

ASSESSMENT

1. SHELL SCRIPTING (BASICS)

A1) Process log + high memory check

Scenario: Ops wants a snapshot of all running processes every time the script runs and a quick flag for any “memory hogs”.

Task:

- 1) Lists all processes and saves them to a dated log file under ~/logs/.
- 2) Identifies any process using > 10% memory and appends those lines under a heading “HIGH MEMORY” in the same log.
- 3) Creates the logs directory if it doesn't exist.

Deliverables: Script file, one sample log.

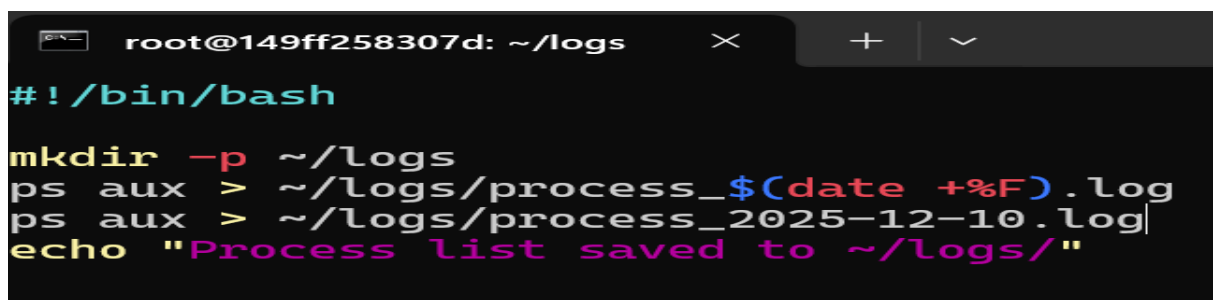
Success criteria: Log exists with timestamped filename; “HIGH MEMORY” section appears when applicable.

Answer:

Now we are try to solve the above problem in Ubuntu machine.

For that we have to follow some steps, They are:

- First we have to open the Docker machine and run that machine in the command prompt.
- For running the docker machine use the command “**docker run -it --name (choose any name) ubuntu /bin/bash**”.
- Now we have to update the machine to run efficiently. For this use the command “**apt update**”. It will update the all the files inside the machine.
- For this shell scripting we required editor to write and edit the scripts according to our requirements. For this installation use the command “**apt install vim -y**”. it will install all the necessary packages and files required to the editor.
- Now we have to select the time zone and place in the editor. Now we are ready to write the scripts.
- For opening the editor use the command “**vi (scriptname).sh**”. the extension of this scriptfile is .sh.
- Now we press “**I**” to enter the text in the editor and then we can write the script.
- For Listing all processes and saves them to a dated log file under ~/logs/, follo the script attached below.



```
root@149ff258307d: ~/logs
#!/bin/bash
mkdir -p ~/logs
ps aux > ~/logs/process_$(date +%F).log
ps aux > ~/logs/process_2025-12-10.log
echo "Process list saved to ~/logs/"
```

- The above will store all the process in the log dated file. We can manually enter the custom date or else it can automatically take the today's date by the command `"$(date +%F)"`.
- `#!/bin/bash` is called a **shebang**. It runs this script using the Bash shell interpreter, located at `/bin/bash`.
- `ps aux` → Lists all running processes.
- `~/logs/` → Folder to store logs (replace with your path).
- `process_` → It will run all the process
- `$(date +%F).log` → Creates a filename with the current date or we can manually enter the particular date and time by using the command `"ps aux > ~/logs/process_$(date +%Y-%m-%d_%H-%M-%S).log"`.
- `>` → Redirects output into the file.
- Now we have to see, whether the process are stored in log file or not. Now use the below commands to see that.

```
root@149ff258307d:/# vi script1.sh
root@149ff258307d:/# chmod 700 script1.sh
root@149ff258307d:/# ./script1.sh
Process list saved to ~/logs/
```

```
root@149ff258307d:/# cd ~/logs/
root@149ff258307d:~/logs# ls
process_2025-08-11.log  process_2025-12-10.log
root@149ff258307d:~/logs# cat process_2025-08-11.log
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1   0.0  0.0   4588  3584 pts/0    Ss   09:14   0:00 /bin/bash
root       344   0.0  0.0   4324  3200 pts/0    S+   09:35   0:00 /bin/bash ./script1.sh
root       346   0.0  0.1   7888  3968 pts/0    R+   09:35   0:00 ps aux
root@149ff258307d:~/logs# cat process_2025-12-10.log
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1   0.0  0.0   4588  3584 pts/0    Ss   09:14   0:00 /bin/bash
root       344   0.0  0.0   4324  3200 pts/0    S+   09:35   0:00 /bin/bash ./script1.sh
root       348   0.0  0.1   7888  3968 pts/0    R+   09:35   0:00 ps aux
```

- See above, all processes are saved in a log file.

Now we identify any process using `> 10%` memory and append those lines under a heading `"HIGH MEMORY"` in the same log.

- For this use the below script.

```
root@9c507856cb7a: /
#!/bin/bash
ps aux > process.txt
awk '$4 > 10' process.txt > HIGH_MEMORY.txt
echo "The memory which is greater than 10 will be stored in HIGH_MEMORY.txt"
~
~
~
```

- Then press **ESC + : + W + Q** to save the script. Then we have to execute the script, use command **chmod +x Scriptname.sh**.
- Now we have to open the script use the command **./Scriptname.sh**
- You will find the HIGH_MEMORY.txt file in the logs. To see the logs use the command **ls**. The image is displayed below.

```
root@9c507856cb7a:/# ls
HIGH_MEMORY.txt  boot  etc  lib  media  opt  process.txt  run  schript1.sh  sys  usr
bin              dev  home lib64 mnt  proc  root        sbin  srv          tmp  var
```

- The logs file is already created by using **mkdir -p ./logs**. **-p** creates an directory with the name which we gave, if it is already present then it will replace by the new one over the existing one.

A2)Create files/dirs with a script

Scenario: A project bootstrap script should lay down a simple workspace.

Task:

- 1) Creates a directory tree: project/{src,bin,logs}.
- 2) Creates an empty file README.md and a file src/app.sh.
- 3) Adds execute permission only to bin and src/app.sh.

Deliverables: Script + screenshot of resulting tree.

Success criteria: Tree structure and permissions match requirement.

