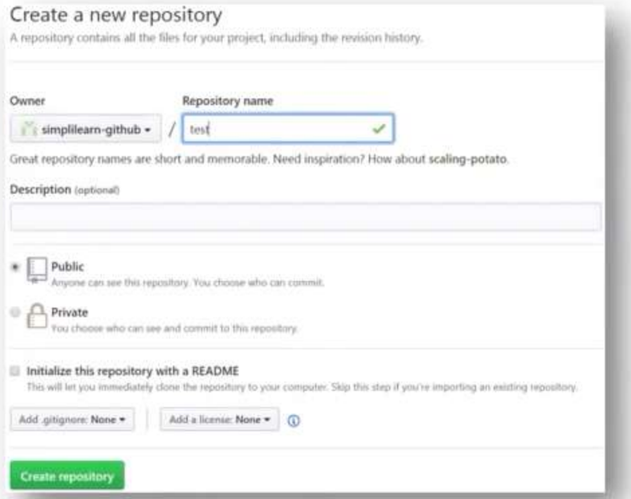
Demo on Git

Let's implement some of the commands on Git using a Git Shell

13. *Add GitHub username to Git Configuration*

```
SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ git config --global user.username simplilearn-github
```

14. *Create a remote repository*



©Simplilearn. All rights reserved.

simplilearn

Playing [H/W] 00:26:16 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4
File View Play Navigate Favourites Help

Demo on Git

Let's implement some of the commands on Git using a Git Shell

15. *Connect the local repository to your remote repository*

```
SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ git remote add origin https://github.com/simplilearn-github/test.git
```

16. *Push the file to the remote repository*

```
SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ git push origin master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 270 bytes | 135.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/simplilearn-github/test.git
 * [new branch]      master -> master
```

©Simplilearn. All rights reserved.

Playing [H/W]
00:27:49 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4
File View Play Navigate Favourites Help

Demo on Git

Let's implement some of the commands on Git using a Git Shell

18. *Create a branch "first_branch" and merge it to the main (master) branch*

```
SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ git branch first_branch
```

------> creating a branch

```
SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ git checkout first_branch
Switched to branch 'first_branch'
M       info.txt
```

------> switching to the new branch from master branch

```
SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (first_branch)
$ git add info3.txt
```

------> create and add "info3.txt" to first_branch

©Simplilearn. All rights reserved.

simplilearn
simplilearn

Playing [H/W]
00:28:53 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4
File View Play Navigate Favourites Help

Demo on Git

Let's implement some of the commands on Git using a Git Shell

18. *Create a branch "first_branch" and merge it to the main (master) branch*

```
SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (first_branch)
$ git commit -m "make some changes to first_branch"
[first_branch b14d609] make some changes to first_branch
 Committer: Simplilearn <Simplilearn@SSPL-LP-DNS-YT01.blrsimplilearn.com>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly:

    git config --global user.name "Your Name"
    git config --global user.email you@example.com

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

 1 file changed, 1 insertion(+)
 create mode 100644 info3.txt
```

------> make a commit to the first_branch

```
SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (first_branch)
$ ls
info.txt  info1.txt  info2.txt  info3.txt
```

------> new branch has access to all the files

©Simplilearn. All rights reserved.

simplilearn
simplilearn

Paused [H/W] 00:30:27 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

```
 Name - Sam
-Registration no.: 12479
\ No newline at end of file
+Registration no.: 12479
+Address - #303, East Main Street, Stamford
\ No newline at end of file

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ git config --global user.username simplilearn-github

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ git remote add origin https://github.com/simplilearn-github/test.git

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ git push origin master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 252 bytes | 252.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
remote:
remote: Create a pull request for 'master' on GitHub by visiting:
remote:      https://github.com/simplilearn-github/test/pull/new/master
remote:
To https://github.com/simplilearn-github/test.git
 * [new branch]      master -> master

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ git branch first_branch

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ git checkout first_branch
Switched to branch 'first_branch'
M       info.txt

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (first_branch)
$ git add info3.txt

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (first_branch)
$ git commit -m "made changes to first branch"
[first_branch 3c408f8] made changes to first branch
 1 file changed, 1 insertion(+)
 create mode 100644 info3.txt

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (first_branch)
$ ls
info.txt  info1.txt  info2.txt  info3.txt

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (first_branch)
$
```

©Simplilearn. All rights reserved.

Playing [H/W] 00:31:18 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

```
remote:      https://github.com/simplilearn-github/test/pull/new/master
remote:
To https://github.com/simplilearn-github/test.git
 * [new branch]      master -> master

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ git branch first_branch

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ git checkout first_branch
Switched to branch 'first_branch'
M       info.txt

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (first_branch)
$ git add info3.txt

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (first_branch)
$ git commit -m "made changes to first branch"
[first_branch 3c408f8] made changes to first branch
 1 file changed, 1 insertion(+)
 create mode 100644 info3.txt

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (first_branch)
$ ls
info.txt  info1.txt  info2.txt  info3.txt

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (first_branch)
$ git checkout master
Switched to branch 'master'
M       info.txt

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ ls
info.txt  info1.txt  info2.txt

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ git merge first_branch
Updating 795666f..3c408f8
Fast-forward
 info3.txt | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 info3.txt

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ ls
info.txt  info1.txt  info2.txt  info3.txt

SSPL-LP-DNS-YT0+Simplilearn@SSPL-LP-DNS-YT01 MINGW64 ~/test (master)
$ |
```

©Simplilearn. All rights reserved.

Playing [H/W] 00:32:08 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

Problem 1: How to maintain workflow visibility and avoid redundancies?

GitHub is not only affordable but also it is a great open source community. It is a cloud based tool so the code becomes more visible across the organization, and anybody could contribute to it now

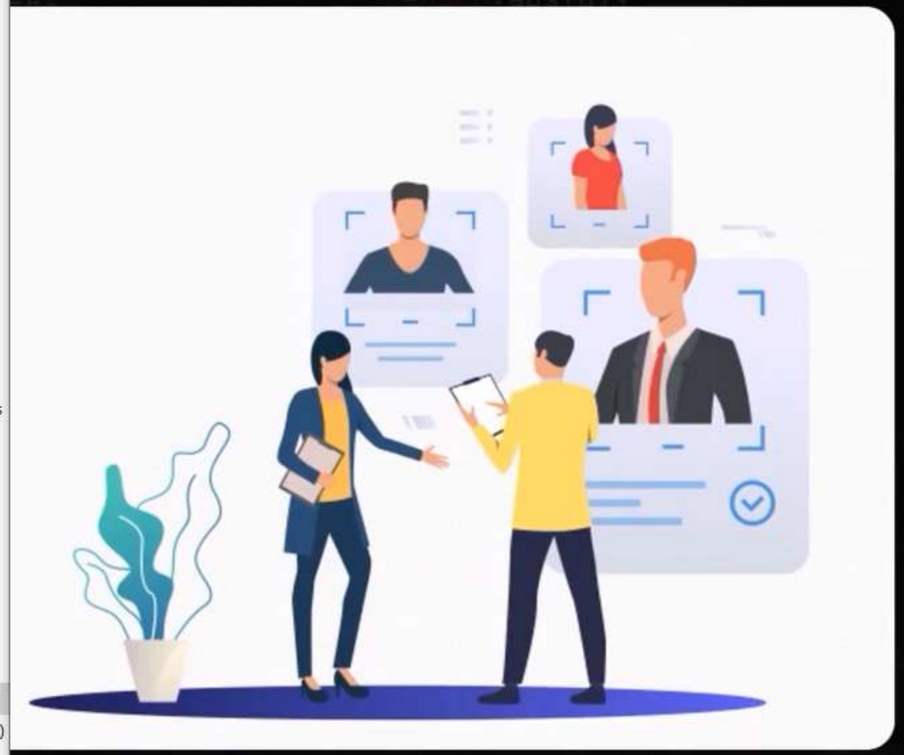
simplilearn

Playing [H/W] 00:33:49 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

Problem 2: How would you hire developers?



GitHub allows collaboration with any developer all over the world. Open source solutions enable potential developers to contribute and share the knowledge to benefit the **Global community**

simplilearn

Playing [H/W] 00:34:08 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

What is GitHub?

