



PARSHVANATH CHARITABLE TRUST'S

A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



Academic Year: 2024-25

Semester: V

Class / Branch: TEIT

Subject: DevOps Lab

Name of Instructor: Ms. Sujata Oak

Experiment No. 2

Aim: To perform installation of Git and work on local and remote Git repositories.

GIT is a Version Control System (VCS) (aka Revision Control System (RCS), Source Code Manager (SCM)). A VCS serves as a Repository (or repo) of program codes, including all the historical revisions. It records changes to files at so-called commits in a log so that you can recall any file at any commit point.

To issue a command, start a "Terminal" (for Ubuntu/Mac) or "Git Bash" (for Windows):

```
$ git <command> <arguments>
```

The commonly-used commands are:

1. `init, clone, config`: for starting a Git-managed project.
2. `add, mv, rm`: for staging file changes.
3. `commit, rebase, reset, tag`:
4. `status, log, diff, grep, show`: show status
5. `checkout, branch, merge, push, fetch, pull`

Getting Started with Local Repo

There are 2 ways to start a Git-managed project:

1. Starting your own project;
2. Cloning an existing project from a GIT host.

Git uses two stages to commit file changes:

1. "git add <file>" to stage file changes into the staging area, and
2. "git commit" to commit ALL the file changes in the staging area to the local repo.



PARSHVANATH CHARITABLE TRUST'S

A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



You need to setup Git on your local machine, as follows:

```
devasc@labvm:~$ cd Desktop/
devasc@labvm:~/Desktop$ cd git-sujata/
devasc@labvm:~/Desktop/git-sujata$ ls
devasc@labvm:~/Desktop/git-sujata$ pwd
/home/devasc/Desktop/git-sujata
```

To update all local package index for ubuntu:

```
$sudo su
#apt update
```

```
devasc@labvm:~/Desktop/git-sujata$ sudo su
root@labvm:/home/devasc/Desktop/git-sujata# apt update
Get:1 https://download.docker.com/linux/ubuntu focal InRelease [57.7 kB]
Ign:2 https://pkg.jenkins.io/debian-stable binary/ InRelease
Hit:3 https://pkg.jenkins.io/debian-stable binary/ Release
Hit:5 http://archive.ubuntu.com/ubuntu focal InRelease
Hit:6 http://archive.ubuntu.com/ubuntu focal-updates InRelease
Hit:7 http://archive.ubuntu.com/ubuntu focal-backports InRelease
Hit:8 http://security.ubuntu.com/ubuntu focal-security InRelease
Fetched 57.7 kB in 1s (67.1 kB/s)
Reading package lists... Done
```

Step1: Download & Install:

```
#apt-get install git
```

```
root@labvm:/home/devasc/Desktop/git-sujata# apt-get install git
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  bridge-utils ubuntu-fan
Use 'sudo apt autoremove' to remove them.
Suggested packages:
```

To check version of git:

```
#git --version
```



PARSHVANATH CHARITABLE TRUST'S

A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
git --version
root@labvm:/home/devasc/Desktop/git-sujata# git --version
git version 2.25.1
```

Step 2: Customize and configure your Git Account:

```
#git config --list
#git config --global --unset user.name
#git config --global --unset user.email

#git config --global user.name "sujataoak799"
```

```
root@labvm:/home/devasc/Desktop/git-sujata# git config --global user.name "sujataoak799"
"
```

```
#git config --global user.email sujataoak2021@gmail.com
```

```
root@labvm:/home/devasc/Desktop/git-sujata# git config --global user.email sujataoak2021@gmail.com
```

To List Global configuration for Git:

```
#git config --list
root@labvm:/home/devasc/Desktop/git-sujata# git config --list
user.name="sujataoak799"
user.email=sujataoak2021@gmail.com
```

Step . 3

To Integrate Git account with Github:

Goto www.github.com

Sign-in to your account

Create a Repository



PARSHVANATH CHARITABLE TRUST'S

A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



Create a new repository Try the new experience

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository](#).

Required fields are marked with an asterisk (*).

Repository template

No template

Start your repository with a template repository's contents.

Owner * sujataoak799 / **Repository name *** gitdemo11072025
gitdemo11072025 is available.

Great repository names are short and memorable. Need inspiration? How about [vigilant-octo-guide](#) ?

Description (optional)

setting a git demo to collaborate from local to remote and viceversa.

Public Anyone on the internet can see this repository. You choose who can commit.

Private You choose who can see and commit to this repository.

Initialize this repository with:

Add a README file This is where you can write a long description for your project. [Learn more about READMEs](#).

Add .gitignore

gitignore template: None

Choose which files not to track from a list of templates. [Learn more about ignoring files](#).

Choose a license

License: None

A license tells others what they can and can't do with your code. [Learn more about licenses](#).

This will set `main` as the default branch. Change the default name in your [settings](#).

Click on Create Repository



The screenshot shows a GitHub repository page. At the top, the URL is `github.com/sujataoak799/gitdemo11072025`. Below the URL, the repository name is `gitdemo11072025` and it is marked as `Public`. The repository has one commit by user `sujataoak799`, which is an initial commit. The README file contains the text: **gitdemo11072025** setting a git demoto collaborate from local to remote and viceversa.

Step . 3:

- In local repository initiate the directory to make it a git repository (.git file must be added inside that folder after initiation)

```
#git init
```

```
root@labvm:/home/devasc/Desktop/git-sujata# git init
Initialized empty Git repository in /home/devasc/Desktop/git-sujata/.git/
```

```
root@labvm:/home/devasc/Desktop/git-sujata# ls -altr
total 12
drwxr-xr-x 7 devasc devasc 4096 Jul 10 16:40 ..
drwxrwxr-x 3 devasc devasc 4096 Jul 10 17:00 .
drwxr-xr-x 7 root root 4096 Jul 10 17:00 .git
```

Now Clone the Remote Repository into your Local Repository ie; Ubuntu Operating System:

Compiled by Prof. Sujata Oak

Department of Information Technology | AP~~S~~IT



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
# git clone https://github.com/sujataoak799/gitdemo11072025.git
```

```
root@labvm:/home/devasc/Desktop/git-sujata# git clone https://github.com/sujataoak799/gitdemo11072025.git
Cloning into 'gitdemo11072025'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (3/3), 925 bytes | 925.00 KiB/s, done.
```

```
root@labvm:/home/devasc/Desktop/git-sujata# ls -ltr
total 4
drwxr-xr-x 3 root root 4096 Jul 10 17:05 gitdemo11072025
```

```
root@labvm:/home/devasc/Desktop/git-sujata# cd gitdemo11072025/
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# ls
README.md
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# cat README.md
# gitdemo11072025
setting a git demoto collaborate from local to remote and viceversa
```

Now Create a python file factorial.py in Local Repository:

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# gedit factorial.py
```

```
num = int(input("Enter a number: "))
factorial = 1

if num < 0:
    print("Factorial does not exist for negative numbers")
elif num == 0:
    print("The factorial of 0 is 1")
else:
    for i in range(1, num + 1):
        factorial = factorial * i
    print("The factorial of", num, "is", factorial)
```



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
# factorial of given number
def factorial(n):
    if n < 0:
        return 0
    elif n == 0 or n == 1:
        return 1
    else:
        fact = 1
        while(n > 1):
            fact *= n
            n -= 1
        return fact

# Driver Code
num = 5
print("Factorial of",num,"is",
factorial(num))
```

Step 4: Now we will apply some git commands to add, commit and push factorial.py file to remote repository: Firstly To View Uncommitted File:

#git status

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git status
On branch main
Your branch is up to date with 'origin/main'.

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

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

The file factorial.py shown in red color is untracked (it means not tracked by git till now). So to add to the git versioning : git add <filename>

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git add factorial.py
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025#
```



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git status
On branch main
Your branch is up to date with 'origin/main'.

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

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

Once file is added to git, the color changes to green color. But it says changes to be committed

Step 5: To start first commit

```
sujat@Sujata-Oak MINGW64 /d/mygithublearning/devopslearn123 (main)
$ git commit -m "my commit file.txt"
Author identity unknown

*** Please tell me who you are.

Run

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

to set your account's default identity.
Omit --global to set the identity only in this repository.

fatal: unable to auto-detect email address (got 'sujat@Sujata-Oak.(none)')
```

git commit -m “First Python File”

```
error: pathspec 'file' did not match any file(s) known to git
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git commit -m "first file added"
[main cc225cc] first file added
 1 file changed, 11 insertions(+)
 create mode 100644 factorial.py
```



PARSHVANATH CHARITABLE TRUST'S

A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
root@labvm:/home/devasc/Desktop/git-sujata/gitm11072025# git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

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

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

Step 6: To Push your changes from local repository to Github/remote Repository:

git push origin master or git push origin main

#git push origin main

Enter username and password

```
root@labvm:/home/devasc/Desktop/git-sujata/gitm11072025# git push origin main
Username for 'https://github.com': sujataoak799
Password for 'https://sujataoak799@github.com':
remote: Support for password authentication was removed on August 13, 2021.
remote: Please see https://docs.github.com/get-started/getting-started-with-git/about-remote-repositories#cloning-with-https-urls for information on currently recommended modes of authentication.
fatal: Authentication failed for 'https://github.com/sujataoak799/gitm11072025.git/'
```

It says support for password authentication was removed on August 13, 2021.
So we need a new kind of method:

Goto your Github Account → User Profile → Settings → Developer
Settings → Personal Access Token → Token(classic) → Generate New token → Generate New Token(Classic)



New personal access token (classic)

Personal access tokens (classic) function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to authenticate to the API over Basic Authentication.

Note

Token for git practical

Expiration

No expiration

Select scopes

Scopes define the access for personal tokens. [Read more about OAuth scopes.](#)

Scope	Description
repo	Full control of private repositories
repo:status	Access commit status
repo_deployment	Access deployment status
public_repo	Access public repositories
repo:invite	Access repository invitations

Click on Generate Token

Copy the token and paste it some location:

#git remote set-url origin https://tokenhere@github.com/user_name/repo_name

#git remote set-url origin

https://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@github.com/sujataoak799/gitdemo11072025

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git remote set-url origin https://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@github.com/sujataoak799/gitdemo11072025
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025#
```

Again try to Push your changes to Github Repository:

#git push origin main



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git push origin main
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 2 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 453 bytes | 453.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/sujataoak799/gitdemo11072025
  73e6d61..cc225cc  main -> main
```

This time it is successfully implemented.

Goto Remote Repository and see the file factorial.py

github.com/sujataoak799/gitdemo11072025

sujataoak799 / gitdemo11072025

Type to search

Code Issues Pull requests Actions Projects Wiki Security

gitdemo11072025 Public

Pin Watch 0

main Go to file + <> Code

sujataoak799 first file added cc225cc · 30 minutes ago 2 Commits

README.md Initial commit 1 hour ago

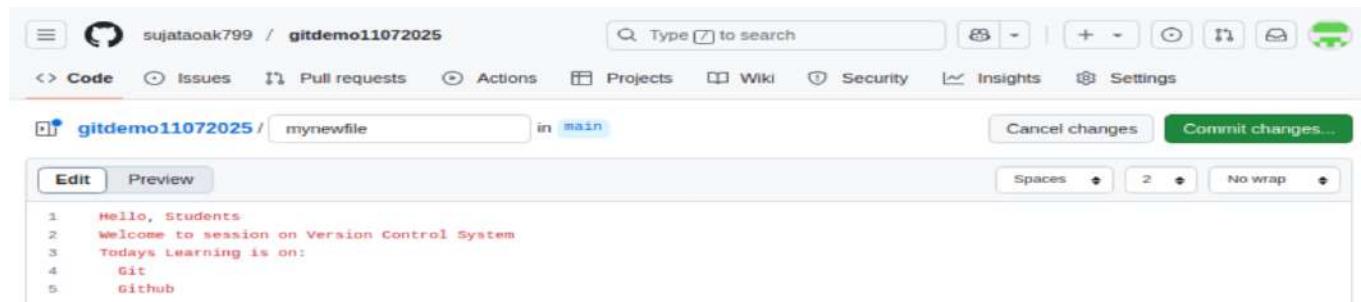
README first file added 30 minutes ago

gitdemo11072025

setting a git demoto collaborate from local to remote and viceversa



Now I create a new file on my github page: Add file->create new file→Commit changes

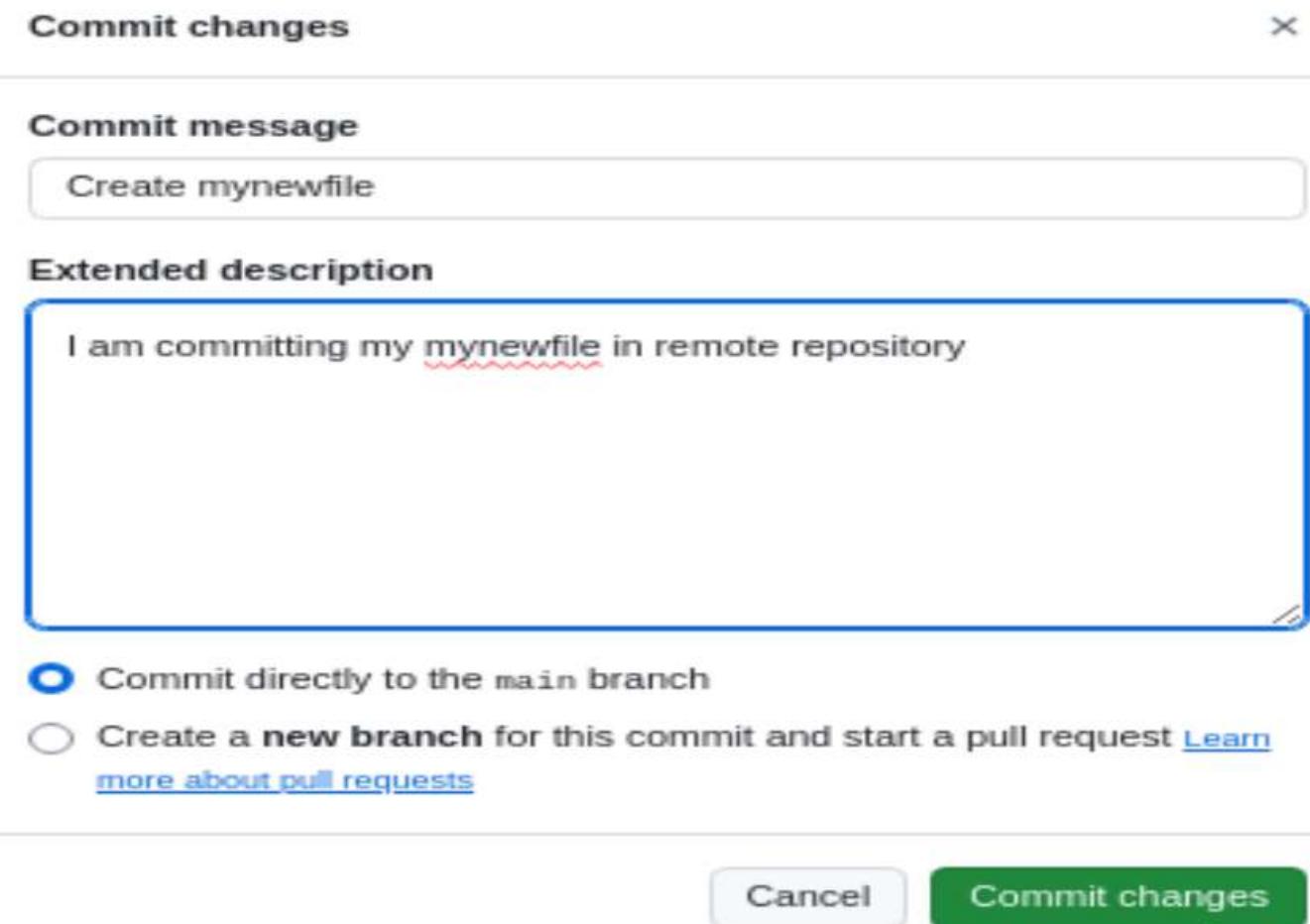


The screenshot shows a GitHub repository interface. At the top, it says "sujataoak799 / gitdemo11072025". Below that is a navigation bar with links for Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. In the main area, there's a search bar with "Type ⌘ to search" and a file list. One file, "mynewfile", is shown with its content:

```
1 Hello, Students
2 Welcome to session on Version Control System
3 Todays Learning is on:
4   Git
5   Github
```

At the bottom right of the code editor, there are buttons for "Cancel changes" and "Commit changes...".

Click on Commit Changes



The screenshot shows the "Commit changes" dialog box. It has a title bar "Commit changes" and a close button "X".

Commit message
Create mynewfile

Extended description
I am committing my mynewfile in remote repository

Commit directly to the main branch
 Create a **new branch** for this commit and start a pull request [Learn more about pull requests](#)

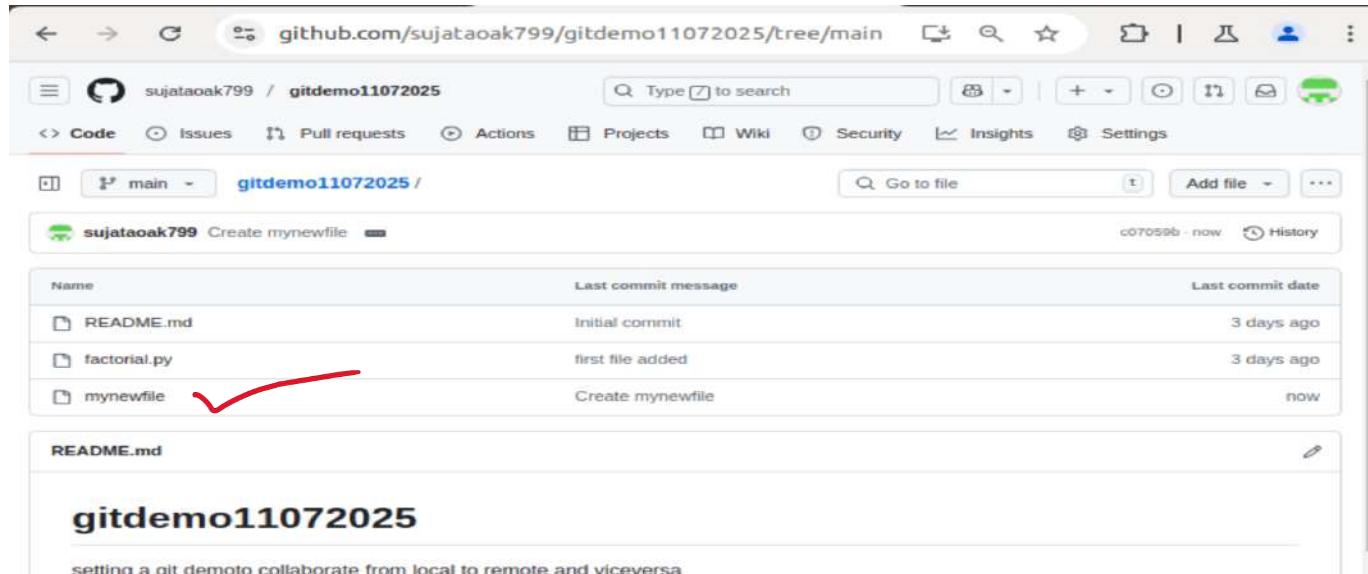
Cancel Commit changes

Click on Commit changes

Compiled by Prof. Sujata Oak



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY
Department of Information Technology
(NBA Accredited)



sujataoak799 / gitdemo11072025

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

main / gitdemo11072025 /

sujataoak799 Create mynewfile c07059b · now History

Name	Last commit message	Last commit date
README.md	Initial commit	3 days ago
factorial.py	first file added	3 days ago
mynewfile	Create mynewfile	now

README.md

gitdemo11072025

setting a git demoto collaborate from local to remote and viceversa

But this new file is not reflected in my local repository

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# ls -ltr
total 8
-rw-r--r-- 1 root root 299 Jul 10 17:22 name.html
-rw-r--r-- 1 root root 299 Jul 10 17:28 factorial.py
```

#git pull

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git pull
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (3/3), 1.06 KiB | 1.06 MiB/s, done.
From https://github.com/sujataoak799/gitdemo11072025
 cc225cc..c07059b main      -> origin/main
Merge made by the 'recursive' strategy.
 mynewfile | 5 ++++++
 1 file changed, 5 insertions(+)
 create mode 100644 mynewfile
```

#ls -ltr

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# ls -ltr
total 12
-rw-r--r-- 1 root root 299 Jul 10 17:22 name.html
-rw-r--r-- 1 root root 299 Jul 10 17:28 factorial.py
-rw-r--r-- 1 root root 99 Jul 13 09:59 mynewfile
```



```
#cat mynewfile
```

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# cat mynewfile
Hello, Students
Welcome to session on Version Control System
Todays Learning is on:
  Git
  Github
```

Step7 : To see the logs in oneline like username, email -id, date, time of creation.

```
#git log
```

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git log
commit 650568337cdc24cb0792f83479739de39113efa0 (HEAD -> main)
Merge: c62d707 c07059b
Author: "sujataoak799" <sujataoak2021@gmail.com>
Date:   Sun Jul 13 09:59:59 2025 +0000

  Merge branch 'main' of https://github.com/sujataoak799/gitdemo11072025 into
main

commit c07059b6d27720f3f9af46e03a8b63df5bb16218 (origin/main, origin/HEAD)
Author: sujataoak799 <79905110+sujataoak799@users.noreply.github.com>
Date:   Sun Jul 13 09:56:31 2025 +0000

  Create mynewfile

  I am committing my mynewfile in remote repository

commit c62d7072911f3f6a0bd0c8b1d4f3217ce086de58
Author: "sujataoak799" <sujataoak2021@gmail.com>
Date:   Thu Jul 10 18:25:44 2025 +0000
```

To Print list of all commits:

```
#git log --oneline
```

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git log --oneline
6505683 (HEAD -> main) Merge branch 'main' of https://github.com/sujataoak799/gitdemo11072025 into main
c07059b (origin/main, origin/HEAD) Create mynewfile
c62d707 Revert "Initial commit"
cc225cc first file added
73e6d61 Initial commit
```

NOTE: Head: The last commit happen on your system

Step 8: To show repository id and other detail or want to know about a particular commit

```
#git show <commit-id>
```



PARSHVANATH CHARITABLE TRUST'S

A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git show cc225cc
commit cc225ccfcfd548eea4d1052a7d79537f4a850092e
Author: "sujataoak799" <sujataoak2021@gmail.com>
Date:   Thu Jul 10 17:36:17 2025 +0000

    first file added

diff --git a/factorial.py b/factorial.py
new file mode 100644
index 0000000..87338e7
--- /dev/null
+++ b/factorial.py
@@ -0,0 +1,11 @@
+num = int(input("Enter a number: "))
+factorial = 1
+
+if num < 0:
+    print("Factorial does not exist for negative numbers")
+elif num == 0:
+    print("The factorial of 0 is 1")
+else:
+    for i in range(1, num + 1):
+        factorial = factorial * i
```

Step 09: To see the difference in the content of file between first and second commit.

```
#git diff <latest commit-id> <commit-id>
```

I will check what is the difference in the file created : mynewfile

```
#cat mynewfile
```

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# cat mynewfile
Hello, Students
Welcome to session on Version Control System
Todays Learning is on:
  Git
  Github
```

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# gedit mynewfile
```



PARSHVANATH CHARITABLE TRUST'S

A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
mynewfile
/home/devasc/Desktop/git-sujata/gitdemo11072025
1 Hello, Students
2 Welcome to session on Version Control System
3 Todays Learning is on:
4 Git
5 Github
6
7 We Will also learn about branching in git
8 Various Commands in branching are:
9 How to create a branch?
10 How to checkout a branch?
11 How to Delete a branch?
```

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git status
On branch main
Your branch is ahead of 'origin/main' by 2 commits.
  (use "git push" to publish your local commits)

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:   mynewfile

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

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

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git add .
[master a4f7201] Add info on Branching
```

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git status
On branch main
Your branch is ahead of 'origin/main' by 2 commits.
  (use "git push" to publish your local commits)

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   mynewfile
    new file:   name.html
```

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git commit -m "Added Info on Branching"
[master a4f7201] Added Info on Branching
  2 files changed, 17 insertions(+)
  create mode 100644 name.html
```



PARSHVANATH CHARITABLE TRUST'S

A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git log --oneline
a4f7201 (HEAD -> main) Added Info on Branching
6505683 Merge branch 'main' of https://github.com/sujataoak799/gitdemo11072025 into main
c07059b (origin/main, origin/HEAD) Create mynewfile
c62d707 Revert "Initial commit"
cc225cc first file added
73e6d61 Initial commit
```

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git diff a4f7201 c07059b
diff --git a/README.md b/README.md
new file mode 100644
index 0000000..c0beeada
--- /dev/null
+++ b/README.md
@@ -0,0 +1,2 @@
+#
+gitdemo11072025
+setting a git demoto collaborate from local to remote and viceversa
diff --git a/mynewfile b/mynewfile
index 45facd4..d4610b0 100644
--- a/mynewfile
+++ b/mynewfile
@@ -3,9 +3,3 @@ Welcome to session on Version Control System
Todays Learning is on:
Git
Github
-
-We WILL also learn about branching in git
-Various Commands in branching are:
-How to create a branch?
-How to checkout a branch?
```

Step 10: Creating a latter commit and reverting back to see the initial/original content

```
git revert<secondcommitID>
```

```
# git revert 73e6d
```



PARSHVANATH CHARITABLE TRUST'S

A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
File Edit View Search Terminal Help
/home/devasc/Desktop/git-sujata/gitdemo11072025/.git/COMMIT EDITMSG
Revert "Initial commit"

This reverts commit 73e6d6181e3557c88c4f12b9cfc1fd76f36cf345.

# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# On branch main
# Your branch is up to date with 'origin/main'.
#
# Changes to be committed:
#       deleted:    README.md
#
# Untracked files:
#       name.html
#
```

Save it

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git revert 73e6d
Removing README.md
[main c62d707] Revert "Initial commit"
 1 file changed, 2 deletions(-)
 delete mode 100644 README.md
```

TASK: push the changes done in local repository of mynewfile into remote repository

Conclusion:

In this experiment, we understood the use case of Version Control System, its benefits in real time scenario which provides a application of reverting the changes when people are in working in a collaborating environment. Different commands were used for the same such as revert (by using its id), diff for displaying the changes between the initial and latter texts.



PARSHVANATH CHARITABLE TRUST'S

A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



Academic Year: 2025-26

Semester: V

Class / Branch: TEIT

Subject: DevOps Lab

Name of Instructor: Ms. Sujata Oak

Experiment No. 3

Aim: To understand and perform version control system / source code management using Git Branch.

Theory:

Branching means creating a separate line of work in your project so you can make changes without affecting the main version.

It's like making a copy of your project to try new things (like adding a feature or fixing a bug), and then later you can combine your changes back into the main project.

When a repository is created, the files are automatically put in a branch called **main**. Whenever possible it is recommended to use branches rather than directly updating the main branch. Branching is used so that you can make changes in another area without affecting the main branch. This is done to help prevent accidental updates that might overwrite existing code.

In this lab, students will implement:

- **create a new branch,**
- **checkout the branch,**
- **make changes in the branch,**
- **stage**
- **commit** the branch
- **merge** the branch changes to the main branch, and
- **delete** the branch.

Implementation:

STEP1:_Goto your working directory

Compiled By Prof. Sujata Oak

Department of Information Technology | AP~~S~~IT



PARSHVANATH CHARITABLE TRUST'S

A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
devasc@labvm:~$ cd Desktop/
devasc@labvm:~/Desktop$ cd git-sujata/
devasc@labvm:~/Desktop/git-sujata$ ls
gitdemo11072025
devasc@labvm:~/Desktop/git-sujata$ sudo su
root@labvm:/home/devasc/Desktop/git-sujata# pwd
/home/devasc/Desktop/git-sujata
root@labvm:/home/devasc/Desktop/git-sujata# █
```

List the files in your working directory:

```
root@labvm:/home/devasc/Desktop/git-sujata# ls -altr
total 16
drwxr-xr-x 7 root    root    4096 Jul 10 17:00 .git
drwxrwxr-x 4 devasc  devasc  4096 Jul 10 17:05 .
drwxr-xr-x 7 devasc  devasc  4096 Jul 10 17:58 ..
drwxr-xr-x 3 root    root    4096 Jul 13 10:21 gitdemo11072025
root@labvm:/home/devasc/Desktop/git-sujata# █
```

Change the directory to your git remote repository:

```
root@labvm:/home/devasc/Desktop/git-sujata# cd gitdemo11072025/
root@labvm:/home/devasc/Desktop/git-sujata/gitm11072025# ls
factorial.py  mynewfile  name.html
```

Step2: Verify Current branch : Use **git branch** command without a branch-name to display all the branches for this repository. Git branch command tells you how many branches you have and you are working on which branch(*)

The "*" next to the main branch indicates that this is the current branch

```
root@labvm:/home/devasc/Desktop/git-sujata/gitm11072025# git branch
* main
```

Step3: Create a new branch feature: **Feature Branch:** Allows you to develop features, fix bugs, or safely experiment with new ideas in a contained area of your repository

Create a new branch called **featureA** using the **git branch <branch-name>**

Compiled By Prof. Sujata Oak

Department of Information Technology | AP~~S~~IT



PARSHVANATH CHARITABLE TRUST'S

A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git branch featureA
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git branch
  featureA
* main
```

Step 4: Checkout the new branch

Use the **git checkout <branch-name>** command to switch to the *feature* branch.

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git checkout featureA
Switched to branch 'featureA'
```

Step 5: Verify the current branch:

- Verify you have switched to the *feature* branch using the **git branch** command. Note the "*" next to the *feature* branch. This is now the *working branch*.

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git branch
* featureA
  main
```

Create a file add.py

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# touch add.py
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# gedit add.py
```

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# cat add.py
# WAP to add two numbers in python???
# Adding two numbers
a = 5
b = 3
sum = a + b

print("The sum is:", sum)
```

Step 6: Stage the modified file in the feature branch

- Stage the updated file to the current *feature* branch.

```
#git status
```



PARSHVANATH CHARITABLE TRUST'S

A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git status
On branch featureA
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    add.py

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

git add add.py

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git add add.py
```

B] Use the **git status** command and notice the modified file **add.py** is staged in the *feature* branch
#git status

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git status
On branch featureA
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   add.py
```

Step 7: Commit the staged file in the feature branch

A] Commit the staged file using the **git commit** command. Notice the new commit ID and your message.