Answer:

- Kepp your ubuntu machine updated and installed all necessary softwares. If it is showing command not fund then we can use the **“apt install”**
- Now we have to create an directory name with project and then we enter into those directory and create sub directories and empty files in the projects directory according to the requirements
- Open the editor and follo the below commands to do this process.

```
root@9c507856cb7a: /
#!/bin/bash

mkdir -p projects
cd projects
mkdir -p src
mkdir -p bin
mkdir -p logs
touch README.md
touch src/app.sh
echo "Hello I am README.md" > README.md
echo "Hello I am src/app.sh" > src/app.sh
chmod 700 README.md
chmod 700 src/app.sh
~
~
```

- The output is showed below like a tree structure for better understanding. If you don't have tree command then install it by using "apt install tree"

```
root@9c507856cb7a: /# tree projects
projects
|-- README.md
|-- bin
|-- logs
`-- src
    |-- app.sh

4 directories, 2 files
root@9c507856cb7a: /# |
```

B. LINUX BASICS

B1) Checking processes

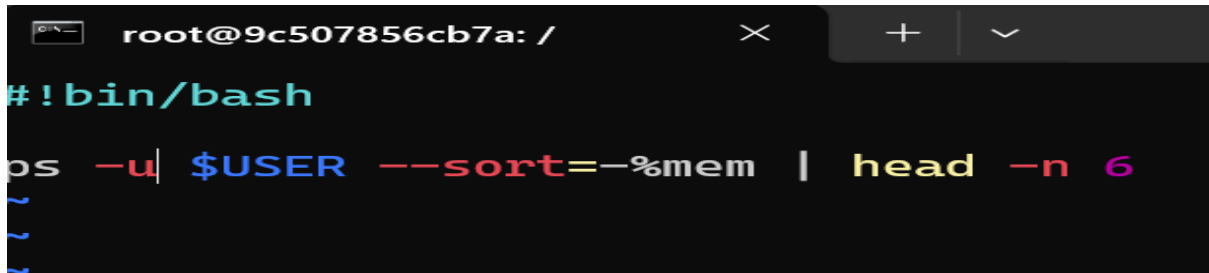
Task: List all processes for the current user, show top 5 memory-consuming processes.

Deliverables: Commands used + screenshots.

Success criteria: Correct process lists visible.

Answer:

- Follow the below commands to do this



```
root@9c507856cb7a: /  
#!/bin/bash  
ps -u $USER --sort=-%mem | head -n 6
```

- The output for this script is:

```
root@9c507856cb7a:/# ./script3.sh  
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND  
root        495  0.0  0.1   7888   3968 pts/0    R+   16:43   0:00 ps -u --sort=-%mem  
root         1  0.0  0.0   4588   3456 pts/0    Ss   15:31   0:00 /bin/bash  
root       494  0.0  0.0   4324   3072 pts/0    S+   16:43   0:00 bin/bash ./script3.sh  
root       496  0.0  0.0   2708   1408 pts/0    S+   16:43   0:00 head -n 6
```

- ps → Shows running processes.
-u \$USER → Only for your username.
--sort=-%mem → Sort by memory usage, biggest first (- means descending).
head -n 6 → First 6 lines (1 header + 5 processes).

B2) File management (cat, touch, vi, nano)

Task: Create a file, add 3 lines using nano or vi, display contents with cat.

Deliverables: Commands + file content screenshot.

Success criteria: File exists with expected content.

Answer:

- For this create an Empty file by using “**touch**” command.
- For insertion of text in the file use “**vi filename.txt**”
- Insert what ever you want and after that save and exit from the editor.
- If you want to see the file content use “**cat**” command to see the content.
- Follow the below commands

```
root@9c507856cb7a:/# touch kumar1.txt
root@9c507856cb7a:/# vi kumar1.txt
root@9c507856cb7a:/# cat kumar1.txt
Hello Buddies, I am Learning DevOps.
My trainer is Akshat Guptha Sir.
Thank you ...
root@9c507856cb7a:/# |
```

B4) cd commands

Task: Navigate from home to /var/log, then back using a single command; print current directory each time.

Deliverables: Commands + outputs.

Success criteria: Correct paths shown

Answer:

- cd /var/log → go from home to /var/log.
- pwd → print current directory (/var/log).
- cd - → go back to the previous directory (home in this case).
- pwd → print current directory again.

```
root@9c507856cb7a:/# pwd
/
root@9c507856cb7a:/# cd /var/log
root@9c507856cb7a:/var/log# cd ..
root@9c507856cb7a:/var# pwd
/var
root@9c507856cb7a:/var# cd ..
root@9c507856cb7a:/# pwd
/
root@9c507856cb7a:/# cd /var/log && pwd && cd .. && pwd
/var/log
/var
```

B5) File editing

Task: Open a config file (create one), add a key=value pair, save, and show the line number where it exists.

Deliverables: Editor steps + proof via command output.

Success criteria: Key=value present with line number.

Answer:

- `grep` → searches for text inside a file.
- `-n` → shows **line numbers** where the match is found.
- `"Key=Lakshmi Kumar"` → the text we are looking for.
- `kumar.config` → the file to search in.

```
root@9c507856cb7a:/# vi kumar.config
root@9c507856cb7a:/# cat kumar.config
Key=Lakshmi Kumar
root@9c507856cb7a:/# grep -n "Key=Lakshmi Kumar" kumar.config
1:Key=Lakshmi Kumar
root@9c507856cb7a:/# |
```

B6) Install package

Task: Install a basic CLI tool (e.g., htop).

Deliverables: Command + version output.

Success criteria: Tool runs and shows version.

Answer:

- Here we are installing apache2 web server using ubuntu.
- First we have to update the machine
- Now we can install the web server by using **"apt install apache2 -y"**.
- It will install the webserver and also we can see the version of the server by using the command **"apache2 -v"**.

```
root@4d065ac629bf:/Mydir# apache2 -v
Server version: Apache/2.4.58 (Ubuntu)
Server built:   2025-07-14T16:22:22
root@4d065ac629bf:/Mydir# |
```

B3) Remove command

Task: Create a temp directory with 3 files inside, then remove only files ending with .tmp.

Deliverables: Commands + before/after listing.

Success criteria: Only .tmp files removed.

Answer:

- Here we are creating an directory and within the directory we create some file with different extensions.
- After creating it, we remove only file with .tmp extension.
- The "*" selects all the file with the .tmp extension
- For this follow the below commands

```
root@4d065ac629bf:/# mkdir Mydir
root@4d065ac629bf:/# cd Mydir
root@4d065ac629bf:/Mydir# touch Kumar.txt kumar.tmp kumar1.tmp kumar2.txt
root@4d065ac629bf:/Mydir# ls
Kumar.txt  kumar.tmp  kumar1.tmp  kumar2.txt
root@4d065ac629bf:/Mydir# rm *.tmp
root@4d065ac629bf:/Mydir# ls
Kumar.txt  kumar2.txt
root@4d065ac629bf:/Mydir#
```

B7) Remove package

Task: Uninstall the same package without removing unrelated deps.

Deliverables: Commands + verification it's gone.

Success criteria: Command not found / removed.