GitHub

- GitHub is a Git repository hosting service, which provides a web-based graphical interface
- GitHub helps every team member to work together on the project from anywhere and makes it easy for them to collaborate

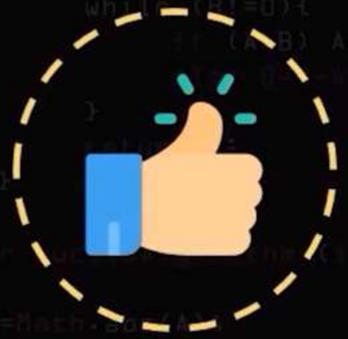
simplilearn

Playing [H/W]

00:34:48 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

Features of GitHub



Easy project management

GitHub is one place where project managers and developers coordinate, track, and update their work so projects stay transparent and on schedule.

simplilearn

Playing [H/W] 00:35:15 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

Features of GitHub

Increased safety with packages

The packages can be published privately, or with in the team or publicly for the open source community. The packages can be used or reused by downloading it from the GitHub

simplilearn

Playing [H/W]

00:35:40 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

Features of GitHub

Effective team management

GitHub helps all the team members to stay on the same page and stay organized. Moderation tools like issue and pull request locking helps the team to focus on the code

simplilearn

Playing [H/W]

00:35:58 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

Features of GitHub

Pull requests help the teams to review, improve and propose new codes on GitHub. The implementations and proposals can be discussed before changing the source code

Improved code writing

simplilearn

Playing [H/W] 00:36:23 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

Features of GitHub

All the codes and documentations are present in one place. There are millions of repositories on GitHub and each repository has their own tools to help you host and release code

Easy code hosting

Playing [H/W] 00:36:45 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

Competitors of GitHub

Bitbucket

FogBugz

Surround SCM

GitLab

Buddy

Beanstalk

simplilearn

Playing [H/W]

00:37:19 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

Git vs GitHub

Installed locally on the system

Hosted on the cloud

Git can be used offline and does not need an internet connection for use.

GitHub cannot be used offline and needs an internet connection.

Playing [H/W]

00:37:41 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

Git vs GitHub

Git can be used without GitHub

GitHub cannot be used without Git

Used for version control and code sharing

Used for centralised source code hosting

Playing [H/W]

00:38:05 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

Git vs GitHub

There is no GUI

It provides an easy to use GUI

Code changes like commit, merge, etc. are done using commands from the command line.

Everything is done through a web-based interface.

Playing [H/W]

00:38:27 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

Git vs GitHub

Open source licensed

Includes a free and pay for use tiers

Competitors – Mercurial, Supervision, IBM, Rational Team Concert and ClearCase

Competitors – Atlassian Bitbucket and GitLab

Playing [H/W]

00:38:58 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

What's in it for you?

- Install Git Bash/Client
- Configure Git Bash/Client for first time use
- Set up new repositories locally
- Push local repositories to GitHub (Free account Registration)
- Git Branching and Merging

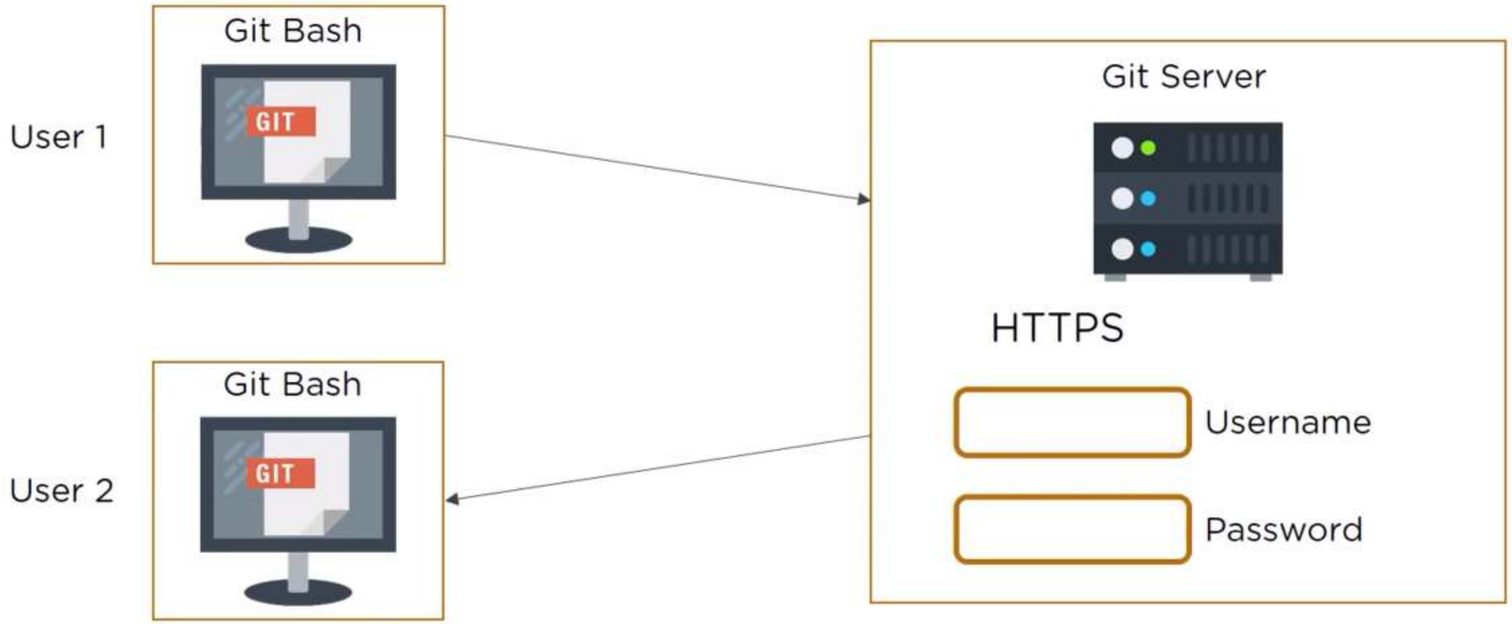
simplilearn

©Simplilearn. All rights reserved.

Playing [H/W] 00:39:11 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4
File View Play Navigate Favourites Help

Git Client Server

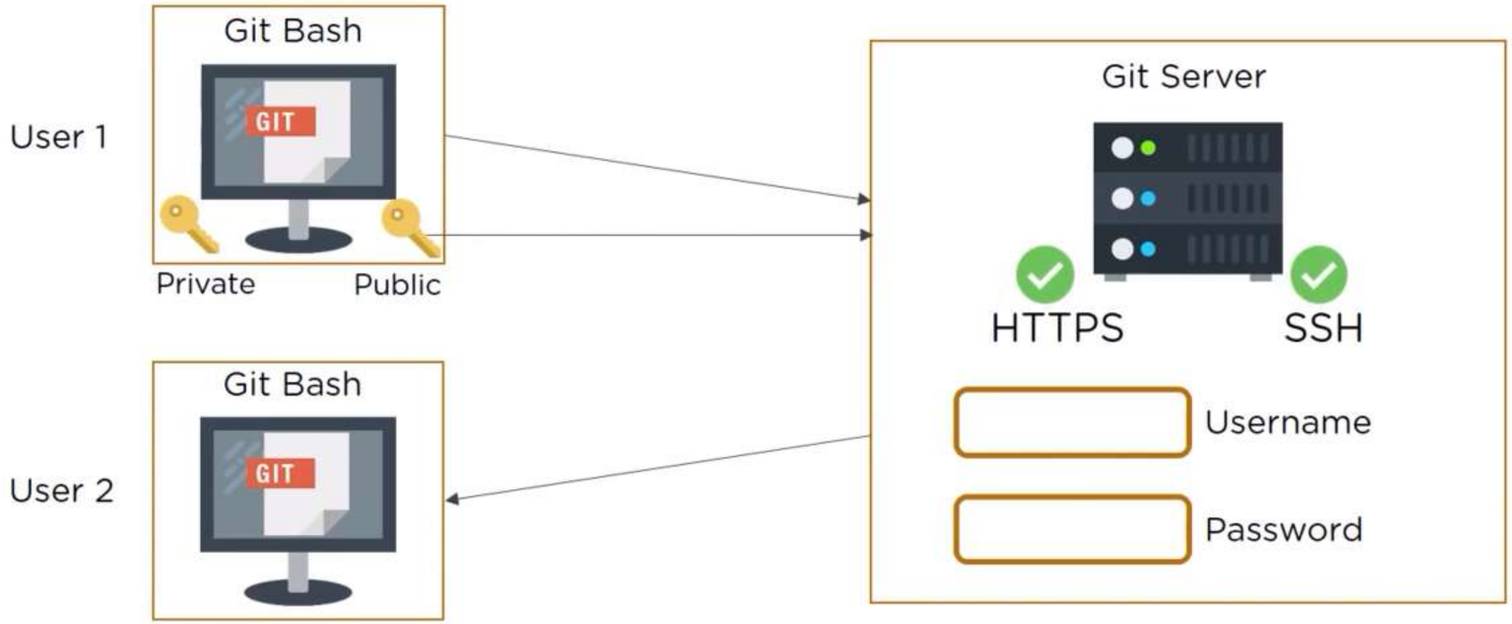


©Simplilearn. All rights reserved.