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git commit -m "Added add.py
file for commit on 18072025"
[featureA 62e5df1] Added add.py file for commit on 18072025
 1 file changed, 7 insertions(+)
 create mode 100644 add.py
```

Step 8: Use the **git log** command to show all commits including the commit you just did to the *feature* branch.
The prior commit was done within the *main* branch.



PARSHVANATH CHARITABLE TRUST'S

A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git log
commit 62e5df132fc714e7b55be031f76536f5b1f5163a (HEAD -> featureA)
Author: "sujataoak799" <sujataoak2021@gmail.com>
Date:   Fri Jul 18 09:00:36 2025 +0000

    Added add.py file for commit on 18072025

commit a4f720130f948281bdfc5950541294bec3238838 (main)
Author: "sujataoak799" <sujataoak2021@gmail.com>
Date:   Sun Jul 13 10:23:10 2025 +0000

    Added Info on Branching

commit 650568337cdc24cb0792f83479739de39113efa0
Merge: c62d707 c07059b
Author: "sujataoak799" <sujataoak2021@gmail.com>
Date:   Sun Jul 13 09:59:59 2025 +0000

    Merge branch 'main' of https://github.com/sujataoak799/gitdemo11072025 into main

commit c07059b6d27720f3f9af46e03a8b63df5bb16218 (origin/main, origin/HEAD)
Author: sujataoak799 <79905110+sujataoak799@users.noreply.github.com>
Date:   Sun Jul 13 09:56:31 2025 +0000

    Create mynewfile

    I am committing my mynewfile in remote repository
```

Press a key **q** to exit from git log screen

Step9: How to push a **featureA** branch from local repository to remote repository?
#git push --set-upstream origin featureA

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git push --set-upstream origin featureA
Enumerating objects: 13, done.
Counting objects: 100% (13/13), done.
Delta compression using up to 2 threads
Compressing objects: 100% (9/9), done.
Writing objects: 100% (10/10), 1.35 KiB | 1.35 MiB/s, done.
Total 10 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), done.
remote:
remote: Create a pull request for 'featureA' on GitHub by visiting:
remote:     https://github.com/sujataoak799/gitdemo11072025/pull/new/featureA
remote:
To https://github.com/sujataoak799/gitdemo11072025
 * [new branch]      featureA -> featureA
Branch 'featureA' set up to track remote branch 'featureA' from 'origin'.
```

Goto github.com account → Refresh the page



gitdemo11072025 / sujataoak799

Type to search

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

gitdemo11072025 Public

Pin Watch Fork

featureA had recent pushes 3 minutes ago

Compare & pull request

main

Go to file

Activity 3 Commits

0 stars 0 watching 0 forks

Releases

No releases published Create a new release

File	Commit Message	Date
README.md	Initial commit	last week
factorial.py	first file added	last week
mynewfile	Create mynewfile	5 days ago
README		

Select featureA branch. You see the same contents in featureA branch as was in main branch.



← → ⌛ github.com/sujataoak799/gitdemo11072025/tree/feat...

featureA had recent pushes 3 minutes ago [Compare & pull request](#)

featureA · ⌛ Go to file + Code

This branch is 4 commits ahead of [main](#). [Contribute](#)

sujataoak799	Added add.py file for commit on 18072025	62e5df1 · 5 hours ago	7 Commits
add.py	Added add.py file for commit on 180...	5 hours ago	
factorial.py	first file added	last week	
mynewfile	Added Info on Branching	5 days ago	
name.html	Added Info on Branching	5 days ago	

[README](#)

Add a README

Click on view all branch:



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY
Department of Information Technology
(NBA Accredited)



sujataoak799 / gitdemo11072025

<> Code Issues Pull requests Actions Projects Wiki Security Insights Settings

Branches

New branch

Overview Yours Active Stale All

Q Search branches...

Default

Branch	Updated	Check status	Behind	Ahead	Pull request
main	5 days ago	Green	0	4	Default

Your branches

Branch	Updated	Check status	Behind	Ahead	Pull request
featureA	5 minutes ago	Green	0	4	...

Active branches

Branch	Updated	Check status	Behind	Ahead	Pull request
featureA	5 minutes ago	Green	0	4	...

Click on add.py file to see the contents of add.py

← → ⌛ github.com/sujataoak799/gitdemo11072025/blob/feat... ⏪ 🔍

<> Code Issues Pull requests Actions Projects Wiki Security Insights Settings

Files

featureA Go to file

- add.py
- factorial.py
- mynewfile
- name.html

gitdemo11072025 / add.py

sujataoak799 Added add.py file for commit on 18072025

Code Blame 7 lines (6 loc) - 118 Bytes

```
1  # WAP to add two numbers in python???
2  # Adding two numbers
3  a = 5
4  b = 3
5  sum = a + b
6
7  print("The sum is:", sum)
```

Now in local repository in featureA branch create 3 files:



```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git branch
* featureA
  main
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# touch featureA-1
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# touch featureA-2
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# touch featureA-3
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# ls
add.py factorial.py featureA-1 featureA-2 featureA-3 mynewfile name.html
```

Suppose the developers, they are done with their coding.

Now they want to **merge** back the content which you have put it in the **featureA** branch to the **main** branch, and then they can delete that branch.

For that they have to **raise the pull request**. That is PR.

```
#git status
```

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git status
On branch featureA
Your branch is up to date with 'origin/featureA'.

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    featureA-1
    featureA-2
    featureA-3

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

```
#git add .
```

```
#git status
```

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git add .
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git status
On branch featureA
Your branch is up to date with 'origin/featureA'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   featureA-1
    new file:   featureA-2
    new file:   featureA-3
```

```
# git commit -m " Added 3 files in featureA branch"
```



PARSHVANATH CHARITABLE TRUST'S

A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git commit -m " Added 3 files in featureA branch"
[featureA 1534631] Added 3 files in featureA branch
 3 files changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 featureA-1
 create mode 100644 featureA-2
 create mode 100644 featureA-3
```

```
# git push -u origin featureA
```

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git push -u origin featureA
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 2 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 385 bytes | 385.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/sujataoak799/gitdemo11072025
  62e5df1..1534631  featureA -> featureA
Branch 'featureA' set up to track remote branch 'featureA' from 'origin'.
```

Goto [github.com account](https://github.com/sujataoak799)



featureA had recent pushes 1 minute ago

Compare & pull request

featureA 2 Branches 0 Tags

Go to file

This branch is 5 commits ahead of main.

Contribute

Author	File	Message	Time
sujataoak799	Added 3 files in featureA branch	1534631 · 3 minutes ago	8 Commits
	add.py	Added add.py file for commit on 18072025	6 hours ago
	factorial.py	first file added	last week
	featureA-1	Added 3 files in featureA branch	3 minutes ago
	featureA-2	Added 3 files in featureA branch	3 minutes ago
	featureA-3	Added 3 files in featureA branch	3 minutes ago
	mynewfile	Added Info on Branching	5 days ago

If you go to **main** branch , files of **feature-A** branch will not be seen. For that we need to do **merging**

For that goto **feature-A** branch → pull requests tab → click on **New Pull Requests**



github.com/sujataoak799/gitdemo11072025/pulls

sujataoak799 / gitdemo11072025

Type to search

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

Label issues and pull requests for new contributors

Now, GitHub will help potential first-time contributors discover issues labeled with good-first-issue.

Filters is:pr is:open Labels Milestones New pull request

github.com/sujataoak799/gitdemo11072025/compare

sujataoak799 / gitdemo11072025

Type to search

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

base: main compare: main

Choose different branches or forks above to discuss and review changes. Learn about pull requests Create pull request

Compare changes

Compare changes across branches, commits, tags, and more below. If you need to, you can also compare across forks.

base: main compare: main

Choose different branches or forks above to discuss and review changes. Learn about pull requests Create pull request

Compare and review just about anything

Branches, tags, commit ranges, and time ranges. In the same repository and across forks.

Example comparisons	
featureA	1 hour ago
main@{1day}...main	24 hours ago

Compare changes from main branch and featureA branch



Comparing changes

Choose two branches to see what's changed or to start a new pull request. If you need to, you can also [compare across forks](#) or [learn more about diff comparisons](#).

base: main compare: featureA

Discuss and review the changes in this comparison with others. [Learn about pull requests](#)

Create pull request

5 commits 7 files changed 1 contributor

Commits on Jul 10, 2025

Revert "Initial commit" sujataoak799 committed last week

Commits on Jul 13, 2025

Merge branch 'main' of https://github.com/sujataoak799/gitdemo11072025 sujataoak799 committed 5 days ago

Added info on Branching sujataoak799 committed 5 days ago

Click on create pull requests

Add a title

Added 3 files in FeatureA branch

Add a description

Merging a branch FeatureA to main branch

Markdown is supported Paste, drop, or click to add files

Create pull request



Click on Create pull request

Added 3 files in FeatureA branch #1

sujataoak799 wants to merge 5 commits into `main` from `FeatureA`

Conversation 0 Commits 5 Checks 0 Files changed 7

sujataoak799 commented 1 minute ago

Merging a branch `FeatureA` to main branch

sujataoak799 added 5 commits last week

- Revert "Initial commit" (1)
- Merge branch 'main' of https://github.com/sujataoak799/gtdemo11072025 (1)
- Added Info on Branching (1)
- Added .add.my file for commit on 18972025 (1)
- Added 3 files in FeatureA branch (1)

No conflicts with base branch

Merge pull request You can also merge this with the command line. View command line instructions

Add a comment

Click on Merge pull request

Commit message

Merge pull request #1 from sujataoak799:featureA

Extended description

Added 3 files in FeatureA branch and merging to main branch from featureA branch

This commit will be authored by 79905110+sjataoak799@users.noreply.github.com.

Confirm merge Cancel

Click on Confirm merge



Added 3 files in FeatureA branch #1

The screenshot shows a GitHub pull request interface. At the top, it says "Merged sujataoak799 merged 5 commits into main from FeatureA now". Below this are tabs for Conversation (selected), Commits (5), Checks (0), and Files changed (7). A comment from "sujataoak799" is shown: "Merging a branch FeatureA to main branch". The commit history shows 5 commits from "sujataoak799" added last week:

- Revert "Initial commit" (commit 6505603)
- Merge branch 'main' of https://github.com/sujataoak799/gitdemo11072025 (commit 6505603)
- Added Info on Branching (commit a4f7201)
- Added add.py file for commit on 18072025 (commit 62e5d71)
- Added 3 files in FeatureA branch (commit 1524633)

The pull request was merged by "sujataoak799" at commit 50b5e27 into main now. A "Revert" button is available. At the bottom, a message says "Pull request successfully merged and closed. You're all set — the FeatureA branch can be safely deleted." A "Delete branch" button is also present.

Goto code → main branch → All files of feature-A branch merges with main branch.

The screenshot shows a GitHub repository page for "gitdemo11072025". The navigation bar includes Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. The repository name "gitdemo11072025" is displayed above the main content area. The main branch dropdown shows "main". There are 2 branches and 0 tags. The search bar has "Type to" and a "Go to file" button. The commit list shows the following entries:

Commit	Message	Time
sujataoak799 Merge pull request #1 from sujataoak799/featureA	50b5e27 - 2 minutes ago	8 Commits
add.py	Added add.py file for commit on 18072025	7 hours ago
factorial.py	first file added	last week
featureA-1	Added 3 files in featureA branch	1 hour ago
featureA-2	Added 3 files in featureA branch	1 hour ago
featureA-3	Added 3 files in featureA branch	1 hour ago
mymewfile	Added Info on Branching	5 days ago
name.html	Added Info on Branching	5 days ago

The commits "featureA-1", "featureA-2", and "featureA-3" are highlighted with a red box. Below the commit list is a "README" section with a "Add a README" button and a note: "Help people interested in this repository understand your project by adding a README."



Step 10: And finally, once it has been merged, I'm going to delete the featureA branch.

Switch to the main branch using the **git checkout main** command and verify the current working branch using the **git branch** command.

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git branch
* featureA
  main
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git checkout main
Switched to branch 'main'
Your branch is ahead of 'origin/main' by 3 commits.
  (use "git push" to publish your local commits)
```

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git branch
  featureA
* main
```

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# ls
factorial.py  mynewfile  name.html
```

Step 11: Merge file contents from featureA to main branch in local repository.

Merge the contents (known as the history) from the feature branch into the main branch using the **git merge <branch-name>** command. The branch-name is the branch that histories are pulled from into the current branch. The output displays that one file was changed with one line inserted.

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git merge featureA
Updating a4f7201..1534631
Fast-forward
 add.py      |  7 ++++++
 featureA-1  |  0
 featureA-2  |  0
 featureA-3  |  0
 4 files changed, 7 insertions(+)
 create mode 100644 add.py
 create mode 100644 featureA-1
 create mode 100644 featureA-2
 create mode 100644 featureA-3
```



PARSHVANATH CHARITABLE TRUST'S

A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
Create Node 100011 featureA
```

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# ls
add.py factorial.py featureA-1 featureA-2 featureA-3 mynewfile name.html
```

Step 12: Deleting a branch

- A] Verify the **feature** branch is still available using the **git branch** command.

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git branch
  featureA
* main
```

- B] Delete the **featureA** branch using the **git branch -d <branch-name>** command

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git branch -d featureA
Deleted branch featureA (was 1534631).
```

- C] Verify the feature branch is no longer available using the **git branch** command

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git branch
* main
```

- D] Note the branch **featureA** is not deleted from remote repository/github

For this we need to :

```
#git push --delete origin featureA
```

```
root@labvm:/home/devasc/Desktop/git-sujata/gitdemo11072025# git push --delete origin fe
atureA
To https://github.com/sujataoak799/gitdemo11072025
 - [deleted]      featureA
```

Refresh your github page



Code Issues Pull requests Actions Projects Wiki Security Insights

gitdemo11072025 Public

Pin Watch 0

main 1 Branch 0 Tags

Switch branches/tags Find or create a branch...

Branches Tags

✓ main default

View all branches

Go to file t + Code

Commit	Date	Author
Added add.py file for commit on 18072025	8 hours ago	sujataoak799/featureA
first file added	last week	
Added 3 files in featureA branch	2 hours ago	
Added 3 files in featureA branch	2 hours ago	

Conclusion:

In this experiment, we understood the use case of branching and merging. Also, different commands were used for the same such as checkout, branch and merge for displaying the changes between the initial and latter commits.



PARSHVANATH CHARITABLE TRUST'S

A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



Academic Year: 2025-26

Semester: V

Class / Branch: TEIT

Subject: DevOps Lab

Name of Instructor: Prof. Sujata Oak

Experiment No. 4

Aim: To install and configure Jenkins to test and deploy an application with Maven.

Theory:

Jenkins is an open-source automation tool written in Java with plugins built for Continuous Integration purpose. Jenkins is used to build and test your software projects continuously making it easier for developers to integrate changes to the project, and making it easier for users to obtain a fresh build. It also allows you to continuously deliver your software by integrating with a large number of testing and deployment technologies.

“Continuous Integration is a software development practice where members of a team integrate their work frequently, usually each person integrates at least daily - leading to multiple integrations per day. Each integration is verified by an automated build (including test) to detect integration errors as quickly as possible.” In simple way, Continuous integration (CI) is the practice of frequently building and testing each change done to your code automatically.

Jenkins is a self-contained, open-source automation server which can be used to automate all sorts of tasks related to building, testing, and delivering or deploying software.

To install Jenkins following software packages are required

- 1) GIT (git-scm.com)
- 2) Notepad++ (<https://notepad-plus-plus.org/downloads/>)
- 3) Latest Java development kit (JDK)
- 4) Jenkins (<https://www.jenkins.io/>)
- 5) Apache Maven (Optional)



Installation Steps for Jenkins on Ubuntu 20.04

Step 1-: Update ubuntu repository

```
sujata@Ubuntu:~$ su root  
Password:
```

```
#apt-get update
```

```
root@Ubuntu:/home/sujata# apt-get update  
Get:1 http://security.ubuntu.com/ubuntu focal-security InRelease [128 kB]  
Hit:2 http://in.archive.ubuntu.com/ubuntu focal InRelease  
Get:3 http://in.archive.ubuntu.com/ubuntu focal-updates InRelease [128 kB]
```

```
Fetched 20.8 MB in 7s (3,072 kB/s)  
Reading package lists... Done
```

Step 2-: Install Java development kit

```
#apt-get install openjdk-11-jdk
```

```
root@Ubuntu:/home/sujata# apt-get install openjdk-11-jdk  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
The following additional packages will be installed:  
  ca-certificates-java fonts-dejavu-extra java-common libatk-wrapper-java  
  libatk-wrapper-java-jni libice-dev libpthread-stubs0-dev libsm-dev libx11-6  
  libx11-dev libxau-dev libxcb1-dev libxdmcp-dev libxt-dev  
  openjdk-11-jdk-headless openjdk-11-jre openjdk-11-jre-headless  
  x11proto-core-dev x11proto-dev xorg-sgml-doctools xtrans-dev  
Suggested packages:  
  default-jre libice-doc libsm-doc libx11-doc libxcb-doc libxt-doc
```



done.

```
Processing triggers for sgml-base (1.29.1) ...
Setting up x11proto-dev (2019.2-1ubuntu1) ...
Setting up libxau-dev:amd64 (1:1.0.9-0ubuntu1) ...
Setting up libice-dev:amd64 (2:1.0.10-0ubuntu1) ...
Setting up libsm-dev:amd64 (2:1.2.3-1) ...
Setting up libxdmcp-dev:amd64 (1:1.1.3-0ubuntu1) ...
Setting up x11proto-core-dev (2019.2-1ubuntu1) ...
Setting up libxcb1-dev:amd64 (1.14-2) ...
Setting up libx11-dev:amd64 (2:1.6.9-2ubuntu1.6) ...
Setting up libxt-dev:amd64 (1:1.1.5-1) ...
```

Step 3-: To test if Java has been installed successfully, run this command:

```
#java -version
```

```
root@Ubuntu:/home/sujata# java -version
openjdk version "11.0.23" 2024-04-16
OpenJDK Runtime Environment (build 11.0.23+9-post-Ubuntu-1ubuntu120.04.2)
OpenJDK 64-Bit Server VM (build 11.0.23+9-post-Ubuntu-1ubuntu120.04.2, mixed mode, sharing)
```

Step 4-: Install Jenkins

Now, we will install Jenkins itself. Issue the following four commands in sequence to initiate the installation from the Jenkins repository:

```
curl -fsSL https://pkg.jenkins.io/debian-stable/jenkins.io.key | sudo tee \
/usr/share/keyrings/jenkins-keyring.asc > /dev/null
```

```
root@Ubuntu:/home/sujata# curl -fsSL https://pkg.jenkins.io/debian-stable/jenkins.io.key | sudo tee \
> /usr/share/keyrings/jenkins-keyring.asc > /dev/null
```

```
echo deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] \
https://pkg.jenkins.io/debian-stable binary/ | sudo tee \
/etc/apt/sources.list.d/jenkins.list > /dev/null
```



PARSHVANATH CHARITABLE TRUST'S

A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
root@Ubuntu:/home/sujata# echo deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] \
> https://pkg.jenkins.io/debian-stable binary/ | sudo tee \
> /etc/apt/sources.list.d/jenkins.list > /dev/null
```

sudo apt-get update

```
root@Ubuntu:/home/sujata# sudo apt-get update
Ign:1 https://pkg.jenkins.io/debian-stable binary/ InRelease
Get:2 https://pkg.jenkins.io/debian-stable binary/ Release [2,044 B]
Get:3 https://pkg.jenkins.io/debian-stable binary/ Release.gpg [833 B]
Get:4 https://pkg.jenkins.io/debian-stable binary/ Packages [27.3 kB]
Hit:5 http://in.archive.ubuntu.com/ubuntu focal InRelease
Hit:6 http://in.archive.ubuntu.com/ubuntu focal-updates InRelease
Hit:7 http://in.archive.ubuntu.com/ubuntu focal-backports InRelease
Hit:8 http://security.ubuntu.com/ubuntu focal-security InRelease
Fetched 30.1 kB in 1s (34.5 kB/s)
Reading package lists... Done
```

sudo apt-get install jenkins

```
root@Ubuntu:/home/sujata# sudo apt-get install jenkins
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  net-tools
The following NEW packages will be installed:
  jenkins net-tools
```

Step 5:- To check Jenkins installed or not:

```
# jenkins --version
```

```
root@Ubuntu:/home/sujata# jenkins --version
2.452.3
```



PARSHVANATH CHARITABLE TRUST'S

A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



Step 6:- Once that's done, start the Jenkins service with the following command:

```
# sudo systemctl start jenkins.service
```

To confirm its status, use:

```
#sudo systemctl status jenkins
```

If its WORKING it will show as active (running)

```
root@Ubuntu:/home/sujata# sudo systemctl status jenkins
● jenkins.service - Jenkins Continuous Integration Server
  Loaded: loaded (/lib/systemd/system/jenkins.service; enabled; vendor preset: enabled)
  Active: active (running) since Sun 2024-07-28 11:03:05 IST; 6min ago
    Main PID: 6542 (java)
      Tasks: 37 (limit: 4542)
     Memory: 411.9M
```

Press ctrl+z to exit

Step 7:- Adjust Firewall and Configuring Jenkins

With Jenkins installed, we can proceed with adjusting the firewall settings. By default, Jenkins will run on port **8080**.

In order to ensure that this port is accessible, we will need to configure the built-in [Ubuntu firewall](#) (ufw). To open the 8080 port and enable the firewall, use the following commands:

```
root@Ubuntu:/home/sujata# sudo ufw status
Status: inactive
```

```
root@Ubuntu:/home/sujata# sudo ufw allow 8080
Rules updated
Rules updated (v6)
```

```
root@Ubuntu:/home/sujata# sudo ufw enable
Firewall is active and enabled on system startup
```



PARSHVANATH CHARITABLE TRUST'S

A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
root@Ubuntu:/home/sujata# sudo ufw status  
Status: active
```

To	Action	From
--	-----	-----
8080	ALLOW	Anywhere
8080 (v6)	ALLOW	Anywhere (v6)

Step 8:- With the firewall configured, it's time to set up Jenkins itself.

Once installation is done, you can test the application on <http://localhost:8080> in browser **OR** <http://127.0.0.1:8080>

Type in the IP of your VPS along with the port number. The Jenkins setup wizard will be shown below:



localhost:8080/login?from=%2F

Getting Started

Unlock Jenkins

To ensure Jenkins is securely set up by the administrator, a password has been written to the log ([not sure where to find it?](#)) and this file on the server:

`/var/lib/jenkins/secrets/initialAdminPassword`

Please copy the password from either location and paste it below.

Administrator password

Continue

An administrator password will be needed to proceed with the configuration. It can be easily found inside the `/var/lib/jenkins/secrets/initialAdminPassword` file. To check the initial password, use the [cat command](#) as indicated below:

```
root@Ubuntu:/home/sujata# sudo cat /var/lib/jenkins/secrets/initialAdminPassword
69e2d2c9995344768c72bfb4e28eba79
```

Copy the password, go back to the setup wizard, paste it and click **Continue**.



Getting Started

Unlock Jenkins

To ensure Jenkins is securely set up by the administrator, a password has been written to the log ([not sure where to find it?](#)) and this file on the server:

`/var/lib/jenkins/secrets/initialAdminPassword`

Please copy the password from either location and paste it below.

Administrator password
.....

Continue

Step 9:- Customize Jenkins window will appear. We recommend simply selecting the **Install suggested plugins** option for this step.



Getting Started

Customize Jenkins

Plugins extend Jenkins with additional features to support many different needs.

Install suggested plugins

Install plugins the Jenkins community finds most useful.

Select plugins to install

Select and install plugins most suitable for your needs.

Jenkins 2.452.3

Step 10-: Give it a couple of minutes for the installation process to complete. Once it's done, specify your username, password, full name, and email address, and click on **Save and Continue** to create an admin user.



Getting Started

Getting Started

✓ Folders	✓ OWASP Markup Formatter	✓ Build Timeout	✓ Credentials Binding	** Trilead API ** Git client ** Pipeline: Input Step ** Pipeline: Declarative Pipeline ** Java JSON Web Token (JJWT) ** OkHttp ** GitHub API Git
✓ Timestamper	✓ Workspace Cleanup	✓ Ant	✓ Gradle	
✓ Pipeline	✓ GitHub Branch Source	✓ Pipeline: GitHub Groovy Libraries	✓ Pipeline: Stage View	
✓ Git	✓ SSH Build Agents	✓ Matrix Authorization Strategy	✓ PAM Authentication	
✓ LDAP	✓ Email Extension	✓ Mailer	⌚ Dark Theme	** Pipeline Graph Analysis ** Pipeline: REST API Pipeline: Stage View Git SSH Build Agents Matrix Authorization Strategy PAM Authentication LDAP Email Extension Mailer ** - required dependency



Getting Started

Create First Admin User

Username

Password

Confirm password

Full name

E-mail address

Jenkins 2.452.3

[Skip and continue as admin](#)

Save and Continue

Then specify the preferred **Jenkins URL** and finish the configuration process.



Getting Started

Instance Configuration

Jenkins URL:

The Jenkins URL is used to provide the root URL for absolute links to various Jenkins resources. That means this value is required for proper operation of many Jenkins features including email notifications, PR status updates, and the `BUILD_URL` environment variable provided to build steps.

The proposed default value shown is **not saved yet** and is generated from the current request, if possible. The best practice is to set this value to the URL that users are expected to use. This will avoid confusion when sharing or viewing links.

Jenkins 2.452.3

Not now Save and Finish



PARSHVANATH CHARITABLE TRUST'S

A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



Getting Started

Jenkins is ready!

Your Jenkins setup is complete.

[Start using Jenkins](#)

Jenkins 2.452.3

Step 11:- After configuration, the Jenkins dashboard will appear, meaning the Jenkins server installation and initial setup were successful.



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY
Department of Information Technology
(NBA Accredited)



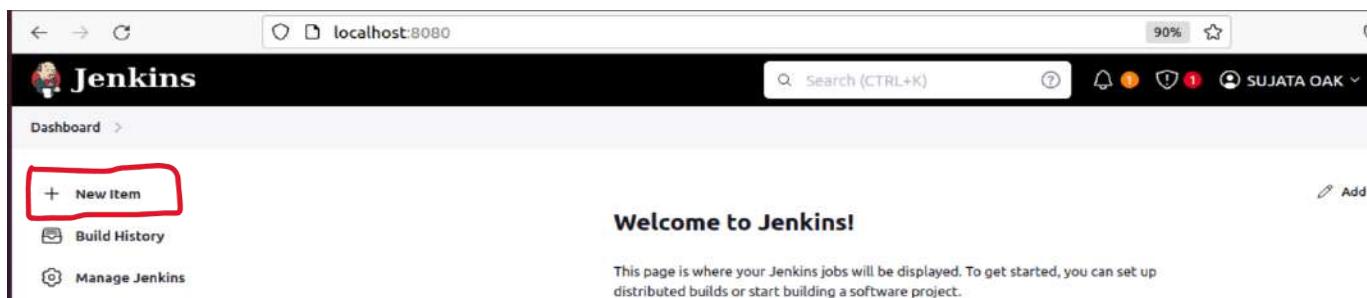


Task to be performed as a part of this experiment:

Jenkins Programming in Java using Version Control Git

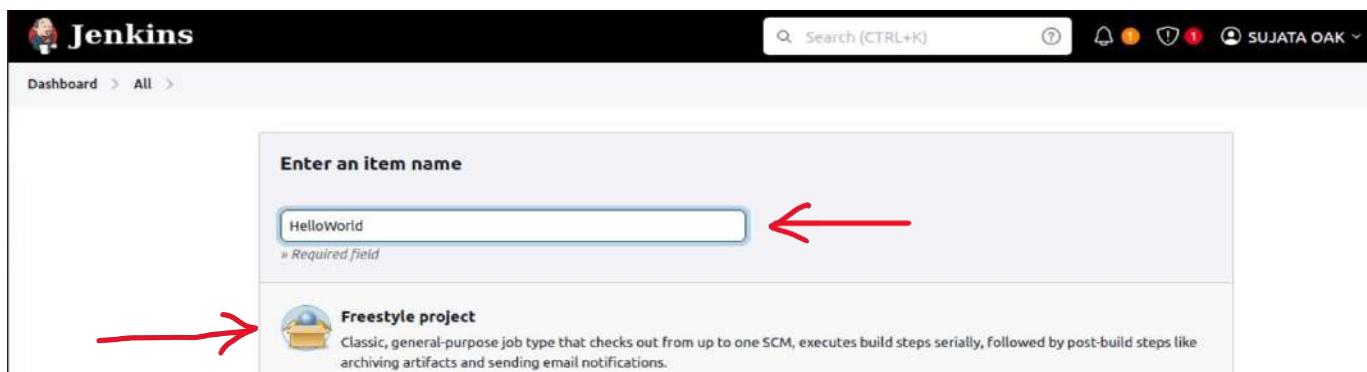
Create and run a job in Jenkins for simple **HelloWorld** in Java

Step 1: Go to the Jenkins dashboard and click on the New Item.



The screenshot shows the Jenkins dashboard at `localhost:8080`. The main header says "Welcome to Jenkins!". Below it, a message reads: "This page is where your Jenkins jobs will be displayed. To get started, you can set up distributed builds or start building a software project." On the left sidebar, there are three buttons: "+ New Item" (highlighted with a red box), "Build History", and "Manage Jenkins". The top right corner shows a user profile for "SUJATA OAK" and some status icons.

Step 2: In the next page, enter the item name, and select the 'Freestyle project' option. And click OK. Here, my item name is HelloWorld.



The screenshot shows the "Enter an item name" dialog. A red arrow points to the input field containing "HelloWorld", which is labeled as a "Required field". Another red arrow points to the "Freestyle project" option, which is described as a "Classic, general-purpose job type that checks out from up to one SCM, executes build steps serially, followed by post-build steps like archiving artifacts and sending email notifications."

Step 3: When you enter the OK, you will get a configuration page. Enter the details of the project in the Description section.



Step 4: On the Source Code Management section, select the **Git** option, and specify the Repository URL.

To do that you should have proper github setup on your system. To do the github setup:

- First, you have to create a project in java. Here, I created a simple **HelloWorld** program and saved it to one folder i.e. **Desktop/JENKINS_LAB**. Compile the `HelloWorld.java` file.

