

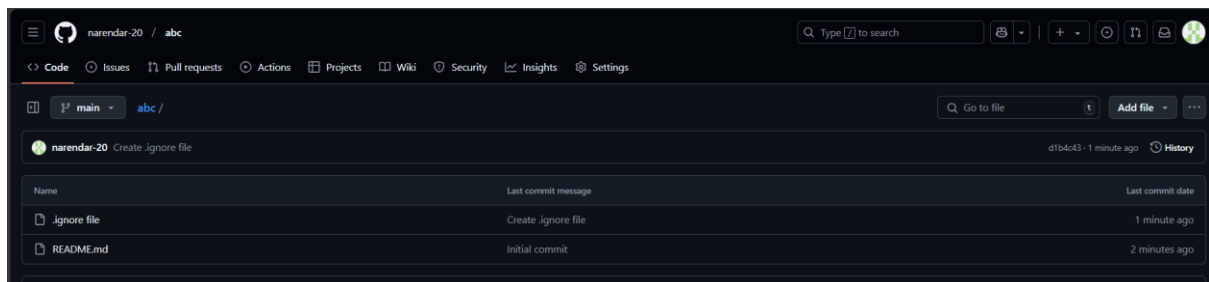
TASK ON GIT AND GITHUB

1)Install git:

```
naren@narendar MINGW64 /d/git_store (master)
$ git --version
git version 2.37.1.windows.1
```

2)Create a repo in github with README.

md and .ignore file:



3)Clone the created repo to local:

```
naren@narendar MINGW64 /d/git_store (naren)
$ git clone git@github.com:narendar-20/abc.git
Cloning into 'abc'...
remote: Enumerating objects: 6, done.
remote: Counting objects: 100% (6/6), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 6 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (6/6), done.

naren@narendar MINGW64 /d/git_store (naren)
$ ls
abc/ alpha f1 f2 hello hotfix sample xyz1 xyz2

naren@narendar MINGW64 /d/git_store (naren)
$ cd abc/

naren@narendar MINGW64 /d/git_store/abc (main)
$ ls
README.md
```

4)Create two files in local repo:

```
naren@narendar MINGW64 /d/git_store (master)
$ touch xyz1 xyz2

naren@narendar MINGW64 /d/git_store (master)
$ ls
alpha f1 f2 hello hotfix xyz1 xyz2

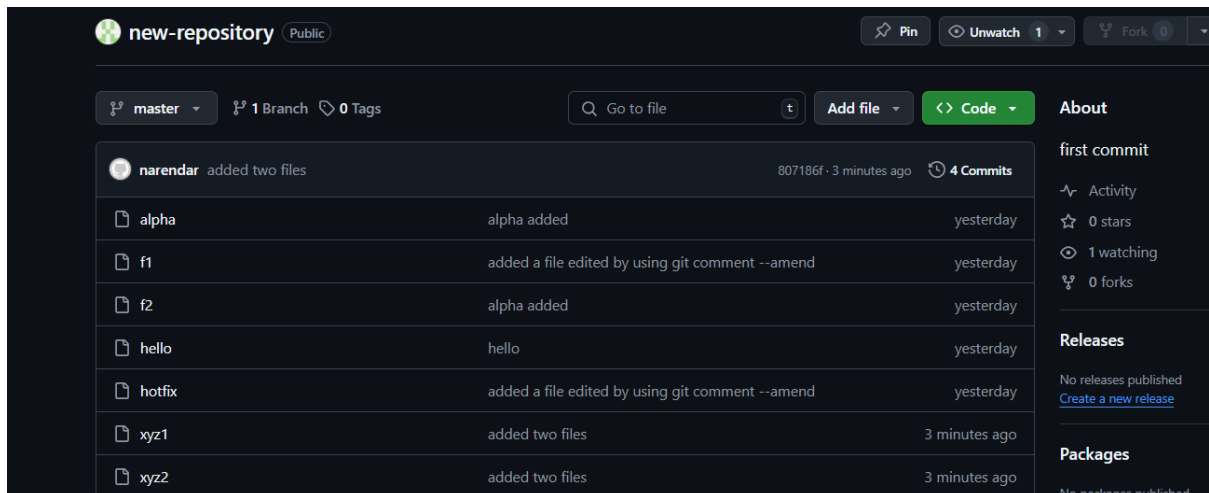
naren@narendar MINGW64 /d/git_store (master)
```