Answer:

- Now we are removing the existed installed package in our ubuntu machine by using the remove command “**apt remove apache2**”
- The output is given below

```
The following packages will be REMOVED:
 apache2
0 upgraded, 0 newly installed, 1 to remove and 0 not upgraded.
After this operation, 465 kB disk space will be freed.
(Reading database ... 7852 files and directories currently installed.)
Removing apache2 (2.4.58-1ubuntu8.7) ...
```

B8) Update

Task: Update package lists and upgrade only security patches (or minimal upgrade).

Deliverables: Commands + summary output.

Success criteria: Update/upgrade completes without errors.

Answer:

- Now we are updating packages and upgrade the packages by using the commands “**apt update -y**” and “**apt upgrade -y**”
- The output is given below

```
Fetched 32.0 MB in 8s (4244 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
4 packages can be upgraded. Run 'apt list --upgradable' to see them.
root@4d065ac629bf:/Mydir# |
```

```
root@4d065ac629bf:/Mydir# apt upgrade -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
The following packages will be upgraded:
 base-files libsystemd0 libudev1 perl-base
4 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
```

C. JIRA

C1) Create & manage project + progress

Task: Create a Kanban or Scrum project, add 4 user stories, assign to 2 users, move across statuses, generate progress report.

Deliverables: Project screenshot, issues list, report/export.

Success criteria: 4 stories exist, assignments correct, progress visible.

Answer:

- Create the project board
- Choose kanban or Scrum what ever you want, here I choosed Scrum (columns: To Do, In Progress, Done).
- Add 4 user stories related to the project
- Assign the stories to the users.
- Change the statuses if the from todo to done

The screenshot displays the Jira web interface for a project named 'My Project'. The view is set to the 'Backlog' tab. A 'MP Sprint 1' is active, containing 4 work items:

Item	Status	Assignee
MP-1 Design	IN PROGRESS	LC
MP-2 Test	IN PROGRESS	
MP-3 QA	DONE	LC
MP-4 Deploy	TO DO	

Below the sprint, there is an empty 'Backlog' section with the message 'Your backlog is empty.' and a 'Create sprint' button. The interface includes a sidebar with navigation options like 'For you', 'Recent', 'Starred', 'Apps', 'Plans', 'Projects', and 'Teams'. The top navigation bar shows the Jira logo, a search bar, and a '+ Create' button. The bottom status bar indicates the system time as 10:04 on 12-08-2025.

Introducing ChatGPT | OpenAI

Kanban project setup

Summary - My Project - Jira

lakshmikumarchalla-1754972492796.atlassian.net/jira/software/projects/MP/summary

GmailYouTubeMaps

Atlassian uses cookies to improve your browsing experience, perform analytics and research, and conduct advertising. Accept all cookies to indicate that you agree to our use of cookies on your device. [Atlassian cookies and tracking notice](#)

PreferencesOnly necessaryAccept all

Jira

Search

Create

See plans

1

3

LC

For you

Recent

Starred

Apps

Plans PREMIUM

Projects

Recent

My Project

View all projects

Teams

More

Projects

My Project

Summary

Timeline

Backlog

Board

Calendar

List

Forms

Goals

All work

Development

Code

More

Filter

1 completed
in the last 7 days

4 updated
in the last 7 days

4 created
in the last 7 days

0 due soon
in the next 7 days

Status overview
Get a snapshot of the status of your work items. [View all work items](#)

4
Total work items

Done: 1

To Do: 1

In Progress: 2

Recent activity
Stay up to date with what's happening across the project.

Today

LC Lakshmi Kumar Challa changed the Assignee to Lakshmi Kumar Challa on MP-3: QA **TO DO** less than a minute ago

LC Lakshmi Kumar Challa changed the Assignee to Lakshmi Kumar Challa on MP-1: Design **IN PROGRESS** 5 minutes ago

LC Lakshmi Kumar Challa updated field "status" on MP-2: Test **IN PROGRESS**

Give feedback on the ne...

29°C
Mostly cloudy

Search

My Project - All work - Jira

lakshmikumarchalla-1754972492796.atlassian.net/jira/software/projects/MP/issues?jql=project%20%3D%20%20MP%20ORDER%20BY%20created%20DESC

GmailYouTubeMaps

Atlassian uses cookies to improve your browsing experience, perform analytics and research, and conduct advertising. Accept all cookies to indicate that you agree to our use of cookies on your device. [Atlassian cookies and tracking notice](#)

PreferencesOnly necessaryAccept all

Jira

Search

Create

See plans

1

3

LC

For you

Recent

Starred

Apps

Plans PREMIUM

Projects

Recent

My Project

View all projects

Teams

More

Projects

My Project

Summary

Timeline

Backlog

Board

Calendar

List

Forms

Goals

All work

Development

Code

More

AI Basic JQL

Search work

Project = My Project

Assignee

Type

Status

Group

More filters

Save filter

	Work	Assignee	Reporter	Priority	Status	
<input type="checkbox"/>	<input checked="" type="checkbox"/> MP-4 Deploy	Unassigned	LC Lakshmi Kumar C...	Medium	TO DO	Unre
<input type="checkbox"/>	<input checked="" type="checkbox"/> MP-3 QA	LC Lakshmi Kumar C...	LC Lakshmi Kumar C...	Medium	DONE	Don
<input type="checkbox"/>	<input checked="" type="checkbox"/> MP-2 Test	Unassigned	LC Lakshmi Kumar C...	Medium	IN PROGRESS	Unre
<input type="checkbox"/>	<input checked="" type="checkbox"/> MP-1 Design	LC Lakshmi Kumar C...	LC Lakshmi Kumar C...	Medium	IN PROGRESS	Unre

+ Create

4 of 4

Give feedback on the ne...

29°C
Mostly cloudy

Search

12-08-2025

C2) Automation rule

Task: Create an automation rule triggered when an issue is moved to Ready for QA, assigning QA lead or sending email.

Deliverables: Rule config screenshots + a test run proof.

Success criteria: Rule fires as expected.

Answer:

- In Jira Cloud: Project settings → Automation → Create rule.
- Trigger → Workitem transitioned.
- From status: Todo
- To status: Done
- Add the action: Send email
- Sub: Work item transferred to..
- Give the name to the rule and turn on the rule.

The screenshot displays the Jira Cloud 'Rule builder' interface for a project named 'My Project'. The rule is titled 'MyRule' and is currently 'ENABLED'. The configuration is as follows:

- When:** Work item transitioned. From: To Do, To: In Progress.
- Then:** Add comment to work item. Comment: 'Hello I am transferring to you Dear...'.
- And:** Send email. Assignee: 'Work Item is Transferred...'.

A notification banner at the top right states: 'Your rule has been turned on. How was your automation experience? (optional)'. The left sidebar shows the navigation menu with 'Automation' selected. The bottom of the screen shows the Windows taskbar with the date '12-08-2025' and time '10:19'.