```
sujata@Ubuntu:~/Desktop$ cd JENKINS_LAB/
sujata@Ubuntu:~/Desktop/JENKINS_LAB$ gedit Simple.java
sujata@Ubuntu:~/Desktop/JENKINS_LAB$ javac Simple.java
sujata@Ubuntu:~/Desktop/JENKINS_LAB$ java Simple
Hello Java
```

- Now create a project in your GitHub account and give the Repository name. Here my repository name is `HelloWorld_29072024`.



Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere?
[Import a repository.](#)

Required fields are marked with an asterisk (*).

Repository template

No template

Start your repository with a template repository's contents.

Owner *



Repository name *

/ HelloWorld_29072024 is available.

Great repository names are short and memorable. Need inspiration? How about [bookish-octo-couscous](#) ?

Description (optional)

For Jenkins Test

Public

Anyone on the Internet can see this repository. You choose who can commit.

Private

You choose who can see and commit to this repository.

Initialize this repository with:

Add a README file

This is where you can write a long description for your project. [Learn more about READMEs.](#)

Add .gitignore

.gitignore template: None

Choose which files not to track from a list of templates. [Learn more about Ignoring files.](#)

Choose a license

License: None

A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

This will set [main](#) as the default branch. Change the default name in your [settings](#).

ⓘ You are creating a public repository in your personal account.

Create repository

- Click on **Create repository**.



The screenshot shows a GitHub repository named 'HelloWorld_29072024'. The repository is public and contains one branch ('main') and no tags. It has one commit from user 'sujataoak799' labeled 'Initial commit'. The README file is present. On the right side, there is a 'Clone' section with options for HTTPS, SSH, and GitHub CLI. The HTTPS link is highlighted with a red box.

Your repository is created. Copy the repository URL. My repository URL is:
https://github.com/sujataoak799>HelloWorld_29072024.git

- Open the command prompt in your Ubuntu and go to the path where your java file is created.
- Then run the following command.

git init

```
sujata@Ubuntu:~/Desktop/JENKINS_LAB$ git init
Initialized empty Git repository in /home/sujata/Desktop/JENKINS_LAB/.git/
git status
```



```
sujata@Ubuntu:~/Desktop/JENKINS_LAB$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    Jenkins-installation-link.odt
    Simple.class
    Simple.java
```

git add .

```
sujata@Ubuntu:~/Desktop/JENKINS_LAB$ git add .
git status
```

```
sujata@Ubuntu:~/Desktop/JENKINS_LAB$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   Jenkins-installation-link.odt
    new file:   Simple.class
    new file:   Simple.java
```

- Configure your GitHub account in your system.

1. git config --global user.email "your@email"
2. git config --global user.name "username"



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
sujata@Ubuntu:~/Desktop/JENKINS_LAB$ git config --global user.name "sujataoak799"
sujata@Ubuntu:~/Desktop/JENKINS_LAB$ git config --global user.email "sujataoak2021@gmail.com"
sujata@Ubuntu:~/Desktop/JENKINS_LAB$ git config --list
user.name=sujataoak799
user.email=sujataoak2021@gmail.com
```

- Commit it and add the repository URL.
- 1. git commit -m "Added HelloWorld Java Program"

```
sujata@Ubuntu:~/Desktop/JENKINS_LAB$ git commit -m "Added HelloWorld Java Program"
[master (root-commit) b38b097] Added HelloWorld Java Program
 3 files changed, 5 insertions(+)
  create mode 100644 Jenkins-installation-link.odt
  create mode 100644 Simple.class
  create mode 100644 Simple.java
```

- 2. git remote add origin https://github.com/sujataoak799>HelloWorld_29072024.git

```
sujata@Ubuntu:~/Desktop/JENKINS_LAB$ git remote add origin https://github.com/sujataoak799>HelloWorld_2907
2024.git
```

- 3. git push -u origin master

```
sujata@Ubuntu:~/Desktop/JENKINS_LAB$ git push -u origin master
Username for 'https://github.com': sujataoak799
Password for 'https://sujataoak799@github.com':
remote: Support for password authentication was removed on August 13, 2021.
remote: Please see https://docs.github.com/get-started/getting-started-with-git/about-remote-repositories#
cloning-with-https-urls for information on currently recommended modes of authentication.
fatal: Authentication failed for 'https://github.com/sujataoak799>HelloWorld_29072024.git/'
```

```
sujata@Ubuntu:~/Desktop/JENKINS_LAB$ git remote set-url origin https://ghp_ZaLvsbTPgxILguomA7MJ2Wwa52NpCX3
iWWij@github.com/sujataoak799>HelloWorld_29072024.git
```



```
sujata@Ubuntu:~/Desktop/JENKINS_LAB$ git push -u origin master
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 8.19 KiB | 8.19 MiB/s, done.
Total 5 (delta 0), reused 0 (delta 0)
remote:
remote: Create a pull request for 'master' on GitHub by visiting:
remote:     https://github.com/sujataoak799/HelloWorld_29072024/pull/new/master
remote:
To https://github.com/sujataoak799/HelloWorld_29072024.git
 * [new branch]      master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
```

- Now, when you refresh your GitHub account, the HelloWorld file will be added in your repository.

The screenshot shows a GitHub repository page for 'HelloWorld_29072024'. The repository is public. The master branch has recent pushes 3 minutes ago. The commit history shows three commits from 'sujataoak799': 'Added HelloWorld Java Program' (b38b097 - 14 minutes ago), 'Jenkins-installation-link.odt' (14 minutes ago), 'Simple.class' (14 minutes ago), and 'Simple.java' (14 minutes ago). The commit for 'Simple.java' is highlighted with a red box.

Commit	Message	Date	Actions
b38b097	Added HelloWorld Java Program	14 minutes ago	1 Commit
Jenkins-installation-link.odt	Added HelloWorld Java Program	14 minutes ago	
Simple.class	Added HelloWorld Java Program	14 minutes ago	
Simple.java	Added HelloWorld Java Program	14 minutes ago	

Step 5: Add the Repository URL in the **Source Code Management** section.



PARSHVANATH CHARITABLE TRUST'S

A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



Source Code Management

None

Git [?](#)

[Repositories ?](#)

[Repository URL ?](#) ✖

`https://github.com/sujataoak799/HelloWorld_29072024.git`

! An internal error occurred during form field validation (HTTP 403). Please reload the page and if the problem persists, ask the administrator for help.

[Credentials ?](#)

- none -

[Save](#) [Apply](#)



Jenkins

Failed to connect to repository

Error performing git command: git.exe ls-remote

Error performing git command: git.exe init

Step 1 - Check git is installed

Step 2 - If not install Git

Step 3 - Provide location of git.exe in Jenkins

Manage Jenkins > Global Tool Configuration > Git

Step 6: Now, it is time to build the code. Click on "**Add build step**" and select the "**Execute Shell**".



Build Steps

≡ Execute shell ?

Command

See [the list of available environment variables](#)

```
javac Simple.java
java Simple
```

Step 7: Enter the following command to compile the java code.

```
javac Simple.java
java Simple
```

Step 8: Click Apply and then Save button.

Step 9: Once you saved the configuration, then now can click on **Build Now** option.

Step 10: After clicking on **Build Now**, you can see the status of the build on the Build History section.

Once the build is completed, a status of the build will show if the build was successful or not. If the build is failed then it will show in red color. Blue symbol is for success.



Dashboard > HelloWorld >

Status **HelloWorld**

Changes Workspace Build Now Configure Delete Project Rename

Author:Sujata Oak
Create and run a job in Jenkins for simple HelloWorld in Java.

Permalinks

- Last build (#2), 8 min 3 sec ago
- Last stable build (#2), 8 min 3 sec ago
- Last successful build (#2), 8 min 3 sec ago
- Last completed build (#2), 8 min 3 sec ago

Build History Filter... /

#	Date
#2	Jul 28, 2024, 11:40 PM
#1	Jul 28, 2024, 11:29 PM

Click on the build number **#2** in the **Build History section** to see the details of the build.

Step 11: Click on **Console Output** from the left side of the screen to see the status of the build you run. It should show the success message.



Dashboard > HelloWorld > #2 > Console Output

</> Changes
Console Output
View as plain text
Edit Build Information
Delete build '#2'
Timings
Git Build Data
← Previous Build

```
Started by user SUJATA OAK ✓
Running as SYSTEM
Building in workspace /var/lib/jenkins/workspace/HelloWorld
The recommended git tool is: NONE
No credentials specified
> git rev-parse --resolve-git-dir /var/lib/jenkins/workspace/HelloWorld/.git # timeout=10
Fetching changes from the remote Git repository
> git config remote.origin.url https://github.com/sujataoak799>HelloWorld_29072024.git # timeout=10
Fetching upstream changes from https://github.com/sujataoak799>HelloWorld_29072024.git
> git --version # timeout=10
> git --version # 'git version 2.25.1'
> git fetch --tags --force --progress -- https://github.com/sujataoak799>HelloWorld_29072024.git
+refs/heads/*:refs/remotes/origin/* # timeout=10
> git rev-parse refs/remotes/origin/master^{commit} # timeout=10
Checking out Revision b38b097da630049c00b0eda65860b67c55600237 (refs/remotes/origin/master)
> git config core.sparsecheckout # timeout=10
> git checkout -f b38b097da630049c00b0eda65860b67c55600237 # timeout=10
Commit message: "Added HelloWorld Java Program"
> git rev-list --no-walk b38b097da630049c00b0eda65860b67c55600237 # timeout=10
[HelloWorld] $ /bin/sh -xe /tmp/jenkins8894639940349829957.sh
+ javac Simple.java
+ java Simple
Hello Java
Finished: SUCCESS
```

Conclusion: In this experiment we understood the installation and implemented the use case of Jenkins in version control system.



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



Academic Year: 2025-26

Semester: V

Class / Branch: TEIT

Subject: DevOps Lab

Name of Instructor: Ms. Sujata Oak

Experiment No. 4

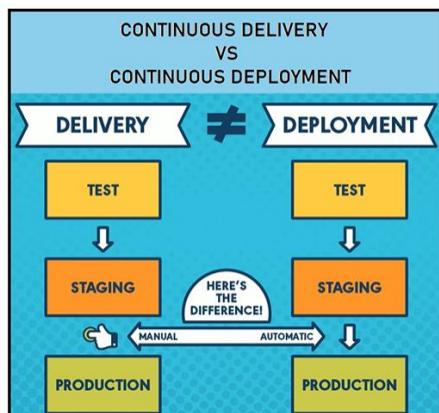
Aim: To install Jenkins and configure parallel jobs for building pipeline.

Theory: Jenkins is an open-source automation server widely used in Continuous Integration (CI) and Continuous Delivery (CD) pipelines. It allows developers to automate building, testing, and deployment of applications. One powerful feature of Jenkins is the ability to execute parallel jobs in a pipeline, which reduces build time and improves efficiency.

Objectives

- Understand the process of installing Jenkins.
- Learn how to configure Jenkins for pipeline execution.
- Implement parallel stages in Jenkins pipelines to run tasks simultaneously.

Continuous Delivery vs Continuous deployment



Continuous Delivery : The entire process is segregated into three phases:



Test, Staging and Production. The test and staging are automatic. But when you try to move your project from staging to production you need manual. So, this is called **continuous delivery**.

Continuous Deployment: All the phases test, staging and production is completely automatic. There is no manual intervention is required. So, this is called **continuous deployment**.

Implementation:

Create 3 jobs: Test , Staging and Production

Test → Freestyle → Ok

Build Steps
Automate your build process with ordered tasks like

Execute shell ?

Command
See [the list of available environment variables](#)

```
sleep 10
echo "This is a Test phase"
```

Advanced ▾

Save **Apply**

Staging → freestyle → Ok



Build Steps

Automate your build process with ordered tasks like

≡ Execute shell ?

Command
See [the list of available environment variables](#)

```
sleep 10
echo "This is a Staging phase"
```

Advanced ▾

Save **Apply**

Production → freestyle → ok

Build Steps

Automate your build process with ordered tasks like

≡ Execute shell ?

Command
See [the list of available environment variables](#)

```
sleep 10
echo "This is a Production Phase"
```

Advanced ▾

Save **Apply**

Part I: Demo on concept of continuous delivery



In continuous delivery you can see the test job and staging is continuous.

Test → Configure → Add post build action → build other project

Post-build Actions

Define what happens after a build completes, like:

Build other projects ?

Projects to build

Staging, |

Trigger only if build is stable
 Trigger even if the build is unstable
 Trigger even if the build fails

Add post-build action ▾

Save **Apply**

In Continuous delivery, test and staging would be automatic. But when you want to move from staging to production it should be manual.

Now goto staging job. In this staging job I have selected this option build other jobs manual step.

Staging → Configure → Post-Build Actions → Build other projects (manual step) → production

So I have created all my jobs and I have interlinked as per the continuous delivery.



Staging

Upstream Projects

Test

Downstream Projects

production

Permalinks

Goto Dashboard,

Now you need to click on this plus icon. New view: Name: continuous delivery.
And type: build pipeline view and click on create.

The screenshot shows the Jenkins dashboard with a dark theme. At the top, there's a header bar with the Jenkins logo, a search icon, a bell icon with a red notification count of 1, and a user profile for 'SUJATA OAK'. Below the header, the main navigation bar has links for 'Dashboard', 'New Item', 'Build History', and 'Project Relationship'. In the center, there's a message: 'Hello. This is an urgent message.' Below the message, there are tabs for 'All', 'Continuous Delivery', and 'myprojectFirstPipeline'. On the far right of the dashboard, there's a red box highlighting the blue '+' button used for creating new views.

The screenshot shows the 'New view' creation page in Jenkins. On the left, there's a sidebar with links for 'New Item', 'Build History', 'Manage Jenkins', and 'My Views'. Below that are two expandable sections: 'Build Queue' (No builds in the queue) and 'Build Executor Status' (0 of 2 executors busy). On the right, the main form is titled 'New view'. It has fields for 'Name' (set to 'Continuous Delivery') and 'Type'. Under 'Type', the 'Build Pipeline View' option is selected, with a description: 'Shows the jobs in a build pipeline which propagates through are shown'. There are also other options: 'List View' (Shows items in a simple list for which view.) and 'My View' (This view automatically displays). At the bottom of the form is a blue 'Create' button.



and scroll down and look for your initial job.
My initial job is the test job.

Pipeline Flow

Layout

Based on upstream/downstream relationship

This layout mode derives the pipeline structure based on the upstream/downstream relationships between jobs. This is the only out-of-the-box supported layout mode.

Upstream / downstream config

Select Initial Job ?

Test

Trigger Options

Build Cards

Standard build card

Use this default build card

Save

Apply

localhost:8080/view/Continuous%20Delivery/

Jenkins

Dashboard > Continuous Delivery >

Build Pipeline

Run History Configure Add Step Delete Manage

This view has no jobs associated with it.
You can either add some existing jobs to this view or create a new job in this view.



The screenshot shows a Jenkins dashboard for a 'Continuous Delivery' pipeline. The pipeline consists of three stages: 'Pipeline #1', 'Test', 'Staging', and 'production'. The 'Test' stage is currently active, showing a single build step named '#1 Test' with a timestamp of 'Jul 1, 2025 6:33:27 PM' and a duration of '3.2 sec and counting'. This step is associated with the user 'sujata'. Below the pipeline stages are various Jenkins management icons: Run, History, Configure, Add Step, Delete, and Manage.

This screenshot shows the same Jenkins dashboard after a refresh. The 'Test' stage has completed successfully, indicated by a green background. The 'Staging' stage is now active, showing a build step '#1 Staging' with a timestamp of 'Jul 1, 2025 6:33:42 PM' and a duration of '10 sec'. This step is also associated with 'sujata'. The 'production' stage remains inactive with a yellow background.

Refresh it , still the production stage will not start Since it is a manual process in continuous delivery. We need to trigger it manually.

The final screenshot shows the Jenkins dashboard after triggering the manual step in the 'Staging' stage. The 'production' stage is now active, showing a build step with a timestamp of 'Jul 1, 2025 6:33:42 PM' and a duration of '10 sec'. This step is also associated with 'sujata'. The entire pipeline is now shown in green, indicating successful completion of all stages.



The screenshot shows the Jenkins Build Pipeline interface. At the top, there are navigation links: Dashboard > Continuous Delivery >. Below this is the title "Build Pipeline". A toolbar with icons for Run, History, Configure, Add Step, Delete, and Manage is visible. The pipeline itself consists of four stages: "#1 Test", "#1 Staging", and "#1 production", which are connected by arrows. Each stage has a timestamp (Jul 1, 2025 6:33:27 PM), a duration (10 sec), and a user (sujata). The first stage, "Pipeline #1", is shown on the left.

Part II: Demo on concept of continuous deployment

All 3 stages has to be automated

Goto Dashboard → Staging → Configure

Post-build Actions

Define what happens after a build completes,

Build other projects ?

Projects to build

production,

Trigger only if build is stable

Trigger even if the build is unstable

Trigger even if the build fails

Add post-build action ▾

Save **Apply**



The screenshot shows the Jenkins Build Pipeline interface. At the top, there are navigation links: 'Dashboard' and 'Continuous Delivery'. Below that is the 'Build Pipeline' section with the title 'Build Pipeline'. There are several buttons: 'Run', 'History', 'Configure', 'Add Step', 'Delete', and 'Manage'. The pipeline consists of three parallel stages: '#2 Test', '#2 Staging', and '#2 production'. Each stage has a timestamp of 'Jul 1, 2025 6:41:57 PM' and a duration of '10 sec'. The stages are labeled with 'sujata'.

Part III: Parallel Jobs in Jenkins Build Pipeline

In Test job → Configure

Post-build Actions

Define what happens after a build completes,

≡ **Build other projects** ?

Projects to build

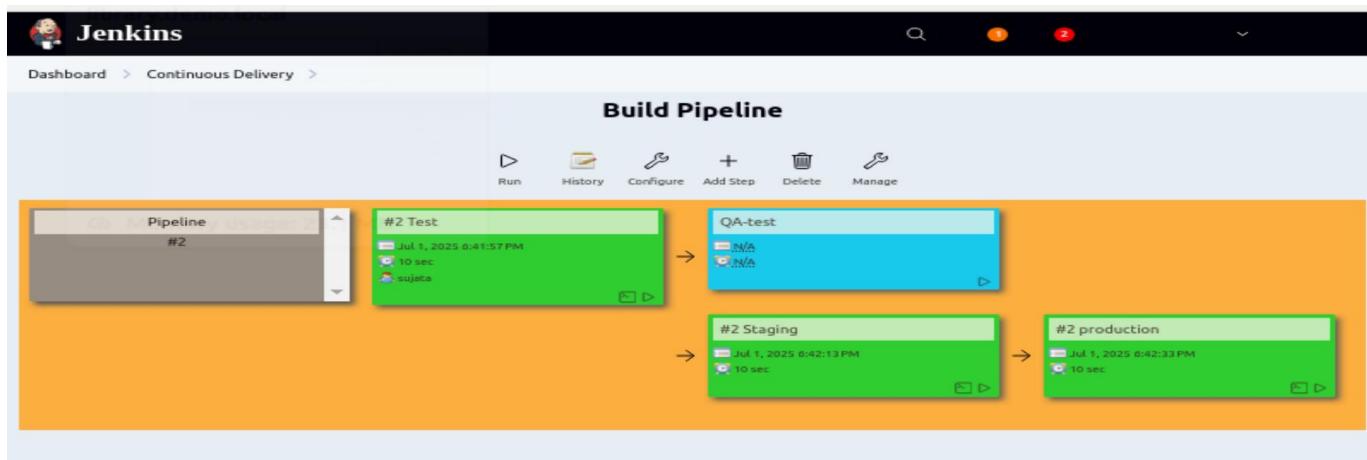
Staging,QA-test

Trigger only if build is stable
 Trigger even if the build is unstable
 Trigger even if the build fails

Add post-build action ▾

Save **Apply**

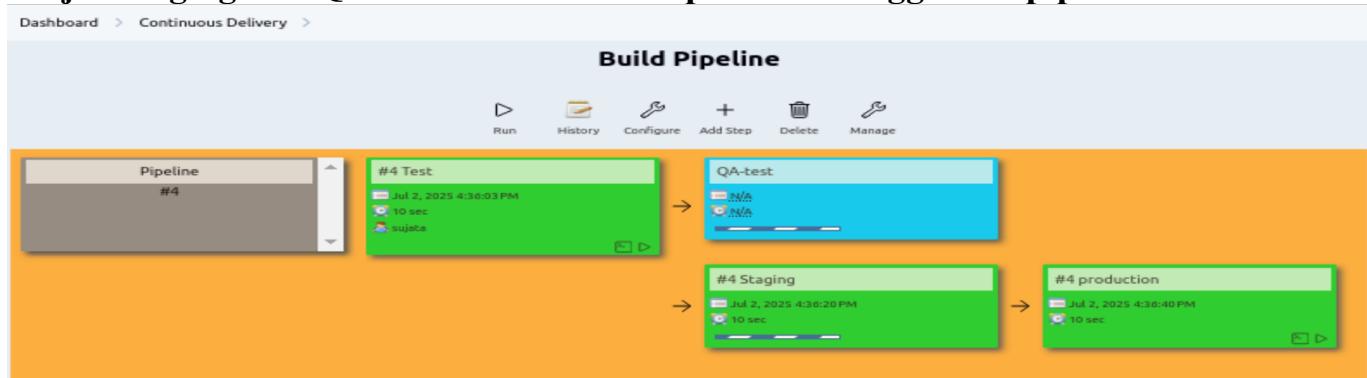
Goto dashboard → Click on Continuous Delivery



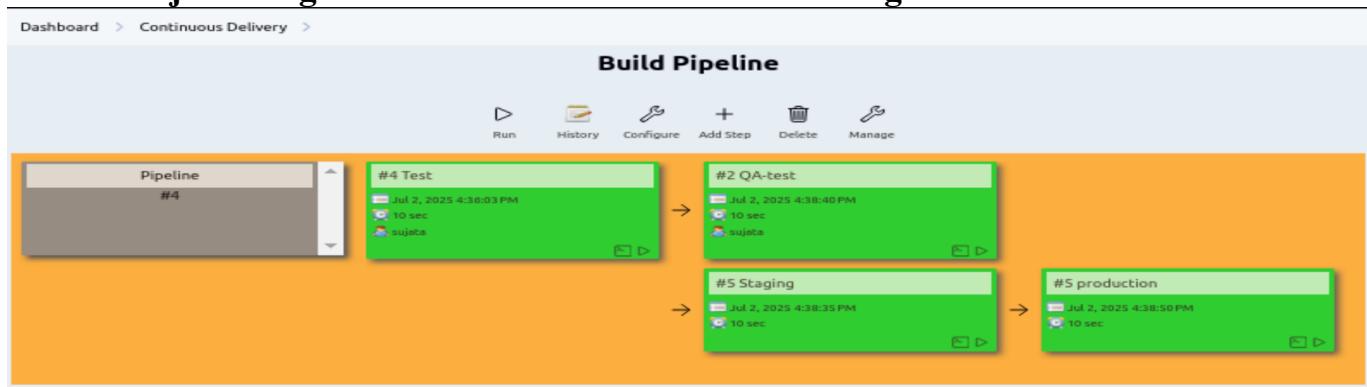
Refresh and Run

1st job executed shown in green color

2nd job staging and QA-test will execute in parallel. Trigger the pipeline



At last all job will get executed and the color should be green.





PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



Conclusion:

In this experiment we successfully learned installing jenkins and setting up parallel jobs in a pipeline which significantly improves CI/CD efficiency. This approach helps in faster feedback cycles, quicker releases, and better utilization of resources. With proper pipeline configuration, teams can ensure high productivity while maintaining quality.



Academic Year: 2025-26

Semester: V

Class / Branch: TEIT

Subject: DevOps Lab

Name of Instructor: Prof. Sujata Oak

Experiment No. 5

Aim: To create and build a CI/CD pipeline in Jenkins to test and deploy an application over the tomcat server.

Theory:

Continuous Integration (CI) and Continuous Deployment (CD) are essential practices in modern software engineering. **Jenkins**, an open-source automation server, is widely used to implement CI/CD pipelines. A **pipeline** in Jenkins defines the automated sequence of tasks needed to build, test, and deploy an application.

Jenkins pipelines are configured to automate the deployment of a **web application** onto an **Apache Tomcat server**.

Jenkins Pipelines

A Jenkins **Pipeline** is a suite of plugins that supports integrating and implementing continuous delivery. Pipelines can be defined as:

- **Declarative Pipeline** – uses a simple, predefined syntax.
- **Scripted Pipeline** – more flexible, written in Groovy scripts.

The pipeline describes **stages** such as:

1. **Checkout Code** (from GitHub or another repository)
2. **Build** (compile code, package into .war)
3. **Test** (unit tests, integration tests)
4. **Deploy** (copy artifact to Tomcat server)

Apache Tomcat Server

Apache Tomcat is an open-source Java Servlet container that runs Java-based web applications packaged as .war (Web Application Archive) files. In deployment pipelines:

- Tomcat provides the runtime environment for the application.
- Jenkins deploys the .war file automatically to Tomcat's webapps directory.

Jenkins–Tomcat Integration

To automate deployment:

Compiled By: Prof. Sujata Oak



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



- Jenkins is configured with **credentials** to connect to the Tomcat server.
- Deployment is done using methods like:
 - **Jenkins Deploy to Container Plugin** (direct WAR deployment).
 - **scp/rsync** to copy .war files into Tomcat's webapps.
 - **Pipeline Groovy scripts** for custom deployment logic.

IMPLEMENTATION: CONFIGURING JENKINS PIPELINE

Once Jenkins is installed follow the below steps:

Step1: Now, we need to specify the Java location to Jenkins. Go back to your server command prompt and use the code below to fetch the directory of Java. Multiple directories will be listed using the below code. In our case, the directory is: '/usr/lib/jvm/java-11-openjdk-amd64/bin/java'.

find / -type f -name java

```
devasc@labvm:~/Desktop/jenkins$ sudo su
root@labvm:/home/devasc/Desktop/jenkins# find / -type f -name java
find: '/run/user/900/gvfs': Permission denied
/var/lib/dpkg/alternatives/java
/etc/apparmor.d/abstractions/ubuntu-browsers.d/java
/usr/share/bash-completion/completions/java
/usr/lib/jvm/java-11-openjdk-amd64/bin/java
/usr/lib/jvm/java-21-openjdk-amd64/bin/java
```

Step 2) Copy the above location and go back to your Jenkins Dashboard. Look for Global Tool Configuration under the Manage Jenkins menu, as shown in the image below.



The screenshot shows the Jenkins management interface. In the sidebar, the 'Manage Jenkins' button is highlighted with a red box. In the main content area, there is a message about a new Jenkins version available for download. Below it, a warning is displayed regarding security vulnerabilities in Jenkins 2.452.3 core and libraries, with a link to a fix. The 'System Configuration' section is expanded, showing the 'System' and 'Tools' tabs. The 'Tools' tab is also highlighted with a red box.

Step 3) Unselect the Install automatically button from the JDK window and fill the fields. Paste the java location path and trim it as shown in the image below. In the below Git window, select the install automatically checkbox.



Dashboard > Manage Jenkins > Tools

JDK installations ^ Edit

Add JDK

≡ JDK

Name

java_home



JAVA_HOME

/usr/lib/jvm/java-11-openjdk-amd64/



Install automatically





≡ **Git**

Name

Default

Path to Git executable ?

git ✓

Install automatically ?

Add Installer ▾

Step 4) Similarly, check the box for Maven and fill the name field. Save all the settings, and now the configuration of Jenkins is completed.



Maven installations ^ Edited

Add Maven

Maven

Name: Mavenlatest

Install automatically ?

Install from Apache

Version: 3.9.10

Add Installer ^

Save **Apply**

Create CI/CD Pipeline Jenkins

We can now start creating pipelines using Jenkins after all the configuration and setup.

Continuous Integration

Step 1) Create New Item, select Freestyle Project and provide a name to your item.



Jenkins

Dashboard > All >

Enter an item name

CICD_Pipeline_Demo ✓

» Required field

Freestyle project ✓

Classic, general-purpose job type that archiving artifacts and sending email n

Pipeline

Orchestrates long-running activities th and/or organizing complex activities th

Multi-configuration project

Suitable for projects that need a large etc.

OK

Step 2) Switch to the Source Code Management window and paste your Github repository link. Specify your branch name of the repository below and Save it.

[Note: The above-linked Github repository '<https://github.com/sujataoak799/hello-world2025.git>' contains a 'pom.xml' file used for Java compilation and generates a web app. It will be deployed to the server]



The **pom.xml** file is used in **Apache Maven** (a popular Java build automation tool).

- **POM → Project Object Model**
- **pom.xml** → An XML file that contains the configuration and metadata about the project.
It usually defines:
 - Project coordinates (groupId, artifactId, version)
 - Dependencies (other libraries required)
 - Build plugins and goals
 - Project info (name, description, URL, licenses, developers, etc.)In short:
pom.xml = the **blueprint** for how Maven builds and manages your Java project.

- **WAR → Web Application Archive** (used to package web apps, including Jenkins)

The screenshot shows the Jenkins Pipeline configuration page for a job named 'CICD_PIPELINE_DEMO'. The 'Configure' section is open, and the 'Source Code Management' tab is selected, indicated by a red box and a red checkmark. The 'Repository URL' field contains 'https://github.com/sujataoak799/hello-world2025.git'. The 'Branch Specifier' field contains '/master'. Other tabs visible include General, Triggers, Environment, Build Steps, and Post-build Actions.

Step 3) Now click on Build Now button from the menu. With this step, all the repository files will be fetched by Jenkins. Click on Configure to go back to the same settings page.



PARSHVANATH CHARITABLE TRUST'S

A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



Dashboard > CICD_Pipeline_Demo >

Status

Changes

Workspace

Build Now

Configure

Delete Project

Rename

Build History

trend

Filter...

/

#1

Aug 11, 2024, 3:48 PM



localhost:8080/job/CICD_PIPELINE_DEMO/6/console

Jenkins

Dashboard > CICD_PIPELINE_DEMO > #6 > Console Output

Status Changes Console Output Edit Build Information Delete build '#0' Timings Git Build Data Previous Build

Console Output

Started by user SUJATA OAK
Running as SYSTEM
Building in workspace /var/lib/jenkins/workspace/CICD_PIPELINE_DEMO
The recommended git tool is: NONE
No credentials specified
> git rev-parse --resolve-git-dir /var/lib/jenkins/workspace/CICD_PIPELINE_DEMO/.git # timeout=10
Fetching changes from the remote Git repository
> git config remote.origin.url https://github.com/sujataoak799/hello-world2825.git # timeout=10
Fetching upstream changes from https://github.com/sujataoak799/hello-world2825.git
> git --version # timeout=10
> git --version # 'git version 2.25.1'
> git fetch --tags --force --progress .. https://github.com/sujataoak799/hello-world2825.git +refs/heads/*:refs/remotes/origin/* # timeout=10
> git rev-parse refs/remotes/origin/master^{commit} # timeout=10
Checking out Revision fa3785e9777df4866e65a298b28f5cc2498df496 (refs/remotes/origin/master)
> git config core.sparsecheckout # timeout=10
> git checkout -f fa3785e9777df4866e65a298b28f5cc2498df496 # timeout=10
Commit message: "Update index.jsp"
> git rev-list --no-walk fa3785e9777df4866e65a298b28f5cc2498df496 # timeout=10
Finished: SUCCESS

Step 4) Click on Build Tab and select build step as ‘Invoke top-level Maven targets’.



