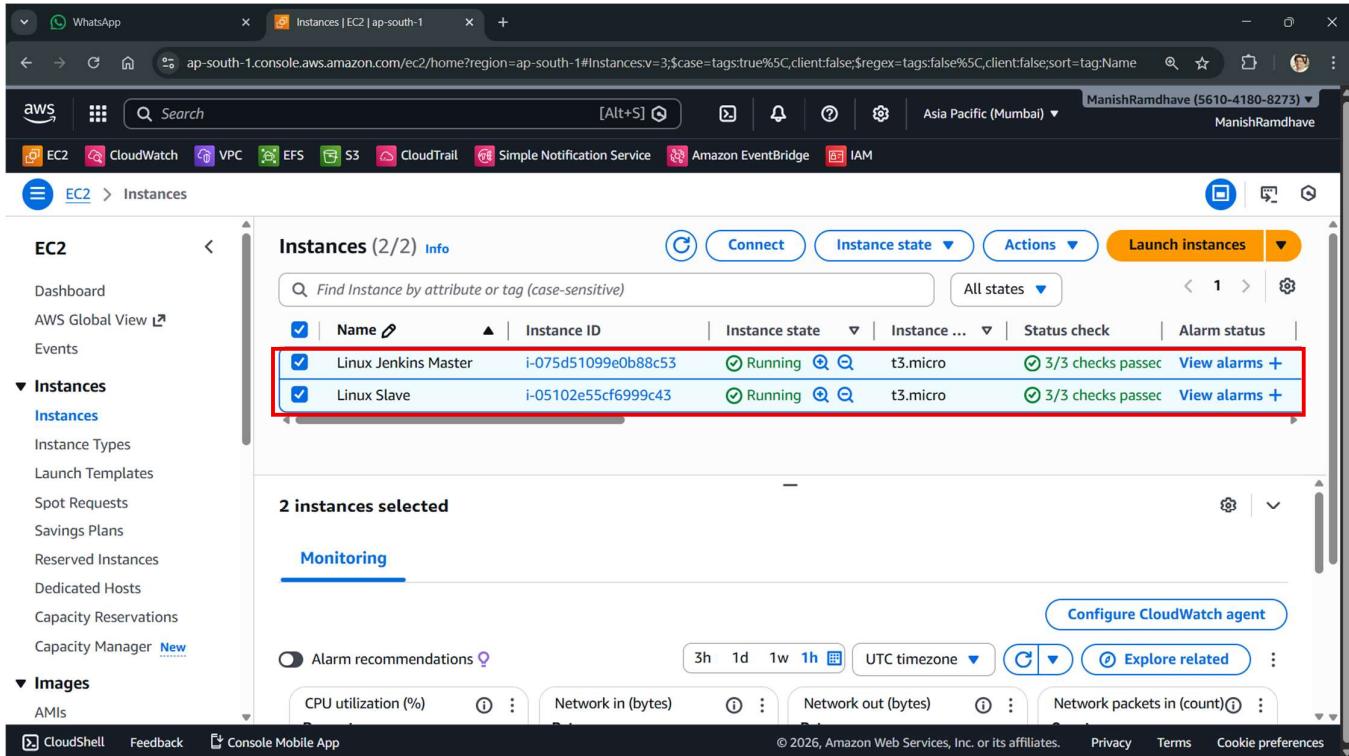


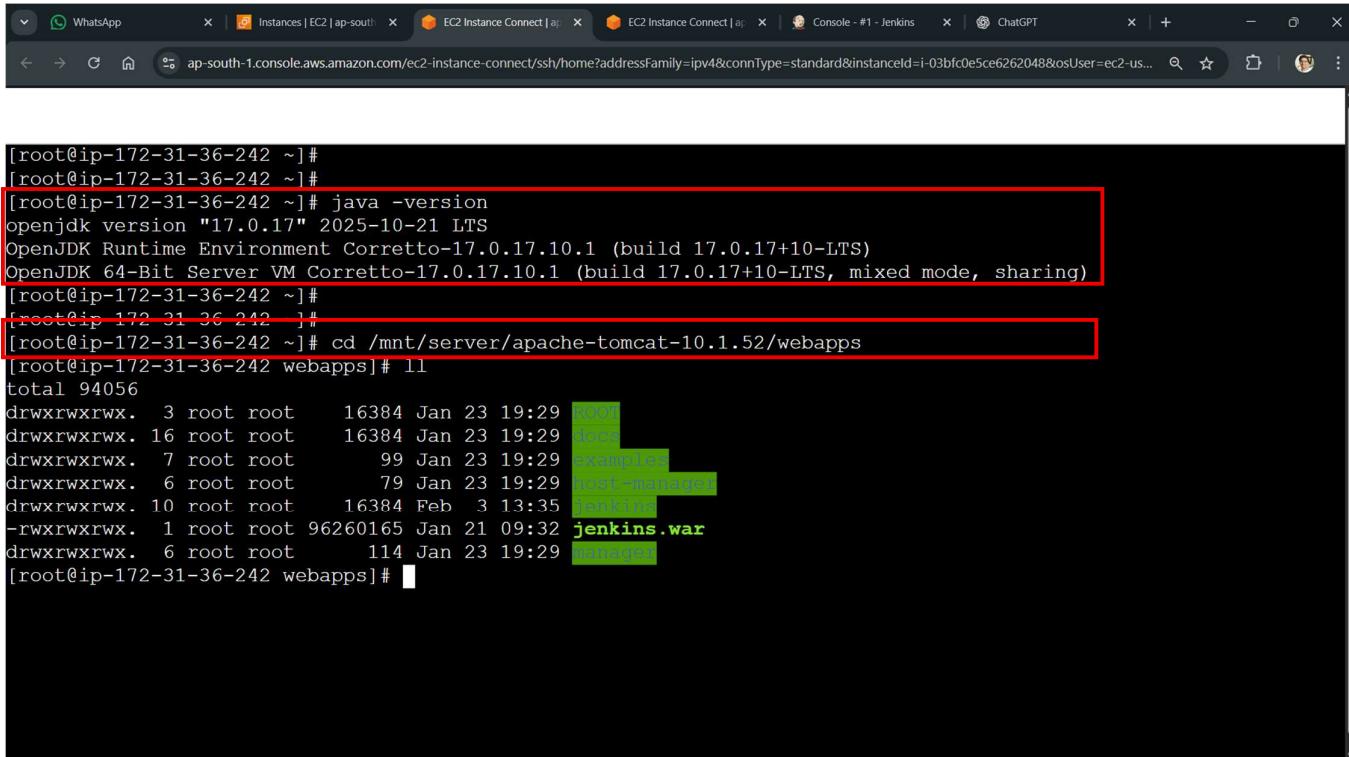
Jenkins Assignment 5

Step 1: Launched two instances, Jenkins Master and Slave1



The screenshot shows the AWS EC2 Instances page. On the left sidebar, under the 'Instances' section, 'Instances' is selected. In the main area, the title 'Instances (2/2) Info' is displayed above a table. The table has columns: Name, Instance ID, Instance state, Status check, and Alarm status. Two rows are present: 'Linux Jenkins Master' (Instance ID: i-075d51099e0b88c53) and 'Linux Slave' (Instance ID: i-05102e55cf6999c43). Both instances are listed as 'Running' with a green checkmark. The 'Status check' column shows '3/3 checks passed' for both. The 'Alarm status' column shows 'View alarms +' for both. A red box highlights both rows. Below the table, it says '2 instances selected'. Under the 'Monitoring' tab, there are sections for 'CPU utilization (%)', 'Network in (bytes)', 'Network out (bytes)', and 'Network packets in (count)'. At the bottom right, there are links for 'Configure CloudWatch agent', 'Explore related', and other options.