5)Commit two files and push to central Repository:

```
naren@narendar MINGW64 /d/git_store (master)
$ git add xyz1 xyz2

naren@narendar MINGW64 /d/git_store (master)
$ git commit -m "added two files"
[master 807186f] added two files
2 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 xyz1
create mode 100644 xyz2

naren@narendar MINGW64 /d/git_store (master)
$ git push -u origin master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 253 bytes | 253.00 KiB/s, done.
Total 2 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To github.com:narendar-20/new-repository.git
   1a932be..807186f  master -> master
branch 'master' set up to track 'origin/master'.
```



The screenshot shows the GitHub interface for a public repository named 'new-repository'. The repository has 1 branch (master) and 0 tags. The commit history shows 4 commits by user 'narendar'. The most recent commit, 807186f, is titled 'added two files' and was made 3 minutes ago. The commit details show two files added: 'alpha' and 'f1'. The file list on the right shows 'alpha', 'f1', 'f2', 'hello', 'hotfix', 'xyz1', and 'xyz2'. The 'About' section on the right shows the repository is a first commit with 0 stars, 1 watching, and 0 forks. The 'Releases' section shows no releases published, and the 'Packages' section shows no packages published.

File	Commit Message	Time
alpha	alpha added	yesterday
f1	added a file edited by using git comment --amend	yesterday
f2	alpha added	yesterday
hello	hello	yesterday
hotfix	added a file edited by using git comment --amend	yesterday
xyz1	added two files	3 minutes ago
xyz2	added two files	3 minutes ago

6) Create a branch in local and create a sample file and push to central:

```
naren@narendar MINGW64 /d/git_store (master)
$ git checkout naren
error: pathspec 'naren' did not match any file(s) known to git

naren@narendar MINGW64 /d/git_store (master)
$ git checkout -b naren
Switched to a new branch 'naren'

naren@narendar MINGW64 /d/git_store (naren)
$ touch sample

naren@narendar MINGW64 /d/git_store (naren)
$ git add sample

naren@narendar MINGW64 /d/git_store (naren)
$ git commit -m "added sample file"
[naren f0448b5] added sample file
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 sample

naren@narendar MINGW64 /d/git_store (naren)
$ git push -u origin naren
fatal: 'origin' does not appear to be a git repository
fatal: Could not read from remote repository.

Please make sure you have the correct access rights
and the repository exists.

naren@narendar MINGW64 /d/git_store (naren)
$ git push -u origin naren
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 229 bytes | 229.00 KiB/s, done.
Total 2 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
remote:
remote: Create a pull request for 'naren' on GitHub by visiting:
remote:   https://github.com/narendar-20/new-repository/pull/new/naren
remote:
To github.com:narendar-20/new-repository.git
 * [new branch]      naren -> naren
branch 'naren' set up to track 'origin/naren'.
```

The screenshot shows the GitHub web interface for a repository named 'narendar-20/new-repository'. At the top, a notification bar indicates 'naren had recent pushes 3 minutes ago' with a 'Compare & pull request' button. Below this, the repository name is shown with '2 Branches' and '0 Tags'. A search bar and 'Add file' button are visible. A status bar indicates 'This branch is 1 commit ahead of master'. The main content area shows a commit history table with 5 commits. The most recent commit, 'narendar: added sample file' (commit hash f0448b5, 6 minutes ago), is highlighted. The commit details show a list of files: 'alpha', 'f1', 'f2', 'hello', 'hotfix', and 'sample', each with a description of the change and the time it was made. On the right sidebar, the 'About' section shows 'first commit', 'Activity' (0 stars, 1 watching, 0 forks), 'Releases' (No releases published), and 'Packages' (No packages published).

