

Experiment No. 05

AIM: To Build the pipeline of jobs using Maven /Gradle /Ant in Jenkins,create a pipeline script to Test and deploy an application over the tomcat server.

THEORY:

Jenkins is an open-source automation server used to implement Continuous Integration (CI) and Continuous Deployment (CD) in software development. It helps automate tasks like building, testing, and deploying code, improving efficiency and reducing human errors.

Key Concepts:

1. **Automated Builds:** Jenkins automatically compiles code whenever changes are made, ensuring that the software is always in a buildable state.
2. **Continuous Integration:** Developers frequently commit code to a shared repository. Jenkins automatically triggers builds and runs tests to detect issues early.
3. **Pipelines:** Jenkins workflows are defined using Pipelines in a Jenkinsfile, which can be scripted or declarative.
4. **Plugins:** Jenkins supports plugins for version control (Git), build tools (Maven, Gradle), deployment (Docker, Kubernetes), and more.
5. **Freestyle Projects vs. Pipelines:**
 - **Freestyle Projects:** Simple, GUI-based jobs for basic automation.
 - **Pipelines:** Code-defined workflows for complex CI/CD tasks.

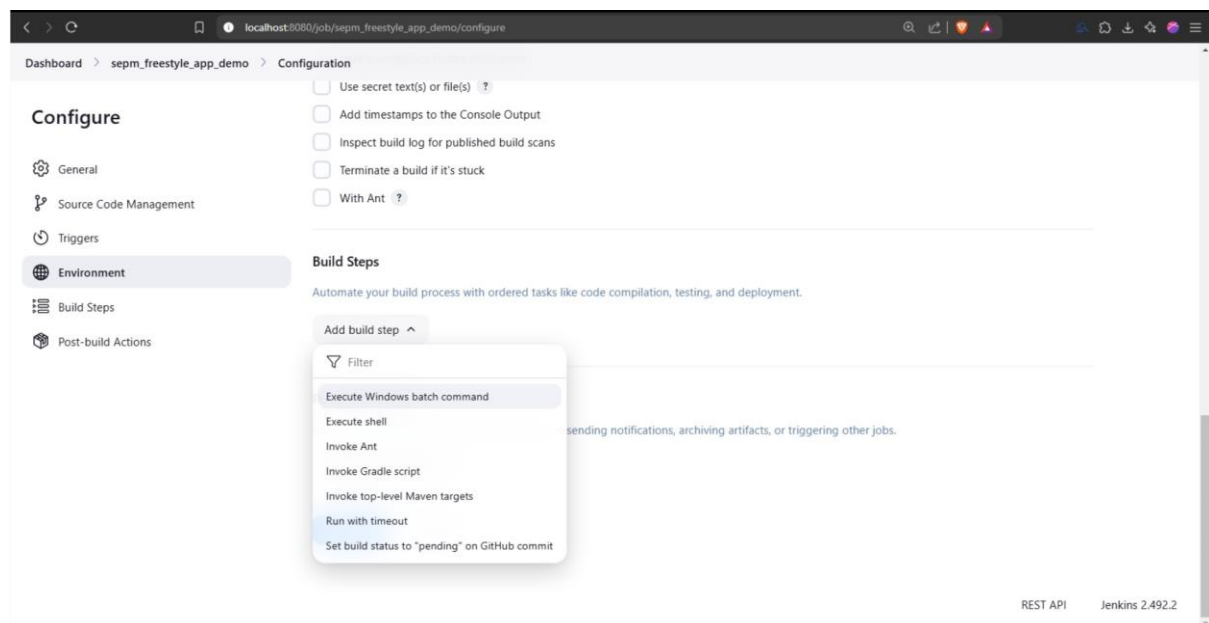
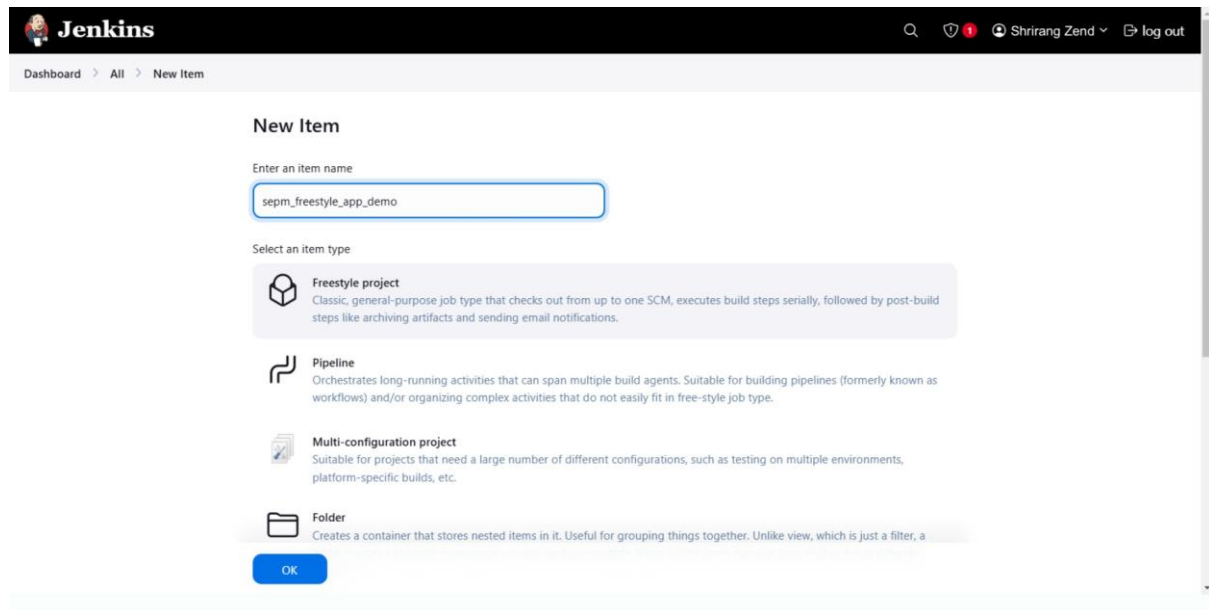
Example of a Simple Jenkins Pipeline (Declarative):

```
pipeline {  
    agent any  
  
    stages {  
        stage('Build') {  
            steps {  
                sh 'echo "Building the project"'  
            }  
        }  
  
        stage('Test') {  
            steps {  
                sh 'echo "Running tests"'  
            }  
        }  
  
        stage('Deploy') {  
            steps {  
                sh 'echo "Deploying the application"'  
            }  
        }  
    }  
}
```

```
}
```

