

GIT Assignment

1. **git init** is used to initialize a git repository. .git folder is created.
2. **git add** To add a file to the staging area.
3. **git commit -m "message"** This command commits the staged changes to the local repository

```
shank@shankar MINGW64 /d/Devops/Project
$ git init
Initialized empty Git repository in D:/Devops/Project/.git/

shank@shankar MINGW64 /d/Devops/Project (master)
$ touch f1

shank@shankar MINGW64 /d/Devops/Project (master)
$ git add f1

shank@shankar MINGW64 /d/Devops/Project (master)
$ git commit -m "added f1"
[master (root-commit) 7653a2f] added f1
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 f1
```

4. **git status** It will show whether files are in workspace or staging are on in git repo

```
shank@shankar MINGW64 /d/Devops/Project (master)
$ touch f2

shank@shankar MINGW64 /d/Devops/Project (master)
$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    f2

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

shank@shankar MINGW64 /d/Devops/Project (master)
$ git add f2

shank@shankar MINGW64 /d/Devops/Project (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   f2

shank@shankar MINGW64 /d/Devops/Project (master)
$ git commit -m "added f2"
[master 2e8418d] added f2
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 f2

shank@shankar MINGW64 /d/Devops/Project (master)
$ git status
On branch master
nothing to commit, working tree clean
```

5. **git log** To list/check the committed file or commit history
6. **git log --oneline** To list all the commit in column format (single column)
7. **git show** To display detailed information about a specific commit, including the commit message, author, date, and the actual changes made to the files.
8. **git show <commit_ID>** To know the modifications in the last commit (prints the line)

```
shank@shankar MINGW64 /d/Devops/Project (master)
$ git log
commit 2e8418de1defb1c5ac57c596205b2684772dfd7 (HEAD -> master)
Author: kavita <kkommar334@gmail.com>
Date:   Mon Aug 5 07:32:43 2024 +0530

    added f2

commit 7653a2f59b1f2151357d83e866035c243548784a
Author: kavita <kkommar334@gmail.com>
Date:   Mon Aug 5 07:30:07 2024 +0530

    added f1

shank@shankar MINGW64 /d/Devops/Project (master)
$ git log --oneline
2e8418d (HEAD -> master) added f2
7653a2f added f1

shank@shankar MINGW64 /d/Devops/Project (master)
$ git show
commit 2e8418de1defb1c5ac57c596205b2684772dfd7 (HEAD -> master)
Author: kavita <kkommar334@gmail.com>
Date:   Mon Aug 5 07:32:43 2024 +0530

    added f2

diff --git a/f2 b/f2
new file mode 100644
index 0000000..e69de29

shank@shankar MINGW64 /d/Devops/Project (master)
$ git show 7653a2f59b1f2151357d83e866035c243548784a
commit 7653a2f59b1f2151357d83e866035c243548784a
Author: kavita <kkommar334@gmail.com>
Date:   Mon Aug 5 07:30:07 2024 +0530

    added f1

diff --git a/f1 b/f1
new file mode 100644
index 0000000..e69de29
```

9. **git commit --amend** It will help you edit the commit message already committed. The old commit is replaced by new commit. It means old commit will not be there

```
shank@shankar: MINGW64 /d/Devops/Project (branch1)
$ git log --oneline
2e8418d (HEAD -> branch1, master) added f2
7653a2f added f1

shank@shankar: MINGW64 /d/Devops/Project (branch1)
$ git commit --amend
[branch1 f1b2117] updated f2 file
Date: Mon Aug 5 07:32:43 2024 +0530
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 f2

shank@shankar: MINGW64 /d/Devops/Project (branch1)
$ git log --oneline
f1b2117 (HEAD -> branch1) updated f2 file
7653a2f added f1
```

```
updated f2 file
# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# Date:      Mon Aug 5 07:32:43 2024 +0530
#
# On branch branch1
#
# Changes to be committed:
#   new file:   f2
#
```

10. **git restore --staged <filename>** it will move the changes from staging area to workspace

```
shank@shankar: MINGW64 /d/project (main)
$ touch f8
shank@shankar: MINGW64 /d/project (main)
$ git add f8
shank@shankar: MINGW64 /d/project (main)
$ git status
On branch main
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   f8

shank@shankar: MINGW64 /d/project (main)
$ git restore --staged f8
shank@shankar: MINGW64 /d/project (main)
$ git status
On branch main
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    f8

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

11. **git reset:** committed changes are reset, but commit history is removed

12. **Git reset --hard HEAD~1:** the committed changes are permanently removed from history and workspace as well

```
shank@shankar: MINGW64 /d/project ((9afdd4c...))
$ git log --oneline
9afdd4c (HEAD) Adding f8
d92d2f1 (tag: v2, main) Added f2
7504bab Initial commit

shank@shankar: MINGW64 /d/project ((9afdd4c...))
$ git reset --hard HEAD~1
HEAD is now at d92d2f1 Added f2

shank@shankar: MINGW64 /d/project ((v2))
$ git log --oneline
d92d2f1 (HEAD, tag: v2, main) Added f2
7504bab Initial commit

shank@shankar: MINGW64 /d/project ((v2))
$ git status
HEAD detached at v2
nothing to commit, working tree clean
```

13. **Git reset --soft HEAD~1:** committed changes will move to staging area but history will be removed

```
shank@shankar: MINGW64 /d/project ((2bb5f99...))
$ git log --oneline
2bb5f99 (HEAD) Adding f1
9afdd4c Adding f8
d92d2f1 (tag: v2, main) Added f2
7504bab Initial commit

shank@shankar: MINGW64 /d/project ((2bb5f99...))
$ git reset --soft HEAD~1
shank@shankar: MINGW64 /d/project ((9afdd4c...))
$ git status
HEAD detached from v2
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   add
    new file:   f1

shank@shankar: MINGW64 /d/project ((9afdd4c...))
$ git log --oneline
9afdd4c (HEAD) Adding f8
d92d2f1 (tag: v2, main) Added f2
7504bab Initial commit
```

14. Git reset --soft <commit id> : committed changes after mentioned commitid will move to staging area but history will be removed

```
shank@shankar MINGW64 /d/project ((1e3fee0...))
$ git log --oneline
1e3fee0 (HEAD) Adding f1
9afdd4c Adding f8
892d2f1 (tag: V2, main) Added f2
7504bab Initial commit

shank@shankar MINGW64 /d/project ((1e3fee0...))
$ git reset --soft 9afdd4c

shank@shankar MINGW64 /d/project ((9afdd4c...))
$ git status
HEAD detached from V2
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   add
        new file:   f1

shank@shankar MINGW64 /d/project ((9afdd4c...))
$ git log --oneline
9afdd4c (HEAD) Adding f8
892d2f1 (tag: V2, main) Added f2
7504bab Initial commit
```

15. git reset --hard <commit id>: committed changes after mentioned commitid will be permanently deleted but history will be removed

```
shank@shankar MINGW64 /d/project ((54c26a5...))
$ git log --oneline
54c26a5 (HEAD) Adding f1
9afdd4c Adding f8
892d2f1 (tag: V2, main) Added f2
7504bab Initial commit

shank@shankar MINGW64 /d/project ((54c26a5...))
$ git reset --hard 9afdd4c
HEAD is now at 9afdd4c Adding f8

shank@shankar MINGW64 /d/project ((9afdd4c...))
$ git log --oneline
9afdd4c (HEAD) Adding f8
892d2f1 (tag: V2, main) Added f2
7504bab Initial commit
```

16. Git revert <commit id>: we can go back to the workspace after committing, but commit history will be stored