Step 2: Installed Java -17 and Apache-Tomcat 10 on both the instance:



```
[root@ip-172-31-36-242 ~]#
[root@ip-172-31-36-242 ~]#
[root@ip-172-31-36-242 ~]# java -version
openjdk version "17.0.17" 2025-10-21 LTS
OpenJDK Runtime Environment Corretto-17.0.17.10.1 (build 17.0.17+10-LTS)
OpenJDK 64-Bit Server VM Corretto-17.0.17.10.1 (build 17.0.17+10-LTS, mixed mode, sharing)
[root@ip-172-31-36-242 ~]#
[root@ip-172-31-36-242 ~]#
[root@ip-172-31-36-242 ~]# cd /mnt/server/apache-tomcat-10.1.52/webapps
[root@ip-172-31-36-242 webapps]# ll
total 94056
drwxrwxrwx. 3 root root 16384 Jan 23 19:29 4001
drwxrwxrwx. 16 root root 16384 Jan 23 19:29 docs
drwxrwxrwx. 7 root root 99 Jan 23 19:29 examples
drwxrwxrwx. 6 root root 79 Jan 23 19:29 host-manager
drwxrwxrwx. 10 root root 16384 Feb 3 13:35 jenkins
-rwxrwxrwx. 1 root root 96260165 Jan 21 09:32 jenkins.war
drwxrwxrwx. 6 root root 114 Jan 23 19:29 manager
[root@ip-172-31-36-242 webapps]#
```

Step 3: Installed Step 2: Installed Java -17 and set an environmental variable and its path too:

The screenshot shows a terminal window within an AWS CloudShell interface. The terminal output is as follows:

```
[root@ip-172-31-38-225 ~]# mvn -v
Apache Maven 3.9.12 (848fb4bf2d427b72bdb2471c22fcfd7ebd9a7a1)
Maven home: /mnt/build-tool/apache-maven-3.9.12
Java version: 17.0.17, vendor: Amazon.com Inc., runtime: /usr/lib/jvm/java-17-amazon-corretto.x86_64
Default locale: en, platform encoding: UTF-8
OS name: "linux", version: "6.1.159-182.297.amzn2023.x86_64", arch: "amd64", family: "unix"
[root@ip-172-31-38-225 ~]# cd /root
[root@ip-172-31-38-225 ~]#
[root@ip-172-31-38-225 ~]# cat .bash_profile
# .bash_profile

# Get the aliases and functions
if [ -f ~/.bashrc ]; then
    . ~/.bashrc
fi

# User specific environment and startup programs
export MAVEN_HOME=/mnt/build-tool/apache-maven-3.9.12

export PATH=$MAVEN_HOME/bin:$PATH
[root@ip-172-31-38-225 ~]#
[root@ip-172-31-38-225 ~]#
```

The terminal window has several sections highlighted with red boxes: the Maven version output, the environment variable setting for MAVEN_HOME, and the .bash_profile content.

Below the terminal, the CloudShell interface shows the instance ID (i-075d51099e0b88c53), public and private IP addresses, and standard navigation links.

Step 4: Cloned the forked repository and installed all dependancies to download the ‘LogiWebApp.war’ application file:

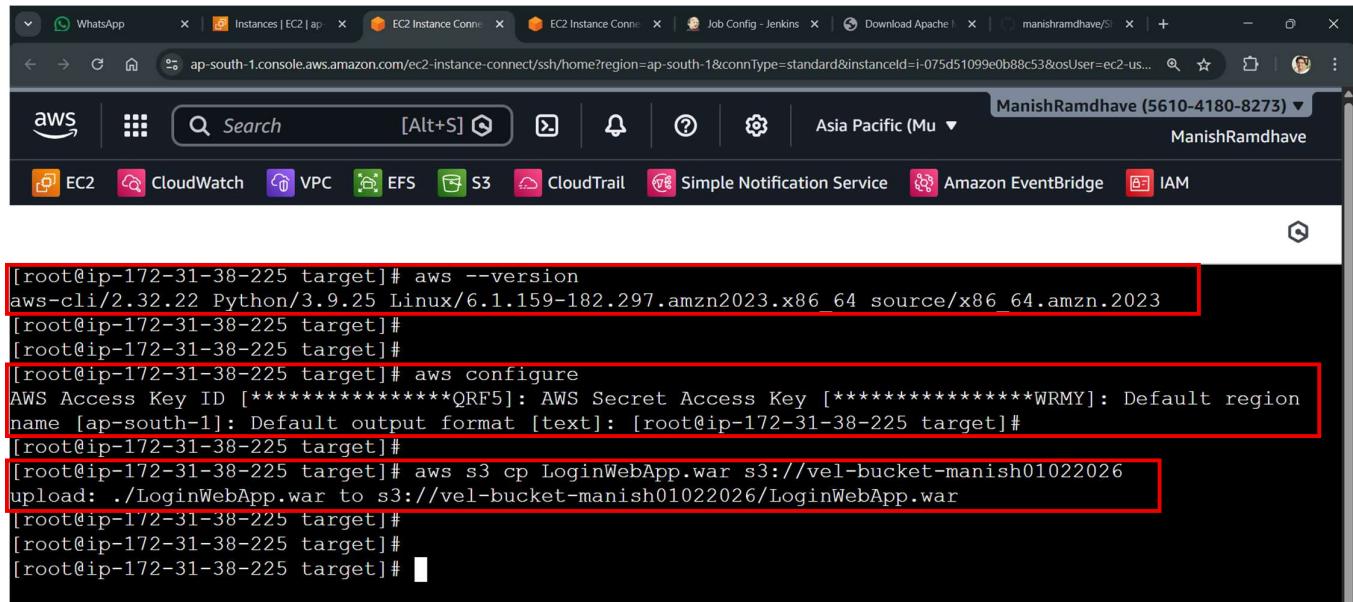
The screenshot shows a terminal window within an AWS CloudShell interface. The terminal output is as follows:

```
[INFO] Installing /mnt/Shantanu_Sir-Repo/pom.xml to /root/.m2/repository/com/javawebtutor/LoginWebApp/1.0-SNAPSHOT/LoginWebApp-1.0-SNAPSHOT.pom
[INFO] Installing /mnt/Shantanu_Sir-Repo/target/LoginWebApp.war to /root/.m2/repository/com/javawebtutor/LoginWebApp/1.0-SNAPSHOT/LoginWebApp-1.0-SNAPSHOT.war
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 7.498 s
[INFO] Finished at: 2026-02-05T08:02:36Z
[INFO] -----
[root@ip-172-31-38-225 Shantanu Sir-Repo]# ll
total 8
-rw-r--r--. 1 root root 507 Feb  5 08:01 Jenkinsfile
-rw-r--r--. 1 root root 949 Feb  5 08:01 pom.xml
drwxr-xr-x. 3 root root  18 Feb  5 08:01 src
drwxr-xr-x. 4 root root  70 Feb  5 08:02 target
[root@ip-172-31-38-225 Shantanu Sir-Repo]# cd target/
[root@ip-172-31-38-225 target]# ll
total 3840
drwxr-xr-x. 4 root root   175 Feb  5 08:02 LoginWebApp
-rw-r--r--. 1 root root 3930146 Feb  5 08:02 LoginWebApp.war
drwxr-xr-x. 2 root root   28 Feb  5 08:02 maven-archiver
[root@ip-172-31-38-225 target]#
```

The terminal window has several sections highlighted with red boxes: the Maven build log, the directory listing after the build, and the final war file in the target folder.

Below the terminal, the CloudShell interface shows the instance ID (i-075d51099e0b88c53), public and private IP addresses, and standard navigation links.

Step 5: On ‘Jenkins Master’, the ‘LoginWebApp.war’ file is uploaded by using the AWS CLI commands. First, we have configured the AWS with Access and Secret Keys. Then we have created a bucket named ‘vel-bucket-manish01022026’ and uploaded the application .war file into the bucket:



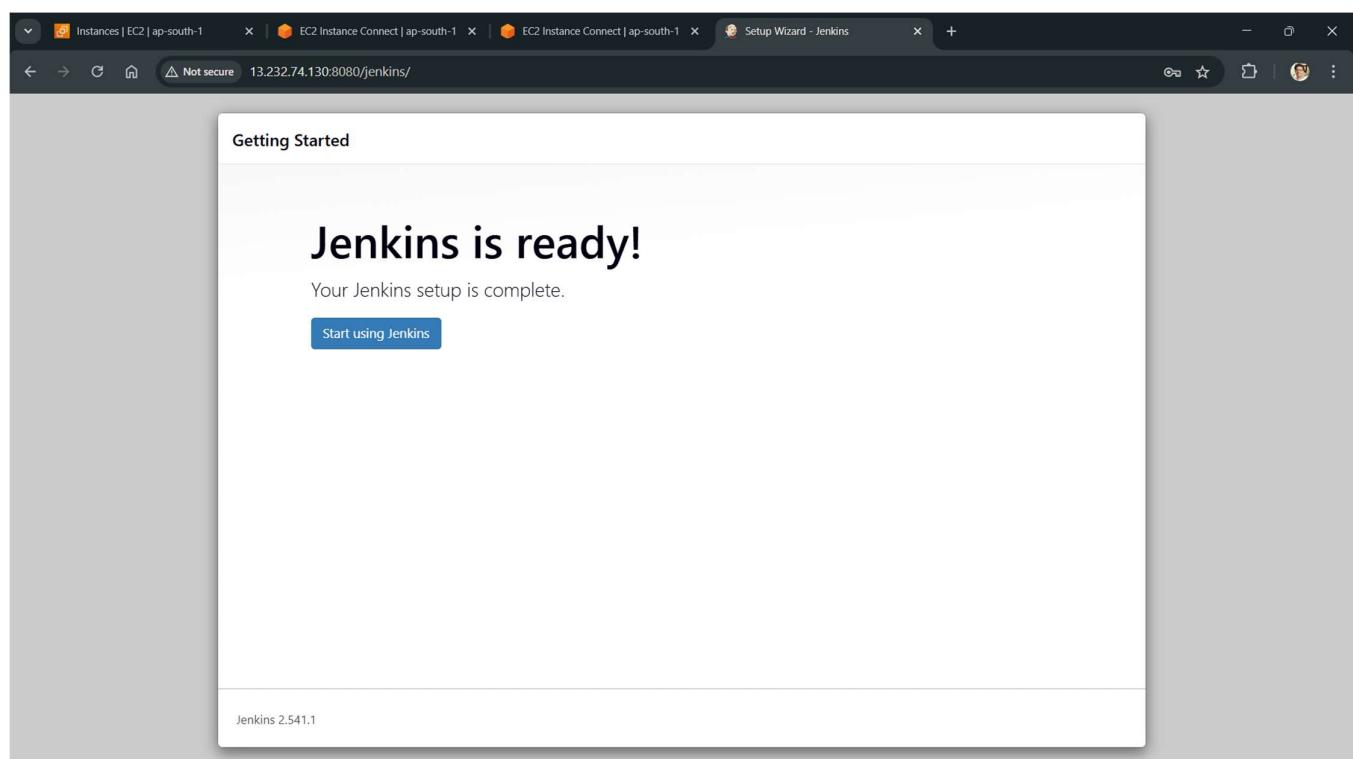
```
[root@ip-172-31-38-225 target]# aws --version
aws-cli/2.32.22 Python/3.9.25 Linux/6.1.159-182.297.amzn2023.x86_64 source/x86_64.amzn.2023
[root@ip-172-31-38-225 target]#
[root@ip-172-31-38-225 target]#
[root@ip-172-31-38-225 target]# aws configure
AWS Access Key ID [*****QRF5]: AWS Secret Access Key [*****WRMY]: Default region
name [ap-south-1]: Default output format [text]: [root@ip-172-31-38-225 target]#
[root@ip-172-31-38-225 target]#
[root@ip-172-31-38-225 target]# aws s3 cp LoginWebApp.war s3://vel-bucket-manish01022026
upload: ./LoginWebApp.war to s3://vel-bucket-manish01022026/LoginWebApp.war
[root@ip-172-31-38-225 target]#
[root@ip-172-31-38-225 target]#
[root@ip-172-31-38-225 target]#
```

i-075d51099e0b88c53 (Linux Jenkins Master) X

[CloudShell](#) [Feedback](#) [Console Mobile App](#) [Privacy](#) [Terms](#) [Cookie preferences](#)

© 2026, Amazon Web Services, Inc. or its affiliates.