Playing [H/W]
00:43:59 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4
File View Play Navigate Favourites Help

Git Client Server



©Simplilearn. All rights reserved.

Playing [H/W]
00:44:41 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

MINGW64:/c/Users/anandr72

```
anandr72@DESKTOP-AFE0KT8 MINGW64 ~
$ pwd
/c/Users/anandr72

anandr72@DESKTOP-AFE0KT8 MINGW64 ~
$ mkdir Git_Demo
mkdir: cannot create directory 'Git_Demo': File exists

anandr72@DESKTOP-AFE0KT8 MINGW64 ~
$ rm -Rf Git_Demo

anandr72@DESKTOP-AFE0KT8 MINGW64 ~
$ mkdir Git_Demo

anandr72@DESKTOP-AFE0KT8 MINGW64 ~
$ cd Git
```

©Simplilearn. All rights reserved.

simplilearn

Playing [H/W] 00:48:53 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

MINGW64:/c/Users/anandr72/Git_Demo/FirstRepo

```
anandr72@DESKTOP-AFE0KT8 MINGW64 ~/Git_Demo
$ pwd
/c/Users/anandr72/Git_Demo

anandr72@DESKTOP-AFE0KT8 MINGW64 ~/Git_Demo
$ mkdir FirstRepo

anandr72@DESKTOP-AFE0KT8 MINGW64 ~/Git_Demo
$ cd FirstRepo/

anandr72@DESKTOP-AFE0KT8 MINGW64 ~/Git_Demo/F
$ pwd
/c/Users/anandr72/Git_Demo/FirstRepo

anandr72@DESKTOP-AFE0KT8 MINGW64 ~/Git_Demo/F
$
```

FirstRepo

File Home Share View

Pin to Quick access | Copy | Paste | Clipboard | Organize | New folder | New | Properties | Open | Select

« Git_De › FirstRepo

Search FirstRepo

Video Recordings
OneDrive
This PC
3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos
Windows (C:)

Name

This folder is empty.

0 items

©Simplilearn. All rights reserved.

simplilearn

Playing [H/W] 00:49:33 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

MINGW64:/c/Users/anandr72/Git_Demo/FirstRepo

```
anandr72@DESKTOP-AFE0KT8
$ touch master.txt

anandr72@DESKTOP-AFE0KT8
$ notepad master.txt
```

master.txt - Notepad

File Edit Format View Help

This is my fi

©Simplilearn. All rights reserved.

simplilearn

Playing [H/W] 00:52:28 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

MINGW64:/c/Users/anandr72/Git_Demo/FirstRepo

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

        master.txt

nothing added to commit but untracked files present (use "git add" to track)

anandr72@DESKTOP-AFEOKT8 MINGW64 ~/Git_Demo/FirstRepo (master)
$ git add .

anandr72@DESKTOP-AFEOKT8 MINGW64 ~/Git_Demo/FirstRepo (master)
$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)

        new file:   master.txt


anandr72@DESKTOP-AFEOKT8 MINGW64 ~/Git_Demo/FirstRepo (master)
$
```

©Simplilearn. All rights reserved.

simplilearn

Playing [H/W] 00:53:28 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

MINGW64:/c/Users/anandr72/Git_Demo/FirstRepo

```
anandr72@DESKTOP-AFEOKT8 MINGW64 ~/Git_Demo/FirstRepo (master)
$ git commit -m "This is my First Git Commit"
[master (root-commit) 131bdad] This is my First Git Commit
 1 file changed, 1 insertion(+)
 create mode 100644 master.txt

anandr72@DESKTOP-AFEOKT8 MINGW64 ~/Git_Demo/FirstRepo (master)
$ git status
On branch master
nothing to commit, working tree clean

anandr72@DESKTOP-AFEOKT8 MINGW64 ~/Git_Demo/FirstRepo (master)
$ git log
commit 131bdad2cbcddb446b27d968be67c15b4440e199 (HEAD -> master)
Author: SimpliLearn GitHub <sandeep.d@simplilearn.net>
Date:   Sun Sep 16 22:33:01 2018 +0530

    This is my First Git Commit

anandr72@DESKTOP-AFEOKT8 MINGW64 ~/Git_Demo/FirstRepo (master)
$
```

©Simplilearn. All rights reserved.

simplilearn

Playing [H/W] 00:54:43 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

MINGW64:/c/Users/anandr72/Git_Demo/FirstRepo

```
anandr72@DESKTOP-AFEOKT8 MINGW64 ~/Git_Demo/FirstRepo (master)
$

anandr72@DESKTOP-AFEOKT8 MINGW64 ~/Git_Demo/FirstRepo (master)
$ git log
commit 7259f7cbaf90498898188fc58224e75806af806a (HEAD -> master)
Author: SimpliLearn GitHub <sandeep.d@simplilearn.net>
Date:   Sun Sep 16 22:35:09 2018 +0530

    This is my Second Commit

commit 131bdad2cbcddb446b27d968be67c15b4440e199
Author: SimpliLearn GitHub <sandeep.d@simplilearn.net>
Date:   Sun Sep 16 22:33:01 2018 +0530

    This is my First Git Commit

anandr72@DESKTOP-AFEOKT8 MINGW64 ~/Git_Demo/FirstRepo (master)
$
```

©Simplilearn. All rights reserved.

simplilearn

Playing [H/W] 00:56:12 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

MINGW64:/c/Users/anandr72/Git_Demo/FirstRepo

```
anandr72@DESKTOP-AFEOKT8 MINGW64 ~/Git_Demo/FirstRepo (master)
$ git remote add origin https://github.com/simplilearn-github/FirstRepo.git

anandr72@DESKTOP-AFEOKT8 MINGW64 ~/Git_Demo/FirstRepo (master)
$ git remote -v
origin  https://github.com/simplilearn-github/FirstRepo.git (fetch)
origin  https://github.com/simplilearn-github/FirstRepo.git (push)

anandr72@DESKTOP-AFEOKT8 MINGW64 ~/Git_Demo/FirstRepo (master)
$ git push -u origin master
```

©Simplilearn. All rights reserved.

simplilearn

Playing [H/W] 01:00:32 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

MINGW64:/c/Users/anandr72/Git_Demo

```
anandr72@DESKTOP-AFE0KT8 MINGW64 ~/Git_Demo
$

anandr72@DESKTOP-AFE0KT8 MINGW64 ~/Git_Demo
$ ssh-keygen -t rsa -C sandeep.d@simplilearn.net
Generating public/private rsa key pair.
Enter file in which to save the key (/c/Users/anandr72/.ssh/id_rsa):
```

©Simplilearn. All rights reserved.

simplilearn

Playing [H/W] 01:03:20 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

MINGW64:/c/Users/anandr72/Git_Demo

```
anandr72@DESKTOP-AFEOKT8 MINGW64 ~/Git_Demo
$ ssh -T git@github.com
The authenticity of host 'github.com (192.30.253.113)' can't be established.
RSA key fingerprint is SHA256:nThbg6kXUpJWGl7E1IGOCspRomTxdCARLviKw6E5SY8.
Are you sure you want to continue connecting (yes/no)? ye
```

©Simplilearn. All rights reserved.

simplilearn

Playing [H/W] 01:05:19 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

MINGW64:/c/Users/anandr72/Git_Demo

```
anandr72@DESKTOP-AFEOKT8 MINGW64 ~/Git_Demo
$ git clone git@github.com:simplilearn-github/SecondRepo.git
Cloning into 'SecondRepo'...
Warning: Permanently added the RSA host key for IP address '192.30.253.112' to the l
ist of known hosts.
Enter passphrase for key '/c/Users/anandr72/.ssh/id_rsa':
remote: Counting objects: 3, done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.

anandr72@DESKTOP-AFEOKT8 MINGW64 ~/Git_Demo
$ ls -ltr
total 0
drwxr-xr-x 1 anandr72 197609 0 Sep 16 22:33 FirstRepo/
drwxr-xr-x 1 anandr72 197609 0 Sep 16 23:03 SecondRepo/

anandr72@DESKTOP-AFEOKT8 MINGW64 ~/Git_Demo
$
```

©Simplilearn. All rights reserved.

simplilearn

Playing [H/W] 01:08:03 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

MINGW64:/c/Users/anandr72/Git_Demo/SecondRepo

```
anandr72@DESKTOP-AFE0KT8 MINGW64 ~/Git_Demo/SecondRepo (master)
$ git add .