Commit Hash	Commit Message	Time
f0448b5	narendar: added sample file	6 minutes ago
	alpha added	yesterday
	added a file edited by using git comment --amend	yesterday
	alpha added	yesterday
	hello	yesterday
	added a file edited by using git comment --amend	yesterday
	added sample file	6 minutes ago

7) Create a branch in github and clone that to local:

```
naren@narendar MINGW64 /d/git_store (naren)
$ cd abc/

naren@narendar MINGW64 /d/git_store/abc (main)
$ git fetch
From github.com:narendar-20/abc
* [new branch]      new-branch -> origin/new-branch

naren@narendar MINGW64 /d/git_store/abc (main)
$ cd ..

naren@narendar MINGW64 /d/git_store (naren)
$ git checkout -b new-branch
Switched to a new branch 'new-branch'

naren@narendar MINGW64 /d/git_store (new-branch)
$ ls
abc/  alpha  f1  f2  hello  hotfix  sample  xyz1  xyz2
```

8) Merge the created branch with master in git local:

```
naren@narendar MINGW64 /d/git_store (master)
$

naren@narendar MINGW64 /d/git_store (master)
$ git merge new-branch
Updating 807186f..f0448b5
Fast-forward
 sample | 0
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 sample

naren@narendar MINGW64 /d/git_store (master)
$ ls
abc/  alpha  f1  f2  hello  hotfix  sample  xyz1  xyz2

naren@narendar MINGW64 /d/git_store (master)
$
```

9) Merge the created branch with master in github by sending a pull request:

The screenshot displays a GitHub pull request interface. At the top, a navigation bar shows 'Conversation 0', 'Commits 1', 'Checks 0', and 'Files changed 1'. Below this, a comment by 'narendar-20' is visible, stating 'No description provided.' with a 'Owner' button and a three-dot menu. The pull request history shows a commit 'a3ef939' merged into 'master' by 'teja384' 1 minute ago, with a 'Revert' button. A green 'Verified' badge and the number '1067113' are also present. A purple box at the bottom left indicates 'Pull request successfully merged and closed', with a message: 'You're all set — the `branch1` branch can be safely deleted.' and a 'Delete branch' button. On the right side, there are sections for 'Reviews' (No reviews, Convert to draft), 'Assignees' (No one—assign yourself), 'Labels' (None yet), 'Projects' (None yet), and 'Milestone'.

10) create a file in local and send that to branch in git hub:

```
naren@narendar MINGW64 /d/git_store (master)
$ touch newfile

naren@narendar MINGW64 /d/git_store (master)
$ git checkout naren
Switched to branch 'naren'
Your branch is up to date with 'origin/naren'.

naren@narendar MINGW64 /d/git_store (naren)
$ ls
abc/ alpha f1 f2 hello hotfix newfile sample xyz1 xyz2

naren@narendar MINGW64 /d/git_store (naren)
$ git add newfile

naren@narendar MINGW64 /d/git_store (naren)
$ git commit -m
error: switch `m' requires a value
'

naren@narendar MINGW64 /d/git_store (naren)
$ git commit -m "add new file"
[naren 741c117] add new file
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 newfile

naren@narendar MINGW64 /d/git_store (naren)
$ git push -u origin naren
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 228 bytes | 228.00 KiB/s, done.
Total 2 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To github.com:narendar-20/new-repository.git
 f0448b5..741c117 naren -> naren
branch 'naren' set up to track 'origin/naren'.
```

The screenshot shows the GitHub web interface for a repository named 'naren'. At the top, it indicates '3 Branches' and '0 Tags'. A search bar and buttons for 'Add file' and 'Code' are visible. Below the repository name, a message states 'This branch is 2 commits ahead of master'. A 'Contribute' button is also present. The main section displays a list of commits by user 'narendar', with the most recent commit being 'add new file' (741c117) from 2 minutes ago. This commit includes files 'alpha', 'f1', 'f2', 'hello', 'hotfix', and 'newfile'. The commit history shows a sequence of additions and amendments. On the right sidebar, the 'Releases' section shows 'No releases published' with a link to 'Create a new release'. The 'Packages' section shows 'No packages published' with a link to 'Publish your first package'.