```
shank@shankar MINGW64 /d/project ((fc82f0c...))
$ git log --oneline
fc82f0c (HEAD) Adding f4
4d2ca4a Adding f3
892d2f1 (tag: V2, main) Added f2
7504bab Initial commit

shank@shankar MINGW64 /d/project ((fc82f0c...))
$ ls
README.md f2 f3 f4

shank@shankar MINGW64 /d/project ((fc82f0c...))
$ git revert fc82f0c
[detached HEAD 44ea2f6] Revert "Adding f4"
1 file changed, 0 insertions(+), 0 deletions(-)
delete mode 100644 f4

shank@shankar MINGW64 /d/project ((44ea2f6...))
$ git log --oneline
44ea2f6 (HEAD) Revert "Adding f4"
fc82f0c Adding f4
4d2ca4a Adding f3
892d2f1 (tag: V2, main) Added f2
7504bab Initial commit

shank@shankar MINGW64 /d/project ((44ea2f6...))
$ ls
README.md f2 f3
```

17. git config --global user.name <username> To update the user name.

18. git config --global user.email <email> To update the email in the terminal.

```
shank@shankar MINGW64 /d/Devops/Project (master)
$ git config --global user.name "kavita"
bash: $: command not found

shank@shankar MINGW64 /d/Devops/Project (master)
$ git config --global user.name "kavita"

shank@shankar MINGW64 /d/Devops/Project (master)
$ git config --global user.email "kkonnur334@gmail.com"

shank@shankar MINGW64 /d/Devops/Project (master)
$ git log
commit 2e8418de51defb1c5ac57c596205b2684772dfd7 (HEAD -> master)
Author: kavita <kkonnur334@gmail.com>
Date: Mon Aug 5 07:32:43 2024 +0530

    added f2
```

19. git branch <name> To create a branch.

20. git branch To list the branches.

21. git checkout <branch name> To switch to other branch.

22. git checkout -b <branch name> To create new branch and switch to new branch.

23. git branch -d <branch name> To delete a branch.

```

shank@shankar MINGW64 /d/Devops/Project (master)
$ git branch
* master

shank@shankar MINGW64 /d/Devops/Project (master)
$ git branch branch1

shank@shankar MINGW64 /d/Devops/Project (master)
$ git branch
branch1
* master

shank@shankar MINGW64 /d/Devops/Project (master)
$ git checkout -b branch2
Switched to a new branch 'branch2'

shank@shankar MINGW64 /d/Devops/Project (branch2)
$ git checkout master
Switched to branch 'master'

shank@shankar MINGW64 /d/Devops/Project (master)
$ git branch
branch1
branch2
* master

shank@shankar MINGW64 /d/Devops/Project (master)
$ git branch -d branch2
Deleted branch branch2 (was 2e8418d).

shank@shankar MINGW64 /d/Devops/Project (master)
$ git branch
branch1
* master

shank@shankar MINGW64 /d/Devops/Project (master)
$ git checkout branch1
Switched to branch 'branch1'

```

24. Git merge <branchname>: to merge specified branch to checked out branch

```

shank@shankar MINGW64 /d/dev1 (branch1)
$ ls
file1 file2 file3 file4 file5

shank@shankar MINGW64 /d/dev1 (branch1)
$ git checkout master
Switched to branch 'master'

shank@shankar MINGW64 /d/dev1 (master)
$ ls
file1 file2 file3 file4

shank@shankar MINGW64 /d/dev1 (master)
$ git log --oneline
453fbc3 (HEAD -> master) Adding file4
d376d2d Adding file3
84250fa Adding file2
8874083 Adding file1

shank@shankar MINGW64 /d/dev1 (master)
$ git merge branch1
Updating 453fbc3..ccc21a0
Fast-forward
 file5 | 0
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 file5

shank@shankar MINGW64 /d/dev1 (master)
$ git log --oneline
ccc21a0 (HEAD -> master, branch1) Adding file5
453fbc3 Adding file4
d376d2d Adding file3
84250fa Adding file2
8874083 Adding file1

```

25. Git rebase: rewrites the history by creating new commits for each commit in source branch.

```

shank@shankar MINGW64 /d/dev1 (master)
$ git checkout -b branch2
Switched to a new branch 'branch2'

shank@shankar MINGW64 /d/dev1 (branch2)
$ touch file6

shank@shankar MINGW64 /d/dev1 (branch2)
$ git add file6

shank@shankar MINGW64 /d/dev1 (branch2)
$ git commit -m "Adding file6"
[branch2 f51ca24] Adding file6
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 file6

shank@shankar MINGW64 /d/dev1 (branch2)
$ git log --oneline
f51ca24 (HEAD -> branch2) Adding file6
ccc21a0 (master, branch1) Adding file5
453fbc3 Adding file4
d376d2d Adding file3
84250fa Adding file2
8874083 Adding file1

shank@shankar MINGW64 /d/dev1 (branch2)
$ git checkout master
Switched to branch 'master'

shank@shankar MINGW64 /d/dev1 (master)
$ git rebase branch2
Successfully rebased and updated refs/heads/master.

shank@shankar MINGW64 /d/dev1 (master)
$ git log --oneline
f51ca24 (HEAD -> master, branch2) Adding file6
ccc21a0 (branch1) Adding file5
453fbc3 Adding file4
d376d2d Adding file3
84250fa Adding file2
8874083 Adding file1

```

26. Git cherry-pick <commitid>: to merge specific commit on branch(currently checkout branch)

```
shank@shankar: MINGW64 /d/project (branch1)
$ git log --oneline
5a27c33 (HEAD -> branch1) Adding f5
b01abd8 Adding f4
23c1ffc Added f1
7504bab Initial commit

shank@shankar: MINGW64 /d/project (branch1)
$ git checkout main
Switched to branch 'main'

shank@shankar: MINGW64 /d/project (main)
$ git log --oneline
d92d2f1 (HEAD -> main, tag: V2) Added f2
7504bab Initial commit

shank@shankar: MINGW64 /d/project (main)
$ git cherry-pick b01abd8
[main 835610a] Adding f4
Date: Fri Aug 2 11:04:10 2024 +0530
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 f4

shank@shankar: MINGW64 /d/project (main)
$ git log --oneline
835610a (HEAD -> main) Adding f4
d92d2f1 (tag: V2) Added f2
7504bab Initial commit
```

27. git tag tagname or git tag -a <commit id> -m "tagging f2" To create a tag name

28. git checkout tagname Switch to tag

29. git tag -d tagname To delete the tag

```
shank@shankar: MINGW64 /d/Devops/Project (branch1)
$ git log --oneline
17cc423 (HEAD -> branch1) Added f8
282face Adding f7
2a6bc16 updated f1
eb53b71 updated f2 file
7653a2f added f1

shank@shankar: MINGW64 /d/Devops/Project (branch1)
$ git tag v1

shank@shankar: MINGW64 /d/Devops/Project (branch1)
$ git log --oneline
17cc423 (HEAD -> branch1, tag: v1) Added f8
282face Adding f7
2a6bc16 updated f1
eb53b71 updated f2 file
7653a2f added f1

shank@shankar: MINGW64 /d/Devops/Project (branch1)
$ git checkout v1
Note: switching to 'v1'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by switching back to a branch.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -c with the switch command. Example:

    git switch -c <new-branch-name>

or undo this operation with:

    git switch -

Turn off this advice by setting config variable advice.detachedHead to false

HEAD is now at 17cc423 Added f8

shank@shankar: MINGW64 /d/Devops/Project ((v1))
$ git checkout master
Previous HEAD position was 17cc423 Added f8
Switched to branch 'master'

shank@shankar: MINGW64 /d/Devops/Project (master)
$ git tag -d v1
Deleted tag 'v1' (was 17cc423)

shank@shankar: MINGW64 /d/Devops/Project (master)
$ git log --oneline
8a0bbd4 (HEAD -> master) Adding .gitignore
9f46ede Squashed f2 f3 f4
7653a2f added f1
```

30. git blame <filename> To show the last modification for each line in a file, it will show who has modified the code(eachline of the code)