anandr72@DESKTOP-AFE0KT8 MINGW64 ~/Git_Demo/SecondRepo (master)
$ git commit -m "This is my First commit for the Second Repo"
[master 9c965b5] This is my First commit for the Second Repo
 1 file changed, 1 insertion(+)
 create mode 100644 second.txt

anandr72@DESKTOP-AFE0KT8 MINGW64 ~/Git_Demo/SecondRepo (master)
$ git remote -v
origin  git@github.com:simplilearn-github/SecondRepo.git (fetch)
origin  git@github.com:simplilearn-github/SecondRepo.git (push)

anandr72@DESKTOP-AFE0KT8 MINGW64 ~/Git_Demo/SecondRepo (master)
$ git push
```

©Simplilearn. All rights reserved.

simplilearn

Playing [H/W] 01:09:44 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

MINGW64:/c/Users/anandr72/Git_Demo/SecondRepo

```
anandr72@DESKTOP-AFE0KT8 MINGW64 ~/Git_Demo/SecondRepo (master)
$ git add .

anandr72@DESKTOP-AFE0KT8 MINGW64 ~/Git_Demo/SecondRepo (master)
$ git commit -m "This is my First commit for the Second Repo"
[master 9c965b5] This is my First commit for the Second Repo
 1 file changed, 1 insertion(+)
 create mode 100644 second.txt

anandr72@DESKTOP-AFE0KT8 MINGW64 ~/Git_Demo/SecondRepo (master)
$ git remote -v
origin  git@github.com:simplilearn-github/SecondRepo.git (fetch)
origin  git@github.com:simplilearn-github/SecondRepo.git (push)

anandr72@DESKTOP-AFE0KT8 MINGW64 ~/Git_Demo/SecondRepo (master)
$ git push origin master
```

©Simplilearn. All rights reserved.

simplilearn

Playing [H/W] 01:09:48 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

Git Commands

git init

git add .

git commit –m “Message”

git status

git log

git add remote origin Server_Repo_URL

git push –u origin master

git clone Server_Repo_URL

simplilearn

©Simplilearn. All rights reserved.

Playing [H/W] 01:12:28 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

MINGW64:/c/Users/anandr72/Git_Demo/UndoRepo

```
anandr72@DESKTOP-AFEOKT8 MINGW64 ~/Git_Demo/UndoRepo (master)
$ git log --pretty=oneline
c4145890f9464038f8854e4faa181db5d5db9250 (HEAD -> master) C5: Added m4.txt
91cffc34adece98510aa867c10c3bcf168ebde02 C4: changes to m1.txt and m2.txt
a91e12303d459f395a679620a8c3a12e9becc1b6 C3: Adding m3.txt
0272db1cd068a981af5cb19df53afb78fec9208d C2: Adding m2.txt
a07aa63a46a19ed130cfec16bd707b4c8d86b599 C1: Adding m1.txt

anandr72@DESKTOP-AFEOKT8 MINGW64 ~/Git_Demo/UndoRepo (master)
$
```

©Simplilearn. All rights reserved.

simplilearn

Playing [H/W] 01:14:00 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

MINGW64:/c/Users/anandr72/Git_Demo/UndoRepo

```
anandr72@DESKTOP-AFEOKT8 MINGW64 ~/Git_Demo/UndoRepo (master)
$ git log --pretty=oneline
c4145890f9464038f8854e4faa181db5d5db9250 (HEAD -> master) C5: Added m4.txt
91cffc34adece98510aa867c10c3bcf168ebde02 C4: changes to m1.txt and m2.txt
a91e12303d459f395a679620a8c3a12e9becc1b6 C3: Adding m3.txt
0272db1cd068a981af5cb19df53afb78fec9208d C2: Adding m2.txt
a07aa63a46a19ed130cfec16bd707b4c8d86b599 C1: Adding m1.txt

anandr72@DESKTOP-AFEOKT8 MINGW64 ~/Git_Demo/UndoRepo (master)
$ git reset --soft a91e12303d459f395a679620a8c3a12e9becc1b6

anandr72@DESKTOP-AFEOKT8 MINGW64 ~/Git_Demo/UndoRepo (master)
$
```

©Simplilearn. All rights reserved.

simplilearn

Playing [H/W] 01:15:32 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

MINGW64:/c/Users/anandr72/Git_Demo/UndoRepo

```
a07aa63a46a19ed130cfec16bd707b4c8d86b599 C1: Adding m1.txt

anandr72@DESKTOP-AFEOKT8 MINGW64 ~/Git_Demo/UndoRepo (master)
$ git reset --soft a91e12303d459f395a679620a8c3a12e9becc1b6

anandr72@DESKTOP-AFEOKT8 MINGW64 ~/Git_Demo/UndoRepo (master)
$ git log --pretty=oneline
a91e12303d459f395a679620a8c3a12e9becc1b6 (HEAD -> master) C3: Adding m3.txt
0272db1cd068a981af5cb19df53afb78fec9208d C2: Adding m2.txt
a07aa63a46a19ed130cfec16bd707b4c8d86b599 C1: Adding m1.txt

anandr72@DESKTOP-AFEOKT8 MINGW64 ~/Git_Demo/UndoRepo (master)
$ git status
On branch master
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

        modified:   m1.txt
        modified:   m2.txt
        new file:   m4.txt


anandr72@DESKTOP-AFEOKT8 MINGW64 ~/Git_Demo/UndoRepo (master)
```

©Simplilearn. All rights reserved.

simplilearn

Playing [H/W] 01:16:22 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

MINGW64:/c/Users/anandr72/Git_Demo/UndoRepo

```
anandr72@DESKTOP-AFEOKT8 MINGW64 ~/Git_Demo/UndoRepo (master)
$ git log --pretty=oneline
c4145890f9464038f8854e4faa181db5d5db9250 (HEAD -> master) C5: Added m4.txt
91cffc34adece98510aa867c10c3bcf168ebde02 C4: changes to m1.txt and m2.txt
a91e12303d459f395a679620a8c3a12e9becc1b6 C3: Adding m3.txt
0272db1cd068a981af5cb19df53afb78fec9208d C2: Adding m2.txt
a07aa63a46a19ed130cfec16bd707b4c8d86b599 C1: Adding m1.txt

anandr72@DESKTOP-AFEOKT8 MINGW64 ~/Git_Demo/UndoRepo (master)
$ ls
m1.txt  m2.txt  m3.txt  m4.txt

anandr72@DESKTOP-AFEOKT8 MINGW64 ~/Git_Demo/UndoRepo (master)
$ git revert a91e12303d459f395a679620a8c3a12e9becc1b6
```

©Simplilearn. All rights reserved.

simplilearn

Playing [H/W] 01:21:57 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

MINGW64:/c/Users/anandr72/Git_Demo/Org_Details

```
[master (root-commit) 951f147] First Commit
 2 files changed, 2 insertions(+)
 create mode 100644 employees.txt
 create mode 100644 name.txt

anandr72@DESKTOP-AFE0KT8 MINGW64 ~/Git_Demo/Org_Details
(master)
$ git log --pretty=oneline
951f1472cc893f982bb317f27db5010d2b157dda (HEAD -> master
) First Commit

anandr72@DESKTOP-AFE0KT8 MINGW64 ~/Git_Demo/Org_Details
(master)
$ git branch Blr_Branch

anandr72@DESKTOP-AFE0KT8 MINGW64 ~/Git_Demo/Org_Details
(master)
$ git branch -v
  Blr_Branch 951f147 First Commit
* master     951f147 First Commit

anandr72@DESKTOP-AFE0KT8 MINGW64 ~/Git_Demo/Org_Details
(master)
$
```

Org_Details

File Home Share View

Pin to Quick access Copy Paste Clipboard New Open Select Organize

« Git_... › Org_Det...

Name

.git

employees.txt

name.txt

3 items

©Simplilearn. All rights reserved.

simplilearn

Playing [H/W] 01:26:42 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

MINGW64:/c/Users/anandr72/Git_Demo/Org_Details

```
$ git push -u origin Blr_Branch
Enter passphrase for key '/c/Users/anandr72/.ssh/id_rsa'
:
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 377 bytes | 377.00 KiB/s, d
one.
Total 3 (delta 0), reused 0 (delta 0)
remote:
remote: Create a pull request for 'Blr_Branch' on GitHub
 by visiting:
remote:      https://github.com/simplilearn-github/Org_D
etails/pull/new/Blr_Branch
remote:
To github.com:simplilearn-github/Org_Details.git
 * [new branch]      Blr_Branch -> Blr_Branch
Branch 'Blr_Branch' set up to track remote branch 'Blr_B
ranch' from 'origin'.