C3) Kanban board

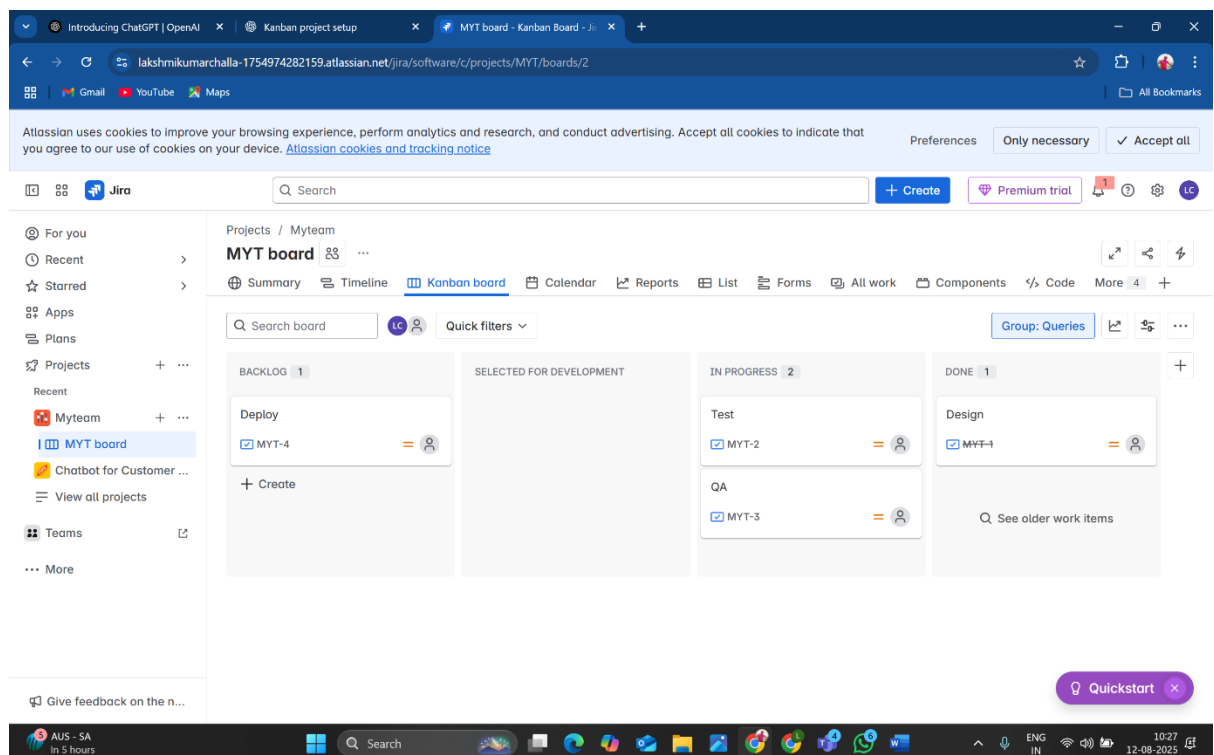
Task: Create a Kanban board with columns: To Do, In Progress, In Review, Done.

Deliverables: Board screenshot with at least 3 issues across columns.

Success criteria: Columns configured; issues visible.

Answer:

- Click Projects → Create project (or use an existing project).
- Select Kanban template.
- Name your project, choose project type (team-managed or company-managed), and create.
- Add issues to the Kanban board. Display the board.



D. YAML

D1) MySQL service descriptor

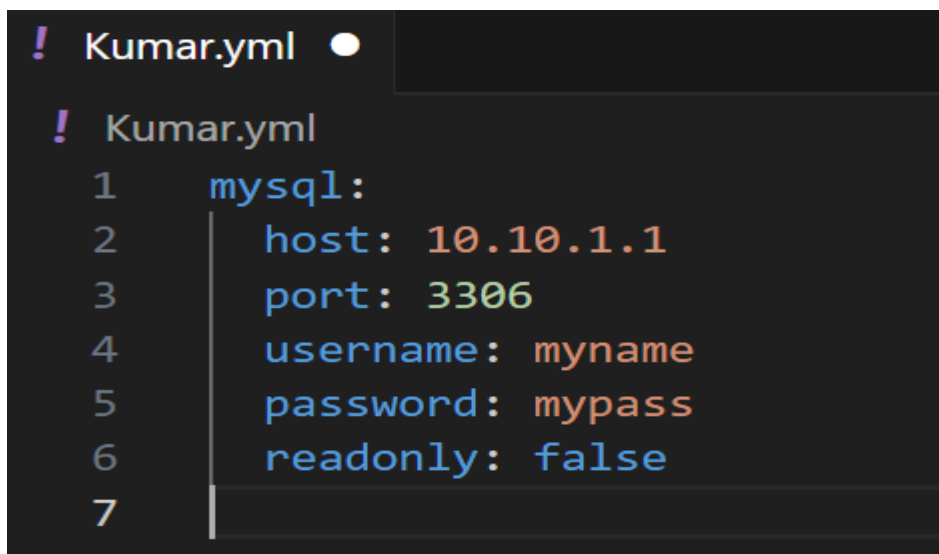
Task: Write YAML describing MySQL server at 10.10.1.1:3306, username=myname, password=mypass, readonly=false.

Deliverables: mysql-config.yaml file.

Success criteria: YAML is valid and human-readable.

Answer:

- Open Vscode create a file with name.yaml and write the below code:



```
! Kumar.yml
! Kumar.yml
1  mysql:
2    host: 10.10.1.1
3    port: 3306
4    username: myname
5    password: mypass
6    readonly: false
7
```