```
shank@shankar: MINGW64 /d/Devops/Project (branch1)
$ git blame f1
2a6bc165 (kavita 2024-08-05 08:07:34 +0530 1) Hey there!

shank@shankar: MINGW64 /d/Devops/Project (branch1)
$ vi f1

shank@shankar: MINGW64 /d/Devops/Project (branch1)
$ git blame f1
2a6bc165 (kavita 2024-08-05 08:07:34 +0530 1) Hey there!
00000000 (Not Committed Yet 2024-08-05 08:08:31 +0530 2) welcome
```

31. git branch -merged :- To check the merged branches.

32. git branch --no-merged:- To check non merged branches

```
shank@shankar: MINGW64 /d/Devops/Project (master)
$ git branch
  branch1
  branch2
* master

shank@shankar: MINGW64 /d/Devops/Project (master)
$ git branch --no-merged
  branch1

shank@shankar: MINGW64 /d/Devops/Project (master)
$ git branch --merged
  branch2
* master
```

33. git clean -n It will show what it would remove Git clean -f To delete all untracked files.

```
shank@shankar: MINGW64 /d/Devops/Project (master)
$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        f3
        f4
        f5

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

shank@shankar: MINGW64 /d/Devops/Project (master)
$ git clean -n
would remove f3
would remove f4
would remove f5

shank@shankar: MINGW64 /d/Devops/Project (master)
$ git clean -f
Removing f3
Removing f4
Removing f5

shank@shankar: MINGW64 /d/Devops/Project (master)
$ git status
git: 'status' is not a git command. See 'git --help'.

The most similar command is
    status
```

34. git/merge conflict A piece of code work parallelly in 2 branches, when they merge, conflictarise.

- a. Initialized the git repository and created file 1 in master branch

Name	Status	Date modified	Type	Size
.git		22/07/2024 10:34	File folder	
file1		22/07/2024 10:34	File	0 KB

```
MINGW64/c/Users/shank/OneDrive/Desktop/New folder
$ git init
Initialized empty Git repository in C:/Users/shank/OneDrive/Desktop/New folder/.git/
shank@shankar: MINGW64 ~/OneDrive/Desktop/new folder (master)
$ ls
shank@shankar: MINGW64 ~/OneDrive/Desktop/new folder (master)
$ touch file1
shank@shankar: MINGW64 ~/OneDrive/Desktop/new folder (master)
$ git add file1
shank@shankar: MINGW64 ~/OneDrive/Desktop/new folder (master)
$ git commit file1
[master (root-commit) a06ee0b] adding file1
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 file1
shank@shankar: MINGW64 ~/OneDrive/Desktop/new folder (master)
$ ls
file1
shank@shankar: MINGW64 ~/OneDrive/Desktop/new folder (master)
$
```

- b. Created a feature1 branch and done some changes in file1 and committed it.

Name	Status	Date modified	Type	Size
.git		22/07/2024 10:46	File folder	
file1		22/07/2024 10:45	File	1 KB

```
MINGW64/c/Users/shank/OneDrive/Desktop/New folder
shank@shankar: MINGW64 ~/OneDrive/Desktop/new folder (master)
$ ls
file1
shank@shankar: MINGW64 ~/OneDrive/Desktop/new folder (master)
$ git branch feature1
shank@shankar: MINGW64 ~/OneDrive/Desktop/new folder (master)
$ git checkout feature1
Switched to branch 'feature1'
shank@shankar: MINGW64 ~/OneDrive/Desktop/new folder (feature1)
$ ls
file1
shank@shankar: MINGW64 ~/OneDrive/Desktop/new folder (feature1)
$ vi file1
shank@shankar: MINGW64 ~/OneDrive/Desktop/new folder (feature1)
$ cat file1
Hello, good morning!!
shank@shankar: MINGW64 ~/OneDrive/Desktop/new folder (feature1)
$ git add file1
warning: in the working copy of 'file1', LF will be replaced by CRLF the next time Git touches it
shank@shankar: MINGW64 ~/OneDrive/Desktop/new folder (feature1)
$ git commit -m "Done changes in file1"
[feature1 a3161a3] Done changes in file1
1 file changed, 1 insertion(+)
shank@shankar: MINGW64 ~/OneDrive/Desktop/new folder (feature1)
$
```

- c. Checkout to master branch and done changes in file1 and committed

Name	Status	Date modified	Type	Size
git		22/07/2024 10:46	File folder	
file1		22/07/2024 10:45	File	1 KB

```

MINGW64~/Users/shank/OneDrive/Desktop/New folder
shank@shankar MINGW64 ~/OneDrive/Desktop/New folder (feature1)
$ git checkout master
switched to branch "master"
shank@shankar MINGW64 ~/OneDrive/Desktop/New folder (master)
$ cat file1
shank@shankar MINGW64 ~/OneDrive/Desktop/New folder (master)
$ vi file1
shank@shankar MINGW64 ~/OneDrive/Desktop/New folder (master)
$ cat file1
welcome to India
shank@shankar MINGW64 ~/OneDrive/Desktop/New folder (master)
$ git commit -m "Done changes in file1 in master branch"
[master 19331b5] Done changes in file1 in master branch
1 file changed, 1 insertion(+)
shank@shankar MINGW64 ~/OneDrive/Desktop/New folder (master)
$

```

d. Now when I try to merge feature1 branch with main branch, merge conflict arises.

Name	Status	Date modified	Type	Size
git		22/07/2024 10:50	File folder	
file1		22/07/2024 10:49	File	1 KB

```

MINGW64~/Users/shank/OneDrive/Desktop/New folder
$ git commit -m "Done changes in file1 in master branch"
[master 19331b5] Done changes in file1 in master branch
1 file changed, 1 insertion(+)
shank@shankar MINGW64 ~/OneDrive/Desktop/New folder (master)
$ git add file1
warning: in the working copy of 'file1', LF will be replaced by CRLF the next time Git touches it
shank@shankar MINGW64 ~/OneDrive/Desktop/New folder (master)
$ git commit -m "Done changes in file1 in master branch"
[master 19331b5] Done changes in file1 in master branch
1 file changed, 1 insertion(+)
shank@shankar MINGW64 ~/OneDrive/Desktop/New folder (master)
$ git merge feature1
Auto-merging file1
CONFLICT (Content): Merge conflict in file1
Automatic merge failed; fix conflicts and then commit the result.
shank@shankar MINGW64 ~/OneDrive/Desktop/New folder (master|MERGING)
$

```

35. Git diff :- The git diff command shows the differences between the files in twocommits or between your current repository and a previous commit.

36. git diff <commitid> <commitid> or git diff CID..CID :- put starting 7 char to check thedifference b/w commit id's