anandr72@DESKTOP-AFE0KT8 MINGW64 ~/Git_Demo/Org_Details
(master)
$
```

©Simplilearn. All rights reserved.

simplilearn

Star 0 Fork 0

Edit

1 contributor

Find file Clone or download

Latest commit 951f147 6 minutes ago

6 minutes ago

6 minutes ago

Add a README

Playing [H/W] 01:32:10 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

MINGW64:/c/Users/anandr72/Git_Demo/Org_Details

```
anandr72@DESKTOP-AFE0KT8 MINGW64 ~/Git_Demo/Org_Details
(master)
$ git checkout Blr_Branch
Switched to branch 'Blr_Branch'
Your branch is up to date with 'origin/Blr_Branch'.

anandr72@DESKTOP-AFE0KT8 MINGW64 ~/Git_Demo/Org_Details
(Blr_Branch)
$ git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.

anandr72@DESKTOP-AFE0KT8 MINGW64 ~/Git_Demo/Org_Details
(master)
$ git merge Blr_Branch
Updating 951f147..3afb319
Fast-forward
 address.txt | 4 ++++
 1 file changed, 4 insertions(+)
 create mode 100644 address.txt

anandr72@DESKTOP-AFE0KT8 MINGW64 ~/Git_Demo/Org_Details
(master)
$ |
```

Org_Details

Name: .git, employees.txt, name.txt

3 items 1 item selected 31 bytes

©Simplilearn. All rights reserved.

Playing [H/W] 01:33:39 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

MINGW64:/c/Users/anandr72/Git_Demo/Org_Details

```
anandr72@DESKTOP-AFE0KT8 MINGW64 ~/Git_Demo/Org_Details
(master)
$ git log --pretty=oneline
3afb319a938158dc204492d4fe8bdfcd730cf47f (HEAD -> master
, origin/Blr_Branch, Blr_Branch) Blr Branch commit
951f1472cc893f982bb317f27db5010d2b157dda (origin/master)
 First Commit

anandr72@DESKTOP-AFE0KT8 MINGW64 ~/Git_Demo/Org_Details
(master)
$ git push origin master
Enter passphrase for key '/c/Users/anandr72/.ssh/id_rsa':
Total 0 (delta 0), reused 0 (delta 0)
To github.com:simplilearn-github/Org_Details.git
   951f147..3afb319  master -> master

anandr72@DESKTOP-AFE0KT8 MINGW64 ~/Git_Demo/Org_Details (master)
$ git branch -D Blr_Branch
Deleted branch Blr_Branch (was 3afb319).

anandr72@DESKTOP-AFE0KT8 MINGW64 ~/Git_Demo/Org_Details (master)
$
```

©Simplilearn. All rights reserved.

simplilearn

Playing [H/W] 01:35:19 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

MINGW64:/c/Users/anandr72/Git_Demo/Org_Details

```
anandr72@DESKTOP-AFE0KT8 MINGW64 ~/Git_Demo/Org_Details (master)
$ git branch -v
* master 3afb319 Blr Branch commit

anandr72@DESKTOP-AFE0KT8 MINGW64 ~/Git_Demo/Org_Details (master)
$
```

Playing [H/W] 01:35:30 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

MINGW64:/c/Users/anandr72/Git_Demo/Org_Details

```
anandr72@DESKTOP-AFE0KT8 MINGW64 ~/Git_Demo/Org_Details (master)
$ git branch -v
* master 3afb319 Blr Branch commit

anandr72@DESKTOP-AFE0KT8 MINGW64 ~/Git_Demo/Org_Details (master)
$ git push origin --delete Blr_Branch
```

©Simplilearn. All rights reserved.

simplilearn

Playing [H/W]

01:36:01 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

Git Branch Commands

```
git branch New_Branch
```

```
git checkout New_Branch
```

```
git push -u origin New_Branch
```

```
git branch -av
```

```
git merge New_Branch
```

```
git branch -D New_Branch
```

```
git push origin -delete New_Branch
```

simplilearn

©Simplilearn. All rights reserved.

Playing [H/W] 01:36:55 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

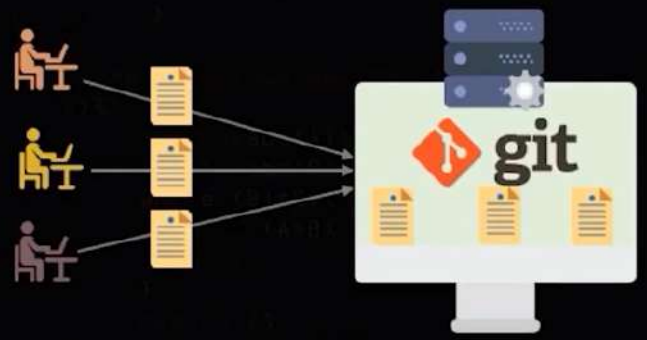
What is Git?

- Git is a version control system for tracking changes in computer files. It is used for coordinating work among several people on a project and tracking progress over time
- It is used for Source Code Management in software development

Playing [H/W] 01:40:52 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

What is Git?



- Git favors both programmers and non-technical users by keeping track of their project files
- It allows multiple users to work together
- Large projects can be handled efficiently

Playing [H/W] 01:43:54 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

Basic commands in Git

Git config	Configure the username and email address
Git init	Initialize a local Git repository
Git add	Add one or more files to staging area
Git diff	View the changes made to the file
Git commit	Commit changes to head but not to the remote repository

Playing [H/W] 01:48:40 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

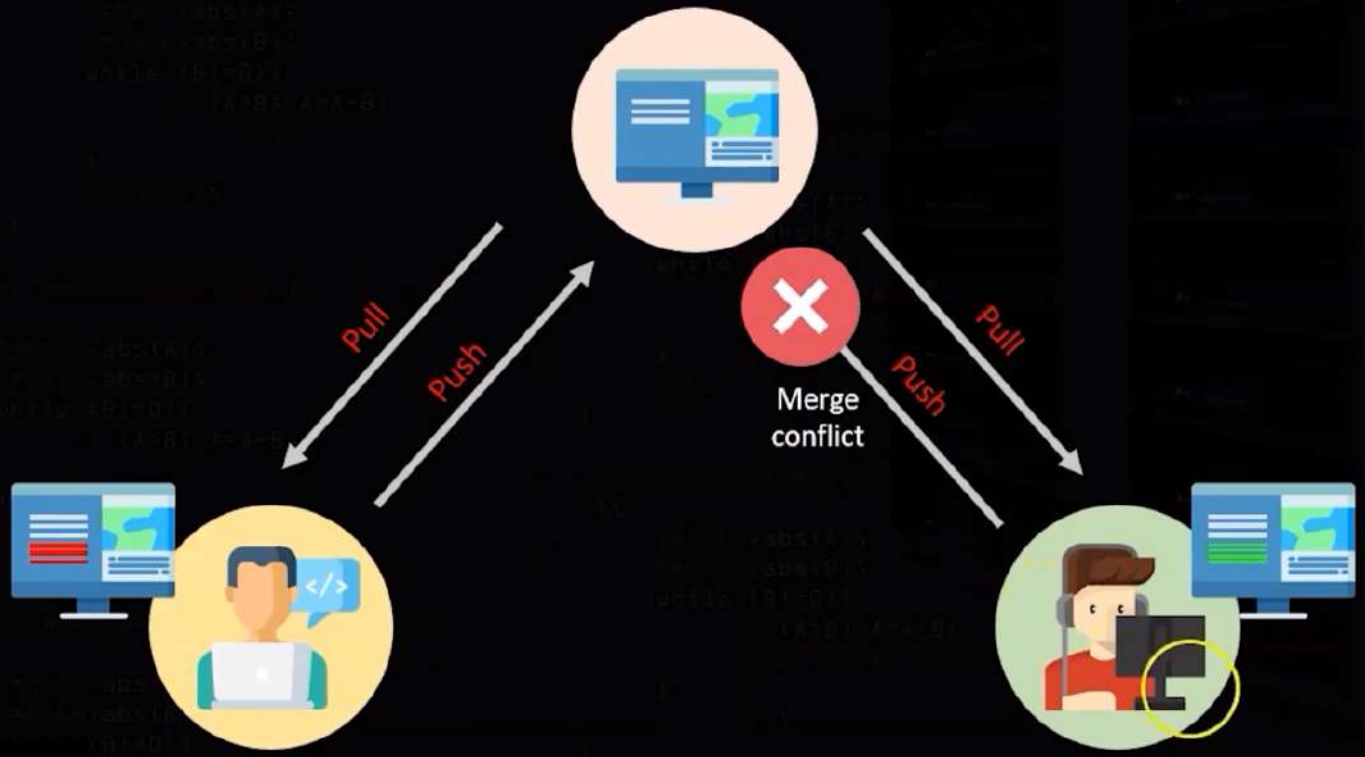
Basic commands in Git

Git reset	Undo local changes to the state of a Git repo
Git status	Displays the state of the working directory and staging area
Git merge	Merge a branch into an active branch
Git push	Upload content from local repository to a remote repository
Git pull	Fetch and download content from a remote repository

Playing [H/W] 01:52:50 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

What is a merge conflict?



Playing [H/W]

01:54:29 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

What is a merge conflict?

To prevent such conflicts developers work in separate isolated branches. The Git merge command combines separate branches and resolves any conflicting edits

Merge conflict

Playing [H/W] 01:55:05 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

What is a merge conflict?

- A merge conflict is an event that takes place when Git is unable to resolve differences in code between the two commits automatically
- Git can automatically merge the changes only if the commits are on different lines or branches

Playing [H/W] 01:55:49 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

What is a merge conflict?

- A merge conflict is an event that takes place when Git is unable to resolve differences in code between the two commits automatically
- Git can automatically merge the changes only if the commits are on different lines or branches

Playing [H/W] 01:55:50 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

Types of merge conflicts

There are two points when a merge can enter a conflicted state

Starting the merge process

- If there are changes in the working directory of the stage area of the current project, merge will fail to start
- In this case conflicts happen due to pending changes which need to be stabilized using different Git commands

During the merge process

- The failure during the merge process indicates that there is a conflict between the local branch and the branch being merged
- In this case Git resolves as much as possible but there are things that have to be resolved manually in the conflicted files

Playing [H/W] 01:58:15 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

How to resolve merge conflicts?

1. The most simple way to resolve the conflicted file is to open it and make the required changes to it
2. After editing the file we can use the **git add** command to stage the new merged content
3. The final step is to create a new commit with the help of the **git commit** command
4. Git will create a new merge commit to finalize the merge

Playing [H/W] 02:00:01 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

Git commands to resolve conflicts

git log --merge

git log --merge command helps in producing the list of commits that are causing the conflict

git diff

git diff command helps in finding the differences between the states of repositories or files

git checkout

git checkout command is used to undo the changes made to the file, or for changing branches

Playing [H/W] 02:01:15 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

Git commands to resolve conflicts

git reset --mixed

git reset –mixed command is used to undo the changes to the working directory and staging area

Playing [H/W]

02:01:46 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

Git commands to resolve conflicts

git reset --mixed

git reset –mixed command is used to undo the changes to the working directory and staging area

git merge --abort

git merge --abort command helps in exiting the merge process and returning back to the state before the merging began

git reset

git reset command is used at the time of merge conflict to reset the conflicted files to their original state

Playing [H/W] 02:02:20 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

MINGW64:/c/Users/anuj_/A