E. PYTHON BASICS

E1) if / elif / else

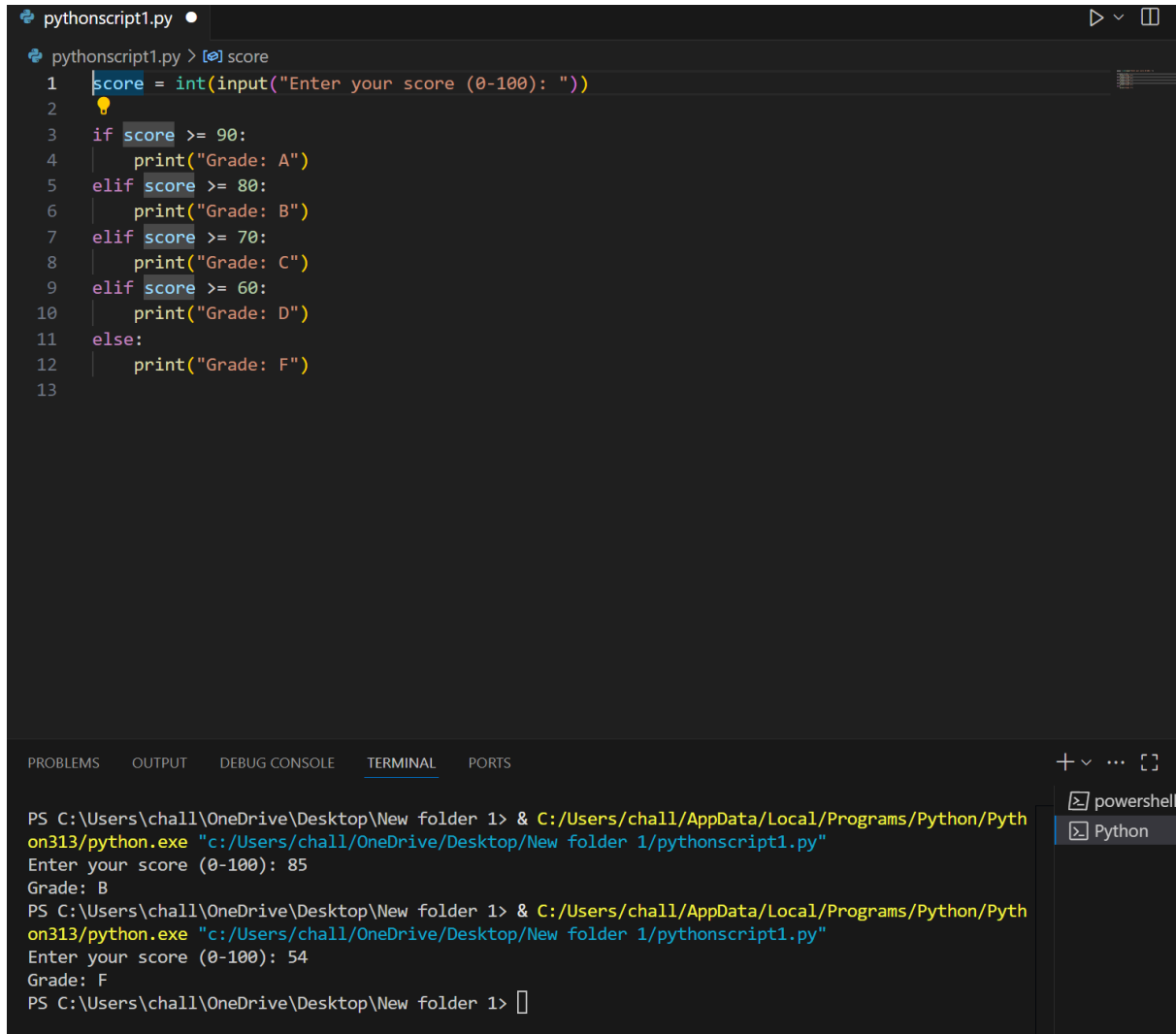
Task: Script to take score (0–100) and print grade A/B/C/D/F.

Deliverables: Script + sample runs.

Success criteria: Correct grades printed.

Answer:

- Open the VS code or Any code editor and return the below code to solve the above question.
- Here I am using VS code.
- The Output will be displayed below



The image shows a screenshot of a Visual Studio Code editor window. The editor is open to a file named `pythonscript1.py`. The code in the editor is a Python script that takes a user input for a score and prints the corresponding grade. The code is as follows:

```
1 score = int(input("Enter your score (0-100): "))
2
3 if score >= 90:
4     print("Grade: A")
5 elif score >= 80:
6     print("Grade: B")
7 elif score >= 70:
8     print("Grade: C")
9 elif score >= 60:
10    print("Grade: D")
11 else:
12    print("Grade: F")
13
```

Below the editor, the TERMINAL panel is visible, showing the execution of the script. The terminal output is as follows:

```
PS C:\Users\chall\OneDrive\Desktop\New folder 1> & C:/Users/chall/AppData/Local/Programs/Python/Python313/python.exe "c:/Users/chall/OneDrive/Desktop/New folder 1/pythonscript1.py"
Enter your score (0-100): 85
Grade: B
PS C:\Users\chall\OneDrive\Desktop\New folder 1> & C:/Users/chall/AppData/Local/Programs/Python/Python313/python.exe "c:/Users/chall/OneDrive/Desktop/New folder 1/pythonscript1.py"
Enter your score (0-100): 54
Grade: F
PS C:\Users\chall\OneDrive\Desktop\New folder 1>
```

E2) for loop

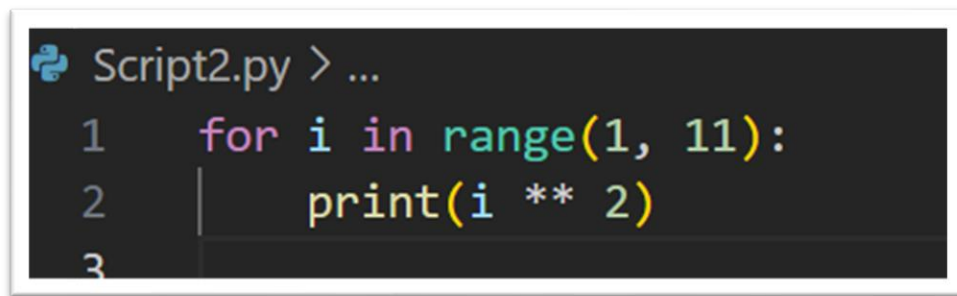
Task: Print squares of numbers 1 to 10.

Deliverables: Script + output.

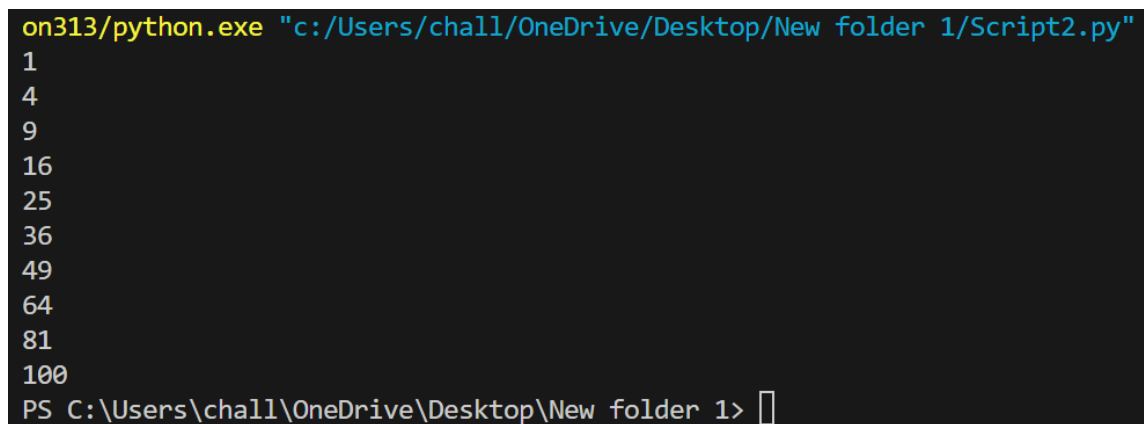
Success criteria: Correct sequence printed.