Step 6: Installed Jenkins on the Jenkins Master Instance:



Step 7: Created a Node for to establish the connection between Jenkins Master and Slave Instance using the '**Credential**' and '**Manually trusted key Verification Strategy**':

The screenshot shows the Jenkins 'Nodes' page. At the top, there are tabs for 'Instances | EC2 | ap-south-1', 'EC2 Instance Connect | ap-south-1', 'EC2 Instance Connect | ap-south-1', and 'Nodes - Manage Jenkins - Jenkins'. The URL in the address bar is '13.232.74.130:8080/jenkins/computer/'. The page title is 'Jenkins / Nodes'. On the right, there are icons for search, gear, and user. Below the title, a button for '+ New Node' and a 'Configure Monitors' button are visible.

Add, remove, control and monitor the various nodes that Jenkins runs jobs on.

S	Name ↓	Architecture	Clock Difference	Free Disk Space	Free Swap Space	Free Temp Space	Response Time
	Built-In Node	Linux (amd64)	In sync	5.72 GiB	0 B	5.72 GiB	0ms
	Slave1		N/A	N/A	N/A	N/A	
	last checked	13 sec	13 sec	13 sec	13 sec	13 sec	13 sec

Icon: S M L

Legend

Step 8: Created an ‘Access.sh’ shell script file for AWS Configuration **on Slave1 instance** for Authentication purpose. This script will be directly run by Jenkins ‘Job’:

Step 9: Launched the Jenkins and created a Freestyle Job:

The screenshot shows the Jenkins dashboard at 13.234.202.85:8080/jenkins/. The main content area displays a table of build items. A specific row for a 'Job' item is highlighted with a red border. The table columns include Status (S), Workstation (W), Name, Last Success, Last Failure, and Last Duration. The 'Job' entry has a green checkmark icon, a yellow cloud icon, and the name 'Job'. It shows a 'Last Success' of 27 min (#27), a 'Last Failure' of 28 min (#26), and a 'Last Duration' of 3.9 sec. Below the table, there are sections for 'Build Queue' (empty), 'Build Executor Status' (Built-In Node 0/2, Slave1 0/10), and 'Icon' selection (S, M, L). At the bottom right, links for 'REST API' and 'Jenkins 2.54.1' are visible.

Step 10: Enabled 'Restrict where this project can be run' on only Slave1 Instance and used a custom workspace to '/mnt/CustomWorkspace':

The screenshot shows the 'Configure' screen for the 'Job' project at 13.234.202.85:8080/jenkins/job/Job/configure. The left sidebar lists configuration sections: General, Source Code Management, Triggers, Environment, Build Steps, and Post-build Actions. The 'General' section is selected and expanded. In the main pane, under 'Advanced' settings, two key configurations are highlighted with red boxes: 'Restrict where this project can be run' (with a 'Label Expression' of 'Slave1') and 'Use custom workspace' (with a 'Directory' of '/mnt/CustomWorkspace'). Both checkboxes are checked. Other options like 'Quiet period', 'Retry Count', and 'Block build when upstream/downstream projects are building' are also listed but not highlighted.

Step 11: Executed a shell script to run AWS CLI commands to run only on the Slave instance. It will first run the ‘Access.sh’ shell script for AWS configuration and Authentication and then, it will download the ‘LoginWebApp.war’ to the Tomcat-10-Server and after that, it will delete the same bucket forcefully:

The screenshot shows the Jenkins job configuration interface. On the left, there's a sidebar with links: General, Source Code Management, Triggers, Environment, Build Steps (which is selected and highlighted in grey), and Post-build Actions. The main area is titled 'Configure' and has a red box around the 'Build Steps' section. Inside this section, there's a heading 'Build Steps' with the sub-instruction 'Automate your build process with ordered tasks like code compilation, testing, and deployment.' Below this, there's a 'Command' section with the text:

```
cd /root
./Access.sh
cd /mnt/server/apache-tomcat-10.1.52/webapps
aws s3 cp s3://vel-bucket-manish01022026/LoginWebApp.war .
aws s3 rb s3://vel-bucket-manish01022026 --force
```

At the bottom of the 'Build Steps' section, there are 'Advanced' and '+ Add build step' buttons, and two primary action buttons: 'Save' and 'Apply'.