```
anuj_@DESKTOP-RTN18AB MINGW64 ~/A (master)
$ vi README.md

anuj_@DESKTOP-RTN18AB MINGW64 ~/A (master)
$ git status
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:   README.md

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

anuj_@DESKTOP-RTN18AB MINGW64 ~/A (master)
$ git a
```

Playing [H/W] 02:05:57 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

MINGW64:/c/Users/anuj_/B

```
anuj_@DESKTOP-RTN18AB MINGW64 ~/B (master)
$

anuj_@DESKTOP-RTN18AB MINGW64 ~/B (master)
$

anuj_@DESKTOP-RTN18AB MINGW64 ~/B (master)
$ git pull --rebase origin master
From https://github.com/anujdevopslearn/MavenBuild
 * branch            master     -> FETCH_HEAD
First, rewinding head to replay your work on top of it...
Applying: Commit from Repo B
Using index info to reconstruct a base tree...
M       README.md
Falling back to patching base and 3-way merge...
Auto-merging README.md
CONFLICT (content): Merge conflict in README.md
error: Failed to merge in the changes.
hint: Use 'git am --show-current-patch' to see the failed patch
Patch failed at 0001 Commit from Repo B
Resolve all conflicts manually, mark them as resolved with
"git add/rm <conflicted_files>", then run "git rebase --continue".
You can instead skip this commit: run "git rebase --skip".
To abort and get back to the state before "git rebase", run "git rebase --abort".

anuj_@DESKTOP-RTN18AB MINGW64 ~/B (master|REBASE 1/1)
$
```

Playing [H/W] 02:10:51 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

MINGW64:/c/Users/anuj_/B

```
anuj_@DESKTOP-RTN18AB MINGW64 ~/B (master)
$

anuj_@DESKTOP-RTN18AB MINGW64 ~/B (master)
$ git pull --rebase origin master
From https://github.com/anujdevopslearn/MavenBuild
 * branch            master     -> FETCH_HEAD
First, rewinding head to replay your work on top of it...
Applying: Commit from Repo B
Using index info to reconstruct a base tree...
M       README.md
Falling back to patching base and 3-way merge...
Auto-merging README.md
CONFLICT (content): Merge conflict in README.md
error: Failed to merge in the changes.
hint: Use 'git am --show-current-patch' to see the failed patch
Patch failed at 0001 Commit from Repo B
Resolve all conflicts manually, mark them as resolved with
"git add/rm <conflicted_files>", then run "git rebase --continue".
You can instead skip this commit: run "git rebase --skip".
To abort and get back to the state before "git rebase", run "git rebase --abort".

anuj_@DESKTOP-RTN18AB MINGW64 ~/B (master|REBASE 1/1)
$ git mergetool
Merging:
README.md

Normal merge conflict for 'README.md':
  {local}: modified file
  {remote}: modified file
4 files to edit

anuj_@DESKTOP-RTN18AB MINGW64 ~/B (master|REBASE 1/1)
$
```

Playing [H/W] 02:14:53 / 02:30:20

Quick access: Desktop, Downloads, Documents, Pictures, PUBG MODELS, Family photos, Github tutorial vi, Images, Navbar, Screenshots

OneDrive

This PC: 3D Objects, Desktop, Documents, Downloads, Music, Pictures, Videos, Adam textures, Blood, Captures, Face angels, fire, fortnight, fruits, Hip pop, thunders, Local Disk (C:), New Volume (E:), New Volume (F:)

Network

2 items 1 item selecte

Type here to search

28°C Mostly cloudy ENG 16:50 21-11-2024

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

MINGW64:/c/Users/anuj_/B

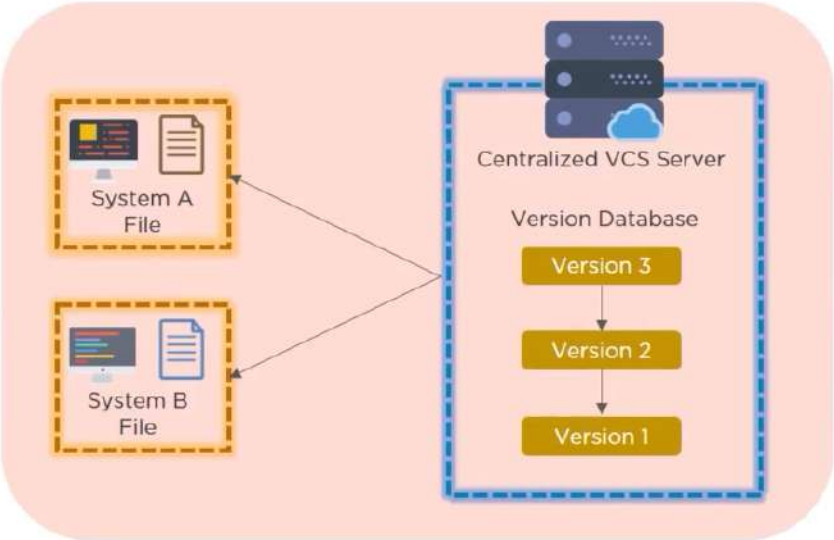
```
anuj_@DESKTOP-RTN18AB MINGW64 ~/B (master)
$

anuj_@DESKTOP-RTN18AB MINGW64 ~/B (master)
$ git log -n 2
commit c788bc59709cbe9330d15ec91669b2f60e24fa42 (HEAD -> master)
Author: Anuj Sharma <abc@abc.com>
Date:   Tue Apr 21 21:36:29 2020 +0530

    Commit from Repo B

commit b94e6ab3ed4835c253dd4b1b79f1e5a7cd7279e8 (origin/master)
Author: Anuj Sharma <abc@abc.com>
Date:   Tue Apr 21 21:26:41 2020 +0530