```

shank@shankar MINGW64 /d/Devops/Project (master)
$ git log --oneline
e8418d (HEAD -> master, branch2) added f2
7653a2f added f1
shank@shankar MINGW64 /d/Devops/Project (master)
$ git diff
warning: in the working copy of 'f1', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'f2', LF will be replaced by CRLF the next time Git touches it
diff --git a/f1 b/f1
index e69de29..78e2ac8 100644
--- a/f1
+++ b/f1
@@ -0,0 +1 @@
+welcome to India
diff --git a/f2 b/f2
index e69de29..c189c6b 100644
--- a/f2
+++ b/f2
@@ -0,0 +1 @@
+welcome to India

```

```

shank@shankar MINGW64 /d/Devops/Project (master)
$ git log
commit c07158e0023adde52028b0543326c4460083b561 (HEAD -> master)
Author: kavita <konnur334@gmail.com>
Date: Mon Aug 5 08:31:59 2024 +0530
    added f4
commit 44b558e26c01b364003f6d0d996e900574bb3a12
Author: kavita <konnur334@gmail.com>
Date: Mon Aug 5 08:31:17 2024 +0530
    added f3
commit 2e8418de61defb1c5ac57c596205b2684772df7 (branch2)
Author: kavita <konnur334@gmail.com>
Date: Mon Aug 5 07:32:43 2024 +0530
    added f2
commit 7653a2f59b1f2151357d83e866035c243548784a
Author: kavita <konnur334@gmail.com>
Date: Mon Aug 5 07:30:07 2024 +0530
    added f1
shank@shankar MINGW64 /d/Devops/Project (master)
$ git diff 44b558e26c01b364003f6d0d996e900574bb3a12...c07158e0023adde52028b0543326c4460083b561
diff --git a/f4 b/f4
new file mode 100644
index 0000000..def5b78
--- /dev/null
+++ b/f4
@@ -0,0 +1,2 @@
+
+

```

37. .gitignore It specifies which files and dir should be ignored or excluded for staging

```
shank@shankar MINGW64 /d/Devops/Project (master)
$ touch f{1..6}

shank@shankar MINGW64 /d/Devops/Project (master)
$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    f1
    f2
    f3
    f4
    f5
    f6

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

shank@shankar MINGW64 /d/Devops/Project (master)
$ touch .gitignore

shank@shankar MINGW64 /d/Devops/Project (master)
$ vi .gitignore

shank@shankar MINGW64 /d/Devops/Project (master)
$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    .gitignore

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

```
f5
f6
~
~
~
~
~
```

38. git squash (git rebase -i HEAD ~ 2 or 3 (no of commit) to merge multiple commits into a single commit, used to cleanup a commit history before merging

```
shank@shankar MINGW64 /d/Devops/Project (master)
$ git log --oneline
c07158e (HEAD -> master) added f4
44b558e added f3
2e8418d (branch2) added f2
7653a2f added f1

shank@shankar MINGW64 /d/Devops/Project (master)
$ git rebase -i HEAD~3
[detached HEAD 9f46ede] Squashed f2 f3 f4
Date: Mon Aug 5 07:32:43 2024 +0530
4 files changed, 5 insertions(-)
create mode 100644 f2
create mode 100644 f3
create mode 100644 f4
Successfully rebased and updated refs/heads/master.

shank@shankar MINGW64 /d/Devops/Project (master)
$ git log --oneline
9f46ede (HEAD -> master) Squashed f2 f3 f4
7653a2f added f1
```

```
pick 2e8418d added f2
squash 44b558e added f3
squash c07158e added f4

# Rebase 7653a2f..c07158e onto 9f46ede (3 commands)
#
# Commands:
#
# This is a combination of 3 commits.
# This is the 1st commit message:
squashed f2 f3 f4|
# This is the commit message #2:
#
# This is the commit message #3:
```

39. git reflog To view the history of all references and changes in the repository, including those that are not visible in the standard commit log.

```
shank@shankar MINGW64 /d/Devops/Project (master)
$ git reflog
9f46ede (HEAD -> master) HEAD@{0}: rebase (finish): returning to refs/heads/master
9f46ede (HEAD -> master) HEAD@{1}: rebase (squash): Squashed f2 f3 f4
2202b3 HEAD@{2}: rebase (squash): # This is a combination of 2 commits.
2e8418d (branch2) HEAD@{3}: rebase (start): checkout HEAD~3
c07158e HEAD@{4}: checkout: moving from 2e8418de61defb1c3ac57c596205b2684772dfd7 to master
2e8418d (branch2) HEAD@{5}: checkout: moving from master to 2e8418de61defb1c3ac57c596205b2684772dfd7
c07158e HEAD@{6}: commit: added f4
44b558e HEAD@{7}: commit: added f3
2e8418d (branch2) HEAD@{8}: checkout: moving from branch1 to master
2a8bc16 (branch1) HEAD@{9}: reset: moving to HEAD
2a8bc16 (branch1) HEAD@{10}: commit: updated f1
a053b71 HEAD@{11}: commit (amend): updated f2 file
f3b2117 HEAD@{12}: commit (amend): updated f2 file
2e8418d (branch2) HEAD@{13}: checkout: moving from master to branch1
2e8418d (branch2) HEAD@{14}: checkout: moving from branch2 to master
2e8418d (branch2) HEAD@{15}: checkout: moving from master to branch2
2e8418d (branch2) HEAD@{16}: commit: added f2
7653a2f HEAD@{17}: commit (initial): added f1
```


40. git stash To hide the file from master/branch (after staging the file).

41. git stash pop To get back files from temporary area

```
shank@shankar MINGW64 /d/Devops/Project (branch1)
$ vi f7

shank@shankar MINGW64 /d/Devops/Project (branch1)
$ git checkout master
error: Your local changes to the following files would be overwritten by checkout:
f7
Please commit your changes or stash them before you switch branches.
Aborting

shank@shankar MINGW64 /d/Devops/Project (branch1)
$ git stash
warning: in the working copy of 'f7', LF will be replaced by CRLF the next time Git touches it
Saved working directory and index state WIP on branch1: 282face Adding f7

shank@shankar MINGW64 /d/Devops/Project (branch1)
$ git checkout master
Switched to branch 'master'

shank@shankar MINGW64 /d/Devops/Project (master)
$ git checkout branch1
Switched to branch 'branch1'

shank@shankar MINGW64 /d/Devops/Project (branch1)
$ cat f7
welcome

shank@shankar MINGW64 /d/Devops/Project (branch1)
$ git stash pop
On branch branch1
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:   f7

no changes added to commit (use "git add" and/or "git commit -a")
Dropped refs/stash@{0} (74fbab2b10ecb5e132e0428a9b5d9c7c7bbe2938)

shank@shankar MINGW64 /d/Devops/Project (branch1)
$ cat f7
welcome
```