PARSHVANATH CHARITABLE TRUST'S

A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



Dashboard > CICD_Pipeline_Demo > Configuration

 Poll SCM ?

Configure

 General Source Code Management Build Triggers Build Environment Build Steps Post-build Actions

Build Environment

 Delete workspace before build starts Use secret text(s) or file(s) ? Filter

Execute Windows batch command

Execute shell

Invoke Ant

Invoke Gradle script

 Invoke top-level Maven targets

Run with timeout

Set build status to "pending" on GitHub commit

 Add build step ^

Step 5) Select your maven name from the drop-down menu. Fill the goals with the multiple jobs you need to perform and separate them with one space. These goals are available in your repository, and you need to invoke them using Maven. Save it and again click on the ‘Build Now’ button from the menu as we did in the previous steps. Now the maven commands will be executed that will generate a war file.



Dashboard > CICD_PIPELINE_DEMO > Configuration

Configure

- General
- Source Code Management
- Triggers
- Environment**
- Build Steps
- Post-build Actions

Add timestamps to the Console Output
 Inspect build log for published build scans
 Terminate a build if it's stuck
 with Ant ?

Build Steps

Automate your build process with ordered tasks like code compilation, testing, and deployment.

Invoke top-level Maven targets ?

Maven Version
Mavenlatest

Goals
clean compile package

Advanced

Save Apply

Click-On Build Now:

Dashboard > CICD_Pipeline_Demo > #2 > Console Output

```
WARNING: Illegal reflective access by com.thoughtworks.xstream.core.util.Fields (file:/var/lib/jenkins/.m2/repository/com/thoughtworks/xstream/xstream/1.3.1/xstream-1.3.1.jar) to field java.util.Properties.defaults
WARNING: Please consider reporting this to the maintainers of com.thoughtworks.xstream.core.util.Fields
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
[INFO] Packaging webapp
[INFO] Assembling webapp [webapp] in [/var/lib/jenkins/workspace/CICD_Pipeline_Demo/webapp/target/webapp]
[INFO] Processing war project
[INFO] Copying webapp resources [/var/lib/jenkins/workspace/CICD_Pipeline_Demo/webapp/src/main/webapp]
[INFO] Webapp assembled in [20 msecs]
[INFO] Building war: /var/lib/jenkins/workspace/CICD_Pipeline_Demo/webapp/target/webapp.war
[INFO] WEB-INF/web.xml already added, skipping
[INFO] -----
[INFO] Reactor Summary for Maven Project 1.0-SNAPSHOT:
[INFO]
[INFO] Maven Project ..... SUCCESS [ 1.927 s]
[INFO] Server ..... SUCCESS [ 8.677 s]
[INFO] Webapp ..... SUCCESS [ 1.673 s]
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 12.544 s
[INFO] Finished at: 2024-08-11T15:58:53+05:30
[INFO]
```

Finished: SUCCESS



Step 6) If you want to check the war file created in the previous steps, visit the workspace on your Jenkins dashboard or just run the directory commands in your server. Your directories and project name can vary, so you can use the ‘ls‘ command to see the list inside that directory and also keep in mind the directory name is case sensitive.

```
# cd /var/lib/jenkins/workspace/  
#ls
```

```
root@labvm:/home/devasc/Desktop/jenkins# cd /var/lib/jenkins/workspace/  
root@labvm:/var/lib/jenkins/workspace# ls  
BuildPack first-demo PRODUCTION1 QA-TEST1 Test  
CICD_PIPELINE_DEMO gitclone project-build second-demo TEST1  
Clone_Git_Project HelloWorld project-deployment Staging UATest  
clone_Git_Project_SCM production QA-test STAGING1
```

```
root@labvm:/var/lib/jenkins/workspace# cd CICD_PIPELINE_DEMO/
```

```
root@labvm:/var/lib/jenkins/workspace/CICD_PIPELINE_DEMO# ls  
azure-pipelines.yml pom.xml server webapp  
Dockerfile README.md sonar-project.properties webapp.war
```

```
root@labvm:/var/lib/jenkins/workspace/CICD_PIPELINE_DEMO# cd webapp
```

```
root@labvm:/var/lib/jenkins/workspace/CICD_PIPELINE_DEMO/webapp# ls  
pom.xml src target
```

```
root@labvm:/var/lib/jenkins/workspace/CICD_PIPELINE_DEMO/webapp# cd target/
```

```
root@labvm:/var/lib/jenkins/workspace/CICD_PIPELINE_DEMO/webapp/target# ls  
maven-archiver surefire webapp webapp.war
```

Step 7) Now go back to Configure and visit the ‘Post Build Actions‘ tab. Click the drop-down and select ‘Archive the Artifacts‘ from the options. In the field, write down ‘**/*.war‘ as shown in the image below. It will fetch all the directories and get the war file wherever it is present. Click again on Build Now button, and you will now see the



Artifacts in the Jenkins dashboard.

Dashboard > CICD_Pipeline_Demo > Configuration

Configure

- General
- Source Code Management
- Build Triggers
- Build Environment
- Build Steps**
- Post-build Actions

☰ Invoke top-level Maven targets ?

Maven Version: maven_home

Goals: clean compile package

Advanced ▾

Add build step ▾

Post-build Actions

Add post-build action ▾ ✓



Configure

General

Advanced ▾

Source Code Management

Add build step ▾

Build Triggers

Post-build Actions

Build Environment

Build Steps

Post-build Actions

Archive the artifacts

Files to archive ?

**/*.war

Advanced ▾

Add post-build action ▾

Save

Apply



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



Dashboard > CICD_Pipeline_Demo > #3 > Console Output

```
thoughtworks/xstream/xstream/1.3.1/xstream-1.3.1.jar) to field java.util.Properties.defaults
WARNING: Please consider reporting this to the maintainers of com.thoughtworks.xstream.core.util.Fields
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
[INFO] Packaging webapp
[INFO] Assembling webapp [webapp] in [/var/lib/jenkins/workspace/CICD_Pipeline_Demo/webapp/target/webapp]
[INFO] Processing war project
[INFO] Copying webapp resources [/var/lib/jenkins/workspace/CICD_Pipeline_Demo/webapp/src/main/webapp]
[INFO] Webapp assembled in [43 msecs]
[INFO] Building war: /var/lib/jenkins/workspace/CICD_Pipeline_Demo/webapp/target/webapp.war
[INFO] WEB-INF/web.xml already added, skipping
[INFO] -----
[INFO] Reactor Summary for Maven Project 1.0-SNAPSHOT:
[INFO]
[INFO] Maven Project ..... SUCCESS [ 0.134 s]
[INFO] Server ..... SUCCESS [ 2.747 s]
[INFO] Webapp ..... SUCCESS [ 0.890 s]
[INFO] -----
[INFO] BUILD SUCCESS ✓
[INFO] -----
[INFO] Total time: 3.912 s
[INFO] Finished at: 2024-08-11T16:14:18+05:30
[INFO] -----
Archiving artifacts ✓
Finished: SUCCESS ✓
```

Continuous Deployment

Step 8) We need to install Apache Tomcat, and for this, you need to visit the [Tomcat Download](#) page. In the core section, hover over the ‘tar.gz’ link and copy it. Now, use the below commands in your server one by one.

- First, four commands will create one temporary directory and user group to access the file. Here, use the command curl -O ‘paste tomcat download link‘ as shown in the command below.
- Use further commands to create a tomcat directory and extract the gzip file. Just cross-check the version number of the Tomcat that you are downloading and extracting.
- Now the permission of the files needs to be configured with the below commands. In the last command, replace it with your username by which you are accessing the server.

sudo groupadd tomcat



PARSHVANATH CHARITABLE TRUST'S

A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
devasc@labvm:~/Desktop/jenkins$ sudo su  
root@labvm:/home/devasc/Desktop/jenkins# sudo groupadd tomcat
```

```
sudo useradd -s /bin/false -g tomcat -d /opt/tomcat tomcat
```

```
root@labvm:/home/devasc/Desktop/jenkins# sudo useradd -s /bin/false -g tomcat -d  
/opt/tomcat tomcat
```

```
cd
```

```
root@labvm:/home/devasc/Desktop/jenkins# cd
```

```
cd /tmp
```

```
root@labvm:~# cd /tmp
```

```
curl -O https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.109/bin/apache-tomcat-9.0.109.tar.gz
```

```
root@labvm:/tmp# curl -O https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.109/bin/apache-tomcat-9.0.109.tar.gz  
% Total    % Received % Xferd  Average Speed   Time     Time     Current  
          Dload  Upload Total   Spent   Left  Speed  
100 12.4M  100 12.4M    0     0  16.2M      0 --:--:-- --:--:-- --:--:-- 16.2M
```

```
sudo mkdir /opt/tomcat
```

```
root@labvm:/tmp# sudo mkdir /opt/tomcat
```

```
sudo tar xzvf apache-tomcat-9.0.109.tar.gz -C /opt/tomcat --strip-components=1
```

```
root@labvm:/tmp# sudo tar xzvf apache-tomcat-9.0.109.tar.gz -C /opt/tomcat --strip-components=1
```



PARSHVANATH CHARITABLE TRUST'S

A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
apache-tomcat-9.0.109/webapps/docs/images/tomcat.png
apache-tomcat-9.0.109/webapps/docs/images/update.gif
apache-tomcat-9.0.109/webapps/docs/images/void.gif
apache-tomcat-9.0.109/webapps/docs/index.html
apache-tomcat-9.0.109/webapps/docs/introduction.html
apache-tomcat-9.0.109/webapps/docs/jasper-howto.html
apache-tomcat-9.0.109/webapps/docs/jaspicapi/index.html
apache-tomcat-9.0.109/webapps/docs/jdbc-pool.html
apache-tomcat-9.0.109/webapps/docs/jndi-datasource-examples-howto.html
apache-tomcat-9.0.109/webapps/docs/jndi-resources-howto.html
apache-tomcat-9.0.109/webapps/docs/jspapi/index.html
apache-tomcat-9.0.109/webapps/docs/logging.html
apache-tomcat-9.0.109/webapps/docs/manager-howto.html
apache-tomcat-9.0.109/webapps/docs/maven-jars.html
apache-tomcat-9.0.109/webapps/docs/mbeans-descriptors-howto.html
apache-tomcat-9.0.109/webapps/docs/mbeans-descriptors.dtd
apache-tomcat-9.0.109/webapps/docs/monitoring.html
apache-tomcat-9.0.109/webapps/docs/proxy-howto.html
apache-tomcat-9.0.109/webapps/docs/realm-howto.html
apache-tomcat-9.0.109/webapps/docs/rewrite.html
apache-tomcat-9.0.109/webapps/docs/security-howto.html
apache-tomcat-9.0.109/webapps/docs/security-manager-howto.html
apache-tomcat-9.0.109/webapps/docs/servletapi/index.html
apache-tomcat-9.0.109/webapps/docs/setup.html
apache-tomcat-9.0.109/webapps/docs/ssi-howto.html
```

cd /opt/tomcat

sudo chgrp -R tomcat /opt/tomcat

sudo chmod -R g+r conf

sudo chmod g+x conf

```
root@labvm:/tmp# cd /opt/tomcat
root@labvm:/opt/tomcat# sudo chgrp -R tomcat /opt/tomcat
root@labvm:/opt/tomcat# sudo chmod -R g+r conf
root@labvm:/opt/tomcat# sudo chmod g+x conf
root@labvm:/opt/tomcat#
```

cd ..

```
root@labvm:/opt/tomcat# cd ..
```

sudo chown -R root:root tomcat/

```
root@labvm:/opt# whoami
root
root@labvm:/opt# sudo chown -R root:root tomcat/
root@labvm:/opt#
```



Step 9) We need to update the port number from 8080 to 8090 in the server.xml file. We are updating it as this port number is already in use by Jenkins, and we have created 8090 in Azure VM for Tomcat. Use the below commands to edit the file. When you enter the file, click the INSERT button to edit. Now search for a similar code, as shown in the image below. Update the port number to 8090. To save the file, press the *Esc* key, type :*wq* and click on *Enter* button.

cd

```
root@labvm:/opt# cd  
root@labvm:~#
```

cd /opt/tomcat/conf

```
root@labvm:~# cd /opt/tomcat/conf  
root@labvm:/opt/tomcat/conf#
```

vi server.xml

```
root@labvm:/opt/tomcat/conf# vi server.xml
```

```
<Connector port="8090" protocol="HTTP/1.1"  
connectionTimeout="20000"  
redirectPort="8443"  
maxParameterCount="1000"  
/>
```

Step 10) Similarly, we need to edit the '*tomcat-users.xml*' file to update the roles that enable us to deploy files using Tomcat. In the file before *tomcat-users* ending code, paste the below roles code. To save the file press, the *Esc* key, type :*wq* and press *Enter*.

cd



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
root@labvm:/opt/tomcat/conf# cd  
root@labvm:~#
```

cd /opt/tomcat/conf

```
root@labvm:~# cd /opt/tomcat/conf  
root@labvm:/opt/tomcat/conf#
```

vi tomcat-users.xml

```
root@labvm:/opt/tomcat/conf# vi tomcat-users.xml
```

```
<role rolename="manager-gui"/>  
<role rolename="manager-script"/>  
<role rolename="manager-jmx"/>  
<role rolename="manager-status"/>  
<user username="admin" password="admin" roles="manager-gui, manager-script, manager-jmx, manager-status"/>  
<user username="deployer" password="deployer" roles="manager-script"/>  
<user username="tomcat" password="s3cret" roles="manager-gui"/>
```

```
<role rolename="manager-gui"/>  
<role rolename="manager-script"/>  
<role rolename="manager-jmx"/>  
<role rolename="manager-status"/>  
<user username="admin" password="admin" roles="manager-gui, manager-script, manager-jmx, manager-status"/>  
<user username="deployer" password="deployer" roles="manager-script"/>  
<user username="tomcat" password="s3cret" roles="manager-gui"/>  
  
</tomcat-users>
```

Step 11) We also need to update the context.xml file to remove the IP restriction. Use the same steps to edit the file with the below commands.

cd



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
root@labvm:/opt/tomcat/conf# cd  
root@labvm:~#
```

cd /opt/tomcat/webapps/manager/META-INF

```
root@labvm:~# cd /opt/tomcat/webapps/manager/META-INF  
root@labvm:/opt/tomcat/webapps/manager/META-INF#
```

vi context.xml

```
root@labvm:/opt/tomcat/webapps/manager/META-INF# vi context.xml
```

Remove all the content present inside ‘context’ as shown in the image below and save it. :wq <enter key>

```
root@labvm:/opt/tomcat/webapps/manager/META-INF  
File Edit View Search Terminal Help  
<?xml version="1.0" encoding="UTF-8"?>  
<!--  
Licensed to the Apache Software Foundation (ASF) under one or more  
contributor license agreements. See the NOTICE file distributed with  
this work for additional information regarding copyright ownership.  
The ASF licenses this file to You under the Apache License, Version 2.0  
(the "License"); you may not use this file except in compliance with  
the License. You may obtain a copy of the License at  
http://www.apache.org/licenses/LICENSE-2.0  
Unless required by applicable law or agreed to in writing, software  
distributed under the License is distributed on an "AS IS" BASIS,  
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
See the License for the specific language governing permissions and  
limitations under the License.  
-->  
<Context antiResourceLocking="false" privileged="true" >  
 </Context>
```

Step 12) Now, all the files are edited successfully. To update the Tomcat, we need to restart the system to accept all our changes. Use the below commands in the server for shutdown and startup of Tomcat. With this step, Tomcat is ready to deploy our container.

cd



PARSHVANATH CHARITABLE TRUST'S

A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
root@labvm:/opt/tomcat/webapps/manager/META-INF# cd  
root@labvm:~#
```

cd /opt/tomcat/bin/

```
root@labvm:~# cd /opt/tomcat/bin/  
root@labvm:/opt/tomcat/bin#
```

./shutdown.sh

```
root@labvm:/opt/tomcat/bin# ./shutdown.sh  
Using CATALINA_BASE:   /opt/tomcat  
Using CATALINA_HOME:   /opt/tomcat  
Using CATALINA_TMPDIR: /opt/tomcat/temp  
Using JRE_HOME:        /usr  
Using CLASSPATH:       /opt/tomcat/bin/bootstrap.jar:/opt/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.lang.invoke=ALL-UNNAMED --add-opens=java.base/java.lang.reflect=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  
Sep 07, 2025 1:34:54 PM org.apache.catalina.startup.Catalina stopServer  
SEVERE: Could not contact [localhost:8005] (base port [8005] and offset [0]). Tomcat may not be running.  
Sep 07, 2025 1:34:54 PM org.apache.catalina.startup.Catalina stopServer  
SEVERE: Error stopping Catalina
```

./startup.sh

```
root@labvm:/opt/tomcat/bin# ./startup.sh  
Using CATALINA_BASE:   /opt/tomcat  
Using CATALINA_HOME:   /opt/tomcat  
Using CATALINA_TMPDIR: /opt/tomcat/temp  
Using JRE_HOME:        /usr  
Using CLASSPATH:       /opt/tomcat/bin/bootstrap.jar:/opt/tomcat/bin/tomcat-juli.jar  
Using CATALINA_OPTS:  
Tomcat started.
```

Step 13) In the Jenkins Dashboard, click on Manage Jenkins and then visit Manage Plugins.



The screenshot shows the Jenkins Manage Jenkins dashboard. On the left, there's a sidebar with links like 'Build History', 'Manage Jenkins' (which is highlighted with a red box), and 'My Views'. Below that are 'Build Queue' and 'Build Executor Status'. The main area has a message about a new Jenkins version available for download. It also shows a warning about security vulnerabilities in Jenkins 2.452.3 and earlier. On the right, there's a 'System Configuration' section with 'System' and 'Tools' options, and a large 'Plugins' section which is also highlighted with a red box.

Step 14) Click on the *Available* tab and search for the ‘Deploy to Container’ plugin. Select the plugin and click on the ‘Install without restart’ button.

The screenshot shows the Jenkins Plugins page under the 'Available plugins' tab. A search bar at the top contains 'deploy to container'. Below it, a table lists the plugin 'Deploy to container' by 'Artifact Uploaders'. The table includes columns for 'Install', 'Name', and 'Released'. A red box highlights the 'Install' button for the 'Deploy to container' plugin.

The screenshot shows the Jenkins Plugins page under the 'Download progress' tab. On the left, there's a sidebar with 'Updates', 'Available plugins' (which is highlighted with a red box), 'Installed plugins', and 'Advanced settings'. The main area shows a 'Download progress' table with rows for 'Preparation', 'SSH server', 'Deploy to container', and 'Loading plugin extensions'. Each row has a status column with green checkmarks and the word 'Success'. A red checkmark is placed next to the 'Deploy to container' row. At the bottom, there are links to 'Go back to the top page' and 'Restart Jenkins when installation is complete and no jobs are running'.



Step 15) Go back to your Configure window and select the Post-build Actions tab. Select the ‘Add post-build action‘ drop-down button and select the ‘Deploy war/ear to a container‘ plugin. Fill the same path of your war file here, as shown in the image below.

The screenshot shows the Jenkins 'Configure' screen for a pipeline named 'CICD_Pipeline_Demo'. The left sidebar lists configuration tabs: General, Source Code Management, Build Triggers, Build Environment, Build Steps, and Post-build Actions. The 'Post-build Actions' tab is selected and highlighted with a blue background. On the right, under 'Post-build Actions', there are two main sections: 'Archive the artifacts' and 'Deploy war/ear to a container'. In the 'Archive the artifacts' section, the 'Files to archive' field contains '**/*.war'. In the 'Deploy war/ear to a container' section, the 'WAR/EAR files' field also contains '**/*.war'. Both fields have a red checkmark to their right, indicating they are correctly configured. At the bottom of the configuration screen are 'Save' and 'Apply' buttons.



Step 16) Now click on the ‘*Add Container*‘ button and select the ‘*Tomcat 9.x Remote*‘ as we are using version 9 of the Tomcat. Fill in the URL of the same virtual machine with the new port number 8090.

Dashboard > CICD_Pipeline_Demo > Configuration

Configure

- General
- Source Code Management
- Build Triggers
- Build Environment
- Build Steps
- Post-build Actions**

Context path ?

Containers

Tomcat 9.x Remote ✓

Credentials

- none -

+ Add ▾

Tomcat URL ?

http://localhost:8090/ ✓

Advanced ▾

Add Container ▾

On the credentials drop-down button, select Jenkins.



Configure

- General
- Source Code Management
- Build Triggers
- Build Environment
- Build Steps
- Post-build Actions

Containers

Tomcat 9.x Remote

Credentials

- none -

+ Add +

Jenkins

http://localhost:8090/

Step 17) In this window, fill in the username and password that we have used in the ‘tomcat-users.xml’ file (in Step 10). Fill in the ID, description and click on the button ‘Add’.

Jenkins Credentials Provider: Jenkins

Global (Jenkins, nodes, items, all child items, etc)

Username ✓

Treat username as secret ✓

Password ✓

ID ✓

Description ✓

Cancel Add

Step 18) Click on the credentials drop-down button and select the recently created credential. Save all the settings and again click on the ‘Build Now’ button from the Jenkins dashboard. If the Build is successful, the war file will get deployed.



Configure

- General
- Source Code Management
- Build Triggers
- Build Environment
- Build Steps
- Post-build Actions

Containers

Tomcat 9.x Remote

Credentials

admin/******** (tomcat)

- none -

admin/******** (tomcat)

sujata/********

Tomcat 9.x

http://localhost:8090/

Advanced ▾

Add Container ▾

Deploy on failure

Add post-build action ▾

Save

Apply



The screenshot shows the Apache Tomcat 9.0.93 welcome page. At the top, there's a green banner with the text "If you're seeing this, you've successfully installed Tomcat. Congratulations!". Below the banner is a cartoon cat icon. To the right of the cat, there's a section titled "Recommended Reading" with links to "Security Considerations How-To", "Manager Application How-To", and "Clustering/Session Replication How-To". On the right side of the page, there are three buttons: "Server Status", "Manager App", and "Host Manager". Below the banner, there's a "Developer Quick Start" section with links to "Tomcat Setup", "First Web Application", "Realms & AAA", "JDBC DataSources", "Examples", "Servlet Specifications", and "Tomcat Versions". The main content area is divided into three columns: "Managing Tomcat" (with links to "Release Notes", "Changelog", "Migration Guide", and "Security Notices"), "Documentation" (with links to "Tomcat 9.0 Documentation", "Tomcat 9.0 Configuration", and "Tomcat Wiki"), and "Getting Help" (with links to "FAQ and Mailing Lists" and descriptions of three mailing lists: "tomcat-announce", "tomcat-users", and "tomcat-dev").

<http://localhost:8090/webapp/>

The screenshot shows a Jenkins pipeline deployment page. The top banner says "Hello, Welcome to APSIT's Jenkins CICD PIPELINE TOMCAT SERVER DevOps Demo !!". Below the banner, there are several bold text statements: "THIS IS THE DEMO OF JENKINS WITH GITHUB AND TOMCAT SERVER", "THIS APPLICATION IS DEVELOPED BY PROF. SUJATA OAK", and "TODAYS DATE IS 09-SEPTEMBER-2025".

Conclusion: By implementing Jenkins pipelines for Tomcat deployment, software teams achieve full automation of the build, test, and deployment lifecycle of web applications. This experiment demonstrates the practical application of CI/CD concepts in a real-world scenario.



PARSHVANATH CHARITABLE TRUST'S

A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
root@labvm:/home/devasc/Desktop/jenkins# cd  
root@labvm:~# cd /opt/tomcat/bin  
root@labvm:/opt/tomcat/bin# ./startup.sh  
root@labvm:/opt/tomcat/bin# ./shutdown.sh
```



PARSHVANATH CHARITABLE TRUST'S

A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



Academic Year: 2025-26

Semester: V

Class / Branch: TEIT

Subject: DevOps Lab

Name of Instructor: Prof. Sujata Oak

Experiment No. 6

Aim: To implement Jenkins Master-Slave Architecture with Scaling.

Theory:

Objective

To understand and implement Jenkins' distributed build architecture where a central Jenkins Master (Controller) coordinates tasks and multiple Slave (Agent) nodes execute builds in parallel.

Purpose of Distributed Builds

- Large projects often require running builds and tests on different operating systems, environments, or hardware.
- A single server can become a bottleneck. Distributed builds increase **scalability, speed, and fault tolerance**.

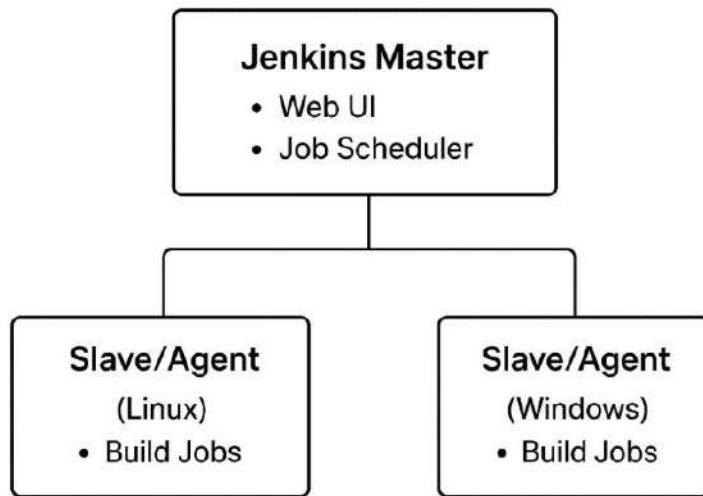
Key Roles

- **Master (Controller):**

- Hosts the Jenkins web UI and job configurations.
- Schedules jobs, monitors nodes, and aggregates build results.
- Decides *what* to build and *where* to build it.

- **Slave (Agent):**

- A remote machine (physical/VM/container) where the actual build steps run.
- Communicates with the master through an SSH or JNLP (Java Web Start) connection.
- Can have specific labels (e.g., linux, windows) to run platform-dependent jobs.



Jenkins Master and Slave Concept

A Jenkins master comes with the basic installation of Jenkins, and in this configuration, the master handles all the tasks for our build system.

If we are working on multiple projects, we may run multiple jobs on each project. Some projects need to run on some nodes, and in this process, we need to configure slaves. [Jenkins slaves connect to the Jenkins master](#) using the Java Network Launch Protocol (JNLP).

The Jenkins master acts to schedule the jobs, assign slaves, and send builds to slaves to execute the jobs.

It will also monitor the slave state (offline or online) and get back the build result responses from slaves and the display build results on the console output. The workload of building jobs is delegated to multiple **slaves**.

Advantages

- Parallel execution → faster CI/CD pipeline.
- Flexibility to run jobs on specific environments.
- Load distribution prevents the master from being overloaded.

Steps to Configure Jenkins Master and Slave Nodes

STEPA: Sign-In to AWS MANAGEMENT CONSOLE



The screenshot shows the AWS EC2 Instance Connect interface. At the top, it displays the instance ID i-09a1cc0a080e9528a (Jenkins-MS) and the connection type as Public IPv4 address (3.108.196.146). The username is set to ubuntu. A note at the bottom states: "Note: In most cases, the default username, ubuntu, is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username." At the bottom right are 'Cancel' and 'Connect' buttons.

Click on Connect

STEP B: Install Jenkins

STEP1: In Jenkins Dashboard Click on Manage Jenkins -> Manage Nodes

The screenshot shows the Jenkins Dashboard. The 'Manage Jenkins' link in the sidebar is highlighted. A prominent warning message states: "New version of Jenkins (2.462.1) is available for download (changelog)." Another message below it says: "Building on the built-in node can be a security issue. You should set the number of executors on the built-in node to 0. See the documentation." A warning about Jenkins 2.452.3 core and libraries is also present. The 'System Configuration' section includes links for System, Tools, Plugins, Nodes, Clouds, and Appearance.



STEP 2: Select New Node and enter the name of the node in the Node Name field.

Select Permanent Agent and click the OK button. Initially, you will get only one option, “Permanent Agent.” Once we have one or more slaves you will get the “Copy Existing Node” option. Click Create

The screenshot shows the Jenkins interface for creating a new node. The top navigation bar includes 'Dashboard', 'Manage Jenkins', 'Nodes', and 'New node'. The main form is titled 'New node'. It has a 'Node name' field containing 'agent2'. Below it is a 'Type' section with two options: 'Permanent Agent' (selected) and 'Copy Existing Node'. At the bottom is a blue 'Create' button.

STEP3: Configure node with below details:

```
sujata@Ubuntu:~/Desktop/JENKINS_LAB$ pwd
/home/sujata/Desktop/JENKINS_LAB
```

```
#find / -type f -name java
```

```
sujata@Ubuntu:~/Desktop/JENKINS_LAB$ su root
Password:
root@Ubuntu:/home/sujata/Desktop/JENKINS_LAB# find / -type f -name java
```

```
/usr/lib/jvm/java-11-openjdk-amd64/bin/java
```



Jenkins

Search (CTRL+K) ? 🔍 🔔 2 🔑 SUJATA OAK ▾

Dashboard > Manage Jenkins > Nodes >

Name ?
agent2

Description ?
This is a demo on Master-Slave Jenkins

Plain text: [Preview](#)

Number of executors ?
1

Remote root directory ?
/home/sujata/Desktop/JENKINS_LAB

Labels ?
agent2

Usage ?
Only build jobs with label expressions matching this node

Launch method ?
Launch agent by connecting it to the controller

Availability ?
Keep this agent online as much as possible

Under 'Node Properties', provide jdk path.



Node Properties

Disable deferred wipeout on this node [?](#)

Disk Space Monitoring Thresholds

Environment variables

List of variables [?](#)

Name

java_home

Value

/usr/lib/jvm/java-11-openjdk-amd64/bin/java

Add

Tool Locations

Save

Jenkins

Dashboard > Manage Jenkins > Nodes >

Nodes

S	Name	Architecture	Clock Difference	Free Disk Space	Free Swap Space	Free Temp Space	Response Time
1	agent1	Linux (amd64)	In sync	5.88 GiB	923.26 MiB	5.88 GiB	189ms
2	agent2		N/A	N/A	N/A	N/A	N/A
3	Built-in Node	Linux (amd64)	In sync	5.88 GiB	923.26 MiB	5.88 GiB	0ms
	Data obtained		21 sec	21 sec	21 sec	21 sec	21 sec

Icon: S M L

Legend



STEP4: On click of ‘Save’ will display the below page with error message. Here Jenkins connect with Slave node using Java Web Start and it needs a port to establish the connection.