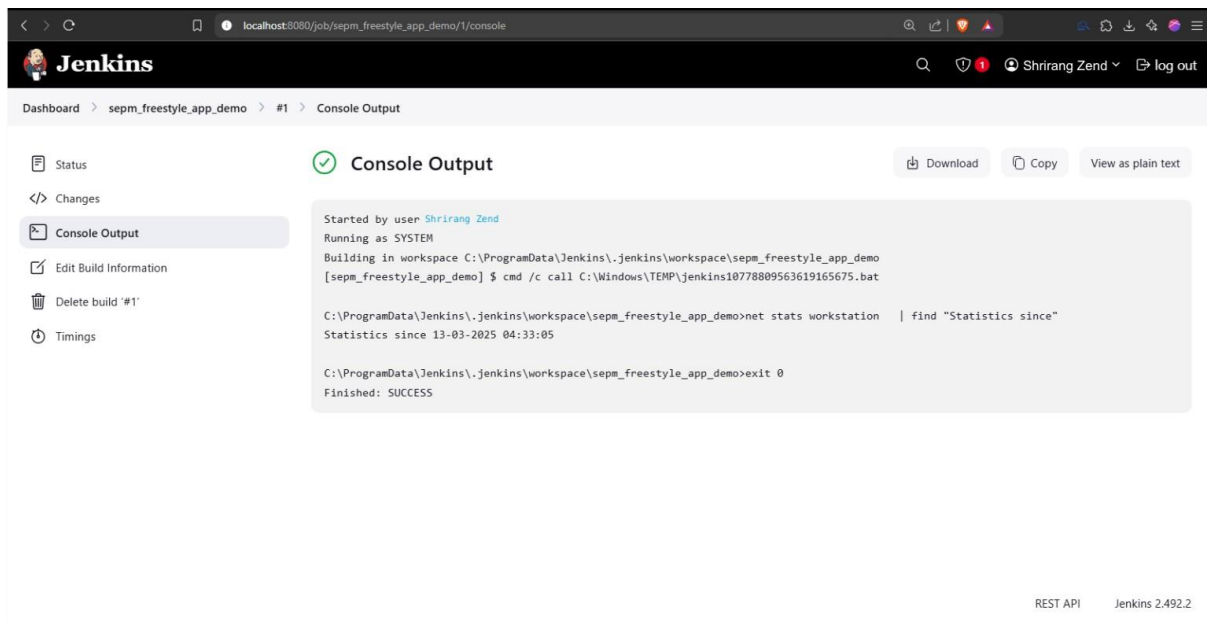
This pipeline defines three stages: **Build**, **Test**, and **Deploy**, each with its own commands.