Commit Hash	Commit Message	Time Ago
741c117	add new file	2 minutes ago
f0448b5	add new file	2 minutes ago
...

11)clone only a branch from github to local:

```
naren@narendar MINGW64 /d/git_store (naren)
$ git clone -b branch1 git@github.com:narendar-20/new-repository.git
Cloning into 'new-repository'...
remote: Enumerating objects: 15, done.
remote: Counting objects: 100% (15/15), done.
remote: Compressing objects: 100% (9/9), done.
remote: Total 15 (delta 4), reused 14 (delta 3), pack-reused 0 (from 0)
Receiving objects: 100% (15/15), done.
Resolving deltas: 100% (4/4), done.

naren@narendar MINGW64 /d/git_store (naren)
$ cd new-repository

naren@narendar MINGW64 /d/git_store/new-repository (branch1)
$ git branch
* branch1
```

12)create a file with all passwords and make that untrackable with git:

```
naren@narendar MINGW64 ~ (master)
$ vi passwordfile

naren@narendar MINGW64 ~ (master)
$ cat passwordfile
password
new password

naren@narendar MINGW64 ~ (master)
$ git status
On branch master
Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
```

```
passwordfile
sedawk.bash
string.bash
"\357\200\233[A\357\200\233[B/"

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

13)make a commit and make that commit reset without savings changes:

```
naren@narendar MINGW64 /d/git_store/new-repository (branch1)
$ git add new

naren@narendar MINGW64 /d/git_store/new-repository (branch1)
$ git status
On branch branch1
Your branch is up to date with 'origin/branch1'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   new

naren@narendar MINGW64 /d/git_store/new-repository (branch1)
$ git commit -m "added new file"
[branch1 4e392ea] added new file
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 new

naren@narendar MINGW64 /d/git_store/new-repository (branch1)
$ git reset

naren@narendar MINGW64 /d/git_store/new-repository (branch1)
$ git status
On branch branch1
Your branch is ahead of 'origin/branch1' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean
```

14) Revert a committed commit to the older version:


```
naren@narendar MINGW64 /d/git_store (master)
$ git log -4
commit f0448b5cbf2e0309867c94bba17d682b39008b8e (HEAD -> master, new-branch)
Author: narendar <narendar27@gmail.com>
Date:   Wed Feb 19 14:27:47 2025 +0530

    added sample file

commit 807186ff5fb4d0123d8af0d23740ff4f9b3e21e6 (origin/master)
Author: narendar <narendar27@gmail.com>
Date:   Wed Feb 19 14:15:35 2025 +0530

    added two files

commit 1a932be4ab772b4f0b61c294a64403dc19b0707e
Author: narendar <narendar27@gmail.com>
Date:   Tue Feb 18 16:43:34 2025 +0530

    alpha added

commit dccbb88b6d9f83c940269b932fe29103b1d9e4f1
Author: narendar <narendar27@gmail.com>
Date:   Tue Feb 18 16:20:02 2025 +0530

    hello

naren@narendar MINGW64 /d/git_store (master)
$ git revert dccbb8
[master 4353e1a] Revert "hello"
 1 file changed, 1 deletion(-)

naren@narendar MINGW64 /d/git_store (master)
$

naren@narendar MINGW64 /d/git_store (master)
$ ls
abc/  f1  f2  hello  hotfix  new-repository/  sample  xyz2
```

15)push a file to stash without savings the changes and work on another file:

```
naren@narendar MINGW64 /d/git_store (master)
$ vi alpha

naren@narendar MINGW64 /d/git_store (master)
$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        alpha