To configure JNLP port in global security. Now goto Manage Jenkins -> Security

Agents

TCP port for inbound agents ?

Fixed
50000

Random

Disable

This port has to be allowed to access across firewall, so from Master terminal run the below command,

```
sudo ufw allow 50000/tcp
```

This command will allow port 50000 to listen for request.

```
root@Ubuntu:/home/sujata/Desktop/JENKINS_LAB# sudo ufw allow 50000/tcp
Rule added
Rule added (v6)
```

STEP5: Again coming back to Jenkins and navigate to Nodes -> agent2 which will display two ways to connect with Agent node.

The screenshot shows the Jenkins interface for managing nodes. The left sidebar lists 'Status', 'Delete Agent', 'Configure', 'Build History', 'Load Statistics', and 'Log'. The main panel is titled 'Agent agent2' and contains the following information:

- Description: This is a demo on Master-Slave Jenkins
- Status: Online
- Run from agent command line: (Unix)
curl -s0 http://127.0.0.1:8080/jnlpJars/agent.jar
java -jar agent.jar -url http://127.0.0.1:8080/ -secret cacd8d769874ea4f1a2a28392ffe62d08add0eeb0ea463cced99faf1f707fad0
agent2 -workDir "/home/sujata/Desktop/JENKINS_LAB"
- Actions: Mark this node temporarily offline, Edit description



PARSHVANATH CHARITABLE TRUST'S

A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



To establish connection, run the below command

```
root@Ubuntu:/home/sujata/Desktop/JENKINS_LAB# curl -s0 http://127.0.0.1:8080/jnlpJars/agent.jar
```

```
root@Ubuntu:/home/sujata/Desktop/JENKINS_LAB# java -jar agent.jar -url http://127.0.0.1:8080/ -secret cacd8d769874ea4f1a2a28392  
ffe62d08add0eeb0ea463cced99fa1f707fad0 -name agent2 -workDir "/home/sujata/Desktop/JENKINS_LAB"
```

OUTPUT:

```
INFO: Both error and output logs will be printed to /home/sujata/Desktop/JENKINS_LAB/remoting
Aug 20, 2024 10:24:53 AM hudson.remoting.Launcher createEngine
INFO: Setting up agent: agent2
Aug 20, 2024 10:24:53 AM hudson.remoting.Engine startEngine
INFO: Using Remoting version: 3206.vb_15dcf73f6a_9
Aug 20, 2024 10:24:53 AM org.jenkinsci.remoting.engine.WorkDirManager initializeWorkDir
INFO: Using /home/sujata/Desktop/JENKINS_LAB/remoting as a remoting work directory
Aug 20, 2024 10:24:54 AM hudson.remoting.Launcher$CuiListener status
INFO: Locating server among [http://127.0.0.1:8080/]
Aug 20, 2024 10:24:54 AM org.jenkinsci.remoting.engine.JnlpAgentEndpointResolver resolve
INFO: Remoting server accepts the following protocols: [JNLP4-connect, Ping]
Aug 20, 2024 10:24:54 AM hudson.remoting.Launcher$CuiListener status
INFO: Agent discovery successful
  Agent address: 127.0.0.1
  Agent port: 50000
  Identity: 80:21:52:35:ca:60:ed:97:f1:2a:65:7a:50:b9:27:77
Aug 20, 2024 10:24:54 AM hudson.remoting.Launcher$CuiListener status
INFO: Handshaking
Aug 20, 2024 10:24:54 AM hudson.remoting.Launcher$CuiListener status
INFO: Connecting to 127.0.0.1:50000
Aug 20, 2024 10:24:54 AM hudson.remoting.Launcher$CuiListener status
INFO: Server reports protocol JNLP4-connect-proxy not supported, skipping
Aug 20, 2024 10:24:54 AM hudson.remoting.Launcher$CuiListener status
INFO: Trying protocol: JNLP4-connect
Aug 20, 2024 10:24:54 AM org.jenkinsci.remoting.protocol.impl.BIONetworkLayer$Reader run
INFO: Waiting for ProtocolStack to start.
Aug 20, 2024 10:24:54 AM hudson.remoting.Launcher$CuiListener status
INFO: Remote identity confirmed: 80:21:52:35:ca:60:ed:97:f1:2a:65:7a:50:b9:27:77
Aug 20, 2024 10:24:54 AM hudson.remoting.Launcher$CuiListener status
INFO: Connected
```

This will establish connection with the configured Slave node.



S	Name ↓	Architecture	Clock Difference	Free Disk Space	Free Swap Space	Free Temp Space	Response Time
	agent1		N/A	5.88 GiB	N/A	5.88 GiB	Timed out for last 1 attempts
	agent2	Linux (amd64)	14 sec behind	5.88 GiB	923.26 MiB	5.88 GiB	30003ms
	Built-in Node	Linux (amd64)	In sync	5.88 GiB	923.26 MiB	5.88 GiB	0ms
	Data obtained	1 min 13 sec	1 min 13 sec	1 min 13 sec	1 min 13 sec	1 min 13 sec	1 min 13 sec

Now Jenkins Slave node is ready to run any job. This node's label name should be mentioned in the corresponding Job configuration as below:

STEP 6: Create a New Job in Jenkins dashboard

Enter an item name

master_slave_jenkins_demo20082024
» Required field

Freestyle project
Classic, general-purpose job type that checks out from up to one SCM, executes build steps serially, followed by post-build steps like archiving artifacts and sending email notifications.

Pipeline
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.

Multi-configuration project
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.

Folder
Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different folders.

OK [Create Pipeline](#)



STEP 7: Configure the page with following:

Dashboard > master_slave_jenkins_demo20082024 > Configuration

Configure

General

Description: Demo On Jenkins Master Slave architecture

Source Code Management: Plain text [Preview](#)

Discard old builds ?

GitHub project

This project is parameterized ?

Throttle builds ?

Execute concurrent builds if necessary ?

Restrict where this project can be run ?

Label Expression ?
agent2

Build Steps

Execute shell ?

Command:
See [the list of available environment variables](#)

```
echo "Hello Students, Welcome to session on MASTER SLAVE ARCHITECTURE IN JENKINS!!!"
```

Save Apply

Click on Build-Now, Console Output



The screenshot shows the Jenkins console output for build #1 of the 'master_slave_jenkins_demo20082024' project. The output indicates the build was started by user SUJATA_OAK and running as SYSTEM. It shows the command being run and a welcome message to students. The build status is SUCCESS.

```
Started by user SUJATA_OAK
Running as SYSTEM
Building remotely on agent2 in workspace /home/sujata/Desktop/JENKINS_LAB/workspace/master_slave_jenkins_demo20082024
[master_slave_jenkins_demo20082024] $ /bin/sh -xe /tmp/jenkins8463428541727822031.sh
+ echo Hello Students, Welcome to session on MASTER SLAVE ARCHITECTURE IN JENKINS!!!
Hello Students, Welcome to session on MASTER SLAVE ARCHITECTURE IN JENKINS!!!
Finished: SUCCESS
```

STEP 8: Goto Jenkins Dashboard->Manage Jenkins->Nodes->agent2

The screenshot shows the Jenkins node configuration for 'agent2'. The node is connected and tied to the 'master_slave_jenkins_demo20082024' project. The table below lists the project details.

S	W	Name ↓	Last Success	Last Failure	Last Duration
🕒	☀️	master_slave_jenkins_demo20082024	3 min 53 sec #1	N/A	0.74 sec ➤

Conclusion: Jenkins Master-Slave (Controller-Agent) architecture allows scalable, parallel, and environment-specific builds. It is essential for real-world CI/CD pipelines where multiple teams and platforms are involved.



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



Academic Year: 2025-26

Semester: V

Class / Branch: TEIT

Subject: DevOps Lab

Name of Instructor: Prof. Sujata Oak

Experiment No. 7

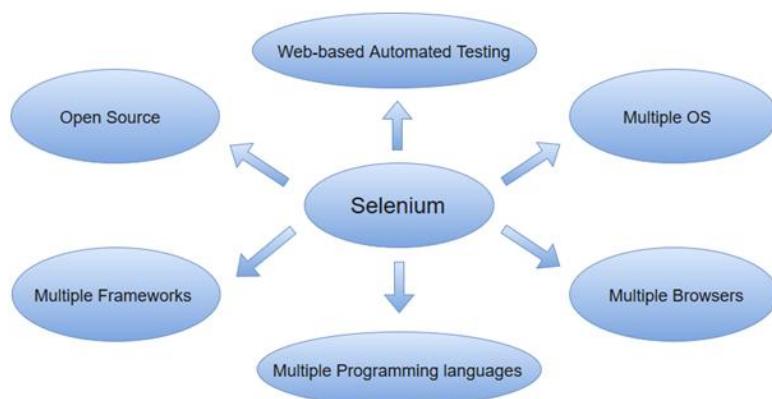
Aim: To implement selenium automation.

Theory:

Selenium is one of the most widely used open source Web UI (User Interface) automation testing suite. It was originally developed by Jason Huggins in 2004 as an internal tool at Thought Works. Selenium supports automation across different browsers, platforms and programming languages.

Selenium can be easily deployed on platforms such as Windows, Linux, Solaris and Macintosh. Moreover, it supports OS (Operating System) for mobile applications like iOS, windows mobile and android.

Selenium supports a variety of programming languages through the use of drivers specific to each language. Languages supported by Selenium include C#, Java, Perl, PHP, Python and Ruby. Currently, Selenium Web driver is most popular with Java and C#. Selenium test scripts can be coded in any of the supported programming languages and can be run directly in most modern web browsers. Browsers supported by Selenium include Internet Explorer, Mozilla Firefox, Google Chrome and Safari.





PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



Selenium can be used to automate functional tests and can be integrated with automation test tools such as **Maven, Jenkins, & Docker** to achieve continuous testing. It can also be integrated with tools such as **TestNG, & JUnit** for managing test cases and generating reports.

Automation Testing

Automation testing uses the specialized tools to automate the execution of manually designed test cases without any human intervention. Automation testing tools can access the test data, controls the execution of tests and compares the actual result against the expected result. Consequently, generating detailed test reports of the system under test.

Steps for Selenium Automation in DevOps on Ubuntu

STEP 1: Selenium IDE-Installation

Selenium IDE is available only as Firefox and Chrome plug-in.

- Launch Mozilla Firefox browser.
- Open URL<https://addons.mozilla.org/en-us/firefox/addon/selenium-ide/> It will redirect you to the official add-on page of Firefox.
- Click on "Add to Firefox" button.



The screenshot shows the Firefox Add-ons page for the "Selenium IDE" extension. The top navigation bar includes the Firefox logo, "Firefox Browser", "ADD-ONS", and links for "Extensions", "Themes", and "More...". Below the navigation is the extension icon, which is a blue square containing a white "Se" monogram with a video camera icon above it. The title "Selenium IDE" is displayed in large, bold, brown font, followed by "by Selenium" in blue. A warning message in a grey box states: "⚠ This add-on is not actively monitored for security by Mozilla. Make sure you trust it before installing." with a "Learn more" link. A descriptive text block explains the extension's purpose: "Selenium IDE is an integrated development environment for Selenium tests. It is implemented as a Firefox extension, and allows you to record, edit, and debug tests." To the right is a blue "Add to Firefox" button. At the bottom of the page, there is a permission dialog box with the heading "Add Selenium IDE? This extension will have permission to:" listing several permissions with a "Learn more" link, and "Cancel" and "Add" buttons.

- A pop-up dialog box will be appeared asking you to add Selenium IDE as extension to your Firefox browser.
- Click on "Add" button.

 **Add Selenium IDE? This extension will have permission to:**

- Access your data for all websites
- Download files and read and modify the browser's download history
- Access browser tabs
- Access browser activity during navigation

[Learn more](#)

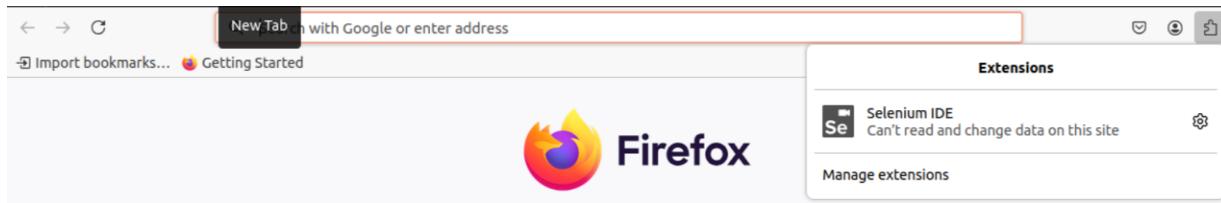
[Cancel](#)

[Add](#)

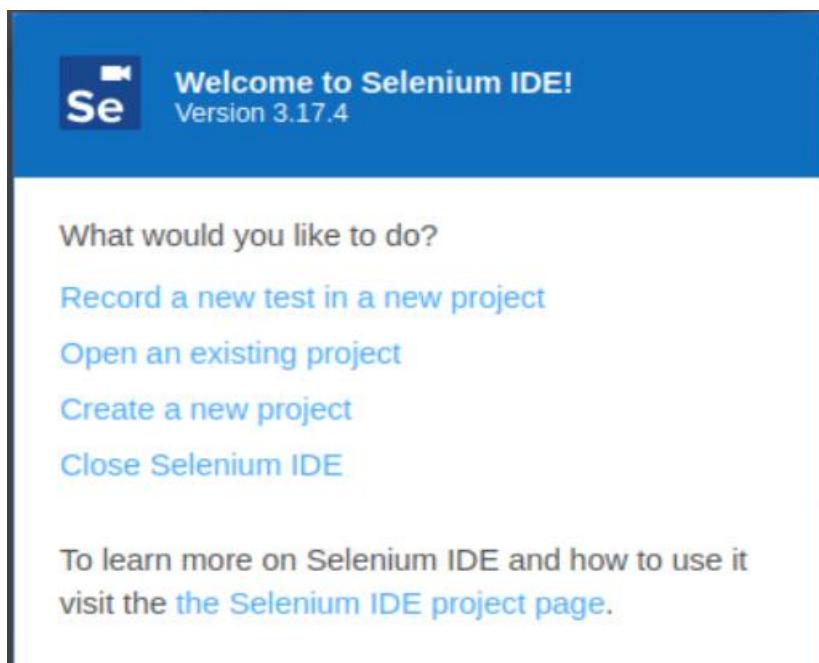
- Restart you Firefox browser.



- Go to the top right corner on your Firefox browser and look for the Selenium IDE icon.



- Click on that icon to launch Selenium IDE.



Click on : Create a new project



PARSHVANATH CHARITABLE TRUST'S

A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



Name your new project



Please provide a name for your new project.

PROJECT NAME

selenium_demo

You can change the name of your project at any time by clicking it and entering a new name.

Cancel

OK

Click OK



Extension: (Selenium IDE) - Selenium IDE - selenium_demo — Mozilla ... X

Project: selenium_demo

Tests + ☰

Search tests... 🔍

Untitled Command Target Value

Playback base URL ▼

Command // []

Target [] 🔍

Value []

Description []

Log Reference ✖

STEP2: Create a basic test case in Selenium ide.

The entire test script creation process in Selenium IDE can be classified into three steps:

1. Recording (recording user interactions with the browser)
 2. Playing back (executing the recorded script)
 3. Saving the test suite
- o Rename the project as "selenium_demo".
 - o Rename the test case as "javaTpoint_test".



Rename test case

TEST CASE NAME

Cancel Rename

Extension: (Selenium IDE) - Selenium IDE - selenium_demo* — Mozilla...

Project: selenium_demo*

Command	Target	Value

seldevops_test*

Search tests...

Playback base URL

Command

Target

Value

Description

Log Reference

- Click on the "Start Recording" Button present on the top right corner on the IDE to start recording the test case.



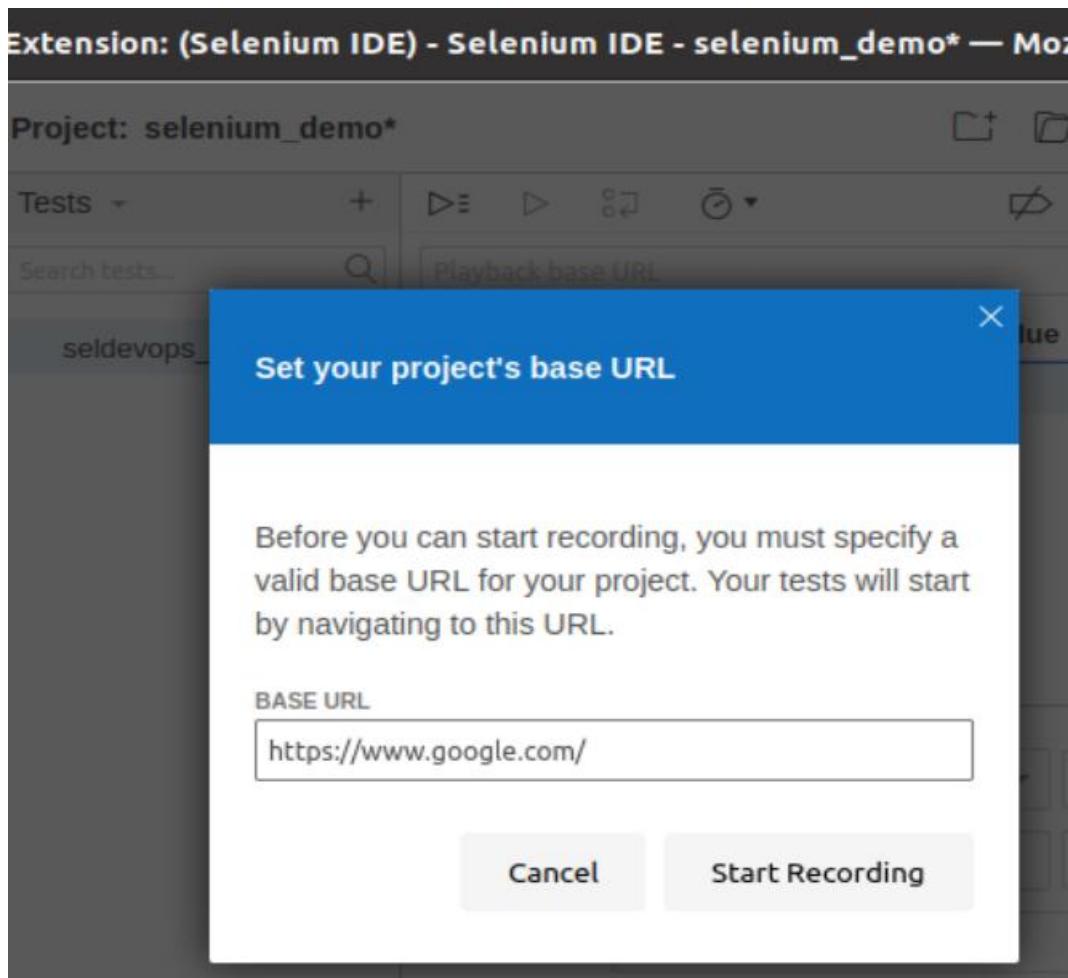
Extension: (Selenium IDE) - Selenium IDE - selenium_demo* — Mozilla...

Project: selenium_demo*



Tests	+	▷Ξ	▷	⟳	⌚	⋮
Search tests...		Playback base URL				
		Command	Target	Value		
seldevops test*						

- Go to your Firefox browser and open URL:www.google.com



- It will redirect you to the Google search engine page.



- Type "Selenium DevOps Tutorials" in the Google search box.
- Hit enter to get the search results.
- Click on the link "How to Use DevOps in Selenium Testing" provided under the URL <https://www.softwaretestinghelp.com/devops-in-selenium-testing/>
- It will redirect you to <https://www.softwaretestinghelp.com/devops-in-selenium-testing/> tutorial web page. Meanwhile, you will get the notifications of the actions performed by the IDE at the extreme right corner of your web browser.



How To Use DevOps In Selenium Testing

By Sruthy · Updated March 10, 2024

This Hands-on Tutorial Explains How to Implement DevOps Practices in Selenium Project and How to Set Up Selenium Project For DevSecOps:

The increasing trend in collaboration has led the Development and the Operation teams to combine their objectives and achieve the organization's goal of shipping software with speed at a higher quality. Quality Engineers also use the shift-left approach and align their activities or tasks with those of developers and operations.

- The Test Editor box now contains the list of all of your interactions with the browser.



Extension: (Selenium IDE) - Selenium IDE - selenium_demo* — Mozilla Firefox

Project: selenium_demo*

Tests + D E ⌂ ⌂ ⌂ ⌂ REC

Search tests... https://www.google.com

	Command	Target	Value
1	✓ open	/	
2	✓ set window size	663x692	
3	✓ type	id=APjFqb	selenium
4	✓ click	css=#jZ2SBf > .wM6W7d > span	
5	✓ run script	window.scrollTo(0,342)	

Command //

Target

Value

- Now, go the IDE and click on the "Stop Recording" button to stop recording your actions further.
- Now, we will proceed to the next step which includes executing the recorded script.

STEP 3: Playing Back

- Click on the "Run Current Test" button present on the tool bar menu of the IDE. It will execute all of your interactions with the browser and gives you an overall summary of the executed test script.
- The Log pane displays the overall summary of the executed test scripts.



Extension: (Selenium IDE) - Selenium IDE - selenium_demo* — Mozilla Firefox

Project: selenium_demo*

Executing: seldevops_test

https://www.google.com

	Command	Target	Value
1	✓ open	/	
2	✓ set window size	663x692	
3	✓ type	id=APjFqb	selenium
4	✓ click	css=#jZ2SBf > .wM6W7d > span	
5	✓ run script	window.scrollTo(0,342)	

Command Target Value Description

Runs: 1 Failures: 0

Log Reference

Running 'seldevops_test'

1. open on / OK 16:19:21

2. setWindowSize on 663x692 OK 16:19:22

3. type on id=APjFqb with value selenium OK 16:19:22

4. click on css=#jZ2SBf > .wM6W7d > span OK 16:19:25

5. runScript on window.scrollTo(0,342) OK 16:19:25

'seldevops_test' completed successfully 16:19:28

Saving the test suite

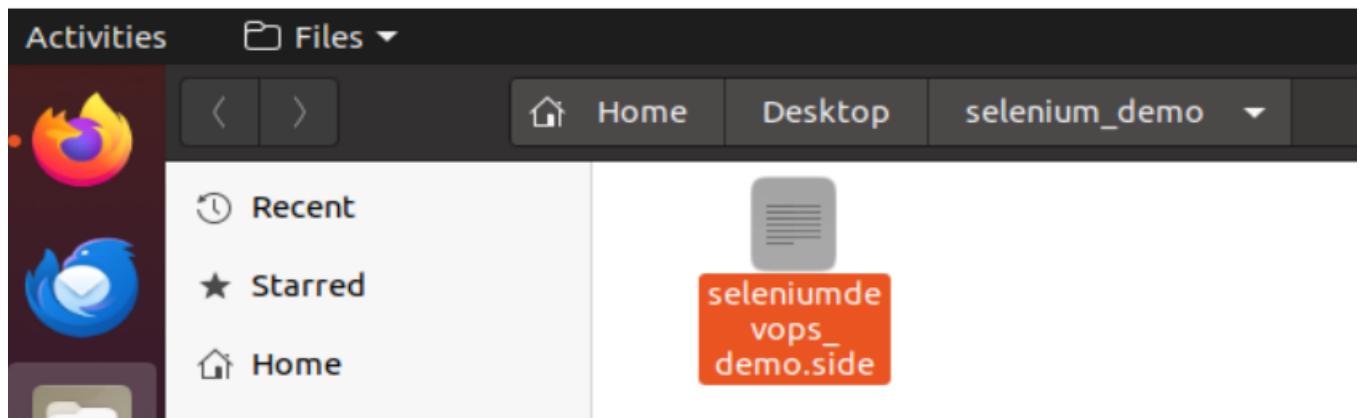
- Click on the save button present on the extreme right corner of the menu bar.
- Save the entire test suite as "SelDevOpsdemo.side" Test.
- The test suite can be found at the location provided in the above steps. Notice that the test script is saved in .side format.



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



Conclusion: This experiment demonstrated how to automate a test case in Selenium ide.



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



Academic Year: 2025-26
Subject: DevOPs Lab (DL)
Subject Lab In-charge: Prof. Sujata Oak

Semester: V Class / Branch: TE IT

EXPERIMENT NO. 08

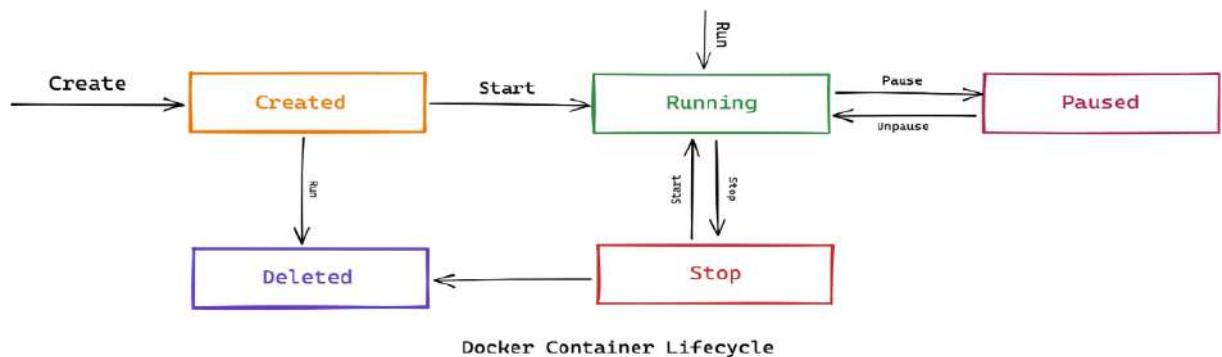
Aim: To demonstrate container lifecycle using various docker commands.

Theory:

Docker is an open-source platform that helps developers **build**, **ship**, and **run** applications in **containers**. A **container** is a lightweight, standalone executable package that includes everything needed to run a piece of software — **code, runtime, libraries, and dependencies**.

There are mainly five states that a container can be in during its lifecycle -

- Created state
- Running state
- Paused state/ Unpaused state
- Stopped state
- Killed/Deleted state



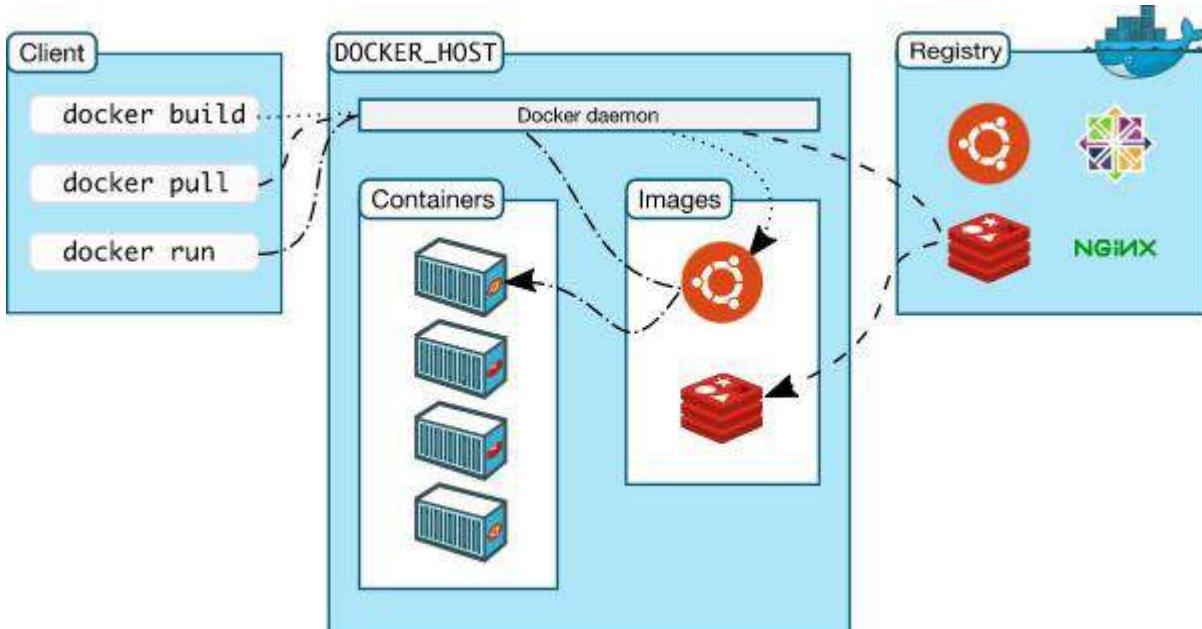


Fig. Architectural overview of Docker

Command

docker --version
docker info
docker login
docker pull <image>
docker images
docker rmi <image>
docker build -t <name> .
docker run <image>
docker run -it <image>
docker run -d <image>
docker run -p 8080:80 <image>
docker ps
docker ps -a
docker stop <container>
docker start <container>
docker restart <container>
docker rm <container>
docker logs <container>
docker exec -it <container> bash

Description

Check Docker version installed
Shows detailed info about Docker system
Login to Docker Hub registry
Download an image from Docker Hub (e.g., docker pull nginx)
List all locally available Docker images
Remove a Docker image from local system
Build an image from a Dockerfile (-t tags the image)
Create and start a container from an image
Run container in interactive mode with terminal
Run container in detached mode (background)
Map port 80 in container to port 8080 on host
List running containers
List all containers (running + stopped)
Stop a running container
Start a stopped container
Restart a container
Remove a stopped container
View logs of a container
Execute command inside a running container (open terminal)



docker inspect <container/image> Get low-level info (JSON) about a container or image
docker top <container> Display processes running inside a container

Check the version of OS: # cat /etc/os-release

Step1: Install Docker using the convenience script

Take sudo privileges.

```
# curl -fsSL https://get.docker.com -o get-docker.sh
```

```
devasc@labvm:~/Desktop/DOCKER_LAB$ sudo su
root@labvm:/home/devasc/Desktop/DOCKER_LAB# curl -fsSL https://get.docker.com -o get-docker.sh
```

```
# sudo sh get-docker.sh
```

```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# sudo sh get-docker.sh
# Executing docker install script, commit: 0d6f72e671ba87f7aa4c6991646a1a5b9f9dae84
Warning: the "docker" command appears to already exist on this system.

If you already have Docker installed, this script can cause trouble, which is
why we're displaying this warning and provide the opportunity to cancel the
installation.

If you installed the current Docker package using this script and are using it
again to update Docker, you can safely ignore this message.

You may press Ctrl+C now to abort this script.
```

```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# ls
get-docker.sh
```

```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# cat get-docker.sh
```

Step2: To verify docker is installed or not:

```
# docker version
```

```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# docker -v
Docker version 27.1.2, build d01f264
```



DOCKER COMMANDS:

1] #docker info: Give details about docker

```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# docker info
Client: Docker Engine - Community
  Version:           27.1.2
  Context:          default
  Debug Mode:       false
  Plugins:
    buildx: Docker Buildx (Docker Inc.)
      Version: v0.16.2
      Path:   /usr/libexec/docker/cli-plugins/docker-buildx
    compose: Docker Compose (Docker Inc.)
      Version: v2.29.1
      Path:   /usr/libexec/docker/cli-plugins/docker-compose

Server:
  Containers: 4
    Running: 0
    Paused: 0
    Stopped: 4
  Images: 8
```

2] #systemctl status docker: To Check docker engine is active or not

#systemctl start docker: To start the docker engine

#systemctl stop docker : To stop the docker engine

```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset: enabled)
   Active: active (running) since Mon 2025-07-28 15:28:05 UTC; 5h 14min left
     TriggeredBy: ● docker.socket
     Docs: https://docs.docker.com
   Main PID: 1401 (dockerd)
     Tasks: 11
       Memory: 107.2M
      CGroup: /system.slice/docker.service
                  └─1401 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock

Jul 28 15:28:04 labvm dockerd[1401]: time="2025-07-28T15:28:04.3291322Z" level=info msg="Docker daemon is running"
Jul 28 15:28:04 labvm dockerd[1401]: time="2025-07-28T15:28:04.6717834Z" level=info msg="Starting containerd"
Jul 28 15:28:05 labvm dockerd[1401]: time="2025-07-28T15:28:05.2305624Z" level=info msg="Containerd is running"
Jul 28 15:28:05 labvm dockerd[1401]: time="2025-07-28T15:28:05.3725995Z" level=info msg="Starting containerd-shim"
Jul 28 15:28:05 labvm dockerd[1401]: time="2025-07-28T15:28:05.3729293Z" level=info msg="Containerd-shim is running"
Jul 28 15:28:05 labvm dockerd[1401]: time="2025-07-28T15:28:05.6039004Z" level=info msg="Starting containerd-remote-api"
Jul 28 15:28:05 labvm dockerd[1401]: time="2025-07-28T15:28:05.6039322Z" level=info msg="Containerd-remote-api is running"
Jul 28 15:28:05 labvm dockerd[1401]: time="2025-07-28T15:28:05.6058814Z" level=info msg="Starting containerd-logs"
Jul 28 15:28:05 labvm systemd[1]: Started Docker Application Container Engine.
```

3] How you login into your Docker Hub Account from CLI?

```
#docker login
```



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# docker login
Log in with your Docker ID or email address to push and pull images from Docker
Hub. If you don't have a Docker ID, head over to https://hub.docker.com/ to crea
te one.
You can log in with your password or a Personal Access Token (PAT). Using a limi
ted-scope PAT grants better security and is required for organizations using SSO
. Learn more at https://docs.docker.com/go/access-tokens/
Username: 18061977
Password:
WARNING! Your password will be stored unencrypted in /root/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credential-stores
Login Succeeded
```

4] docker run: It helps you to run a container on top of your docker engine, but the ingredients that it needs is image name.