Output:



The screenshot shows the Jenkins Configuration page for a job named 'sepm_freestyle_app_demo'. The left sidebar contains a 'Configure' section with a list of tabs: General, Source Code Management, Triggers, Environment (selected), Build Steps, and Post-build Actions. The main content area is divided into two sections. The top section, 'Configure', contains several checkboxes: 'Use secret text(s) or file(s)', 'Add timestamps to the Console Output', 'Inspect build log for published build scans', 'Terminate a build if it's stuck', and 'With Ant'. The bottom section, 'Build Steps', has a description: 'Automate your build process with ordered tasks like code compilation, testing, and deployment.' Below this is a dashed box containing a step titled 'Execute Windows batch command'. The 'Command' field is populated with 'net stats workstation | find "Statistics since"'. There is a 'See the list of available environment variables' link above the command field. At the bottom of the dashed box is an 'Advanced' dropdown menu. Below the dashed box are 'Save' and 'Apply' buttons.

The screenshot shows the Jenkins Build #1 page for a job named 'sepm_freestyle_app_demo'. The top header bar includes the Jenkins logo, a search icon, a shield icon, a user profile icon for 'Shrirang Zend', and a 'log out' link. The breadcrumb trail is 'Dashboard > sepm_freestyle_app_demo > #1'. The left sidebar contains a list of links: 'Status' (selected), 'Changes', 'Console Output', 'Edit Build Information', 'Delete build #1', and 'Timings'. The main content area displays the build status as a green checkmark and '#1 (Apr 2, 2025, 11:43:13 AM)'. To the right of the status are links for 'Add description' and 'Keep this build forever'. Below the status, it says 'Started by user Shrirang Zend' and 'Started 4.9 sec ago Took 0.37 sec'. A section titled 'This run spent:' lists the following times: '12 ms waiting;', '0.37 sec build duration;', and '0.38 sec total from scheduled to completion.'. Below this is a code icon and the text 'No changes.'. At the bottom right, it says 'REST API Jenkins 2.492.2'.



The screenshot shows the Jenkins web interface. The top navigation bar includes the Jenkins logo, a search icon, a notification bell, and the user 'Shrirang Zend' with a 'log out' link. The breadcrumb trail is 'Dashboard > sepm_freestyle_app_demo > #1 > Console Output'. On the left sidebar, the 'Console Output' tab is selected, with other options like 'Status', 'Changes', 'Edit Build Information', 'Delete build #1', and 'Timings'. The main area displays the console output for build #1, which is in a 'SUCCESS' state. The output text shows the build starting as 'SYSTEM', running in the workspace 'C:\ProgramData\Jenkins\jenkins\workspace\sepm_freestyle_app_demo', and executing a command to call a batch file. It then shows the output of 'net stats workstation' and 'exit 0'. At the bottom right, there are links for 'REST API' and 'Jenkins 2.492.2'.