42. git stash drop When you are done with the stashed item or want to remove it from the list. It will remove the last added stash item by default

```
shank@shankar MINGW64 /d/Devops/Project (branch1)
$ vi f7

shank@shankar MINGW64 /d/Devops/Project (branch1)
$ git stash
Saved working directory and index state WIP on branch1: 17cc423 Added f8

shank@shankar MINGW64 /d/Devops/Project (branch1)
$ vi f8

shank@shankar MINGW64 /d/Devops/Project (branch1)
$ git stash
warning: in the working copy of 'f8', LF will be replaced by CRLF the next time Git touches it
Saved working directory and index state WIP on branch1: 17cc423 Added f8

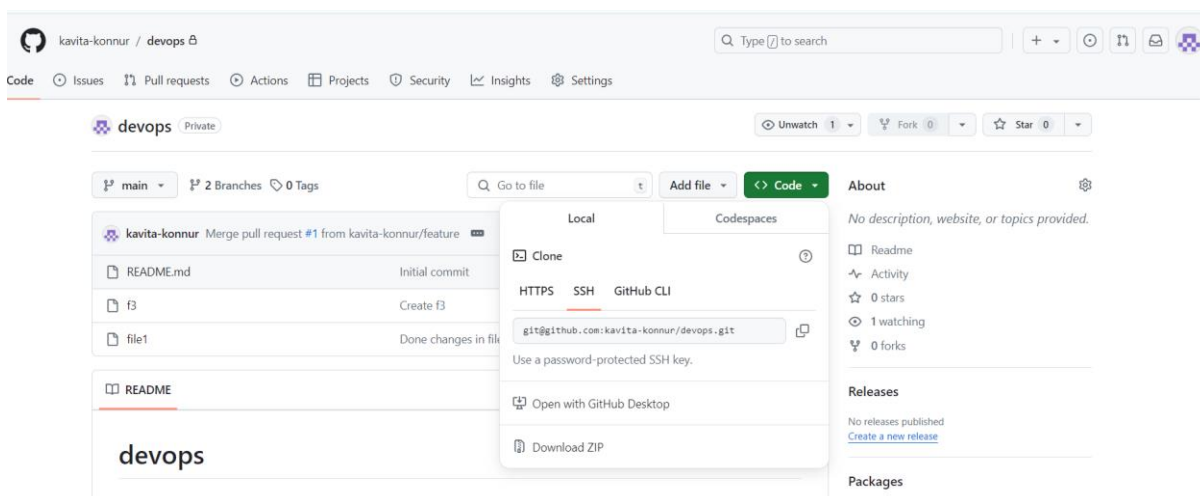
shank@shankar MINGW64 /d/Devops/Project (branch1)
$ git stash drop
Dropped refs/stash@{0} (8ab9448ede7a595cc11b45655fd95bd24f0aa435)

shank@shankar MINGW64 /d/Devops/Project (branch1)
$ git stash pop
On branch branch1
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:   f7

no changes added to commit (use "git add" and/or "git commit -a")
Dropped refs/stash@{0} (b4b47a4708c70f20c31ca45a75efbba7fced8600)
```

43. Git clone <URL> synch central repo to local repo only 1st time

---central/remote repo



--- cloned central repo using 'git clone' command, created and committed file2 and then pushed to central repo using 'git push' command.

```
shank@shankar MINGW64 /d/Devops/Project (master)
$ git clone git@github.com:kavita-konnur/devops.git
Cloning into 'devops'...
remote: Enumerating objects: 31, done.
remote: Counting objects: 100% (31/31), done.
remote: Compressing objects: 100% (22/22), done.
remote: Total 31 (delta 5), reused 11 (delta 1), pack-reused 0
Receiving objects: 100% (31/31), 8.63 KiB | 1.44 MiB/s, done.
Resolving deltas: 100% (5/5), done.

shank@shankar MINGW64 /d/Devops/Project (master)
$ ls
devops/  f1  f2  f3  f4

shank@shankar MINGW64 /d/Devops/Project (master)
$ cd devops/

shank@shankar MINGW64 /d/Devops/Project/devops (main)
$ ls
README.md  f3  file1

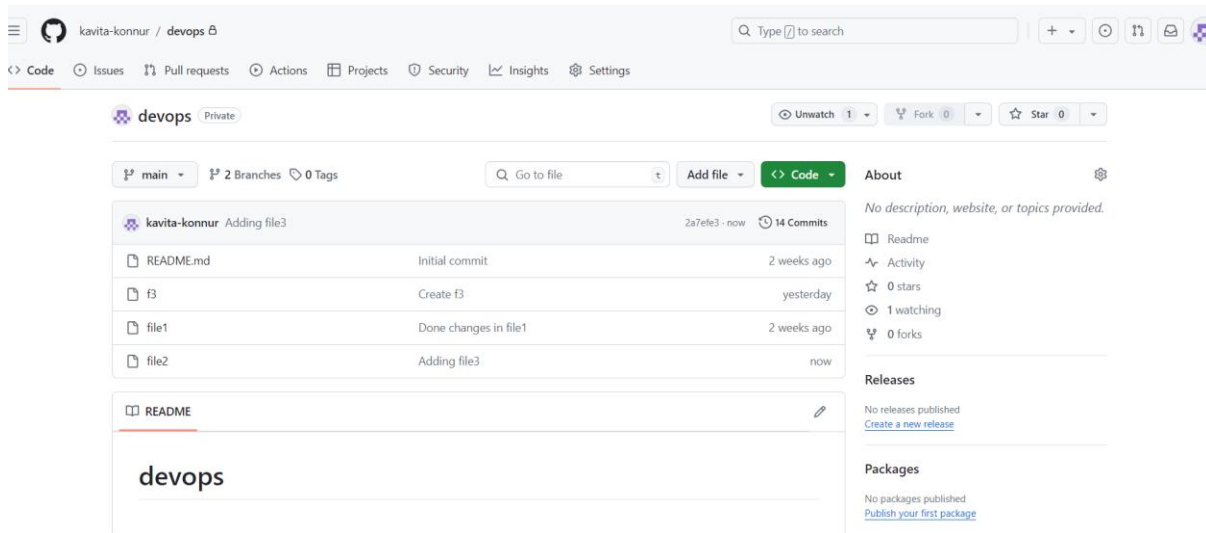
shank@shankar MINGW64 /d/Devops/Project/devops (main)
$ touch file2

shank@shankar MINGW64 /d/Devops/Project/devops (main)
$ git add file2

shank@shankar MINGW64 /d/Devops/Project/devops (main)
$ git commit -m "Adding file3"
(main 2a7efe3) Adding file3
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 file2

shank@shankar MINGW64 /d/Devops/Project/devops (main)
$ git push
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 562 bytes | 262.00 KiB/s, done.
Total 3 (delta 1), reused 1 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To github.com:kavita-konnur/devops.git
 96114b4..2a7efe3  main -> main
```

--- file2 has been added in central repo



44. Git fetch It is same as pull but the commits are not merged, it is in a temporary branch fetch head.

```
shank@shankar MINGW64 /d/Devops/Project/devops (main)
$ git fetch
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
Unpacking objects: 100% (3/3), 914 bytes | 1024 bytes/s, done.
From github.com:kavita-konnur/devops
 2a7efe3..7492a49  main       -> origin/main

shank@shankar MINGW64 /d/Devops/Project/devops (main)
$ ls
README.md  f3  file1  file2

shank@shankar MINGW64 /d/Devops/Project/devops (main)
$ git merge
Updating 2a7efe3..7492a49
Fast-forward
 f2 | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 f2

shank@shankar MINGW64 /d/Devops/Project/devops (main)
$ ls
README.md  f2  f3  file1  file2
```

45. git pull: the changes from the central has been directly merged with local repo

```
shank@shankar: MINGW64 /d/Devops/Project/devops (main)
$ ls
README.md f2 f3 file1 file2

shank@shankar: MINGW64 /d/Devops/Project/devops (main)
$ 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
Unpacking objects: 100% (3/3), 900 bytes | 5.00 KiB/s, done.
From github.com:kavita-konnur/devops
  7492a49..3637534  main    -> origin/main
Updating 7492a49..3637534
Fast-forward
 f5 | 1
 1 file changed, 1 insertion(+)
 create mode 100644 f5

shank@shankar: MINGW64 /d/Devops/Project/devops (main)
$ ls
README.md f2 f3 f5 file1 file2
```

46. Maven Installation

ec2-user@ip-172-31-83-197:~

```
[ec2-user@ip-172-31-83-197 ~]$ sudo yum install maven
```

```
gdox.noarch 0:1.12.1-10.amzn2
sisu-inject-bean.noarch 0:2.3.0-11.amzn2
slf4j.noarch 0:1.7.4-4.amzn2
xalan-j2.noarch 0:2.7.1-23.1.amzn2
xerces-j2.noarch 0:2.11.0-17.amzn2
xml-commons-resolver.noarch 0:1.2-15.amzn2
```

Complete!

```
[ec2-user@ip-172-31-83-197 ~]$ mvn -version
```

Apache Maven 3.0.5 (Red Hat 3.0.5-17)

Maven home: /usr/share/maven

Java version: 17.0.12, vendor: Amazon.com Inc.