```
# docker container run ubuntu cat /etc/os-release
```

```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# docker container run ubuntu cat /etc/os-rel
ease
Unable to find image 'ubuntu:latest' locally
latest: Pulling from library/ubuntu
32f112e3802c: Pull complete
Digest: sha256:a08e551cb33850e4740772b38217fc1796a66da2506d312abe51acda354ff061
Status: Downloaded newer image for ubuntu:latest
PRETTY_NAME="Ubuntu 24.04.2 LTS"
NAME="Ubuntu"
VERSION_ID="24.04"
VERSION="24.04.2 LTS (Noble Numbat)"
VERSION_CODENAME=noble
ID=ubuntu
ID_LIKE=debian
HOME_URL="https://www.ubuntu.com/"
SUPPORT_URL="https://help.ubuntu.com/"
BUG_REPORT_URL="https://bugs.launchpad.net/ubuntu/"
PRIVACY_POLICY_URL="https://www.ubuntu.com/legal/terms-and-policies/privacy-policy"
UBUNTU_CODENAME=noble
LOGO=ubuntu-logo
```

5] #docker images : Lists all images

```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# docker images
REPOSITORY      TAG          IMAGE ID      CREATED        SIZE
ubuntu          latest       65ae7a6f3544   2 weeks ago   78.1MB
```

6] # docker container run ubuntu cat /etc/os-release #Again Hit the same command.



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# docker container run ubuntu cat /etc/os-release
PRETTY_NAME="Ubuntu 24.04.2 LTS"
NAME="Ubuntu"
VERSION_ID="24.04"
VERSION="24.04.2 LTS (Noble Numbat)"
VERSION_CODENAME=noble
ID=ubuntu
ID_LIKE=debian
HOME_URL="https://www.ubuntu.com/"
SUPPORT_URL="https://help.ubuntu.com/"
BUG_REPORT_URL="https://bugs.launchpad.net/ubuntu/"
PRIVACY_POLICY_URL="https://www.ubuntu.com/legal/terms-and-policies/privacy-policy"
UBUNTU_CODENAME=noble
LOGO=ubuntu-logo
root@labvm:/home/devasc/Desktop/DOCKER_LAB#
```

7] To list docker images: #docker images

```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# docker images
REPOSITORY      TAG      IMAGE ID      CREATED      SIZE
ubuntu          latest   65ae7a6f3544   2 weeks ago   78.1MB
```

8] To List Container: #docker container ls

```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# docker container ls
CONTAINER ID      IMAGE      COMMAND      CREATED      STATUS      PORTS      NAMES
530ef56a0a90     ubuntu     "cat /etc/os-release"  8 minutes ago   Exited (0) 8 mi
```

Here the container started and exited at same time.

9] To check container have exited or NOT

```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# docker container ls -a
CONTAINER ID      IMAGE      COMMAND      CREATED      STATUS
PORTS      NAMES
530ef56a0a90     ubuntu     "cat /etc/os-release"  8 minutes ago   Exited (0) 8 mi
```

Open another terminal:

```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# docker container ls -a
CONTAINER ID      IMAGE      COMMAND      CREATED      STATUS
PORTS      NAMES
baad61dfa5c7     ubuntu     "sleep 60"    28 seconds ago  Up 27 se
conds      stupefied_kepler
101cc947a3e8     ubuntu     "sleep 60"    4 minutes ago   Exited (
0) 3 minutes ago      fervent_pike
a1a3a06f8b5e     ubuntu     "cat /etc/os-release..."  5 minutes ago   Exited (
1) 5 minutes ago      awesome_jones
530ef56a0a90     ubuntu     "cat /etc/os-release"  18 minutes ago   Exited (
0) 18 minutes ago      sharp_perlman
499fb530aa8e     ubuntu     "cat /etc/os-release"  22 minutes ago   Exited (
0) 22 minutes ago      boring_herschel
```



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



#docker run hello-world

```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
c1ec31eb5944: Pull complete
Digest: sha256:53cc4d415d839c98be39331c948609b659ed725170ad2ca8eb36951288f81b75
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)
 3. The Docker daemon created a new container from that image which runs the
 executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
 to your terminal.
```

The command used to access the running container Is:

You Can run a ubuntu container with following command:

#docker run -it ubuntu bash

```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# docker run -it ubuntu bash
Unable to find image 'ubuntu:latest' locally
latest: Pulling from library/ubuntu
31e907dcc94a: Pull complete
Digest: sha256:8a37d68f4f73ebf3d4efafbcf66379bf3728902a8038616808f04e34a9ab63ee
Status: Downloaded newer image for ubuntu:latest
root@1d1a286d11ea:/# █
```

3] How to see the image that I just downloaded whether it is available on my machine or not?

docker images: This command is used to show all the pulled images from docker

docker images

```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# docker images
REPOSITORY      TAG      IMAGE ID      CREATED      SIZE
ubuntu          latest   edbfe74c41f8   3 weeks ago   78.1MB
mysql           latest   a82a8f162e18   4 weeks ago   586MB
hello-world     latest   d2c94e258dc8   16 months ago 13.3kB
```

Try to launch a docker image for testing purpose, you can find the images in docker public repository at <https://hub.docker.com>

4] docker pull: This command is used to pull images from the docker repository(hub.docker.com)
Usage: docker pull <image name>

#docker pull mysql



```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# docker pull mysql
Using default tag: latest
latest: Pulling from library/mysql
6e839ac3722d: Pull complete
ad912193ad5f: Pull complete
25d13d87fd8d: Pull complete
004d383c75ef: Pull complete
6d9bbbc82a0b8: Pull complete
81fec07ea550: Pull complete
83357cb2d3a5: Pull complete
8ffe968b82c1: Pull complete
30df9a7ed57: Pull complete
35844ae33cbe: Pull complete
Digest: sha256:86cdfe832c81e39a89cfb63c3fde1683c41cc00ef91e67653c9c1df0ba80f454
Status: Downloaded newer image for mysql:latest
docker.io/library/mysql:latest
```

OR

#docker container run nginx

```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# docker container run nginx
Unable to find image 'nginx:latest' locally
latest: Pulling from library/nginx
59e22667830b: Pull complete
140da4f89dc9: Pull complete
96e47e70491e: Pull complete
2ef442a3816e: Pull complete
4b1e45a9989f: Pull complete
1d9f51194194: Pull complete
f30ffbee4c54: Pull complete
Digest: sha256:84ec966e61a8c7846f509da7eb081c55c1d56817448728924a87ab32f12a72fb
Status: Downloaded newer image for nginx:latest
/docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
/docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
/docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
10-listen-on-ipv6-by-default.sh: info: Getting the checksum of /etc/nginx/conf.d/default.conf
10-listen-on-ipv6-by-default.sh: info: Enabled listen on IPv6 in /etc/nginx/conf.d/default.conf
/docker-entrypoint.sh: Sourcing /docker-entrypoint.d/15-local-resolvers.envsh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-processes.sh
/docker-entrypoint.sh: Configuration complete; ready for start up.
```

Open another terminal:

#docker container run nginx



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY



Department of Information Technology

(NBA Accredited)

```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# docker container ls -a
CONTAINER ID IMAGE COMMAND CREATED STATUS
PORTS NAMES
3739e1fd10a8 nginx "/docker-entrypoint...." 34 seconds ago Up 31 se
conds pedantic_kirch
baad61dfa5c7 ubuntu "sleep 60" 8 minutes ago Exited (
0) 7 minutes ago stupefied_kepler
101cc947a3e8 ubuntu "sleep 60" 12 minutes ago Exited (
0) 11 minutes ago fervent_pike
a1a3a06f8b5e ubuntu "cat /etc/os-release..." 13 minutes ago Exited (
1) 12 minutes ago awesome_jones
530ef56a0a90 ubuntu "cat /etc/os-release" 26 minutes ago Exited (
0) 26 minutes ago sharp_perlman
499fb530aa8e ubuntu "cat /etc/os-release" 29 minutes ago Exited (
0) 29 minutes ago boring_herschel
81cf4991575a 59ab366372d5 "cat /etc/os-release" 38 minutes ago Exited (
0) 38 minutes ago silly_hodgkin
41aa6f73a6bd f1e530d55abc "nginx -g 'daemon of..." 9 months ago Exited (
255) 6 months ago 0.0.0.0:4032->80/tcp, :::4032->80/tcp ssjcoeserver
659dd47038a0 d2c94e258dcf "/hello" 9 months ago Exited (
0) 9 months ago gifted_visvesvaraya
bc8a1f0b194f d2c94e258dcf "/hello" 11 months ago Exited (
0) 11 months ago peaceful_aryabhata
fbfd148039aee d2c94e258dcf "/hello" 11 months ago Exited (
```

Ctrl C to come out of the current screen and again check whether nginx image container is exited.

```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# docker container ls -a
CONTAINER ID IMAGE COMMAND CREATED STATUS
PORTS NAMES
3739e1fd10a8 nginx "/docker-entrypoint...." 4 minutes ago Exited (
0) 13 seconds ago pedantic_kirch
baad61dfa5c7 ubuntu "sleep 60" 12 minutes ago Exited (
```

```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
nginx latest 2cd1d97f893f 13 days ago 192MB
ubuntu latest 65ae7a6f3544 2 weeks ago 78.1MB
root@labvm:/home/devasc/Desktop/DOCKER_LAB#
```

5] docker ps : This command lists the running containers on my system

```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
```

6] docker ps -a : This command list all containers running or exited from the system.

```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# docker ps -a
CONTAINER ID IMAGE COMMAND CREATED STATUS
PORTS NAMES
1d1a286d11ea ubuntu "bash" 12 minutes ago Exited (0) 8 minutes ag
o
fbfd148039aee hello-world "/hello" 22 minutes ago Exited (0) 22 minutes a
go infallible_cerf
```

Now to get the container running : #docker run -it ubuntu bash

```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# docker run -it ubuntu bash
root@a0baee026da8:/#
```



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



In New terminal: docker ps -a

```
devasc@labvm:~/Desktop/DOCKER_LAB$ docker ps -a
CONTAINER ID        IMAGE               COMMAND      CREATED          STATUS
a0baee026da8        ubuntu              "bash"       39 seconds ago   Up 39 seconds
1d1a286d11ea        friendly_easley     "bash"       18 minutes ago  Exited (0) 13 minutes ago
go
fb1d148039aee      hello-world        "/hello"    27 minutes ago  Exited (0) 27 minutes ago
go
```

In First Terminal: ls

You will see the lists of directories available in ubuntu container

```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# docker run -it ubuntu bash
root@19151f20756a:#
root@19151f20756a:#
bin  dev  home  lib64  mnt  proc  run  srv  tmp  var
boot  etc  lib  media  opt  root  sbin  sys  usr
```

Now get exit from ubuntu container

```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# docker run -it ubuntu bash
root@a0baee026da8:#
exit
```

In New terminal: docker ps -a

So You see the container has exited 3 minutes ago.

```
devasc@labvm:~/Desktop/DOCKER_LAB$ docker ps -a
CONTAINER ID        IMAGE               COMMAND      CREATED          STATUS
a0baee026da8        ubuntu              "bash"       3 minutes ago   Exited (0) 6 seconds ago
o                    friendly_easley
```

NOTE: Every container created has a unique container id. That is, from a single image multiple containers can be created. Also, every container will be independent of itself, will be isolated from other container

NOTE: If you don't provide name to your container , the docker-engine gives fancy name to your container.

7] **docker exec:** This command is used to executes the container.

Usage: **docker exec -it <container id> bash**

8] To delete the container: #**docker rm <container-name/container-id>**

```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# docker rm 191
191
```

```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# docker rm a0b 1d1
a0b
1d1
```



```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# docker ps -a
CONTAINER ID        IMAGE               COMMAND             CREATED            STATUS              PORTS
PORTS
fbd148039aee      hello-world        "/hello"          48 minutes ago   Exited (0) 48 minutes ago
go                 infallible_cerf
```

9] To delete the image: #docker rmi <image-name/image-id>

```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# docker images
REPOSITORY          TAG      IMAGE ID      CREATED        SIZE
ubuntu              latest   edbfe74c41f8    3 weeks ago   78.1MB
mysql               latest   a82a8f162e18   4 weeks ago   586MB
hello-world         latest   d2c94e258dcb   16 months ago  13.3kB
```

```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# docker rmi edb
Untagged: ubuntu:latest
Untagged: ubuntu@sha256:8a37d68f4f73ebf3d4efafbcf66379bf3728902a8038616808f04e34a9ab63ee
Deleted: sha256:edbfe74c41f8a3501ce542e137cf28ea04dd03e6df8c9d66519b6ad761c2598a
Deleted: sha256:f36fd4bb7334b7ae3321e3229d103c4a3e7c10a263379cc6a058b977edfb46de
```

```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# docker images
REPOSITORY          TAG      IMAGE ID      CREATED        SIZE
mysql               latest   a82a8f162e18   4 weeks ago   586MB
hello-world         latest   d2c94e258dcb   16 months ago  13.3kB
```

10] Commands related to containers:

docker start <container-id> : Start the stop container
docker stop <container-id> : Stop the running container
docker pause <container-id> : Pause the processes in running container
docker kill <container id> : Kill the container.

Task 2: HOW TO SETUP AND CONFIGURE MYSQL DATABASE INSIDE DOCKER CONTAINER?

MySQL is the single most popular relational database tool.

MySQL is popular because it is simple yet powerful. Here are its best features:

- Relational: follows the relational model and uses SQL to manage databases.
- Open-source (GNU license): the community loves it. Companies love it.
- Scalable: can handle applications from small-sized to enterprise-level.
- Secure: offers user authentication, access management, and encryption.
- High-performance: known for its speed and efficiency in handling complex queries and large volumes of data.
- Replication and backup: it has options for data replication and backup, allowing for disaster recovery strategies.

Step 1: To pull the image of mysql from docker hub



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



#docker pull mysql:latest

```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# docker pull mysql
Using default tag: latest
latest: Pulling from library/mysql
6e839ac3722d: Pull complete
ad912193ad5f: Pull complete
25d13d87fd8d: Pull complete
004d383c75ef: Pull complete
6d9bbc82a0b8: Pull complete
81fec07ea550: Pull complete
83357cb2d3a5: Pull complete
8ffe968b82c1: Pull complete
30dfd9a7ed57: Pull complete
35844ae33cbe: Pull complete
Digest: sha256:86cdfe832c81e39a89cfb63c3fde1683c41cc00ef91e67653c9c1df0ba80f454
Status: Downloaded newer image for mysql:latest
docker.io/library/mysql:latest
```

EXPLANATION: The code is a Docker command, not SQL.

- It's used to download the latest version of the MySQL Docker image.
- "docker pull" is a command that tells Docker to download an image from Docker Hub.
- "mysql:latest" specifies the image to download.
- "mysql" is the name of the image and "latest" is the tag.

Step 2: List the mysql images

#docker images

```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# docker images
REPOSITORY      TAG      IMAGE ID      CREATED       SIZE
mysql           latest   a82a8f162e18   4 weeks ago   586MB
hello-world     latest   d2c94e258dcf   16 months ago  13.3kB
```

Docker images are blueprints for building containers. Just like a blueprint allows you to build a house, a Docker image contains all the necessary instructions and components to create a running instance of an application or service.

STEP 3: Running and Managing a MySQL Server Container

Now, let's create our first container from the mysql image. Here is the command we will use:

```
$ docker run --name test-mysql -e MYSQL_ROOT_PASSWORD=strong_password -d mysql
```

EXPLANATION:

- ✓ **run:** creates a new container or starts an existing one
- ✓ **--name CONTAINER_NAME:** gives the container a name. The name should be readable and short. In our case, the name is test-mysql.
- ✓ **-e ENV_VARIABLE=value:** the -e tag creates an environment variable that will be accessible within the container. It is crucial to set **MYSQL_ROOT_PASSWORD** so that we can run SQL commands later



from the container. Make sure to store your strong password somewhere safe (not your brain).

- ✓ -d: short for detached, the -d tag makes the container run in the background. If you remove this tag, the command will keep printing logs until the container stops.
- ✓ image_name: the final argument is the image name the container will be built from. In this case, our image is mysql.
- ✓

If the command returns a long string of gibberish (the container ID), it means the container has started. You can check its status with docker ps:

```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# docker run --name test-mysql -e MYSQL_ROOT_PASSWORD=sujata -d mysql
0e2b27de5979ae2bfa7c4e464841157796500458dfb194c18bd4716b63440de4
```

In New Terminal:

```
devasc@labvm:~/Desktop/DOCKER_LAB$ docker ps
CONTAINER ID        IMAGE       COMMAND             CREATED          STATUS
PORTS               NAMES
0e2b27de5979        mysql      "docker-entrypoint.s..."   35 seconds ago   Up 34 seconds
            3306/tcp, 33060/tcp   test-mysql_
```

Step 4: To access the terminal inside your container, you can use the following command:

```
$ docker exec -it container_name bash
```

This will launch a bash session.

Connecting to the MySQL Server Container Locally :

All MySQL containers launch a MySQL server that includes everything to create and manage databases using SQL. To connect to the server, containers also come with a MySQL client that lets us run SQL queries. The client is just a fancy name for the mysql terminal command. Let's use it inside test-mysql's terminal:

1. Open the bash terminal of test-mysql:

```
$ docker exec -it test-mysql bash
```

```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# docker exec -it test-mysql bash
bash-5.1#
```

2. Connect to the client as a root user:



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



We are using the -u tag to specify the username (root) and adding the -p tag to enter the password when prompted.

```
$ mysql -u root -p
```

```
Enter password: ...
```

```
mysql>
```

```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# docker exec -it test-mysql bash
bash-5.1# mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 9.0.1 MySQL Community Server - GPL

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
mysql>
```

```
mysql> SELECT 'hello-world!!!';
```

```
mysql> SELECT 'hello-world!!!';
+-----+
| hello-world!!! |
+-----+
| hello-world!!! |
+-----+
1 row in set (0.00 sec)
```

```
mysql> show databases;
```



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
4 rows in set (0.00 sec)
```

```
mysql> use mysql
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A
Database changed
```

```
mysql> SELECT DATABASE();
+-----+
| DATABASE() |
+-----+
| mysql |
+-----+
1 row in set (0.00 sec)
```

```
mysql> SHOW TABLES;
+-----+
| Tables_in_mysql |
+-----+
| columns_priv
| component
| db
| default_roles
| engine_cost
| func
| general_log
| global_grants
| gtid_executed
+-----+
```



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



Field	Type	Null	Key	Default	Extra
Host	char(255)	NO	PRI		
Db	char(64)	NO	PRI		
User	char(32)	NO	PRI		
Select_priv	enum('N','Y')	NO		N	
Insert_priv	enum('N','Y')	NO		N	
Update_priv	enum('N','Y')	NO		N	
Delete_priv	enum('N','Y')	NO		N	
Create_priv	enum('N','Y')	NO		N	
Drop_priv	enum('N','Y')	NO		N	
Grant_priv	enum('N','Y')	NO		N	
References_priv	enum('N','Y')	NO		N	
Index_priv	enum('N','Y')	NO		N	

```
mysql> DESCRIBE db;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| Host  | char(255) | NO  | PRI |        |       |
| Db    | char(64)  | NO  | PRI |        |       |
| User  | char(32)  | NO  | PRI |        |       |
| Select_priv | enum('N','Y') | NO  |      |        |       |
| Insert_priv | enum('N','Y') | NO  |      |        |       |
| Update_priv | enum('N','Y') | NO  |      |        |       |
| Delete_priv | enum('N','Y') | NO  |      |        |       |
| Create_priv | enum('N','Y') | NO  |      |        |       |
| Drop_priv  | enum('N','Y') | NO  |      |        |       |
| Grant_priv  | enum('N','Y') | NO  |      |        |       |
| References_priv | enum('N','Y') | NO  |      |        |       |
| Index_priv  | enum('N','Y') | NO  |      |        |       |
+-----+-----+-----+-----+-----+-----+
mysql> SELECT NOW();
+-----+
| NOW() |
+-----+
| 2024-08-25 18:10:04 |
+-----+
1 row in set (0.00 sec)
```

Conclusion: In this experiments student have learnt how to deal with containerization technology using various docker commands.



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



Academic Year: 2024-25

Subject: DevOPs Lab (DL)

Subject Lab In-charge: Prof. Sujata Oak

ter: V Class / Branch: TE IT

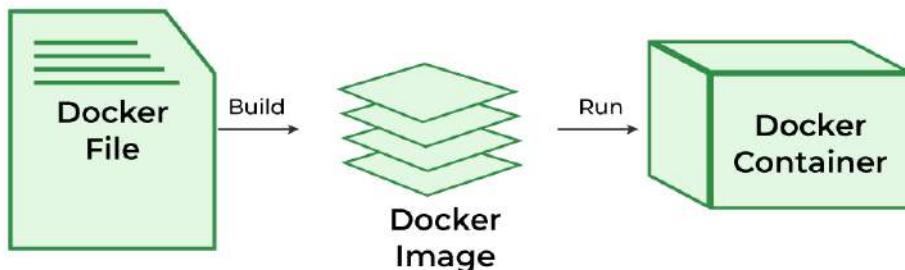
EXPERIMENT NO. 09

Aim: To build an image for a sample web application from a CLI and docker file using various docker file instructions

Theory: The Dockerfile uses DSL (Domain Specific Language) and contains instructions for generating a Docker image. Dockerfile will define the processes to quickly produce an image. While creating your application, you should create a Dockerfile in order since the Docker daemon runs all of the instructions from top to bottom.

An artifact with several layers and a lightweight, compact stand-alone executable package that contains all of the components required to run a piece of software, including the code, a runtime, libraries, environment variables, and configuration files is called a [Docker image](#).

A container is a runtime instance of an image. Containers make development and deployment more efficient since they contain all the dependencies and parameters needed for the application it runs completely isolated from the host environment.



Dockerfile commands/Instructions

1. FROM

- Represents the base image(OS), which is the command that is executed first before any other commands.

Syntax

FROM <ImageName>

2. COPY

- *The copy command is used to copy the file/folders to the image while building the image.*



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



Syntax:

COPY <Source> <Destination>

3] RUN

- Scripts and commands are run with the RUN instruction. The execution of RUN commands or instructions will take place while you create an image on top of the prior layers (Image).

Syntax

RUN <Command + ARGS>

4] CMD

- *The main purpose of the CMD command is to start the process inside the container and it can be overridden.*

Syntax

CMD [command + args]

Stages of Creating Docker Image from Dockerfile

The following are the stages of creating docker image form Dockerfile:

1. Create a file named Dockerfile.
2. Add instructions in Dockerfile.
3. Build Dockerfile to create an image.
4. Run the image to create a container.

IMPLEMENTATION:

PART I: Containerize an application using docker CLI Commands:

Let's create an nginx webserver, it is a web server platform which helps to host your web applications.

STEP1: Download nginx official image and then containerized your web application in it.

#docker images

```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# docker images
REPOSITORY      TAG      IMAGE ID      CREATED      SIZE
mysql           latest   245a6c909dc0   11 days ago   921MB
nginx           latest   2cd1d97f893f   2 weeks ago   192MB
ubuntu          latest   65ae7a6f3544   2 weeks ago   78.1MB
```



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
# docker rmi mysql nginx ubuntu
```

```
#docker images
```

```
devasc@labvm:~/Desktop/sujata-docker$ sudo su
root@labvm:/home/devasc/Desktop/sujata-docker# docker images
REPOSITORY      TAG      IMAGE ID      CREATED      SIZE
REPOSITORY      TAG      IMAGE ID      CREATED      SIZE
```

```
#docker ps -a
```

```
root@labvm:/home/devasc/Desktop/sujata-docker# docker ps -a
CONTAINER ID      IMAGE      COMMAND      CREATED      STATUS      PORTS      NAMES
root@labvm:/home/devasc/Desktop/sujata-docker#
```

```
#docker pull nginx
```

```
root@labvm:/home/devasc/Desktop/sujata-docker# docker pull nginx:latest
latest: Pulling from library/nginx
59e22667830b: Pull complete
140da4f89dc5: Pull complete
96e47e70491e: Pull complete
2ef442a3816e: Pull complete
4b1e45a9989f: Pull complete
1d9f51194194: Pull complete
f30ffbee4c54: Pull complete
Digest: sha256:84ec966e61a8c7846f509da7eb081c55c1d56817448728924a87ab32f12a72fb
Status: Downloaded newer image for nginx:latest
docker.io/library/nginx:latest
```

```
#docker images
```

```
root@labvm:/home/devasc/Desktop/sujata-docker# docker images
REPOSITORY      TAG      IMAGE ID      CREATED      SIZE
nginx          latest    2cd1d97f893f    2 weeks ago   192MB
```

STEP2: Run the container from nginx image

```
# docker run --name webserver1 5ef
```

```
root@labvm:/home/devasc/Desktop/sujata-docker# docker run --name webserver1 nginx
/docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
/docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
/docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
10-listen-on-ipv6-by-default.sh: info: Getting the checksum of /etc/nginx/conf.d/default.conf
10-listen-on-ipv6-by-default.sh: info: Enabled listen on IPv6 in /etc/nginx/conf.d/default.conf
/docker-entrypoint.sh: Sourcing /docker-entrypoint.d/15-local-resolvers.envsh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-processes.sh
/docker-entrypoint.sh: Configuration complete; ready for start up
2025/08/02 18:53:58 [notice] 1#1: using the "epoll" event method
2025/08/02 18:53:58 [notice] 1#1: nginx/1.29.0
2025/08/02 18:53:58 [notice] 1#1: built by gcc 12.2.0 (Debian 12.2.0-14+deb12u1)
2025/08/02 18:53:58 [notice] 1#1: OS: Linux 5.4.0-37-generic
2025/08/02 18:53:58 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1048576:1048576
2025/08/02 18:53:58 [notice] 1#1: start worker processes
2025/08/02 18:53:58 [notice] 1#1: start worker process 28
2025/08/02 18:53:58 [notice] 1#1: start worker process 29
```



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



In another terminal

```
#docker ps -a
```

```
devasc@labvm:~/Desktop/sujata-docker$ sudo su
root@labvm:/home/devasc/Desktop/sujata-docker# docker ps -a
CONTAINER ID   IMAGE      COMMAND           CREATED          STATUS
              PORTS     NAMES
1007983ca28c  nginx      "/docker-entrypoint...."   54 seconds ago   Up 54 seconds
               80/tcp    webserver1
```

In previous terminal: ctrl+C ie; exit from container

In another terminal

```
#docker ps -a
```

```
root@labvm:/home/devasc/Desktop/sujata-docker# docker ps -a
CONTAINER ID   IMAGE      COMMAND           CREATED          STATUS
              PORTS     NAMES
1007983ca28c  nginx      "/docker-entrypoint...."   About a minute ago   Exited (0)
               7 seconds ago                  webserver1
```

Remove the container:

```
root@labvm:/home/devasc/Desktop/sujata-docker# docker container rm 100
100
root@labvm:/home/devasc/Desktop/sujata-docker# docker ps -a
CONTAINER ID   IMAGE      COMMAND           CREATED          STATUS      PORTS     NAMES
```

In terminal 1:

```
# docker run -it -p 3031:80 --name server1 nginx:latest bash
```

```
root@labvm:/home/devasc/Desktop/DOCKER_LAB# docker run -it -p 3031:80 --name nse
rver1 nginx:latest bash
```