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

naren@narendar MINGW64 /d/git_store (master)
$ git add .
warning: in the working copy of 'alpha', LF will be replaced by CR
LF the next time Git touches it

naren@narendar MINGW64 /d/git_store (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   alpha

naren@narendar MINGW64 /d/git_store (master)
$ git stash
Saved working directory and index state WIP on master: 38af69e add
ed a file edited by using git comment --amend

naren@narendar MINGW64 /d/git_store (master)
$ ls
f1 hello hotfix

naren@narendar MINGW64 /d/git_store (master)
$ vi hello
```

16)undo the stash file and start working on that again:

```
naren@narendar MINGW64 /d/git_store (master)
$ git stash list
stash@{0}: WIP on master: 38af69e added a file edited by using git
comment --amend

naren@narendar MINGW64 /d/git_store (master)
$ git stash pop
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   alpha

Dropped refs/stash@{0} (6e4a6211cef0ee3401ef3c5471682d54989c0b81)

naren@narendar MINGW64 /d/git_store (master)
$ ls
alpha  f1  hello  hotfix
```

17)generate a ssh-keygen and configure into git hub:

```
naren@narendar MINGW64 ~/Desktop/hiring-app (master)
$ cat /c/Users/naren/.ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQGCyC050E6Qh7h1ttfzgUdAyyWSBRPivmOH9XALZog7Q61Xwd9UCEOXK
70X+HU0GIyXt+Gs5mQOYItjp5BFsdokxHNIgX/jtT/Hee7PgbzHP5SbRr3mmadK9PagIDxiIwY13l1Ra2D7OJQJBQ6p2
7Xkj1qpDo6njFkgC/tLjVqSHLOtVpOE7Z8Nah1+vUBoLJw3AvDN2pBM2PhQ/0e1F8K9wnEuZ+DhCj+PkuieXR5H5IiSc
wDwkvfAp4Z2ZX58cpbf159PTyefYVof3owo9+goYlSIQcBS06YODs7jvXt2hbMFRaMxDVbyaVqc1q0neRFHt8Gwmf6Qz
sAdvL00IH9D53N9wXj2Wakx51x9NzAQcj+uCTUKWKTIUjMcQlT2US8LFtSFonu9kwb8Nub3TEF2T3YPajlQjPGxFQ6P8
jKObKHlCJoMOS4GnD3Nqv2rVF1HMZoNh+ugVEqSXPHgOI0oo85Ac1fous2X6MY5Zmj8WuHWIu2qHJwJ0dcQ+M= nare
@narendar

naren@narendar MINGW64 ~/Desktop/hiring-app (master)
$ git push -u origin master
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 4 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (6/6), 502 bytes | 251.00 KiB/s, done.
Total 6 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), done.
To github.com:narendar-20/narendar-01.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.

naren@narendar MINGW64 ~/Desktop/hiring-app (master)
$ ls
b  c  d 'docker file'  e  f  g  h  i  j  k  l  m  n  na  z

naren@narendar MINGW64 ~/Desktop/hiring-app (master)
$ git pull
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (3/3), 928 bytes | 132.00 KiB/s, done.
From github.com:narendar-20/narendar-01
  96591a9..abcf8d  master    -> origin/master
Updating 96591a9..abcf8d
Fast-forward
 abc | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 abc