Result:

- When Build is done by ‘Job’, our ‘LoginWebApp.war’ file is first downloaded from the S3 Bucket (vel-bucket-manish01022026) and deployed on Tomcat Server of a **Slave1 instance**:

The screenshot shows the Jenkins console output for build #8. The left sidebar includes links for Status, Changes, Console Output (which is selected and highlighted in grey), Edit Build Information, Delete build #8, Timings, and Previous Build. The main area is titled 'Console' and contains a red box around the log output. The log shows the following steps:

```
Started by user manish
Running as SYSTEM
Building remotely on Slave1 in workspace /mnt/Custom Workspace
[Custom Workspace] $ /bin/sh -xe /tmp/jenkins10223802183903098355.sh
+ cd /root
+ ./Access.sh
AWS CLI credentials configured
+ cd /mnt/server/apache-tomcat-10.1.52/webapps
+ aws s3 cp s3://vel-bucket-manish01022026/LoginWebApp.war .
Completed 256.0 KiB/3.7 MiB (378.2 KiB/s) with 1 file(s) remaining
Completed 512.0 KiB/3.7 MiB (755.1 KiB/s) with 1 file(s) remaining
Completed 768.0 KiB/3.7 MiB (1.1 MiB/s) with 1 file(s) remaining
Completed 1.0 MiB/3.7 MiB (1.5 MiB/s) with 1 file(s) remaining
Completed 1.2 MiB/3.7 MiB (1.8 MiB/s) with 1 file(s) remaining
Completed 1.5 MiB/3.7 MiB (2.2 MiB/s) with 1 file(s) remaining
Completed 1.8 MiB/3.7 MiB (2.6 MiB/s) with 1 file(s) remaining
Completed 2.0 MiB/3.7 MiB (2.9 MiB/s) with 1 file(s) remaining
Completed 2.2 MiB/3.7 MiB (3.3 MiB/s) with 1 file(s) remaining
Completed 2.5 MiB/3.7 MiB (3.7 MiB/s) with 1 file(s) remaining
Completed 2.8 MiB/3.7 MiB (4.0 MiB/s) with 1 file(s) remaining
Completed 3.0 MiB/3.7 MiB (4.4 MiB/s) with 1 file(s) remaining
Completed 3.2 MiB/3.7 MiB (4.7 MiB/s) with 1 file(s) remaining
Completed 3.5 MiB/3.7 MiB (5.1 MiB/s) with 1 file(s) remaining
Completed 3.7 MiB/3.7 MiB (5.4 MiB/s) with 1 file(s) remaining
download: s3://vel-bucket-manish01022026/LoginWebApp.war to ./LoginWebApp.war
+ aws s3 rb s3://vel-bucket-manish01022026 --force
delete: s3://vel-bucket-manish01022026/LoginWebApp.war
delete: s3://vel-bucket-manish01022026/SampleWebApp.war
remove_bucket: vel-bucket-manish01022026
Finished: SUCCESS
```

2. Our LoginWebApp Application running successfully on Tomcat Server:

The screenshot shows a web browser window with the URL `13.235.17.110:8080/LoginWebApp/` in the address bar. The page title is "Login Page". It contains two input fields: "Username" and "Password", each with a corresponding "Reset" button. Below the password field is a "Login" button. At the bottom left, there is a link "New User [Register Here](#)". The browser has several tabs open in the background, including WhatsApp, Instances | EC2, EC2 Instance Co..., Job - Jenkins, Download Ap..., manishramdh..., and JSP Example.

Login Page

Username

Password

New User [Register Here](#)