In Another Terminal:

```
#docker ps -a
```

```
root@labvm:/home/devasc/Desktop/sujata-docker# docker ps -a
CONTAINER ID   IMAGE      COMMAND           CREATED          STATUS
              PORTS     NAMES
0b847b3b176c  nginx:latest  "/docker-entrypoint...."   30 seconds ago   Up 30 seconds
               0.0.0.0:3031->80/tcp, :::3031->80/tcp   server1
```



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



Lets create a static website inside container. I need to go to the location where my index.html file is:

```
# cd /usr/share/nginx/html/
```

```
root@0b847b3b176c:/# cd /usr/share/nginx/html/
```

```
root@0b847b3b176c:/usr/share/nginx/html#ls
```

```
root@0b847b3b176c:/usr/share/nginx/html# ls
50x.html  index.html
root@0b847b3b176c:/usr/share/nginx/html#
```

Rename the default index.html to index.html_backup

```
root@0b847b3b176c:/usr/share/nginx/html#
```

```
root@0b847b3b176c:/usr/share/nginx/html# mv index.html index.html_backup
```

```
#nano index.html
```

```
root@0b847b3b176c:/usr/share/nginx/html# nano index.html
```

Nano not found: Because the container that I am running inside the shell says that nano application is not available inside the container. So first install nano: apt install nano

```
root@0b847b3b176c:/usr/share/nginx/html# nano index.html
bash: nano: command not found
```

```
root@0b847b3b176c:/usr/share/nginx/html# apt install nano
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Package nano is not available, but is referred to by another package.
This may mean that the package is missing, has been obsoleted, or
is only available from another source

E: Package 'nano' has no installation candidate
```

```
root@0b847b3b176c:/usr/share/nginx/html# apt update
```



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
root@0b847b3b176c:/usr/share/nginx/html# apt update
Get:1 http://deb.debian.org/debian bookworm InRelease [151 kB]
Get:2 http://deb.debian.org/debian bookworm-updates InRelease [55.4 kB]
Get:3 http://deb.debian.org/debian-security bookworm-security InRelease [48.0 kB]
Get:4 http://deb.debian.org/debian bookworm/main amd64 Packages [8793 kB]
Get:5 http://deb.debian.org/debian bookworm-updates/main amd64 Packages [6916 B]
Get:6 http://deb.debian.org/debian-security bookworm-security/main amd64 Packages [272 kB]
Fetched 9327 kB in 2s (4129 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
2 packages can be upgraded. Run 'apt list --upgradable' to see them.
```

#apt install nano

```
root@0b847b3b176c:/usr/share/nginx/html# apt install nano
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libgpm2 libncursesw6
Suggested packages:
  gpm hunspell
```

```
root@0b847b3b176c:/usr/share/nginx/html# nano index.html
```

```
<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<title> Login Page </title>
<style>
Body {
  font-family: Calibri, Helvetica, sans-serif;
  background-color: pink;
}
button {
  background-color: #4CAF50;
  width: 100%;
  color: orange;
  padding: 15px;
  margin: 10px 0px;
  border: none;
  cursor: pointer;
}
form {
  border: 3px solid #f1f1f1;
}
input[type=text], input[type=password] {
```



```
width: 100%;  
margin: 8px 0;  
padding: 12px 20px;  
display: inline-block;  
border: 2px solid green;  
box-sizing: border-box;  
}  
button:hover {  
    opacity: 0.7;  
}  
.cancelbtn {  
    width: auto;  
    padding: 10px 18px;  
    margin: 10px 5px;  
}  
  
.container {  
    padding: 25px;  
    background-color: lightblue;  
}  
</style>  
</head>  
<body>  
    <center> <h1> <b>Student Login Form Designed by Sujata Oak</b> </h1> </center>  
    <form>  
        <div class="container">  
            <label>Username : </label>  
            <input type="text" placeholder="Enter Username" name="username" required>  
            <label>Password : </label>  
            <input type="password" placeholder="Enter Password" name="password" required>  
            <button type="submit">Login</button>  
            <input type="checkbox" checked="checked"> Remember me  
            <button type="button" class="cancelbtn"> Cancel</button>  
            Forgot <a href="#"> password? </a>  
        </div>  
    </form>  
</body>  
</html>
```



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



To check nginx service status:

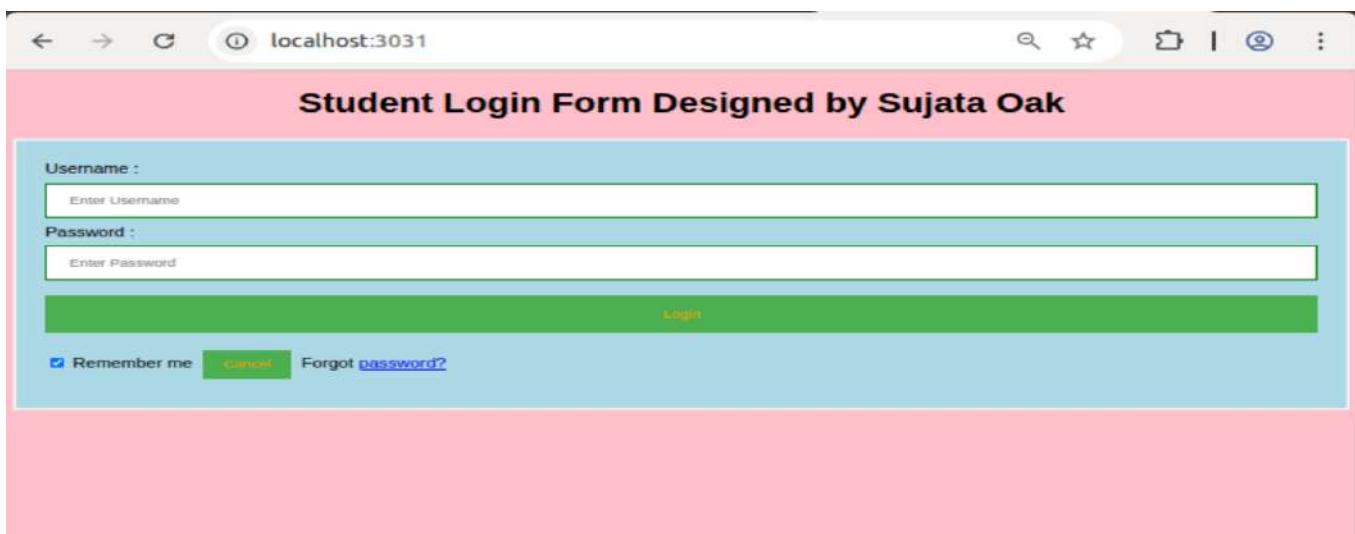
```
root@0b847b3b176c:/usr/share/nginx/html# service nginx status
nginx is not running ... failed!
```

```
#service nginx start
```

```
nginx: [warn] Pid file /var/run/nginx.pid doesn't exist or is not accessible.
root@0b847b3b176c:/usr/share/nginx/html# service nginx start
2025/08/02 19:21:08 [notice] 177#177: using the "epoll" event method
2025/08/02 19:21:08 [notice] 177#177: nginx/1.29.0
2025/08/02 19:21:08 [notice] 177#177: built by gcc 12.2.0 (Debian 12.2.0-1)
2025/08/02 19:21:08 [notice] 177#177: OS: Linux 5.4.0-37-generic
2025/08/02 19:21:08 [notice] 177#177: getrlimit(RLIMIT_NOFILE): 1048
2025/08/02 19:21:08 [notice] 178#178: start worker processes
2025/08/02 19:21:08 [notice] 178#178: start worker process 179
2025/08/02 19:21:08 [notice] 178#178: start worker process 180
```

STEP 3:

GOTO BROWSER: localhost:3031



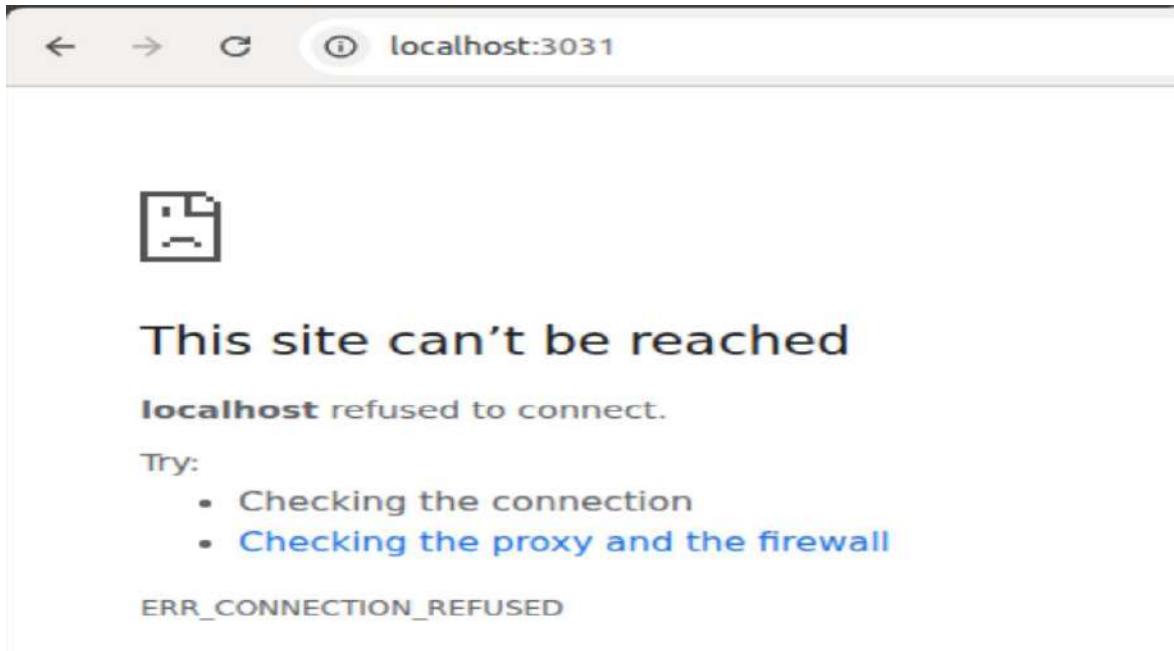
To See the logs on first terminal:

```
root@0b847b3b176c:/usr/share/nginx/html# 172.17.0.1 - - [02/Aug/2025:19:21:44 +0000] "GET / HTTP/1.1" 200 1758 "-" "Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/138.0.0.0 Safari/537.36" "-"
2025/08/02 19:21:44 [error] 180#180: *1 open() "/usr/share/nginx/html/favicon.ico" failed (2: No such file or directory), client: 172.17.0.1, server: localhost, request: "GET /favicon.ico HTTP/1.1", host: "localhost:3031", referrer: "http://localhost:3031/"
172.17.0.1 - - [02/Aug/2025:19:21:44 +0000] "GET /favicon.ico HTTP/1.1" 404 555 "http://localhost:3031/" "Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/138.0.0.0 Safari/537.36" "-"
```

```
root@labvm:/home/devasc/Desktop/sujata-docker# docker stop 0b8
0b8
```



Goto browser → Refresh page . Your Container is stopped now



The screenshot shows a web browser window with the URL `localhost:3031`. The page content includes a large error icon, the text "This site can't be reached", and the message "localhost refused to connect". It also provides troubleshooting steps: "Try:" followed by "Checking the connection" and "Checking the proxy and the firewall". The error code "ERR_CONNECTION_REFUSED" is visible at the bottom.

```
root@labvm:/home/devasc/Desktop/sujata-docker# docker start 0b8
0b8
```

Goto browser → Refresh page . Your Container is not started

```
root@labvm:/home/devasc/Desktop/sujata-docker# docker restart 0b8
0b8
```

Goto browser → Refresh page . Your Container is not restarted



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



This site can't be reached

localhost refused to connect.

Try:

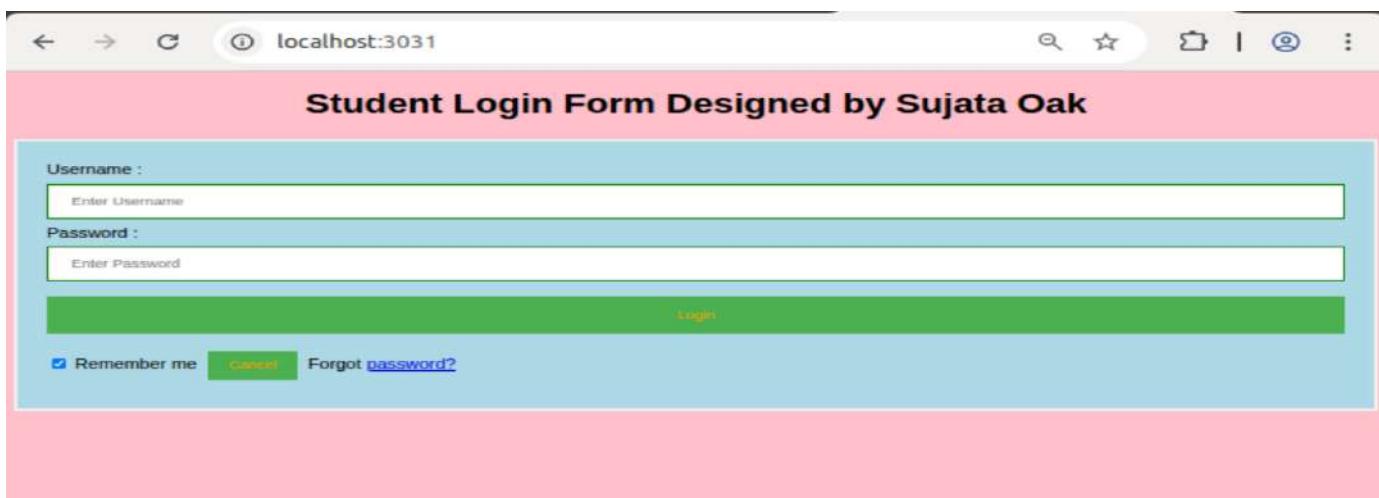
- Checking the connection
- [Checking the proxy and the firewall](#)

ERR_CONNECTION_REFUSED

```
# docker exec 0b8 service nginx start
```

```
root@labvm:/home/devasc/Desktop/sujata-docker# docker exec 0b8 service nginx start
2025/08/02 19:29:29 [notice] 18#18: using the "epoll" event method
2025/08/02 19:29:29 [notice] 18#18: nginx/1.29.0
2025/08/02 19:29:29 [notice] 18#18: built by gcc 12.2.0 (Debian 12.2.0-14+deb12u1)
2025/08/02 19:29:29 [notice] 18#18: OS: Linux 5.4.0-37-generic
2025/08/02 19:29:29 [notice] 18#18: getrlimit(RLIMIT_NOFILE): 1048576:1048576
2025/08/02 19:29:29 [notice] 19#19: start worker processes
2025/08/02 19:29:29 [notice] 19#19: start worker process 20
2025/08/02 19:29:29 [notice] 19#19: start worker process 21
```

Goto Browser and refresh it:





#docker pause 0b8

```
2023/08/02 19:19:17.00 [process] 19819: start worker process 21
root@labvm:/home/devasc/Desktop/sujata-docker# docker pause 0b8
0b8
```

School Library Login Page

localhost:3031

Student Login Form Designed by Sujata Oak

Username :

Password :

Remember me [Cancel](#) [Forgot password?](#)

#docker unpause 0b8

```
root@labvm:/home/devasc/Desktop/sujata-docker# docker unpause 0b8
0b8
```

Firstly stop the container:

```
root@labvm:/home/devasc/Desktop/sujata-docker# docker stop 0b8
0b8
```

Then, Remove the Container

```
root@labvm:/home/devasc/Desktop/sujata-docker# docker container rm 0b8
0b8
```

To Verify container is removed or not:

```
root@labvm:/home/devasc/Desktop/sujata-docker# docker ps -a
CONTAINER ID        IMAGE               COMMAND             CREATED            STATUS              PORTS      NAMES
```



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



PART II: DOCKERFILE

Creating a Docker Image for your Application:

This is the recommended workflow for creating your own Docker image for your application:

1. Write a Dockerfile for your application.
2. Build the image with docker build command.
3. Host your Docker image on a registry.
4. Pull and run the image on the target machine.

Docker builds images automatically by reading the instructions from a Dockerfile. It is a text file that contains all commands needed to build a given image.

STEP 1: # git clone https://github.com/sujataoak799/nginx-dockerfile.git

```
root@labvm:/home/devasc/Desktop/sujata-docker# git clone https://github.com/sujataoak799/nginx-dockerfile.git
Cloning into 'nginx-dockerfile'...
remote: Enumerating objects: 8, done.
remote: Counting objects: 100% (8/8), done.
remote: Compressing objects: 100% (6/6), done.
remote: Total 8 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (8/8), 2.63 KiB | 674.00 KiB/s, done.
```

```
root@labvm:/home/devasc/Desktop/sujata-docker# ls
nginx-dockerfile
```

```
root@labvm:/home/devasc/Desktop/sujata-docker# cd nginx-dockerfile/
root@labvm:/home/devasc/Desktop/sujata-docker/nginx-dockerfile# ls
Dockerfile  index.html  README.md  style.css
```

Step 2:

```
root@labvm:/home/devasc/Desktop/sujata-docker/nginx-dockerfile# nano Dockerfile
```



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
GNU nano 4.8
FROM ubuntu
LABEL author="Sujata Oak"
RUN apt-get update
RUN apt-get install nginx -y
COPY . /var/www/html/
EXPOSE 80
CMD ["nginx","-g","daemon off;"]
```

#docker build -t sujatadocker2025/websitetest25 .

```
root@labvm:/home/devasc/Desktop/sujata-docker/nginx-dockerfile# docker build -t sujatadocker2025/websitetest25 .
[+] Building 24.9s (10/10) FINISHED                                            docker:default
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 190B                                         0.0
=> [internal] load metadata for docker.io/library/ubuntu:latest               4.2
=> [auth] library/ubuntu:pull token for registry-1.docker.io                  0.6
=> [internal] load .dockerignore                                              0.1
=> => transferring context: 2B                                               0.0
=> [1/4] FROM docker.io/library/ubuntu:latest@sha256:a08e551cb33850e4740772b382 6.1
=> => resolve docker.io/library/ubuntu:latest@sha256:a08e551cb33850e4740772b382 0.1
=> => sha256:a08e551cb33850e4740772b38217fc1796a66da2506d312abe 6.69kB / 6.69kB 0.0
=> => sha256:4f1db91d9560cf107b5832c0761364ec64f46777aa4ec637cca300 424B / 424B 0.0
=> => sha256:65ae7a6f3544bd2d2b6d19b13bfc64752d776bc92c510f8741 2.30kB / 2.30kB 0.0
=> => sha256:32f112e3802cadcab3543160f4d2aa607b3cc1c62140d57b 29.72MB / 29.72MB 3.0
=> => extracting sha256:32f112e3802cadcab3543160f4d2aa607b3cc1c62140d57b4f54413 2.2
=> [internal] load build context                                              0.1
=> => transferring context: 28.65kB                                         0.0
=> [2/4] RUN apt-get update                                                 8.2
=> [3/4] RUN apt-get install nginx -y                                         5.0
```

docker images

```
root@labvm:/home/devasc/Desktop/sujata-docker/nginx-dockerfile# docker images
REPOSITORY          TAG      IMAGE ID      CREATED       SIZE
sujatadocker2025/websitetest25    latest   b4bdc0855a68  50 seconds ago  136MB
nginx              latest   2cd1d97f893f  2 weeks ago   192MB
```

Step 3: Run the container now:

```
# docker run -d -p 3032:80 --name sujata_webcontainer b4b
```

```
root@labvm:/home/devasc/Desktop/sujata-docker/nginx-dockerfile# docker run -d -p 3032:80 --name sujata_webcontainer b4b
86b2ebd73cc97f6d98b55fc6eb0f0e4cf0ae086937903d517fb83137b9f51bcb2
```

```
root@labvm:/home/devasc/Desktop/sujata-docker/nginx-dockerfile# docker ps -a
CONTAINER ID        IMAGE           COMMAND            CREATED          STATUS           PORTS
S NAMES
86b2ebd73cc9        b4b           "nginx -g 'daemon of..."   57 seconds ago   Up 56 seconds   0.0.
0.0:3032->80/tcp, :::3032->80/tcp   sujata_webcontainer
```



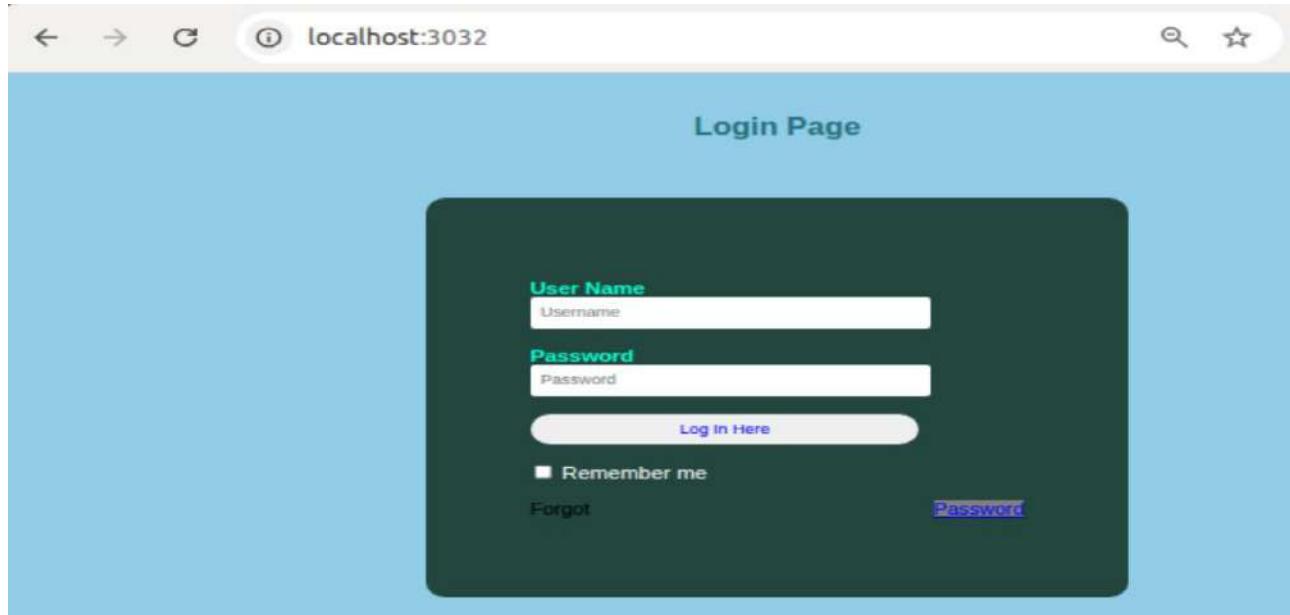
PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



Step 4: Goto Browser: localhost:3032



STEP 5: How to push this image to your dockerhub :

```
root@labvm:/home/devasc/Desktop/sujata-docker/nginx-dockerfile# docker images
REPOSITORY          TAG      IMAGE ID   CREATED        SIZE
sujatadocker2025/websitetest25    latest    b4bdc0855a68  8 minutes ago  136MB
nginx               latest    2cd1d97f893f   2 weeks ago   192MB
# docker push sujatadocker2025/websitetest25
```

```
root@labvm:/home/devasc/Desktop/sujata-docker/nginx-dockerfile# docker push sujatadocker2025/websitetest25
Using default tag: latest
The push refers to repository [docker.io/sujatadocker2025/websitetest25]
3ec9475e5d94: Preparing
5b06bfd1f90d: Preparing
97d8e323fdac: Preparing
107cbdaeecc04: Preparing
denied: requested access to the resource is denied
# docker tag b4b 18061977/apsitsujatacontainer25:v1
```

```
root@labvm:/home/devasc/Desktop/sujata-docker/nginx-dockerfile# docker tag b4b 18061977
/apsitsujatacontainer25:v1
```

```
root@labvm:/home/devasc/Desktop/sujata-docker/nginx-dockerfile# docker images
REPOSITORY          TAG      IMAGE ID   CREATED        SIZE
18061977/apsitsujatacontainer25    v1      b4bdc0855a68  14 minutes ago  136MB
sujatadocker2025/websitetest25    latest    b4bdc0855a68  14 minutes ago  136MB
nginx               latest    2cd1d97f893f   2 weeks ago   192MB
```



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
root@labvm:/home/devasc/Desktop/sujata-docker/nginx-dockerfile# docker login
Authenticating with existing credentials...
WARNING! Your password will be stored unencrypted in /root/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credential-stores
```

Login Succeeded

```
# docker push 18061977/apsitsujatacontainer25:v1
```

```
root@labvm:/home/devasc/Desktop/sujata-docker/nginx-dockerfile# docker push 18061977/apsitsujatacontainer25:v1
The push refers to repository [docker.io/18061977/apsitsujatacontainer25]
3ec9475e5d94: Pushed
5b06bfd1f90d: Pushed
97d8e323fdac: Pushed
107cbdaeecc04: Pushed
v1: digest: sha256:763b31bf15297e1b8f1cf18f04f72a7903346228b5d16dea9d6e1c0aa04c11c5 size: 1161
```

Goto Docker hub page and refresh it:

The screenshot shows the Docker Hub interface. On the left, there's a sidebar for the user '18061977'. The main area is titled 'Repositories' and shows one entry: '18061977/apsitsujatacontainer25'. This entry has a timestamp '2 minutes ago', a visibility status 'Public', and a 'Scout' status.

This screenshot shows the detailed view of the repository '18061977/apsitsujatacontainer25'. It includes sections for 'General' (with a note 'Last pushed 1 minute ago'), 'Add a description', 'Add a category', and a command box containing 'docker push 18061977/apsitsujatacontainer25:tagname'. There are also links for 'Docker commands' and 'Public view'.

Conclusion: In the experiment, we used various docker commands to pull images that were already built, also we created our own images by using docker file instructions for a sample web application and atlast we have pushed the image to docker hub account for others to use the repository.



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



Academic Year: 2025-26

Semester: V Class /

Branch: TE IT

Subject: DevOPs Lab (DL)

Subject Lab In-charge: Prof. Sujata Oak

EXPERIMENT NO.10

Aim: Installation of Ansible on top of AWS instance. Configure SSH access to Ansible master/slave and setup ansible host and test the connection.

Theory: Ansible is an open source IT Configuration Management, Deployment & Orchestration tool. It aims to provide large productivity gains to a wide variety of automation challenges.

Ansible Terms:

- **Controller Machine:** The machine where Ansible is installed, responsible for running the provisioning on the servers you are managing.
- **Inventory:** An initialization file that contains information about the servers you are managing.
- **Playbook:** The entry point for Ansible provisioning, where the automation is defined through tasks using YAML format.
- **Task:** A block that defines a single procedure to be executed, e.g. Install a package.
- **Module:** A module typically abstracts a system task, like dealing with packages or creating and changing files. Ansible has a multitude of built-in modules, but you can also create custom ones.
- **Role:** A pre-defined way for organizing playbooks and other files in order to facilitate sharing and reusing portions of a provisioning.
- **Play:** A provisioning executed from start to finish is called a play. In simple words, execution of a playbook is called a play.
- **Facts:** Global variables containing information about the system, like network interfaces or operating system.
- **Handlers:** Used to trigger service status changes, like restarting or stopping a service.

STEP1: Connect AWS Instances

**STEP 2: Connect to Instances: ansible-master**

```
devasc@labvm:~$ cd Desktop/
devasc@labvm:~/Desktop$ cd ansible_lab/
devasc@labvm:~/Desktop/ansible_lab$ chmod 400 "ansible-key.pem"
devasc@labvm:~/Desktop/ansible_lab$ ssh -i "ansible-key.pem" ubuntu@ec2-54-196-134-80.compute-1.amazonaws.com
```

```
devasc@labvm:~/Desktop/ansible_lab$ ssh -i "ansible-key.pem" ubuntu@ec2-54-196-134-80.compute-1.amazonaws.com
The authenticity of host 'ec2-54-196-134-80.compute-1.amazonaws.com (54.196.134.80)' can't be established.
ECDSA key fingerprint is SHA256:NDn3OD2bYHeoA/SigJ3484lN+ZjFZsdpNDFmGpQIB74.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-54-196-134-80.compute-1.amazonaws.com,54.196.134.80' (EDSA) to the list of known hosts.
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-1012-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

System information as of Sat Sep 14 17:18:37 UTC 2024

 System load:  0.0          Processes:           105
 Usage of /:   22.8% of 6.71GB   Users logged in:      0
 Memory usage: 20%           IPv4 address for enX0: 172.31.18.177
 Swap usage:   0%          

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status
```

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

```
ubuntu@ip-172-31-18-177:~$
```

Connect to Instances: ansible-slave

```
devasc@labvm:~$ cd Desktop/
devasc@labvm:~/Desktop$ cd ansible_lab/
devasc@labvm:~/Desktop/ansible_lab$ chmod 400 "ansible-key.pem"
devasc@labvm:~/Desktop/ansible_lab$ ssh -i "ansible-key.pem" ubuntu@ec2-3-84-176-161.compute-1.amazonaws.com
```

**Department of Information Technology**

(NBA Accredited)

```
devasc@labvm:~/Desktop/ansible_lab$ ssh -i "ansible-key.pem" ubuntu@ec2-3-84-176-161.compute-1.amazonaws.com
The authenticity of host 'ec2-3-84-176-161.compute-1.amazonaws.com (3.84.176.161)' can't be established.
ECDSA key fingerprint is SHA256:YopQszxHVveP3+lp8lzwC+BAp7TuiPK46dceLc5ncW4.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-3-84-176-161.compute-1.amazonaws.com,3.84.176.161' (ECDSA) to the list of known hosts.
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-1012-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

System information as of Sat Sep 14 17:22:55 UTC 2024

System load: 0.0          Processes:           104
Usage of /: 22.8% of 6.71GB   Users logged in:      0
Memory usage: 19%          IPv4 address for enX0: 172.31.16.10
Swap usage:  0%         

Expanded Security Maintenance for Applications is not enabled.
```

Enable ESM Apps to receive additional future security updates.
See <https://ubuntu.com/esm> or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

```
ubuntu@ip-172-31-16-10:~$ █
```

STEP3: To Ping master and slave machine:

ansible-master

```
ubuntu@ip-172-31-18-177:~$ sudo su
root@ip-172-31-18-177:/home/ubuntu# █
```

To ping master to slave

```
ubuntu@ip-172-31-18-177:~$ sudo su
root@ip-172-31-18-177:/home/ubuntu# ping 172.31.16.10
PING 172.31.16.10 (172.31.16.10) 56(84) bytes of data. █
```



To ping slave to master