    Commit Done from Repo A

anuj_@DESKTOP-RTN18AB MINGW64 ~/B (master)
$
```

Playing [H/W] 02:15:41 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

11 Explain the difference between Centralized and Distributed Version Control System.



Centralized Version Control System

- Uses a central server to store all the versions of files
- No developer has a copy of all the files in the local system
- If the central server crashes, entire data of the project will be lost

simplilearn

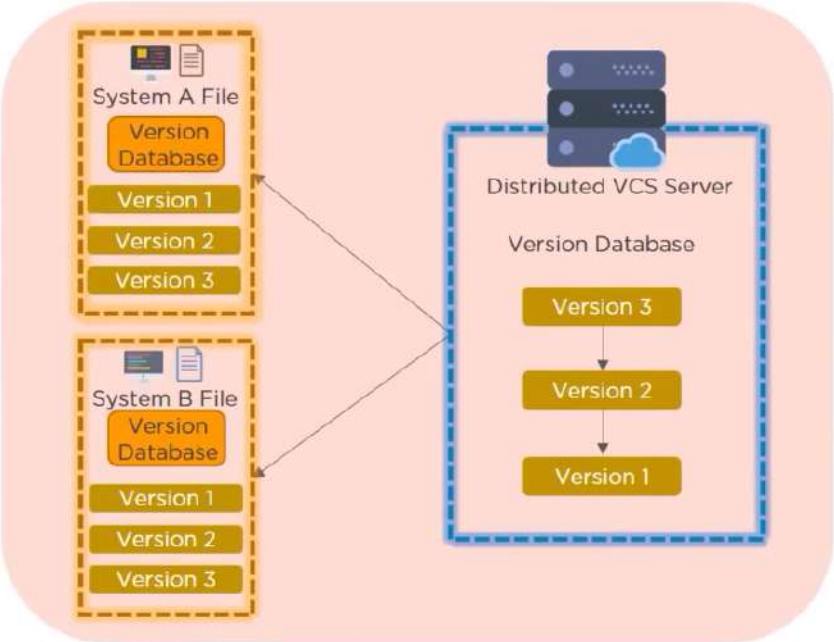
©Simplilearn. All rights reserved.

Playing [H/W] 02:18:10 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

11 Explain the difference between Centralized and Distributed Version Control System.



Distributed Version Control System

- Every developer has a copy of all the versions of the code on their system
- Improves the ability to work offline and does not rely on single location for backups
- There is no threat even if the server crashes

simplilearn

©Simplilearn. All rights reserved.

Playing [H/W] 02:18:40 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

13

How to push a file from your local system on to GitHub repository using Git?

First, connect the local repository to your remote repository:

git remote add origin [copied web address] → *git remote add origin https://github.com/Simplilearn-github/test.git*

Second, push your file to the remote repository:

git push origin master

simplilearn

©Simplilearn. All rights reserved.

Playing [H/W] 02:20:11 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

14 How is bare repository different from the standard way of initializing a Git repository?

Using the standard way

git init

- You create a working directory with git init
- A .git subfolder is created with all the git related revision history

Using the bare way

git init --bare

- Does not contain any working or checked out copy of source files
- Bare repositories store git revision history in the root folder of your repository instead of .git subfolder

simplilearn

©Simplilearn. All rights reserved.

Playing [H/W] 02:21:06 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

16 What is the process to revert a commit that has already been pushed and made public?

There are two processes through which you can revert a commit:

1. Remove or fix the bad file in a new commit and push it to the remote repository. Then commit it to the remote repository using:

   git commit -m "commit message"

2. Create a new commit that undoes all the changes that were made in the bad commit. Use the following command:

   git revert <commit id>

   Example: git revert 56de0938f

simplilearn

©Simplilearn. All rights reserved.

Playing [H/W] 02:21:53 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

15

Which of the following CLI command can be used to rename files?

(a) git rm

(b) git mv

(c) git rm -r

(d) None of the above

simplilearn

©Simplilearn. All rights reserved.

Playing [H/W] 02:21:50 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

17 Explain the difference between git fetch and git pull.

Git fetch

- Git fetch only downloads new data from a remote repository
- Does not integrate any of this new data into your working files
- Git fetch can be done any time to update the remote-tracking branches
- *Command - git fetch origin*
  git fetch --all

Git pull

- Git pull updates the current HEAD branch with the latest changes from the remote server
- It downloads new data and integrates it with the current working files
- It tries to merge remote changes with your local ones
- *Command - git pull origin master*

simplilearn

©Simplilearn. All rights reserved.

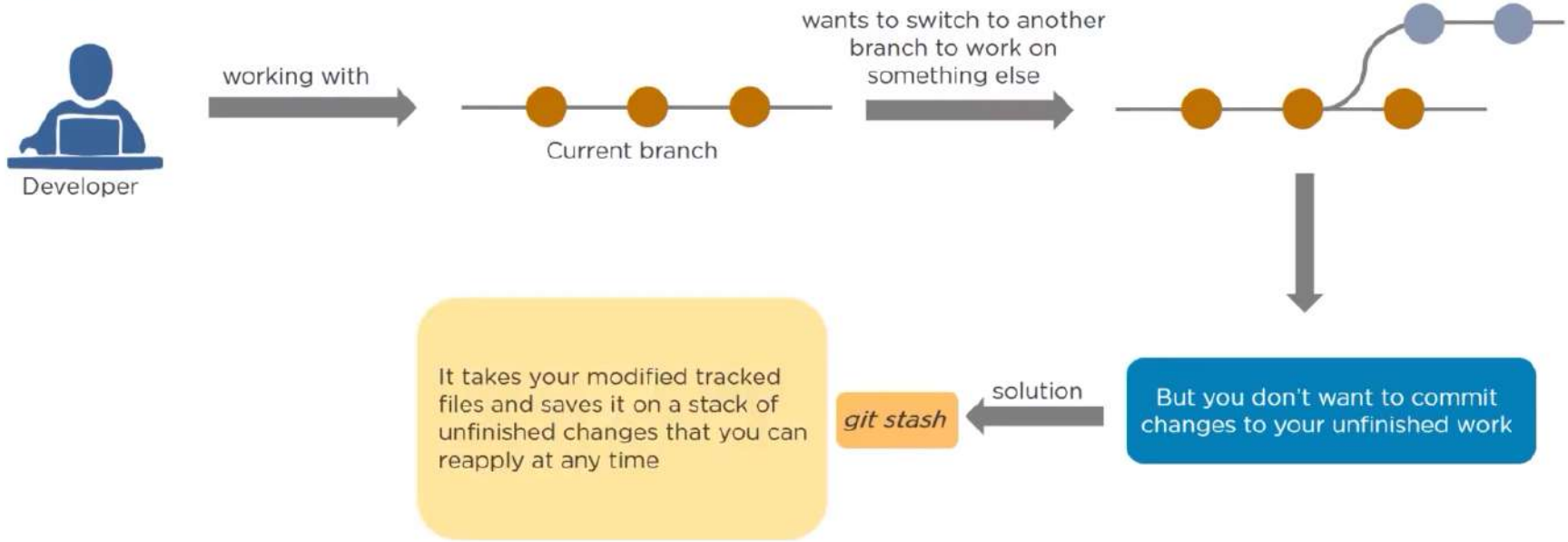
Playing [H/W] 02:22:53 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

18

What is Git stash?



simplilearn

©Simplilearn. All rights reserved.

Playing [H/W] 02:24:03 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

19 Explain the concept of branching in Git.

You can create a new branch and build the new feature on that branch

- By default, you always work on the master branch
- The circles on the branch represent various commits made on the branch
- After you are done with all the changes, you can merge it to the master branch

simplilearn

©Simplilearn. All rights reserved.

Playing [H/W] 02:24:14 / 02:30:20

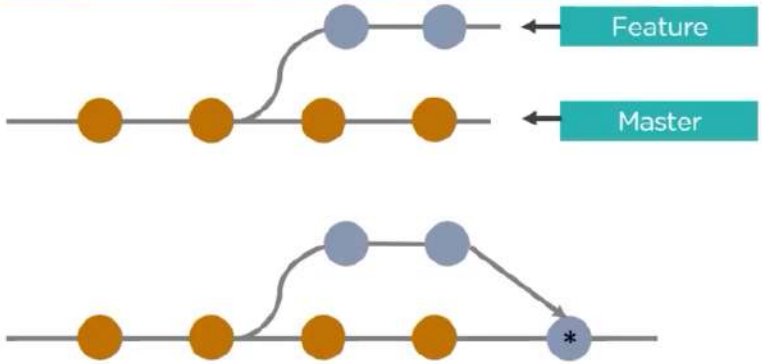
Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

20 What is the difference between git merge and git rebase.

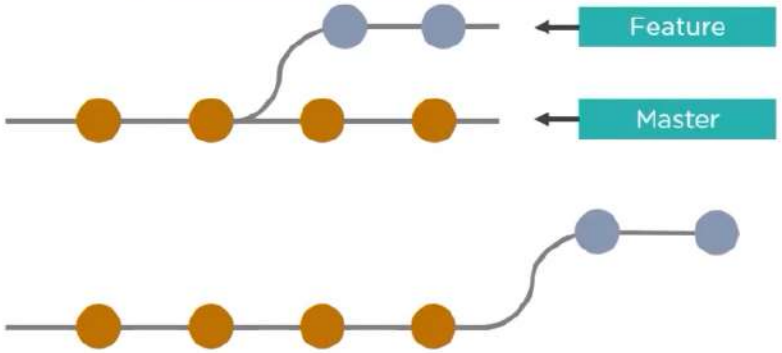
Suppose you are working on a new feature in a dedicated branch and another team member updates the master branch with new commits

To incorporate the new commits into your feature branch, you use *merge*



- Creates an extra merge commit every time you need to incorporate changes
- Pollutes your feature branch history

As an alternative to merging, you can rebase the feature branch onto master



- Incorporates all the new commits in master branch
- Re-writes the project history by creating brand new commits for each commit in the original branch

simplilearn

©Simplilearn. All rights reserved.

Playing [H/W] 02:25:12 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

21

How do you find a list of files that has been changed in a particular commit?

The command to get a list of files that has been changed in a particular commit is:

git diff-tree -r {commit hash}

Example: git diff-tree -r 87e673f21b

- *-r* flag allows the command to list individual files
- *commit hash* will list all the files that were changed or added in that commit

simplilearn

©Simplilearn. All rights reserved.

Playing [H/W] 02:25:46 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

22

What is a merge conflict in Git and how can it be resolved?

Git merge conflict — It arises when you have merge branches that have competing commits and Git needs your help to decide which changes to incorporate in the final merge

Manually edit the conflicted file to select the changes that you want to keep in the final merge

©Simplilearn. All rights reserved.

simplilearn

Playing [H/W] 02:26:10 / 02:30:20

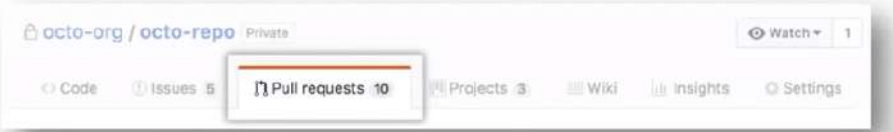
Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

22 What is a merge conflict in Git and how can it be resolved?

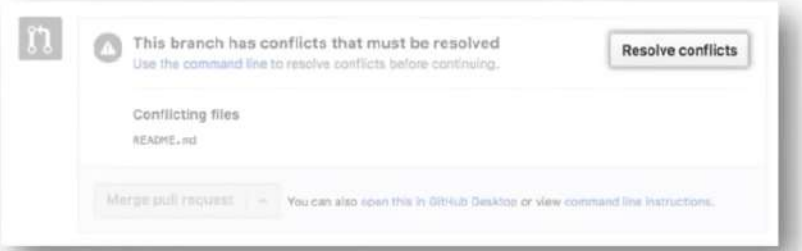
Resolving a merge conflict using conflict editor

1. Under your repository name, click Pull requests.



2. In "**Pull Requests**" list, click the pull request with a merge conflict that you'd like to resolve

3. Near the bottom of your pull request, click Resolve conflicts



4. Decide if you want to keep only your branch's changes, keep only the other branch's changes or make a brand new change, which may incorporate changes from both branches

simplilearn

©Simplilearn. All rights reserved.

Playing [H/W] 02:26:44 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

22 What is a merge conflict in Git and how can it be resolved?

5. Delete the conflict markers `<<<<<<<`, `=======`, `>>>>>>>` and make changes you want in the final merge

6. If you have more than one merge conflict in your file, scroll down to the next set of conflict markers and repeat steps four and five to resolve your merge conflict

7. Once you have resolved all the conflicts in the file, click **Mark as resolved**

8. If you have more than one file with a conflict, select the next file you want to edit on the left side of the page under "**conflicting files**" and repeat steps 4 to 7 until you've resolved all of your pull request's merge conflicts

simplilearn

©Simplilearn. All rights reserved.

Paused [H/W] 02:26:57 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

22 What is a merge conflict in Git and how can it be resolved?

5. Delete the conflict markers `<<<<<<<`, `=======`, `>>>>>>>` and make changes you want in the final merge

6. If you have more than one merge conflict in your file, scroll down to the next set of conflict markers and repeat steps four and five to resolve your merge conflict

7. Once you have resolved all the conflicts in the file, click **Mark as resolved**

8. If you have more than one file with a conflict, select the next file you want to edit on the left side of the page under "**conflicting files**" and repeat steps 4 to 7 until you've resolved all of your pull request's merge conflicts

simplilearn

©Simplilearn. All rights reserved.

Playing [H/W] 02:27:49 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

22 What is a merge conflict in Git and how can it be resolved?

9 Once you've resolved all your merge conflicts, click **Commit merge**. This merges the entire base branch into your head branch

10 To merge your pull request, click Merge pull request

simplilearn

©Simplilearn. All rights reserved.

Playing [H/W] 02:28:31 / 02:30:20

Quick access: Desktop, Downloads, Documents, Pictures, PUBG MODELS, Family photos, Github tutorial vi, Images, Navbar, Screenshots
OneDrive
This PC: 3D Objects, Desktop, Documents, Downloads, Music, Pictures, Videos, Adam textures, Blood, Captures, Face angels, fire, fortnight, fruits, Hip pop, thunders
Local Disk (C:), New Volume (E:), New Volume (F:)
Network

2 items 1 item selecte

Type here to search 28°C Mostly cloudy ENG 17:04 21-11-2024

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

22 What is a merge conflict in Git and how can it be resolved?

Resolving a merge conflict using the command line

1. Open Git Bash

2. Navigate into the local Git repository that has the merge conflict