Dashboard > sepm_freestyle_app_demo > #1 > Console Output

Status
Changes
Console Output
Edit Build Information
Delete build #1
Timings

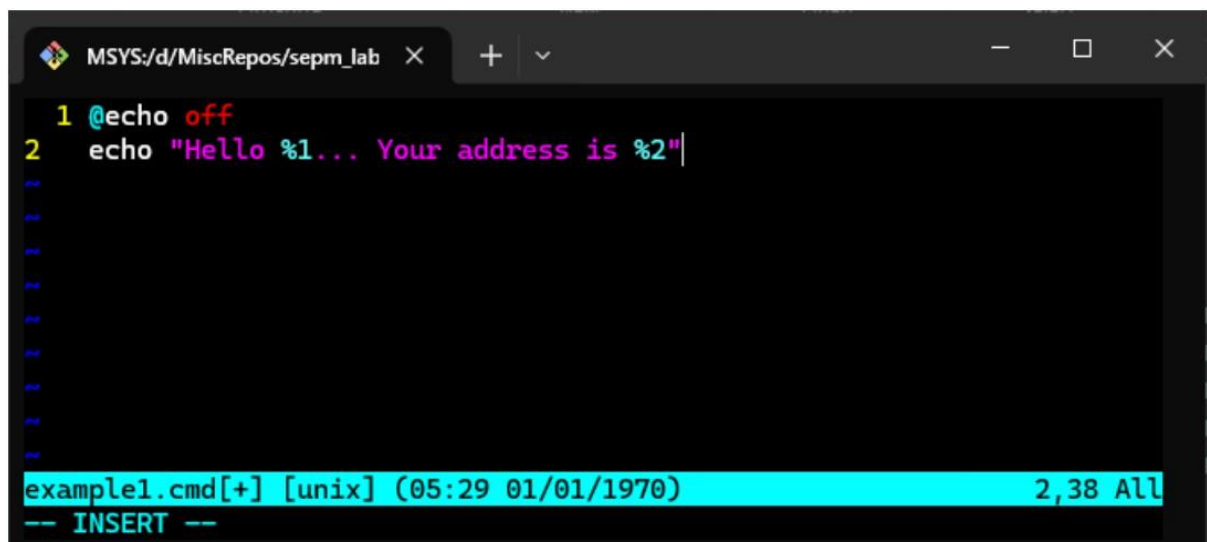
Console Output

Started by user [Shrirang Zend](#)
Running as SYSTEM
Building in workspace C:\ProgramData\Jenkins\jenkins\workspace\sepm_freestyle_app_demo
[sepm_freestyle_app_demo] \$ cmd /c call C:\Windows\TEMP\jenkins1077809563619165675.bat

C:\ProgramData\Jenkins\jenkins\workspace\sepm_freestyle_app_demo>net stats workstation | find "Statistics since"
Statistics since 13-03-2025 04:33:05

C:\ProgramData\Jenkins\jenkins\workspace\sepm_freestyle_app_demo>exit 0
Finished: SUCCESS

REST API Jenkins 2.492.2



The screenshot shows a Windows command prompt window titled 'MSYS:/d/MiscRepos/sepm_lab'. The prompt is at line 1 with '@echo off' and line 2 with 'echo "Hello %1... Your address is %2"'. The command prompt is currently in a state where it is waiting for input. At the bottom, there is a status bar that reads 'example1.cmd[+] [unix] (05:29 01/01/1970) 2,38 All -- INSERT --'.

```
1 @echo off
2 echo "Hello %1... Your address is %2"
```