Java home: /usr/lib/jvm/java-17-amazon-corretto.x86_64

Default locale: en_US, platform encoding: UTF-8

OS name: "linux", version: "5.10.220-209.867.amzn2.x86_64", arch: "amd64", family: "unix"

```
[ec2-user@ip-172-31-83-197 ~]$
```

47. Java Installation

ec2-user@ip-172-31-83-197:~

```
[ec2-user@ip-172-31-83-197 ~]$ sudo yum install java -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core | 3.6 kB 00:00:00
Package 1:java-17-amazon-corretto-17.0.12+7-1.amzn2.1.x86_64 already installed and latest version
Nothing to do
[ec2-user@ip-172-31-83-197 ~]$ java -version
openjdk version "17.0.12" 2024-07-16 LTS
OpenJDK Runtime Environment Corretto-17.0.12.7.1 (build 17.0.12+7-LTS)
OpenJDK 64-Bit Server VM Corretto-17.0.12.7.1 (build 17.0.12+7-LTS, mixed mode, sharing)
[ec2-user@ip-172-31-83-197 ~]$
```

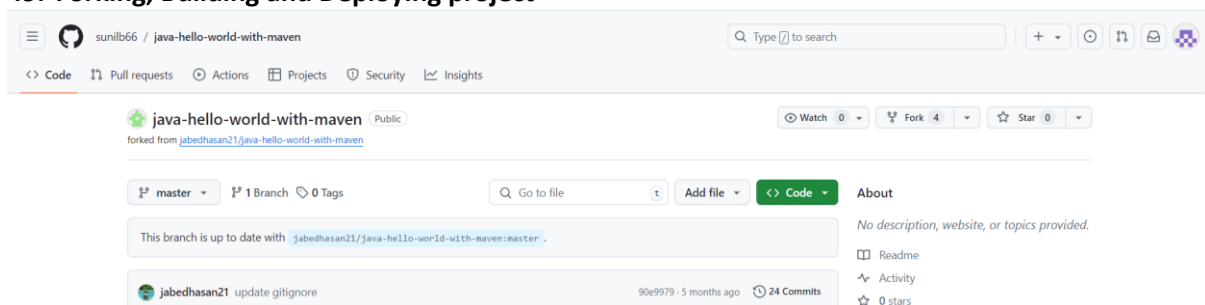
48. Git installation

ec2-user@ip-172-31-83-197:~

```
[ec2-user@ip-172-31-83-197 ~]$ sudo yum instal git
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
No such command: instal. Please use /bin/yum --help
[ec2-user@ip-172-31-83-197 ~]$ clear
```

```
ec2-user@ip-172-31-83-197:~  
[ec2-user@ip-172-31-83-197 ~]$ git --version  
git version 2.40.1  
[ec2-user@ip-172-31-83-197 ~]$
```

49. Forking, Building and Deploying project



a. Forking project

sunilb66 / java-hello-world-with-maven

Q Type [f] to search

+ -

<> Code

Pull requests

Actions

Projects

Security

Insights

java-hello-world-with-maven Public

Watch 0 Fork 4 Star 0

master 1 Branch 0 Tags

Go to file

Add file

<> Code

About

This branch is up to date with javedhasan21/java-hello-world-with-maven:master.

No description, website, or topics provided.

javedhasan21 update gitignore 90e9979 · 5 months ago 24 Commits

Readme

Activity

0 stars

Create a new fork

A fork is a copy of a repository. Forking a repository allows you to freely experiment with changes without affecting the original project. [View existing forks.](#)

Required fields are marked with an asterisk (*).

Owner *

Repository name *

kavita-konnur

/ java-hello-world-with-ma

java-hello-world-with-maven is available.

By default, forks are named the same as their upstream repository. You can customize the name to distinguish it further.

Description (optional)

☒ Copy the master branch only

Contribute back to sunilb66/java-hello-world-with-maven by adding your own branches. [Learn more.](#)

☐ You are creating a fork in your personal account.