```
ubuntu@ip-172-31-16-10:~$ sudo su
root@ip-172-31-16-10:/home/ubuntu# ping 172.31.18.177
PING 172.31.18.177 (172.31.18.177) 56(84) bytes of data.
```

STEP 4: Ansible-master : Ansible Installation

```
#apt update -y
```

```
root@ip-172-31-18-177:/home/ubuntu# apt update -y
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:4 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Packages [15.0 MB]
```

Add Ansible Repository

```
# apt-add-repository ppa:ansible/ansible
```

```
root@ip-172-31-18-177:/home/ubuntu# apt-add-repository ppa:ansible/ansible
Repository: 'Types: deb
URIs: https://ppa.launchpadcontent.net/ansible/ansible/ubuntu/
Suites: noble
Components: main
'
Description:
Ansible is a radically simple IT automation platform that makes your applications and systems easier to deploy. Avoid writing scripts or custom code to deploy and update your applications— automate in a language that approaches plain English, using SSH, with no agents to install on remote systems.

http://ansible.com/

If you face any issues while installing Ansible PPA, file an issue here:
https://github.com/ansible-community/ppa/issues
More info: https://launchpad.net/~ansible/+archive/ubuntu/ansible
Adding repository.
Press [ENTER] to continue or Ctrl-c to cancel.
```

```
Press [ENTER] to continue or Ctrl-c to cancel.
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu noble-security InRelease
Get:5 https://ppa.launchpadcontent.net/ansible/ansible/ubuntu noble InRelease [17.8 kB]
Get:6 https://ppa.launchpadcontent.net/ansible/ansible/ubuntu noble/main amd64 Packages [776 B]
Get:7 https://ppa.launchpadcontent.net/ansible/ansible/ubuntu noble/main Translation-en [472 B]
Fetched 19.1 kB in 3s (5994 B/s)
Reading package lists... Done
```

```
#apt update -y
```



PARSHVANATH CHARITABLE TRUST'S

A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
root@ip-172-31-18-177:/home/ubuntu# apt update -y
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:4 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:5 http://ppa.launchpadcontent.net/ansible/ansible/ubuntu noble InRelease [15.0 MB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe Translation-en [5982 kB]
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
133 packages can be upgraded. Run 'apt list --upgradable' to see them.
```

apt-get install ansible -y

```
root@ip-172-31-18-177:/home/ubuntu# apt-get install ansible -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  ansible-core python3-jmespath python3-kerberos python3-nacl python3-ntlm-auth
  python3-packaging python3-paramiko python3-requests-ntlm python3-resolvelib
  python3-winrm python3-xmllodict sshpass
Suggested packages:
  python-nacl-doc python3-gssapi python3-invoke
The following NEW packages will be installed:
  ansible ansible-core python3-jmespath python3-kerberos python3-nacl
  python3-ntlm-auth python3-packaging python3-paramiko python3-requests-ntlm
  python3-resolvelib python3-winrm python3-xmllodict sshpass
0 upgraded, 13 newly installed, 0 to remove and 133 not upgraded.
```

ansible --version

```
root@ip-172-31-18-177:/home/ubuntu# ansible --version
ansible [core 2.16.11]
  config file = /etc/ansible/ansible.cfg
  configured module search path = ['/root/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']
    ansible python module location = /usr/lib/python3/dist-packages/ansible
    ansible collection location = /root/.ansible/collections:/usr/share/ansible/collections
  executable location = /usr/bin/ansible
  python version = 3.12.3 (main, Apr 10 2024, 05:33:47) [GCC 13.2.0] (/usr/bin/python3)
  jinja version = 3.1.2
  libyaml = True
```

Ansible-slave :

#apt update -y

```
root@ip-172-31-16-10:/home/ubuntu# apt update -y
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:4 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Packages [15.0 MB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe Translation-en [5982 kB]
```



STEP5:

Ansible-master:

```
# nano /etc/ansible/hosts
```

```
root@ip-172-31-18-177:/home/ubuntu# nano /etc/ansible/hosts

GNU nano 7.2                               /etc/ansible/hosts *

# Ex 1: Ungrouped hosts, specify before any group headers:

## green.example.com
## blue.example.com
## 192.168.100.1
## 192.168.100.10

# Ex 2: A collection of hosts belonging to the 'webservers' group:

## [webservers]
## alpha.example.org
## beta.example.org
## 192.168.1.100
## 192.168.1.110
[client_1]
172.31.16.10
```

Add ip address of ansible-slave machine

Save the file

STEP6: To create SSH Key:

Ansible-master:

```
root@ip-172-31-18-177:/home/ubuntu# ssh-keygen -t rsa
```

```
root@ip-172-31-18-177:/home/ubuntu# ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa
Your public key has been saved in /root/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:+2Ydets/gm0rpyBWOT4wd7wBLenG5xzA/lINvXmWqUQ root@ip-172-31-18-177
The key's randomart image is:
+---[RSA 3072]----+
|          . o .
|          * o E
|          + * + o o
| oSX B = =
| B.O * +
| o.. Boo
| ..Boo* .
| o.o*o+.o
+---[SHA256]----+
```

```
root@ip-172-31-18-177:/home/ubuntu# cd /root/.ssh/
```



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



root@ip-172-31-18-177:~/ssh#

```
root@ip-172-31-18-177:/home/ubuntu# cd /root/.ssh/  
root@ip-172-31-18-177:~/ssh# ls  
authorized_keys  id_rsa  id_rsa.pub
```

root@ip-172-31-18-177:~/ssh# cat id_rsa.pub

```
root@ip-172-31-18-177:~/ssh# cat id_rsa.pub  
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQgQDeOKZYM2cg4a4f60LTTYTNAUiHuHCaj2n5YpU3gSbs94AkBw  
XBj9xC6lSdn0uZ0xJQ601TxrGc6ZanyHFFSPgzCg88kMXUaaq/CD75rcvCmKID3EYxtCAw6v0jmJMDWIRC5Hcw  
TPRT0Fouja5Htr02MA89jaJ6wMGicX1emEB0oF2j7J+2uPu6jki5goSt+Yjb2+zdAt03cV/LtmAOYEZ9d9eJZfd  
AiW+6ZICP4wLXkOHVbUTllN3TYj9Q550ltW0IenFvacKBHYHI8YsNgVZ+Kw1SUU3ycEDTODQuPmnBm5+ZtsJVUT  
PTs4gmCSbqQZtbaQAEFqmZl5A9MoOkSNDYBFJXbJVO4wwofGnM68G2gGSbv6UztUNolyzAZV1wcFYI90v7p2yVD  
1KDnOfEcTH0IJtu61UJULXZpqalEqp1IiinpySe7/zntdzskT3DrkSJ8AQvIcYcX2TV9SJkvCudH6o075ToTbhCx  
oB/KCAY+TKojp4Qqb5gt9ha5aIxJk= root@ip-172-31-18-177
```

Copy this key into ansible-slave machine

Ansible-slave:

root@ip-172-31-16-10:/home/ubuntu# cd /root/.ssh/

root@ip-172-31-16-10:~/ssh# ls

```
root@ip-172-31-16-10:/home/ubuntu# cd /root/.ssh/  
root@ip-172-31-16-10:~/ssh# ls  
authorized_keys
```

[root@ip-172-31-16-10:~/ssh# nano authorized_keys](#)

```
root@ip-172-31-16-10:~/ssh# nano authorized_keys  
File Edit View Search Terminal Help  
GNU nano 7.2  authorized_keys *  
no-port-forwarding,no-agent-forwarding,no-X11-forwarding,command="echo 'Please >  
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQgQDeOKZYM2cg4a4f60LTTYTNAUiHuHCaj2n5YpU3gS>
```

Save it.

root@ip-172-31-16-10:~/ssh# nano /etc/ssh/sshd_config

```
root@ip-172-31-16-10:~/ssh# nano /etc/ssh/sshd_config
```



```
root@ip-172-31-16-10: ~/.ssh
File Edit View Search Terminal Help
GNU nano 7.2                               /etc/ssh/sshd_config *
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
#PermitRootLogin prohibit-password
PermitRootLogin yes
#StrictModes yes
#MaxAuthTries 6
```

Save it.

STEP7 :

Ansible-master:

1] root@ip-172-31-18-177:~/.ssh# ansible -m ping all

```
root@ip-172-31-18-177:~/.ssh# ansible -m ping all
The authenticity of host '172.31.16.10 (172.31.16.10)' can't be established.
ED25519 key fingerprint is SHA256:2qlJKjwxmY/FOPpFgKW6lKr4R+R+YwewVnZkfqRitzQ8.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
172.31.16.10 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
```

2] root@ip-172-31-18-177:~/.ssh# ansible client_1 -m setup



```
"ansible_user_shell": "/bin/bash",
"ansible_user_uid": 0,
"ansible_userspace_architecture": "x86_64",
"ansible_userspace_bits": "64",
"ansible_virtualization_role": "guest",
"ansible_virtualization_tech_guest": [
    "xen"
],
"ansible_virtualization_tech_host": [],
"ansible_virtualization_type": "xen",
"discovered_interpreter_python": "/usr/bin/python3",
"gather_subset": [
    "all"
],
"module_setup": true
},
"changed": false
```

3]

Ansible-slave:

```
root@ip-172-31-16-10:~/ssh# git --version
```

```
root@ip-172-31-16-10:~/ssh# git --version
git version 2.43.0
```

So let me remove it

```
root@ip-172-31-16-10:~/ssh# apt remove git
```

```
root@ip-172-31-16-10:~/ssh# apt remove git
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  git-man liberror-perl
Use 'sudo apt autoremove' to remove them.
The following packages will be REMOVED:
  git ubuntu-server
0 upgraded, 0 newly installed, 2 to remove and 132 not upgraded.
After this operation, 22.2 MB disk space will be freed.
Do you want to continue? [Y/n] y
(Reading database ... 67739 files and directories currently installed.)
Removing ubuntu-server (1.539) ...
Removing git (1:2.43.0-1ubuntu7.1) ...
```

```
root@ip-172-31-16-10:~/ssh# git --version
```

```
root@ip-172-31-16-10:~/ssh# git --version
bash: /usr/bin/git: No such file or directory
```

So now I want to install git on all slave machine



Ansible-master:

```
root@ip-172-31-18-177:~/ssh# ansible client -m apt -a "name=git state=present" --become
```

Ansible-slave:

```
root@ip-172-31-16-10:~/ssh# git --version
```

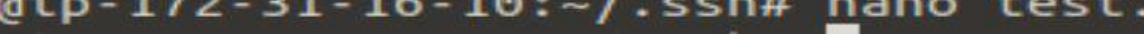
```
root@ip-172-31-16-10:~/.ssh# git --version  
git version 2.43.0
```

How to uninstall package from a ansible-master machine?

In ansible-slave machine :

```
root@ip-172-31-16-10:~/ssh# nano test.txt
```

```
root@ip-172-31-16-10:~/ssh# nano test.txt
```



In ansible-master machine:

```
root@ip-172-31-18-177:~/ssh# ansible client_1 -m apt -a "name=nano state=absent" --become
```

**Department of Information Technology**

(NBA Accredited)

```
root@ip-172-31-18-177:~/ssh# ansible client_1 -m apt -a "name=nano state=absent" --become
172.31.16.10 | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": true,
    "stderr": "",
    "stderr_lines": [],
    "stdout": "Reading package lists...\nBuilding dependency tree...\nReading state information...\n\nThe following packages will be REMOVED:\nnano\n0 upgraded, 0 newly installed, 1 to remove and 132 not upgraded.\nAfter this operation, 856 kB disk space will be freed"
}
```

In ansible-slave machine:

root@ip-172-31-16-10:~/ssh# nano test.txt

```
root@ip-172-31-16-10:~/ssh# nano test.txt
bash: /usr/bin/nano: No such file or directory
```

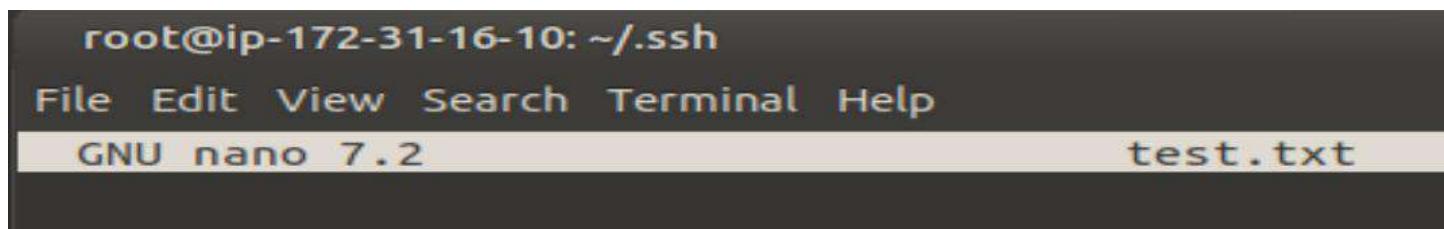
In ansible-master machine:

root@ip-172-31-18-177:~/ssh# ansible client_1 -m apt -a "name=nano state=present" --become

```
root@ip-172-31-18-177:~/ssh# ansible client_1 -m apt -a "name=nano state=present" --become
172.31.16.10 | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "cache_update_time": 1726335400,
    "cache_updated": false,
    "changed": true,
    "stderr": "",
    "stderr_lines": [],
    "stdout": "Reading package lists...\nBuilding dependency tree...\nReading state information...\nSuggested packages:\nhunspell\nThe following NEW packages will be installed:\nnano\n0 upgraded, 1 newly installed, 0 to remove and 132 not upgraded.\nNeed to get 281 kB of archives.\nAfter this operation, 856 kB of additional disk space will be freed"
}
```

In ansible-slave machine:

root@ip-172-31-16-10:~/ssh# nano test.txt



Conclusion: In the experiment, we successfully installed Ansible on top of AWS Instance. Also, configured SSH access to Ansible slave and setup ansible host and tested connection.

**Academic Year: 2025-26****Semester: V Class /****Branch: TE IT****Subject: DevOPs Lab (DL)****Subject Lab In-charge: Prof. Sujata Oak**

EXPERIMENT NO.11

Aim: To deploy a web application by provisioning LAMP Stack using ansible playbook.

Theory: A LAMP stack is a bundle of four different software technologies that developers use to build websites and web applications. LAMP is an acronym for the operating system, Linux; the web server, Apache; the database server, MySQL; and the programming language, PHP. All four of these technologies are open source, which means they are community maintained and freely available for anyone to use. Developers use LAMP stacks to create, host, and maintain web content. It is a popular solution that powers many of the websites you commonly use today.

ANSIBLE PLAYBOOK:

Ansible playbooks are a vital part of Ansible and the core component of every Ansible configuration. An Ansible playbook is a file that contains a set of instructions that Ansible can use to automate tasks on remote hosts. Playbooks are written in YAML, a human-readable markup language.

A playbook typically consists of one or more plays, a collection of tasks run in sequence. Each task is a single instruction that Ansible can execute, such as installing a package, configuring a service, or copying a file.

By using Ansible playbooks, IT operations teams can automate infrastructure provisioning, configuration management, application deployment, and other operational tasks. Playbooks provide a concise and human-readable way to describe the desired automation workflows, making managing and scaling infrastructure configurations easier.

STEP1: Clone ansible code from my github repository

Ansible-master:

```
root@ip-172-31-18-177:~/ssh# cd ~
root@ip-172-31-18-177:~# ls
snap
root@ip-172-31-18-177:~# mkdir ansible-lab
root@ip-172-31-18-177:~# cd ansible-lab/
root@ip-172-31-18-177:~/ansible-lab# git clone https://github.com/sujataoak799/ansible-codes.git
Cloning into 'ansible-codes'...
```



```
remote: Enumerating objects: 23, done.
remote: Counting objects: 100% (6/6), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 23 (delta 1), reused 4 (delta 1), pack-reused 17 (from 1)
Receiving objects: 100% (23/23), 8.63 KiB | 1.73 MiB/s, done.
Resolving deltas: 100% (6/6), done.
root@ip-172-31-18-177:~/ansible-lab# ls
ansible-codes
root@ip-172-31-18-177:~/ansible-lab# cd ansible-codes/
root@ip-172-31-18-177:~/ansible-lab/ansible-codes# ls
config.php      lampstack_1.yml mysqlmodule.yml reset-password.php
deploywebsite.yml login.php      readme.txt    users.sql
index.html      logout.php     register.php   welcome.php

root@ip-172-31-18-177:~/.ssh# cd ~
root@ip-172-31-18-177:~# ls
snap
root@ip-172-31-18-177:~# mkdir ansible-lab
root@ip-172-31-18-177:~# cd ansible-lab/
root@ip-172-31-18-177:~/ansible-lab# git clone https://github.com/sujataoak799/ansible-codes.git
Cloning into 'ansible-codes'...
remote: Enumerating objects: 23, done.
remote: Counting objects: 100% (6/6), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 23 (delta 1), reused 4 (delta 1), pack-reused 17 (from 1)
Receiving objects: 100% (23/23), 8.63 KiB | 1.73 MiB/s, done.
Resolving deltas: 100% (6/6), done.
root@ip-172-31-18-177:~/ansible-lab# ls
ansible-codes
root@ip-172-31-18-177:~/ansible-lab# cd ansible-codes/
root@ip-172-31-18-177:~/ansible-lab/ansible-codes# ls
config.php      lampstack_1.yml mysqlmodule.yml reset-password.php
deploywebsite.yml login.php      readme.txt    users.sql
index.html      logout.php     register.php   welcome.php
```

STEP2:

Now all my files are in ansible-master machine and I need to deploy it on my ansible-slave machine. So we will be configuring our ansible-slave machine to host our full stack application.

The first playbook which I am going to setup on ansible-slave machine is lampstack_1.yml

```
root@ip-172-31-18-177:~/ansible-lab/ansible-codes# nano lampstack_1.yml
```

```
root@ip-172-31-18-177:~/ansible-lab/ansible-codes# nano lampstack_1.yml
```



```
File Edit View Search Terminal Help
GNU nano 7.2                                         lampstack 1.yml
---#
# Setup LAMP Stack
- hosts: client_1
  tasks:
    - name: install lamp stack
      become: yes
      become_user: root
      apt:
        pkg:
          - apache2
          - mysql-server
          - php
          - libapache2-mod-php
          - php-mysql
        state: present
        update_cache: yes

    - name: start apache service
      become: yes
      become_user: root
      service:
        name: apache2
        state: started
        enabled: yes

    - name: start mysql service
      become: yes
      become_user: root
      service:
        name: mysql
        state: started
        enabled: yes

    - name: create target directory
      file: path=/var/www/html state=directory mode=0755

    - name: deploy index.html
      become: yes
      become_user: root
      copy:
        src: index.html
        dest: /var/www/html/index.html
```

Save it.

STEP3: How to Run/Execute a playbook.



```
root@ip-172-31-18-177:~/ansible-lab/ansible-codes# ansible-playbook lampstack_1.yml
```

```
root@ip-172-31-18-177:~/ansible-lab/ansible-codes# ansible-playbook lampstack_1.yml
PLAY [client_1] ****
TASK [Gathering Facts] ****
ok: [172.31.16.10]
TASK [install lamp stack] ****
changed: [172.31.16.10]
TASK [start apache service] ****
ok: [172.31.16.10]
TASK [start mysql service] ****
ok: [172.31.16.10]
TASK [create target directory] ****
ok: [172.31.16.10]
TASK [deploy index.html] ****
changed: [172.31.16.10]
PLAY RECAP ****
172.31.16.10 : ok=6    changed=2    unreachable=0    failed=0    skipped=0
                rescued=0   ignored=0
```

Ansible-slave:

```
root@ip-172-31-16-10:~# mysql
```

```
root@ip-172-31-16-10:~# mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.0.39-0ubuntu0.24.04.2 (Ubuntu)

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> █
```

```
root@ip-172-31-16-10:~# php --version
```

```
root@ip-172-31-16-10:~# php --version
PHP 8.3.6 (cli) (built: Jun 13 2024 15:23:20) (NTS)
Copyright (c) The PHP Group
Zend Engine v4.3.6, Copyright (c) Zend Technologies
    with Zend OPcache v8.3.6, Copyright (c), by Zend Technologies
```

```
root@ip-172-31-16-10:~# service apache2 status
```



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
root@ip-172-31-16-10:~# service apache2 status
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: >
   Active: active (running) since Sat 2024-09-14 18:44:55 UTC; 4min 45s ago
     Docs: https://httpd.apache.org/docs/2.4/
   Process: 11759 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/S>
 Main PID: 11762 (apache2)
   Tasks: 6 (limit: 1130)
  Memory: 10.9M (peak: 11.1M)
    CPU: 63ms
   CGroup: /system.slice/apache2.service
           └─11762 /usr/sbin/apache2 -k start
             ├─11765 /usr/sbin/apache2 -k start
             ├─11766 /usr/sbin/apache2 -k start
             ├─11767 /usr/sbin/apache2 -k start
             ├─11768 /usr/sbin/apache2 -k start
             └─11769 /usr/sbin/apache2 -k start

Sep 14 18:44:55 ip-172-31-16-10 systemd[1]: Starting apache2.service - The Apache>
```

Once apache service status is active. Copy IPv4 address of ansible-slave machine in browser and you can see the deployment of index.html page.



The screenshot shows the AWS CloudWatch Instances interface. It displays the details of an instance named 'i-Odca9416241c968f4 (ansible_slave)'. The public IPv4 address '3.84.176.161' is highlighted with a green border, indicating it has been copied. Below the interface, a browser window is open to the URL '3.84.176.161'. The page content reads: 'Welcome to Ansible Playbook Session by Prof. Sujata Oak' and 'Deploy a web application by provisioning LampStack'.

Conclusion: In the experiment, successfully implemented provisioning lamp stack on ubuntu machine using ansible playbook.

<https://aws.amazon.com/what-is/lamp-stack/>

[https://www.simplilearn.com/what-is-ansible-playbook-article#how to write an ansible playbook](https://www.simplilearn.com/what-is-ansible-playbook-article#how_to_write_an_ansible_playbook)



Academic Year: 2025-26

Semester: V Class /

Branch: TE IT

Subject: DevOPs Lab (DL)

Subject Lab In-charge: Prof. Sujata Oak

EXPERIMENT NO.12

Aim: Deploy a website code on the node by provisioning mysql server and database using ansible playbook.

Theory: MySQL Database is a client/server system that consists of a multithreaded SQL server that supports different back ends, several different client programs and libraries, administrative tools, and a wide range of application-programming interfaces (APIs).

Primary Terminologies

- **MySQL:** For the storage and management of structured data, a lot of people use MySQL, an open-source relational database management system (RDBMS). It offers components, for instance, SQL support, data security, versatility, and execution.
- **Ansible** is a configuration management tool. it is a suit of software tools that enables infrastructure as code.it is an open source and suit includes software provisioning, configuration management and application deployment functionality. There is no need to install run time, as it is a stand-alone tool.
- **Ansible Playbooks-** playbooks are the basis for really a simple configuration management and multi-machine deployment system. [Ansible playbooks](#) are YAML documents containing a set of instructions for Ansible to execute on remote hosts. Playbooks automate tasks like software installation, service configuration, and file management by defining the desired state of systems.
- **Modules for Ansible-** [Ansible](#) modules are little projects that perform tasks on remote hosts. For common tasks like package management, file manipulation, and service control, Ansible has a lot of built-in modules. The Ansible engine runs modules on the target hosts and sends back the results to the control node.

STEP1:

Ansible-master:

root@ip-172-31-18-177:~/ansible-lab/ansible-codes# nano mysqlmodule.yml

```
root@ip-172-31-18-177:~/ansible-lab/ansible-codes# nano mysqlmodule.yml
```



```
GNU nano 7.2                               mysqlmodule.yml
- hosts: client_1
  remote_user: root
  tasks:
    - name: 2. Start Mysql Service
      service: name=mysql state=started enabled=true

    - name: Install python package #required for mysql_db tasks
      apt: name=python3-pip state=present

    - name: Install Mysql-python package #required for mysql_db tasks
      apt: name=python3-mysqldb state=present

    - name: 3. Create a new database
      mysql_db: name=demo state=present collation=utf8_general_ci

    - name: 4. Create a database user
      mysql_user: name=sujata password=123456 priv=*.*:ALL host=localhost state=present

    - name: 5a. Copy sample data
      copy: src=users.sql dest=/tmp/dump.sql

    - name: 5b. Insert sample data
      shell: cat /tmp/dump.sql | mysql -u sujata -p123456 demo

^G Help   ^Q Write Out   ^W Where Is   ^K Cut
^X Exit   ^R Read File   ^A Replace   ^U Paste   ^T Execute   ^J Justify
^C Location   ^I Go To Line
```

root@ip-172-31-18-177:~/ansible-lab/ansible-codes# cat users.sql

```
root@ip-172-31-18-177:~/ansible-lab/ansible-codes# cat users.sql
-- phpMyAdmin SQL Dump
-- version 4.1.14
-- http://www.phpmyadmin.net
--
-- Host: 127.0.0.1
-- Generation Time: Apr 28, 2017 at 02:20 PM
-- Server version: 5.6.17
-- PHP Version: 5.5.12

SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
SET time_zone = "+00:00";


/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8 */;

--
-- Database: `demo`
--

-- Table structure for table `users`
```

STEP2:

root@ip-172-31-18-177:~/ansible-lab/ansible-codes# ansible-playbook mysqlmodule.yml



```
root@ip-172-31-18-177:~/ansible-lab/ansible-codes# ansible-playbook mysqlmodule.yml

PLAY [client_1] ****
TASK [Gathering Facts] ****
ok: [172.31.16.10]

TASK [2. Start Mysql Service] ****
ok: [172.31.16.10]

TASK [Install python package] ****
changed: [172.31.16.10]

TASK [Install Mysql-python package] ****
changed: [172.31.16.10]

TASK [3. Create a new database] ****
changed: [172.31.16.10]

TASK [4. Create a database user] ****
[WARNING]: Option column_case_sensitive is not provided. The default is now false, so
the column's name will be uppercased. The default will be changed to true in
community.mysql 4.0.0.
changed: [172.31.16.10]

TASK [5a. Copy sample data] ****
changed: [172.31.16.10]

TASK [5b. Insert sample data] ****
changed: [172.31.16.10]

PLAY RECAP ****
172.31.16.10 : ok=8    changed=6      unreachable=0    failed=0    skipped=0
                  rescued=0   ignored=0
```

STEP3: ansible-slave

```
root@ip-172-31-16-10:~# mysql -u sujata -p123456
```

```
root@ip-172-31-16-10:~# mysql -u sujata -p123456
mysql: [Warning] Using a password on the command line interface can be insecure.
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 11
Server version: 8.0.39-Ubuntu0.24.04.2 (Ubuntu)

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
mysql> ■
```

```
mysql> show databases;
```



```
mysql> show databases;
+-----+
| Database |
+-----+
| demo      |
| information_schema |
| mysql     |
| performance_schema |
| sys       |
+-----+
5 rows in set (0.00 sec)
```

```
mysql> use mysql;
```

```
mysql> use mysql;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A
Database changed
```

```
mysql> show tables;
```

```
mysql> show tables;
+-----+
| Tables_in_mysql |
+-----+
| columns_priv   |
| component      |
| db             |
| default_roles  |
| engine_cost    |
| func           |
| general_log    |
| global_grants  |
| gtid_executed  |
| help_category  |
| help_keyword   |
| help_relation  |
| help_topic     |
| innodb_index_stats |
| innodb_table_stats |
| password_history |
| plugin          |
| procs_priv     |
+-----+
```



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



mysql> select * from db;

```
mysql> select * from db;
+-----+-----+-----+-----+-----+-----+-----+-----+
| Host      | Db       | User      | Select_priv | Insert_priv | U
| Update_priv | Delete_priv | Create_priv | Drop_priv | Grant_priv | References_priv
| Index_priv | Alter_priv | Create_tmp_table_priv | Lock_tables_priv | Create_
| View_priv | Show_view_priv | Create_routine_priv | Alter_routine_priv | Execute_
| Priv | Event_priv | Trigger_priv |
+-----+-----+-----+-----+-----+-----+-----+-----+
```

mysql> exit

```
mysql> exit
Bye
```

STEP4: ansible-master

root@ip-172-31-18-177:~/ansible-lab/ansible-codes# nano deploywebsite.yml

```
GNU nano 7.2                               deploywebsite.yml
-
- name: copy
hosts: client_1
become: true
become_user: root
gather_facts: true
tasks:
- name: copy file
copy: src=login.php dest=/var/www/html/login.php
- name: copy file
copy: src=reset-password.php dest=/var/www/html/reset-password.php
- name: copy file
copy: src=logout.php dest=/var/www/html/logout.php
- name: copy file
copy: src=register.php dest=/var/www/html/register.php
- name: copy file
copy: src=config.php dest=/var/www/html/config.php
- name: copy file
copy: src=welcome.php dest=/var/www/html/welcome.php

[ Read 19 lines ]
^G Help      ^O Write Out   ^W Where Is    ^K Cut        ^T Execute   ^C Location
^X Exit      ^R Read File   ^\ Replace     ^U Paste      ^J Justify   ^/ Go To Line
```

root@ip-172-31-18-177:~/ansible-lab/ansible-codes# ansible-playbook deploywebsite.yml



PARSHVANATH CHARITABLE TRUST'S
A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



```
root@ip-172-31-18-177:~/ansible-lab/ansible-codes# ansible-playbook deploywebsite.yml
PLAY [copy] ****
TASK [Gathering Facts] ****
ok: [172.31.16.10]

TASK [copy file] ****
changed: [172.31.16.10]

PLAY RECAP ****
172.31.16.10 : ok=7    changed=6    unreachable=0    failed=0    skipped=0
                  rescued=0   ignored=0
```

Ansible-slave:

```
root@ip-172-31-16-10:~# cd /var/www/html
root@ip-172-31-16-10:/var/www/html# ls
config.php  login.php  register.php  welcome.php
index.html  logout.php  reset-passowrd.php
```

```
root@ip-172-31-16-10:~# cd /var/www/html
root@ip-172-31-16-10:/var/www/html# ls
config.php  login.php  register.php  welcome.php
index.html  logout.php  reset-passowrd.php
```

STEP5: Goto Browser: ansible-slave machine IP address/login.php

Login

Please fill in your credentials to login.

Username: sujata

Password: *****

Login

IP address/welcome.php



Welcome

Not secure | 34.201.114.138/welcome.php

Reset Your Password Sign Out of Your Account

Hi, **sujata**. Welcome to our site.

IP address/reset-password.php

Reset Password

Please fill out this form to reset your password.

New Password

Confirm Password

Submit Cancel

Conclusion: In the experiment, we successfully deploy a website code on the node by provisioning mysql server and database using ansible playbook.