example1.cmd[+] [unix] (05:29 01/01/1970) 2,38 All
-- INSERT --

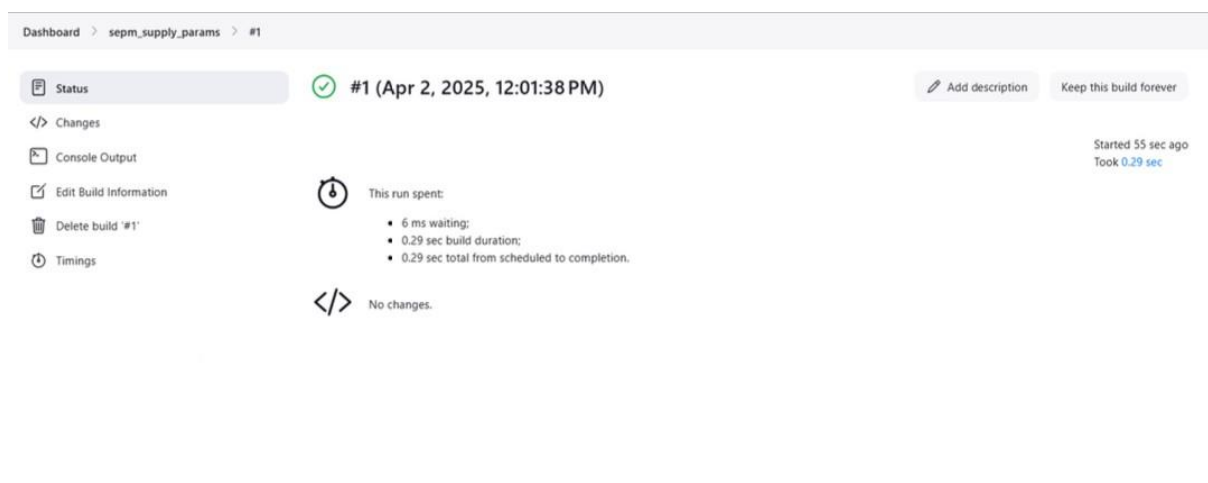
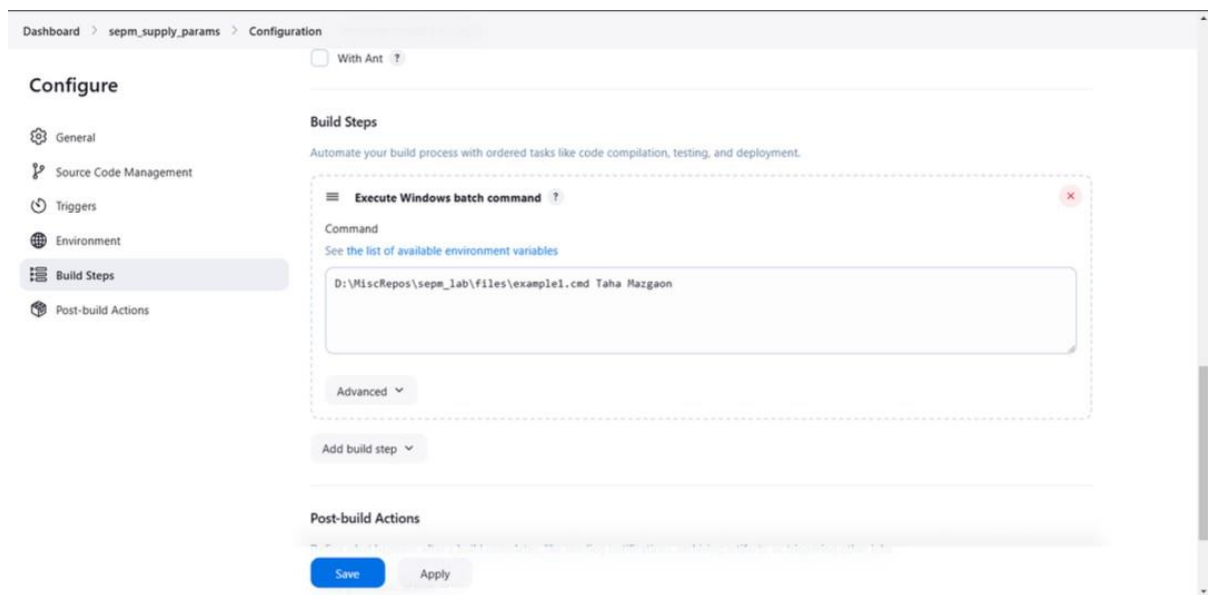
```
Command Prompt
Microsoft Windows [Version 10.0.19045.5608]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Admin>cd "D:\MiscRepos\sepm_lab\files"

C:\Users\Admin>example1.cmd
'example1.cmd' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\Admin>example1.cmd Taha
'example1.cmd' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\Admin>
```



```
vi JavaInJenkins.java + -
6 class JavaInJenkins
5 {
4     public static void main(String[] args)
3     {
2         System.out.println("Hello, from the java file");
1     }
7 }
```

```
/d/MiscRepos/sepm_lab/files git:(master)±5 (5.232s)
```

```
vi JavaInJenkins.java
```

```
/d/MiscRepos/sepm_lab/files git:(master)±5 (1.298s)
```

```
javac JavaInJenkins.java
```

```
/d/MiscRepos/sepm_lab/files git:(master)±5 (1.296s)
```

```
java JavaInJenkins.java
```

```
Hello, from the java file
```

```
/d/MiscRepos/sepm_lab/files git:(master)✓ ±5
```

```
|
```