Create fork

b. Cloning, building and deploying project.

```
[ec2-user@ip-172-31-83-197 project]$ git clone https://github.com/kavita-konnur/java-hello-world-with-maven.git  
Cloning into 'java-hello-world-with-maven'...  
remote: Enumerating objects: 81, done.  
remote: Total 81 (delta 0), reused 0 (delta 0), pack-reused 81  
Receiving objects: 100% (81/81), 14.96 KiB | 4.99 MiB/s, done.  
Resolving deltas: 100% (27/27), done.  
[ec2-user@ip-172-31-83-197 project]$ ls  
java-hello-world-with-maven  README.md  
[ec2-user@ip-172-31-83-197 project]$ cd java-hello-world-with-maven  
[ec2-user@ip-172-31-83-197 java-hello-world-with-maven]$ ls  
Jenkinsfile  pom.xml  README.md  src  
[ec2-user@ip-172-31-83-197 java-hello-world-with-maven]$ mvn clean install  
[INFO] Scanning for projects...  
[INFO]  
[INFO] -----  
[INFO] Building jb-hello-world-maven 0.2.0  
[INFO] -----  
Downloading: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-clean-plugin/2.4.1/maven-clean-plugin-2.4.1.pom  
Downloaded: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-clean-plugin/2.4.1/maven-clean-plugin-2.4.1.pom (5 KB at 8.9 KB/sec)  
Downloading: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-plugins/18/maven-plugins-18.pom  
Downloaded: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-plugins/18/maven-plugins-18.pom (13 KB at 325.5 KB/sec)  
Downloading: https://repo.maven.apache.org/maven2/org/apache/maven/maven-parent/16/maven-parent-16.pom  
Downloaded: https://repo.maven.apache.org/maven2/org/apache/maven/maven-parent/16/maven-parent-16.pom (23 KB at 668.4 KB/sec)
```

```
[ec2-user@ip-172-31-83-197 java-hello-world-with-maven]$ ls
dependency-reduced-pom.xml  Jenkinsfile  pom.xml  README.md  src  target
[ec2-user@ip-172-31-83-197 java-hello-world-with-maven]$ cd target
[ec2-user@ip-172-31-83-197 target]$ ls
classes  generated-sources  jb-hello-world-maven-0.2.0.jar  maven-archiver  original-jb-hello-world-maven-0.2.0.jar  surefire
```

```
[ec2-user@ip-172-31-83-197 target]$ cd ../
[ec2-user@ip-172-31-83-197 java-hello-world-with-maven]$ vi pom.xml
[ec2-user@ip-172-31-83-197 java-hello-world-with-maven]$ mvn clean validate
[INFO] Scanning for projects...
[INFO]
[INFO] -----
[INFO] Building jb-hello-world-maven 0.2.0
[INFO] -----
[INFO]
[INFO] --- maven-clean-plugin:2.4.1:clean (default-clean) @ jb-hello-world-maven ---
[INFO] Deleting /home/ec2-user/project/java-hello-world-with-maven/target
[INFO]
[INFO] BUILD SUCCESS
[INFO]
[INFO] -----
[INFO] Total time: 0.415s
[INFO] Finished at: Mon Jul 29 05:31:05 UTC 2024
[INFO] Final Memory: 5M/15M
[INFO]
[INFO]
[INFO]
```

```
[ec2-user@ip-172-31-83-197 java-hello-world-with-maven]$ mvn clean compile
[INFO] Scanning for projects...
[INFO]
[INFO] -----
[INFO] Building jb-hello-world-maven 0.2.0
[INFO] -----
[INFO]
[INFO] --- maven-clean-plugin:2.4.1:clean (default-clean) @ jb-hello-world-maven ---
[INFO]
[INFO] --- maven-resources-plugin:2.5:resources (default-resources) @ jb-hello-world-maven ---
[debug] execute contextualize
[WARNING] Using platform encoding (UTF-8 actually) to copy filtered resources, i.e. build is platform dependent!
[INFO] skip non existing resourceDirectory /home/ec2-user/project/java-hello-world-with-maven/src/main/resources
[INFO]
[INFO] --- maven-compiler-plugin:2.3.2:compile (default-compile) @ jb-hello-world-maven ---
[WARNING] File encoding has not been set, using platform encoding UTF-8, i.e. build is platform dependent!
[INFO] Compiling 2 source files to /home/ec2-user/project/java-hello-world-with-maven/target/classes
[INFO]
[INFO] BUILD SUCCESS
[INFO]
[INFO] -----
[INFO] Total time: 2.040s
[INFO] Finished at: Mon Jul 29 05:31:23 UTC 2024
[INFO] Final Memory: 4M/20M
[INFO]
[INFO]
[INFO]
```

```
[ec2-user@ip-172-31-83-197 java-hello-world-with-maven]$ mvn test
[INFO] Scanning for projects...
[INFO]
[INFO] -----
[INFO] Building jb-hello-world-maven 0.2.0
[INFO] -----
[INFO]
[INFO] --- maven-resources-plugin:2.5:resources (default-resources) @ jb-hello-world-maven ---
[debug] execute contextualize
[WARNING] Using platform encoding (UTF-8 actually) to copy filtered resources, i.e. build is platform dependent!
[INFO] skip non existing resourceDirectory /home/ec2-user/project/java-hello-world-with-maven/src/main/resources
[INFO]
[INFO] --- maven-compiler-plugin:2.3.2:compile (default-compile) @ jb-hello-world-maven ---
[INFO] Nothing to compile - all classes are up to date
[INFO]
[INFO] --- maven-resources-plugin:2.5:testResources (default-testResources) @ jb-hello-world-maven ---
[debug] execute contextualize
[WARNING] Using platform encoding (UTF-8 actually) to copy filtered resources, i.e. build is platform dependent!
[INFO] skip non existing resourceDirectory /home/ec2-user/project/java-hello-world-with-maven/src/test/resources
[INFO]
[INFO] --- maven-compiler-plugin:2.3.2:testCompile (default-testCompile) @ jb-hello-world-maven ---
[INFO] No sources to compile
[INFO]
[INFO] --- maven-surefire-plugin:2.10:test (default-test) @ jb-hello-world-maven ---
[INFO] No tests to run.
[INFO] Surefire report directory: /home/ec2-user/project/java-hello-world-with-maven/target/surefire-reports

T E S T S

Results :

Tests run: 0, Failures: 0, Errors: 0, Skipped: 0

[INFO]
[INFO] BUILD SUCCESS
[INFO]
[INFO] -----
[INFO] Total time: 1.726s
[INFO] Finished at: Mon Jul 29 05:35:24 UTC 2024
[INFO] Final Memory: 7M/17M
[INFO]
[INFO]
```

```

[ec2-user@ip-172-31-83-197 java-hello-world-with-maven]$ mvn package
[INFO] Scanning for projects...
[INFO]
[INFO] -----
[INFO] Building jb-hello-world-maven 0.2.0
[INFO] -----
[INFO]
[INFO] --- maven-resources-plugin:2.5:resources (default-resources) @ jb-hello-world-maven ---
[debug] execute contextualize
[WARNING] Using platform encoding (UTF-8 actually) to copy filtered resources, i.e. build is platform dependent!
[INFO] skip non existing resourceDirectory /home/ec2-user/project/java-hello-world-with-maven/src/main/resources
[INFO]
[INFO] --- maven-compiler-plugin:2.3.2:compile (default-compile) @ jb-hello-world-maven ---
[INFO] Nothing to compile - all classes are up to date
[INFO]
[INFO] --- maven-resources-plugin:2.5:testResources (default-testResources) @ jb-hello-world-maven ---
[debug] execute contextualize
[WARNING] Using platform encoding (UTF-8 actually) to copy filtered resources, i.e. build is platform dependent!
[INFO] skip non existing resourceDirectory /home/ec2-user/project/java-hello-world-with-maven/src/test/resources
[INFO]
[INFO] --- maven-compiler-plugin:2.3.2:testCompile (default-testCompile) @ jb-hello-world-maven ---
[INFO] No sources to compile
[INFO]
[INFO] --- maven-surefire-plugin:2.10:test (default-test) @ jb-hello-world-maven ---
[INFO] No tests to run.
[INFO] Surefire report directory: /home/ec2-user/project/java-hello-world-with-maven/target/surefire-reports

-----
T E S T S
-----

Results :

Tests run: 0, Failures: 0, Errors: 0, Skipped: 0

[INFO]
[INFO] --- maven-jar-plugin:2.3.2:jar (default-jar) @ jb-hello-world-maven ---
[INFO] Building jar: /home/ec2-user/project/java-hello-world-with-maven/target/jb-hello-world-maven-0.2.0.jar
[INFO]
[INFO] --- maven-shade-plugin:2.1:shade (default) @ jb-hello-world-maven ---
[INFO] Including joda-time:joda-time:jar:2.2 in the shaded jar.
[INFO] Replacing original artifact with shaded artifact.
[INFO] Replacing /home/ec2-user/project/java-hello-world-with-maven/target/jb-hello-world-maven-0.2.0.jar with /home/ec2-user/project/java-hello-world-with-maven/target/jb-hello-world-maven-0.2.0-shaded.jar
[INFO] Dependency-reduced POM written at: /home/ec2-user/project/java-hello-world-with-maven/dependency-reduced-pom.xml
[INFO] -----
[INFO] BUILD SUCCESS