naren@narendar MINGW64 ~/Desktop/hiring-app (master)
$ ls
abc  b  c  d 'docker file'  e  f  g  h  i  j  k  l  m  n  na  z
```

18)configure webhooks to git hub:

→Install and start Jenkins:

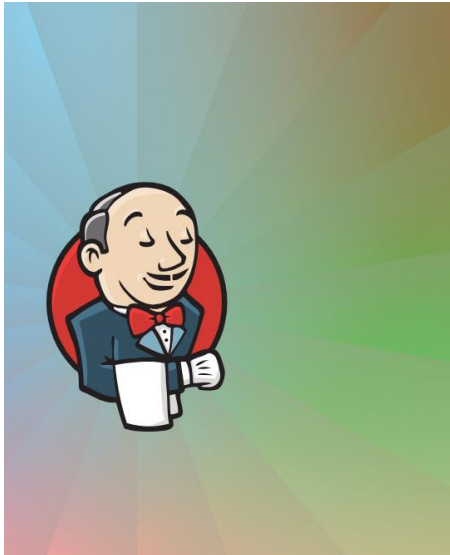
```
[ec2-user@ip-172-31-35-9 ~]$ sudo yum update -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
No packages marked for update
[ec2-user@ip-172-31-35-9 ~]$ sudo -i
[root@ip-172-31-35-9 ~]# sudo yum install java-11-amazon-corretto -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Resolving Dependencies
--> Running transaction check
----> Package java-11-amazon-corretto.x86_64 1:11.0.26+4-1.amzn2 will be installed
--> Processing Dependency: java-11-amazon-corretto-headless(x86-64) = 1:11.0.26+4-1.amzn2 for package: 1:java-11-amazon-corretto-11.0.26+4-1.amzn2.x86_64
--> Processing Dependency: libX11 for package: 1:java-11-amazon-corretto-11.0.26+4-1.amzn2.x86_64
--> Processing Dependency: libXi for package: 1:java-11-amazon-corretto-11.0.26+4-1.amzn2.x86_64
--> Processing Dependency: libXinerama for package: 1:java-11-amazon-corretto-11.0.26+4-1.amzn2.x86_64
--> Processing Dependency: libXt for package: 1:java-11-amazon-corretto-11.0.26+4-1.amzn2.x86_64
--> Processing Dependency: libXrender for package: 1:java-11-amazon-corretto-11.0.26+4-1.amzn2.x86_64
--> Processing Dependency: libXrandr for package: 1:java-11-amazon-corretto-11.0.26+4-1.amzn2.x86_64
--> Processing Dependency: libXtst for package: 1:java-11-amazon-corretto-11.0.26+4-1.amzn2.x86_64
--> Processing Dependency: alsa-lib for package: 1:java-11-amazon-corretto-11.0.26+4-1.amzn2.x86_64
--> Processing Dependency: giflib for package: 1:java-11-amazon-corretto-11.0.26+4-1.amzn2.x86_64
--> Processing Dependency: dejavu-sans-fonts for package: 1:java-11-amazon-corretto-11.0.26+4-1.amzn2.x86_64
--> Processing Dependency: dejavu-serif-fonts for package: 1:java-11-amazon-corretto-11.0.26+4-1.amzn2.x86_64
--> Processing Dependency: dejavu-sans-mono-fonts for package: 1:java-11-amazon-corretto-11.0.26+4-1.amzn2.x86_64
--> Running transaction check
----> Package alsa-lib.x86_64 0:1.1.4-1-2.amzn2 will be installed
----> Package dejavu-sans-fonts.noarch 0:2.33-6.amzn2 will be installed
--> Processing Dependency: dejavu-fonts-common = 2.33-6.amzn2 for package: dejavu-sans-fonts-2.33-6.amzn2.noarch
--> Running transaction check
----> Package dejavu-sans-mono-fonts.noarch 0:2.33-6.amzn2 will be installed
----> Package dejavu-serif-fonts.noarch 0:2.33-6.amzn2 will be installed
----> Package giflib.x86_64 0:4.1.6-9.amzn2.0.2 will be installed
--> Processing Dependency: libICE.so.6()(64bit) for package: giflib-4.1.6-9.amzn2.0.2.x86_64
--> Processing Dependency: libSM.so.6()(64bit) for package: giflib-4.1.6-9.amzn2.0.2.x86_64
--> Package java-11-amazon-corretto-headless.x86_64 1:11.0.26+4-1.amzn2 will be installed
```

```
[root@ip-172-31-35-9 ~]# sudo systemctl start jenkins
[root@ip-172-31-35-9 ~]# sudo systemctl enable jenkins
Created symlink from /etc/systemd/system/multi-user.target.wants/jenkins.service to /usr/lib/systemd/system/jenkins.service.
[root@ip-172-31-35-9 ~]# sudo systemctl status jenkins
● jenkins.service - Jenkins Continuous Integration Server
   Loaded: loaded (/usr/lib/systemd/system/jenkins.service; enabled; vendor preset: disabled)
   Active: active (running) since Tue 2025-03-11 08:42:54 UTC; 10s ago
     Main PID: 4375 (java)
    CGroup: /system.slice/jenkins.service
            └─4375 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.war --webroot=%C/jenk...