Dashboard > All > New Item

New Item

Enter an item name

sepm_java_in_jenkins

Select an item type

-  **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

OK

localhost:8080/job/sepm_java_in_jenkins/configure

Dashboard > sepm_java_in_jenkins > Configuration

Configure

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

Build Steps

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

Execute Windows batch command ?

Command

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

```
javac D:\MiscRepos\sepm_lab\files\JavaInJenkins.java
java D:\MiscRepos\sepm_lab\files\JavaInJenkins.java
```

Advanced ▾

Add build step ▾

Post-build Actions

Define what happens after a build completes, like sending notifications, archiving artifacts, or triggering other jobs.

Add post-build action ▾

Save Apply

Dashboard > sepm_parameterized_build > Configuration

Configure

A demo for parameterized builds

Plain text [Preview](#)

- ☐ Discard old builds ?
- ☐ GitHub project
- ☒ This project is parameterized ?

Add Parameter ^

- Filter
- Boolean Parameter
- Choice Parameter
- Credentials Parameter
- File Parameter
- Multi-line String Parameter
- Password Parameter
- Run Parameter
- String Parameter

Source Code Management

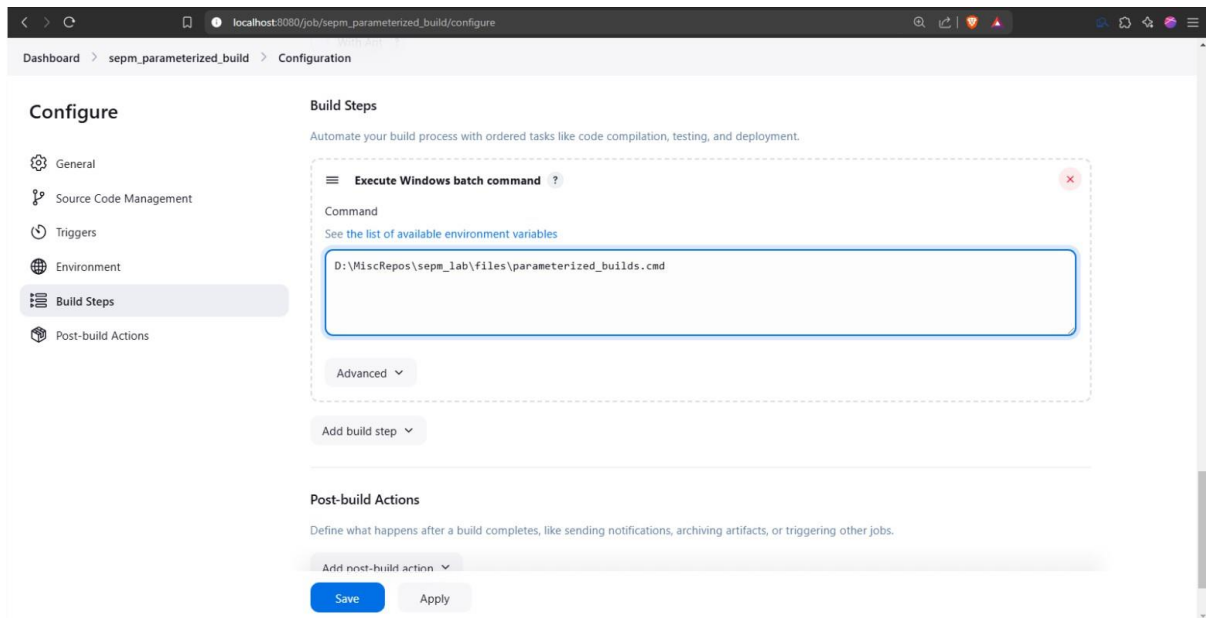
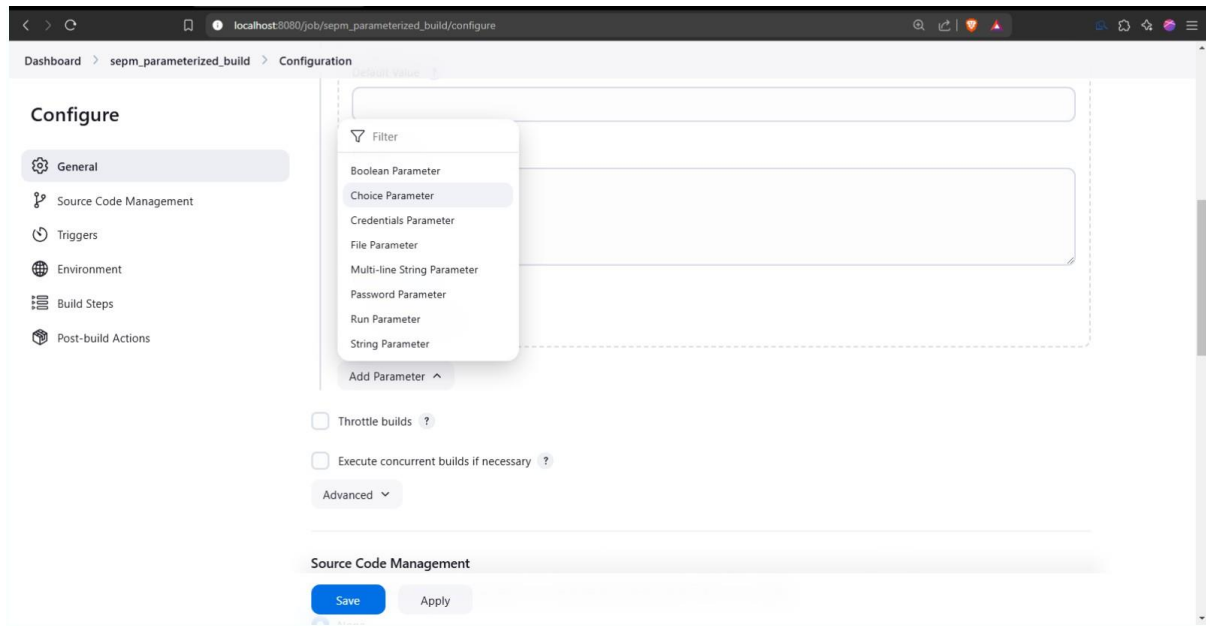
Triggers