Answer;

- Open the VS code or Any code editor and return the below code to solve the above question.
- Here I am using VS code.
- The Output will be displayed below

A screenshot of a code editor window showing a Python script. The title bar reads 'Script2.py > ...'. The code consists of three lines: a for loop that iterates from 1 to 10 (range(1, 11)), and a print statement that prints the square of the current value (i ** 2).

```
Script2.py > ...  
1  for i in range(1, 11):  
2      print(i ** 2)  
3
```

A screenshot of a terminal window showing the execution of a Python script. The command 'python.exe "c:/Users/chall/OneDrive/Desktop/New folder 1/Script2.py"' has been executed, resulting in the output of the script: the squares of numbers from 1 to 10. The prompt 'PS C:\Users\chall\OneDrive\Desktop\New folder 1>' is visible at the bottom.

```
on313/python.exe "c:/Users/chall/OneDrive/Desktop/New folder 1/Script2.py"  
1  
4  
9  
16  
25  
36  
49  
64  
81  
100  
PS C:\Users\chall\OneDrive\Desktop\New folder 1>
```

E3) while loop

Task: Sum numbers until total exceeds 50; print how many numbers were added.

Deliverables: Script + output.

Success criteria: Shows count and sum.

Answer:

- Open the VS code or Any code editor and return the below code to solve the above question.
- Here I am using VS code.
- The Output will be displayed below

```
Script2.py > ...
1  total = 0
2  count = 0
3
4  while total <= 50:
5      count += 1
6      total += count
7
8  print(f"Numbers added: {count}")
9  print(f"Final sum: {total}")
10
```

```
on313/python.exe "c:/Users/chall/OneDrive/Desktop/New folder 1/Script2.py"
Numbers added: 10
Final sum: 55
PS C:\Users\chall\OneDrive\Desktop\New folder 1> 
```

E4) print formatting

Task: Print formatted receipt with 3 items, quantities, unit prices, total.

Deliverables: Script + output.

Success criteria: Aligned columns.

Answer:

- Open the VS code or Any code editor and return the below code to solve the above question.
- Here I am using VS code.
- The Output will be displayed below

```
Script2.py > ...
1  print("Item      Qty  Unit Price  Total")
2  print("-----")
3
4  item1 = "Apples"
5  qty1 = 2
6  price1 = 3.50
7  total1 = qty1 * price1
8  print(item1, qty1, price1, total1)
9
10 item2 = "Bananas"
11 qty2 = 5
12 price2 = 1.20
13 total2 = qty2 * price2
14 print(item2, qty2, price2, total2)
15
16 item3 = "Milk"
17 qty3 = 1
18 price3 = 2.75
19 total3 = qty3 * price3
20 print(item3, qty3, price3, total3)
21
22 grand_total = total1 + total2 + total3
23 print("-----")
24 print("Grand Total:", grand_total)
25
```

```

on313/python.exe "c:/Users/chall/OneDrive/Desktop/New folder 1/Script2.py"
Item      Qty  Unit Price  Total
-----
Apples 2 3.5 7.0
Bananas 5 1.2 6.0
Milk 1 2.75 2.75
-----
Grand Total: 15.75
PS C:\Users\chall\OneDrive\Desktop\New folder 1> 

```

F. GIT & GITHUB

F1) Repo, branch, PR, merge

Task: Create GitHub repo, push to feature branch, raise PR, merge.

Deliverables: Command history, GitHub PR screenshots, merged commit link.

Success criteria: PR merged; branch history shows change.

Answer:

- First create an folder and open the git bash in the folder after that initialize the git and do the below commands to solve the above

```

mkdir project1
cd project1
git init
echo "# Assessment1" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/Challakumar241/Assessment1.git
git push -u origin main
git checkout -b feature-branch
echo "This is a Feature Branch" > feature.txt
git add feature.txt
git add feature.txt
git commit -m "Add a new feature file"
git push -u origin feature-branch
git checkout master
git checkout main
git pull origin main

```

Challakumar241 / Assessment1

Type / to search

CodeIssuesPull requests1ActionsProjectsWikiSecurityInsightsSettings

Pulling #1

Merged

Challakumar241 merged 1 commit into `main` from `feature-branch` 1 minute ago

Conversation0Commits1Checks0Files changed1

+1-0

Challakumar241 commented 1 minute ago

Owner

...

I am pulling the request to you

8f28d93

Add a new feature file

Challakumar241 merged commit 5a22b3e into `main` 1 minute ago

Revert

Pull request successfully merged and closed

Delete branch

You're all set — the `feature-branch` branch can be safely deleted.

Reviewers

No reviews

Still in progress? Convert to draft

Assignees

No one—assign yourself

Labels

None yet

Projects

None yet

Milestone

Assessment1

Public

PinWatch0

main2 Branches0 Tags

Go to file

Add file

Code

Challakumar241 Merge pull request #1 from Challakumar241/feature-branch 5a22b3e · 2 minutes ago 3 Commits

README.md

first commit

12 minutes ago

feature.txt

Add a new feature file

6 minutes ago

README

Assessment1

Assessment1

Public

PinWatch0

main2 Branches0 Tags

Go to file

Add file

Code

Challakumar241 Merge pull request #1 from Challakumar241/feature-branch 5a22b3e · 2 minutes ago 3 Commits

README.md

first commit

12 minutes ago

feature.txt

Add a new feature file

6 minutes ago

README

Assessment1

G. MAVEN

G1) Build/deploy Address Book project

Task: Clone repo, run clean build, produce artifact, run locally.

Deliverables: Build logs, artifact path, running proof.