Mar 11 08:42:46 ip-172-31-35-9.ap-southeast-2.compute.internal jenkins[4375]: *****
Mar 11 08:42:46 ip-172-31-35-9.ap-southeast-2.compute.internal jenkins[4375]: *****
Mar 11 08:42:46 ip-172-31-35-9.ap-southeast-2.compute.internal jenkins[4375]: *****
Mar 11 08:42:54 ip-172-31-35-9.ap-southeast-2.compute.internal jenkins[4375]: 2025-03-11 08:42:54.521+0000...n
Mar 11 08:42:54 ip-172-31-35-9.ap-southeast-2.compute.internal jenkins[4375]: 2025-03-11 08:42:54.540+0000...g
Mar 11 08:42:54 ip-172-31-35-9.ap-southeast-2.compute.internal systemd[1]: Started Jenkins Continuous Inte...
Mar 11 08:42:56 ip-172-31-35-9.ap-southeast-2.compute.internal jenkins[4375]: 2025-03-11 08:42:56.130+0000...r
Mar 11 08:42:56 ip-172-31-35-9.ap-southeast-2.compute.internal jenkins[4375]: 2025-03-11 08:42:56.130+0000...1
Mar 11 08:42:57 ip-172-31-35-9.ap-southeast-2.compute.internal systemd[1]: [/usr/lib/systemd/system/jenkin...'
Mar 11 08:42:57 ip-172-31-35-9.ap-southeast-2.compute.internal systemd[1]: [/usr/lib/systemd/system/jenkin...'

Hint: Some lines were ellipsized, use -l to show in full.
[root@ip-172-31-35-9 ~]# yum install git -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Resolving Dependencies
```

→ Access the Jenkins UI by copying PUBLIC IP of ec2 instance with 8080:



Sign in to Jenkins

Username

Password

☐ Keep me signed in

Sign in

→ Created a job in Jenkins

The screenshot shows the Jenkins Dashboard. On the left, there's a sidebar with links: New Item, Build History, Manage Jenkins, My Views, Build Queue, and a dropdown for Build Queue (No builds in the queue). The main area shows a table of jobs. The 'webhooks' job is listed with a status of 'Success' (green checkmark), a duration of '35 min', and a last success time of '#2'. The table has columns for S, W, Name, Last Success, Last Failure, and Last Duration. A 'Add description' button is visible in the top right.

S	W	Name	Last Success	Last Failure	Last Duration
✓	☀	webhooks	35 min #2	N/A	0.79 sec

→ Added github webhooks in github

The screenshot shows the GitHub Webhooks configuration page. The left sidebar lists various settings: Branches, Tags, Rules, Actions, Webhooks (selected), Environments, Codespaces, Pages, Security, Code security, Deploy keys, Secrets and variables, Integrations, GitHub Apps, and Email notifications. The main content area is for configuring a webhook. It includes fields for 'Payload URL' (http://3.25.206.82:8080/github-webhooks/), 'Content type' (application/json), and a 'Secret' field. There's a section for 'SSL verification' with options to 'Enable' or 'Disable' (selected). Below that, there's a section for 'Which events would you like to trigger this webhook?' with options: 'Just the push event.' (selected), 'Send me everything.', and 'Let me select individual events.'. At the bottom, there's a checkbox for 'Active' (checked) and two buttons: 'Update webhook' and 'Delete webhook'.

→ push a file to github repo:

Name	Last commit message	Last commit date
Horizon	Create Horizon	1 hour ago
TEST	Create TEST	1 hour ago
TH	Create TH	53 minutes ago
Techie	Create Techie	1 hour ago

→Go to Jenkins Dashboard, the build starts automatically:

Dashboard > webhooks >

Status

</> Changes

Workspace

Build Now

Configure

Delete Project

GitHub Hook Log

Rename

webhooks

Add description

Permalinks

- Last build (#2), 54 min ago
- Last stable build (#2), 54 min ago
- Last successful build (#2), 54 min ago
- Last completed build (#2), 54 min ago

Builds

Filter

Today

- #2 8:55 AM
- #1 8:53 AM

Jenkins

Dashboard > webhooks > #2

Status

</> Changes

Console Output

Edit Build Information

Delete build '#2'

Timings

Git Build Data

Previous Build

#2 (Mar 11, 2025, 8:55:56 AM)

Add description

Keep this build forever

Started 54 min ago

Took 0.79 sec

Started by user [admin](#)

This run spent:

- 14 ms waiting;
- 0.79 sec build duration;
- 0.8 sec total from scheduled to completion.

Revision: 2f659cb36158f11902dec48b16d3f15cf1d9e30d

Repository: <https://github.com/isyedriyan/test.git>

- refs/remotes/origin/main

Changes

1. Create TH (details / [githubweb](#))

19)basic understanding of .git file:

```
naren@narendar MINGW64 /d/git_store (master)
$ rm -rf .git/

naren@narendar MINGW64 /d/git_store
$ git init
Initialized empty Git repository in D:/git_store/.git/
```

20)Check all the logs of git:

```
naren@narendar MINGW64 /d/git_store (master)
$ git log
commit 4cb969eb171b4c6d3802d23a557625dc7f915cd6 (HEAD -> master)
Author: narendar <narendar27@gmail.com>
Date: Tue Feb 18 13:13:13 2025 +0530

    add in

commit 609602fa7e2f6e5baffc5b191d2e62f14aa5a0e5
Author: narendar <narendar27@gmail.com>
Date: Tue Feb 18 12:39:47 2025 +0530

    added hello in master
```


21)Rename the commit message:

```
naren@narendar MINGW64 /d/git_store (master)
$ git log
commit 6608969270255673f5ee2b159124c1d9690291ee (HEAD -> master)
Author: narendar <narendar27@gmail.com>
Date: Tue Feb 18 16:00:31 2025 +0530

    added a file

naren@narendar MINGW64 /d/git_store (master)
$ git --amend
unknown option: --amend
usage: git [-v | --version] [-h | --help] [-c <path>] [-c <name>=<value>]
          [--exec-path[=<path>]] [--html-path] [--man-path] [--info-path]
          [-p | --paginate | -P | --no-pager] [--no-replace-objects] [--bare]
          [--git-dir=<path>] [--work-tree=<path>] [--namespace=<name>]
          [--super-prefix=<path>] [--config-env=<name>=<envvar>]
          <command> [<args>]

naren@narendar MINGW64 /d/git_store (master)
$ git commit --amend
[master 38af69e] added a file edited by using git comment --amend
Date: Tue Feb 18 16:00:31 2025 +0530
3 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 f1
create mode 100644 hello
create mode 100644 hotfix

naren@narendar MINGW64 /d/git_store (master)
$ git log
commit 38af69e652de2c1adb0646a67f267a088793148d (HEAD -> master)
Author: narendar <narendar27@gmail.com>
Date: Tue Feb 18 16:00:31 2025 +0530

    added a file edited by using git comment --amend
```

22) Merge multiple commits into single commit:

```
naren@narendar MINGW64 /d/git_store (release)
$ ls
hello  release

naren@narendar MINGW64 /d/git_store (release)
$ git checkout hotfix
Switched to branch 'hotfix'

naren@narendar MINGW64 /d/git_store (hotfix)
$ ls
hello

naren@narendar MINGW64 /d/git_store (hotfix)
$ git merge release
Updating 609602f..c969d63
Fast-forward
 release | 0
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 release

naren@narendar MINGW64 /d/git_store (hotfix)
$ ls
hello  release
```