Environment

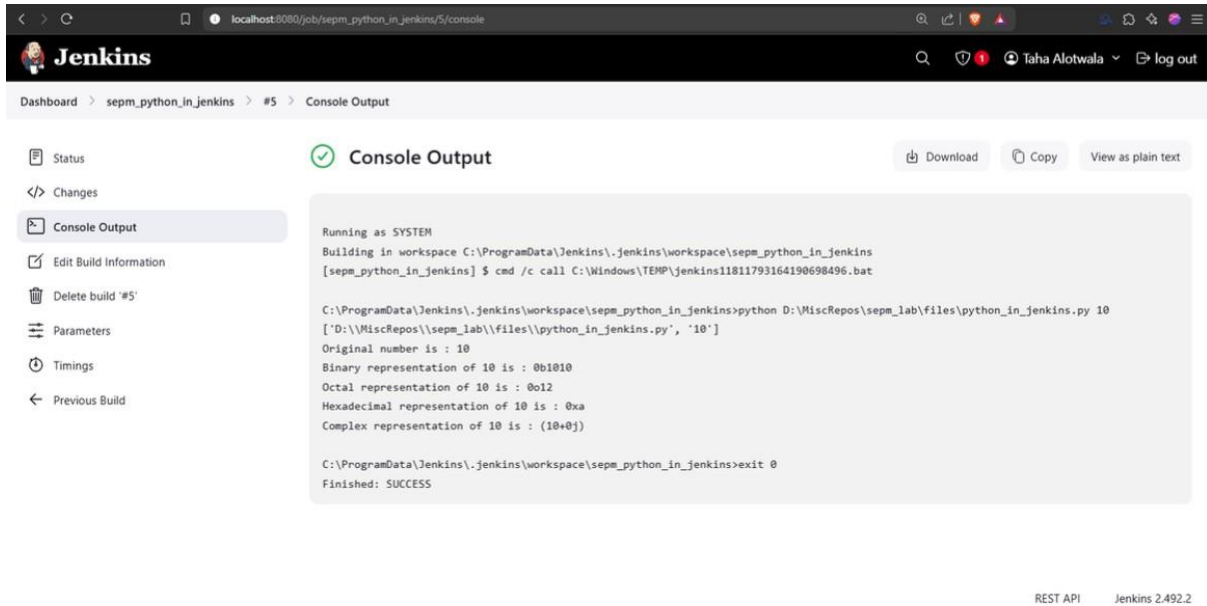
Build Steps

Post-build Actions

Connect to GitHub repository to automatically pull the latest code for your builds.



```
.. parameterized_builds.cmd
1 @echo off
2 echo "Hello your name is : %name%, and your city is %city%."
```



The screenshot shows the Jenkins web interface. The top navigation bar includes the Jenkins logo, a search icon, a shield icon, a user profile icon for 'Taha Alotwala', and a 'log out' button. The breadcrumb trail is 'Dashboard > sepm_python_in_jenkins > #5 > Console Output'. On the left sidebar, there are links for 'Status', 'Changes', 'Console Output' (which is selected), 'Edit Build Information', 'Delete build #5', 'Parameters', 'Timings', and 'Previous Build'. The main area is titled 'Console Output' with a green checkmark icon and buttons for 'Download', 'Copy', and 'View as plain text'. The console output text is as follows:

```
Running as SYSTEM
Building in workspace C:\ProgramData\Jenkins\jenkins\workspace\sepm_python_in_jenkins
[sepm_python_in_jenkins] $ cmd /c call C:\Windows\TEMP\jenkins11811793164190698496.bat

C:\ProgramData\Jenkins\jenkins\workspace\sepm_python_in_jenkins>python D:\MiscRepos\sepm_lab\files\python_in_jenkins.py 10
['D:\MiscRepos\sepm_lab\files\python_in_jenkins.py', '10']
Original number is : 10
Binary representation of 10 is : 0b1010
Octal representation of 10 is : 0o12
Hexadecimal representation of 10 is : 0xa
Complex representation of 10 is : (10+0j)

C:\ProgramData\Jenkins\jenkins\workspace\sepm_python_in_jenkins>exit 0
Finished: SUCCESS
```

At the bottom right of the console output area, it says 'REST API' and 'Jenkins 2.492.2'.

CONCLUSION: Hence, we have successfully created a VPC Service for launching the Instances.