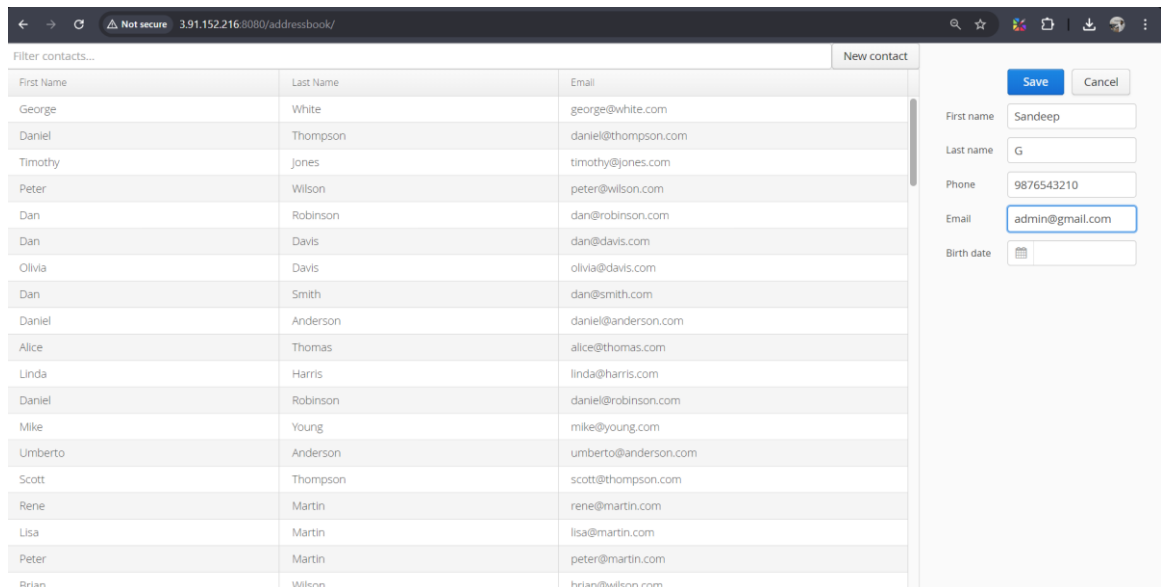
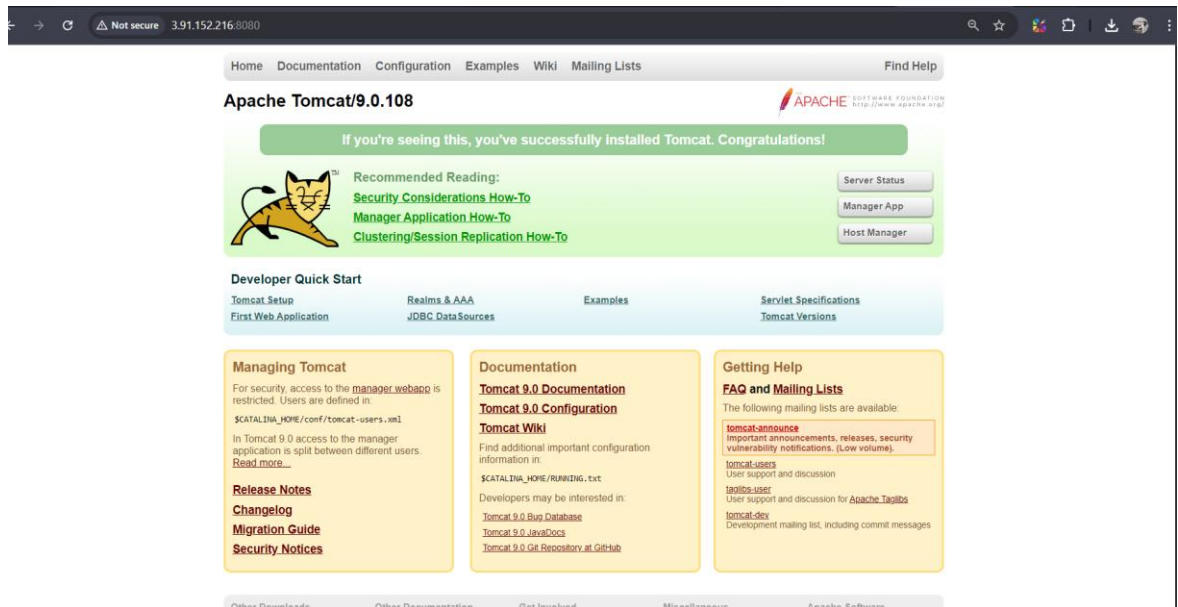
Success criteria: Build succeeds, artifact produced, app runs.

Answer:

- First you need to create an EC2 Instance to do this. After that connect the server to and perform the below tasks.

```
1 apt update
2 apt install maven -y
3 mvn --version
4 git clone https://github.com/akshu20791/addressbook-cicd-project
5 ls
6 cd addressbook-cicd-project
7 ls
8 mvn compile
9 mvn package
10 cd
11 cd..
12 cd addressbook-cicd-project
13 cd
14 cd addressbook-cicd-project
15 cd /home/ubuntu
16 cd addressbook-cicd-project
17 cd /home/ubuntu/addressbook-cicd-project/target/
18 ls
19 cd /home/ubuntu
20 wget https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.108/bin/apache-tomcat-9.0.108.zip
21 apt install unzip -y
22 unzip apache-tomcat-9.0.108.zip
23 cd apache-tomcat-9.0.108
24 cd bin
25 chmod 700 *.sh
26 ./startup.sh
27 cd ..
28 ls
29 cd webapps
30 pwd
31 cd ..
32 ls
33 cd addressbook-cicd-project
34 ls
35 cd target
36 pwd
37 ls
38 cd /home/ubuntu
39 cp /home/ubuntu/addressbook-cicd-project/target/addressbook.war /home/ubuntu/apache-tomcat-9.0.108/webapps
40
```

- You will get an Output like below



H. SONARQUBE

H1) QA report for Address Book

Task: Analyze Address Book project with SonarQube, capture metrics.

Deliverables: Dashboard screenshots or report.

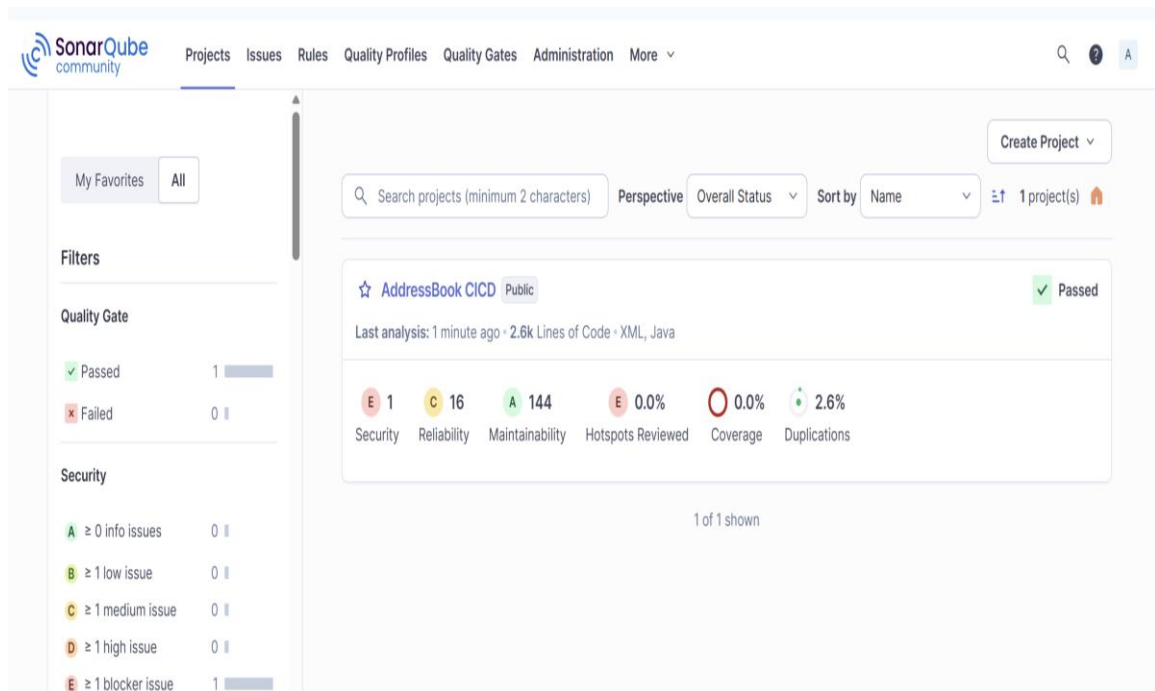
Success criteria: Analysis completed; metrics visible.