```
cd REPOSITORY-NAME
```

3. Generate a list of the files affected by the merge conflict. In this example, the file **styleguide.md** has a merge conflict

```
$ git status
# On branch branch-b
# You have unmerged paths.
#   (fix conflicts and run "git commit")
#
# Unmerged paths:
#   (use "git add ..." to mark resolution)
#
# both modified:      styleguide.md
#
no changes added to commit (use "git add" and/or "git commit -a")
```

simplilearn

©Simplilearn. All rights reserved.

Playing [H/W] 02:28:53 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

22

What is a merge conflict in Git and how can it be resolved?

Resolving a merge conflict using the command line

4 Open any text editor, such as Sublime text or Atom, and navigate to the file that has merge conflicts

5 To see the beginning of the merge conflict in your file, search the file for the conflict marker `<<<<<<<`

open → You'll see the changes from the base branch after the line `<<<<<<< HEAD`

6 Next you'll see `=======`, which divides your changes from the changes in the other branch, followed by `>>>>>>> BRANCH-NAME`

```
If you have questions, please
<<<<<<< HEAD
open an issue
=======
ask your question in IRC.
>>>>>>> branch-a
```

simplilearn

©Simplilearn. All rights reserved.

Playing [H/W] 02:29:20 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

22

What is a merge conflict in Git and how can it be resolved?

Resolving a merge conflict using the command line

7 Decide if you want to keep only your branch's changes, keep only the other branch's changes or make a brand new change, which may incorporate changes from both branches

8 Delete the conflict markers `<<<<<<<`, `=======`, `>>>>>>>` and make changes you want in the final merge

In this example, both the changes are incorporated into the final merge:

```
If you have questions, please open an issue or ask in our IRC channel if it's more urgent.
```

simplilearn

©Simplilearn. All rights reserved.

Paused [H/W] 02:29:29 / 02:30:20

Git Tutorial For Beginners 2022 _ Git & Github Complete Tutorial _ Learn Git in 2 hrs _ Simplilearn.mp4

File View Play Navigate Favourites Help

22

What is a merge conflict in Git and how can it be resolved?

Resolving a merge conflict using the command line

9 Add or stage your changes

```
$ git add .
```

10 Commit your changes with a comment

```
$ git commit -m "Resolved merge conflict by incorporating both suggestions."
```

Now you can merge the branches on the command line or **push** your changes to your remote repository on GitHub and merge your changes in a **pull request**

simplilearn

©Simplilearn. All rights reserved.

Playing [H/W] 02:29:40 / 02:30:20