```

```

[ec2-user@ip-172-31-83-197 java-hello-world-with-maven]$ mvn install
[INFO] Scanning for projects...
[INFO]
[INFO] -----
[INFO] Building jb-hello-world-maven 0.2.0
[INFO] -----
[INFO]
[INFO] Replacing /home/ec2-user/project/java-hello-world-with-maven/target/jb-hello-world-maven-0.2.0.jar with /home/ec2-user/project/java-hello-world-with-maven/target/jb-hello-world-maven-0.2.0-shaded.jar
[INFO] Dependency-reduced POM written at: /home/ec2-user/project/java-hello-world-with-maven/dependency-reduced-pom.xml
[INFO]
[INFO] --- maven-install-plugin:2.3.1:install (default-install) @ jb-hello-world-maven ---
[INFO] Installing /home/ec2-user/project/java-hello-world-with-maven/target/jb-hello-world-maven-0.2.0.jar to /home/ec2-user/.m2/repository/org/springframework/jb-hello-world-maven/0.2.0/jb-hello-world-maven-0.2.0.jar
[INFO] Installing /home/ec2-user/project/java-hello-world-with-maven/dependency-reduced-pom.xml to /home/ec2-user/.m2/repository/org/springframework/jb-hello-world-maven/0.2.0/jb-hello-world-maven-0.2.0.pom
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 3.201s
[INFO] Finished at: Mon Jul 29 05:37:35 UTC 2024
[INFO] Final Memory: 9M/22M
[INFO] -----

```

50. Accessing manager app, host manager and server in tomcat.

```

ubuntu@ip-172-31-41-221:~$ ls
addusers.sh  dir1          example.sh  file3          filesize.sh  hostreach.sh  p2.sh      test      wordcount
backup       dir3          factfor.sh  file_operator.sh  folder1      logscript.sh  project    tomcat    wordcount.sh
backup1.sh   diskspace.sh  file1.sh    file_upper_lower.sh  folder2      p1.sh         rxwpermission.sh  tomcat.sh

ubuntu@ip-172-31-41-221:~$ cd tomcat/webapps/manager/META-INF$ vi context.xml
ubuntu@ip-172-31-41-221:~/tomcat/webapps/manager/META-INF$ cd ../../
ubuntu@ip-172-31-41-221:~/tomcat/webapps$ ls
ROOT  docs  examples  host-manager  manager
ubuntu@ip-172-31-41-221:~/tomcat/webapps$ cd host-manager/META-INF
ubuntu@ip-172-31-41-221:~/tomcat/webapps/host-manager/META-INF$ vi context.xml
ubuntu@ip-172-31-41-221:~/tomcat/webapps/host-manager/META-INF$ cd ../../
ubuntu@ip-172-31-41-221:~/tomcat$ cd conf
ubuntu@ip-172-31-41-221:~/tomcat/conf$ vi tomcat-users.xml
ubuntu@ip-172-31-41-221:~/tomcat/conf$ cd ../
ubuntu@ip-172-31-41-221:~/tomcat$ cd bin
ubuntu@ip-172-31-41-221:~/tomcat/bin$ sh shutdown.sh
Using CATALINA_BASE:   /home/ubuntu/tomcat
Using CATALINA_HOME:   /home/ubuntu/tomcat
Using CATALINA_TMPDIR: /home/ubuntu/tomcat/temp
Using JRE_HOME:        /usr
Using CLASSPATH:        /home/ubuntu/tomcat/bin/bootstrap.jar:/home/ubuntu/tomcat/bin/tomcat-juli.jar
Using CATALINA_OPTS:
NOTE: Picked up JDK_JAVA_OPTIONS:  --add-opens=java.base/java.lang=ALL-UNNAMED --add-opens=java.base/java.io=ALL-UNNAMED --add-opens=java.base/java.util=ALL-UNNAMED --add-opens=java.base/java.util.concurrent=ALL-UNNAMED --add-opens=java.rmi/sun.rmi.transport=ALL-UNNAMED
ubuntu@ip-172-31-41-221:~/tomcat/bin$ sh startup.sh
Using CATALINA_BASE:   /home/ubuntu/tomcat
Using CATALINA_HOME:   /home/ubuntu/tomcat
Using CATALINA_TMPDIR: /home/ubuntu/tomcat/temp
Using JRE_HOME:        /usr
Using CLASSPATH:        /home/ubuntu/tomcat/bin/bootstrap.jar:/home/ubuntu/tomcat/bin/tomcat-juli.jar
Using CATALINA_OPTS:
Tomcat started.
ubuntu@ip-172-31-41-221:~/tomcat/bin$

```



Server Status

Manager

[List Applications](#) [HTML Manager Help](#) [Manager Help](#) [Complete Server Status](#)

Server Information

Tomcat Version	JVM Version	JVM Vendor	OS Name	OS Version	OS Architecture	Hostname	IP Address
Apache Tomcat/9.0.90	21.0.4+7-Ubuntu-1ubuntu224.04	Ubuntu	Linux	6.8.0-1012-aws	amd64	21	1

JVM

Free Memory: 7.24 MiB Total Memory: 26.75 MiB Max Memory: 232.00 MiB

Memory Pool	Type	Initial	Total	Maximum	Used
Eden Space	Heap memory	4.31 MiB	7.50 MiB	64.00 MiB	4.60 MiB (7%)
Survivor Space	Heap memory	0.50 MiB	0.87 MiB	8.00 MiB	0.63 MiB (7%)
Tenured Gen	Heap memory	10.68 MiB	18.37 MiB	160.00 MiB	14.26 MiB (8%)
CodeHeap 'non-nmethods'	Non-heap memory	2.43 MiB	2.43 MiB	5.55 MiB	1.24 MiB (22%)
CodeHeap 'non-profiled nmethods'	Non-heap memory	2.43 MiB	2.43 MiB	117.22 MiB	1.34 MiB (1%)
CodeHeap 'profiled nmethods'	Non-heap memory	2.43 MiB	7.62 MiB	117.21 MiB	7.57 MiB (6%)
Compressed Class Space	Non-heap memory	0.00 MiB	2.56 MiB	1024.00 MiB	2.37 MiB (0%)
Metaspace	Non-heap memory	0.00 MiB	25.56 MiB	-0.00 MiB	25.19 MiB

"http-nio-8080"

Max threads: 200 Current thread count: 10 Current threads busy: 1 Keep alive sockets count: 1
Max processing time: 1695 ms Processing time: 2.285 s Request count: 11 Error count: 1 Bytes received: 0.00 MiB Bytes sent: 0.08 MiB



Tomcat Web Application Manager

Message: OK

Manager

[List Applications](#) [HTML Manager Help](#) [Manager Help](#) [Server Status](#)

Applications

Path	Version	Display Name	Running	Sessions	Commands
	None specified	Welcome to Tomcat	true	0	Start Stop Reload Undeploy Expire sessions with idle <input type="text" value="30"/> minutes
docs	None specified	Tomcat Documentation	true	0	Start Stop Reload Undeploy Expire sessions with idle <input type="text" value="30"/> minutes
examples	None specified	Servlet and JSP Examples	true	0	Start Stop Reload Undeploy Expire sessions with idle <input type="text" value="30"/> minutes
host-manager	None specified	Tomcat Host Manager Application	true	1	Start Stop Reload Undeploy Expire sessions with idle <input type="text" value="30"/> minutes
manager	None specified	Tomcat Manager Application	true	2	Start Stop Reload Undeploy Expire sessions with idle <input type="text" value="30"/> minutes

Deploy

Deploy directory or WAR file located on server



Tomcat Virtual Host Manager

Message: OK

Host Manager

[List Virtual Hosts](#) [HTML Host Manager Help](#) [Host Manager Help](#) [Server Status](#)

Host name

Host name	Host aliases	Commands
localhost		Host Manager installed - commands disabled

Add Virtual Host

Host

Name:

Aliases:

App base:

AutoDeploy ☒

DeployOnStartup ☒

DeployXML ☒

UnpackWARs ☒

Manager App ☒

CopyXML ☐

[Add](#)