Answer:

- First you need to create an EC2 Instance to do this. After that connect the server to and perform the below tasks.

```
1 sudo apt update -y && sudo apt upgrade -y
2 sudo apt install -y docker.io docker-compose unzip git
3 mkdir sonarqube-docker && cd sonarqube-docker
4 nano docker-compose.yml
5 sudo docker-compose up -d
6 cd ~
7 wget https://binaries.sonarsource.com/Distribution/sonar-scanner-cli/sonar-scanner-cli-5.0.1.3006-linux.zip
8 unzip sonar-scanner-cli-5.0.1.3006-linux.zip
9 sudo mv sonar-scanner-5.0.1.3006-linux /opt/sonar-scanner
10 echo 'export PATH=$PATH:/opt/sonar-scanner/bin' >> ~/.bashrc
11 source ~/.bashrc
12 git clone https://github.com/akshu20791/addressbook-cicd-project.git
13 cd addressbook-cicd-project
14 sudo apt install -y openjdk-17-jdk maven
15 mvn clean compile
16 #compile will convert your code from user readable to machine readable
17 sonar.projectKey=addressbook
18 sonar.projectName=AddressBook CICD
19 sonar.projectVersion=1.0
20 sonar.sources=.
21 sonar.java.binaries=target/classes
```

```
Downloaded from central: https://repo.maven.apache.org/maven2/junit/junit/3.8.2/junit-3.8.2.jar (121 kB at 3.9 MB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/log4j/log4j/1.2.12/log4j-1.2.12.jar (358 kB at 11 MB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/com/google/collections/google-collections/1.0/google-collections-1.0.jar (640
kB at 16 MB/s)
[INFO] Changes detected - recompiling the module!
[INFO] Compiling 13 source files to /root/addressbook-cicd-project/target/classes
[INFO] /root/addressbook-cicd-project/src/main/java/com/edurekademo/tutorial/addressbook/AddressbookUI.java: Some input files use or overrid
e a deprecated API.
[INFO] /root/addressbook-cicd-project/src/main/java/com/edurekademo/tutorial/addressbook/AddressbookUI.java: Recompile with -Xlint:deprecati
on for details.
[INFO] /root/addressbook-cicd-project/src/main/java/com/edurekademo/tutorial/addressbook/backend/ContactService.java: /root/addressbook-cicd
-project/src/main/java/com/edurekademo/tutorial/addressbook/backend/ContactService.java uses unchecked or unsafe operations.
[INFO] /root/addressbook-cicd-project/src/main/java/com/edurekademo/tutorial/addressbook/backend/ContactService.java: Recompile with -Xlint:
unchecked for details.
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 6.107 s
[INFO] Finished at: 2025-08-12T06:12:05Z
[INFO] -----
root@ip-172-31-81-140:~/addressbook-cicd-project#
```



I. JENKINS

11) Install Jenkins on Ubuntu EC2

Task: Provision EC2, install Jenkins, ensure starts on boot, access UI.

Deliverables: Commands, service status, UI screenshot.

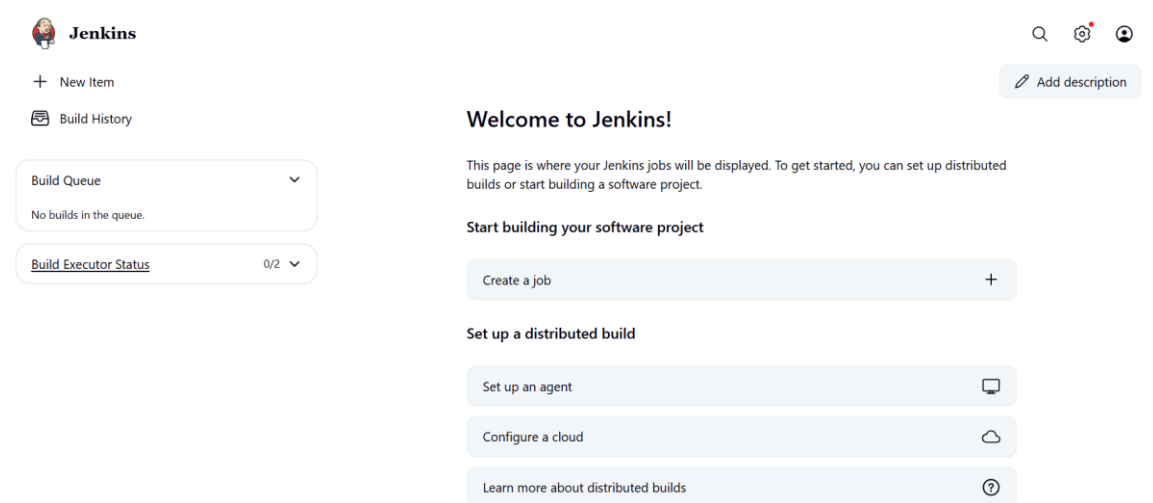
Success criteria: Jenkins reachable and setup completed.

Answer;

- First you need to create an EC2 Instance to do this. After that connect the server to and perform the below tasks


```
No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
4cbdc2185a8c447995dc9ab38c6b7d13
root@ip-172-31-42-58:/home/ubuntu# cat 4cbdc2185a8c447995dc9ab38c6b7d13
cat: 4cbdc2185a8c447995dc9ab38c6b7d13: No such file or directory
root@ip-172-31-42-58:/home/ubuntu# cat /var/lib/jenkins/secrets/initialAdminPassword
4cbdc2185a8c447995dc9ab38c6b7d13
root@ip-172-31-42-58:/home/ubuntu# ^C
root@ip-172-31-42-58:/home/ubuntu# history
1 wget https://raw.githubusercontent.com/akshu20791/Deployment-script/refs/heads/main/jenkins.sh
2 ls
3 chmod 700 jenkins.sh
4 ls
5 ./jenkins.sh
6 ./jenkins.sh
7 cat 4cbdc2185a8c447995dc9ab38c6b7d13
8 cat /var/lib/jenkins/secrets/initialAdminPassword
9 history
```



PREPARED BY,

CHALLA LAKSHMI KUMAR