

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:


 **Jenkins**


Dashboard > All > New Item


New Item


Enter an item name

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 folder is a real container for items.


OK


localhost:8080/job/sepm_freestyle_app_demo/configure


Dashboard > sepm_freestyle_app_demo > Configuration


Configure


 General

 Source Code Management

 Triggers

 Environment

 Build Steps

 Post-build Actions

☐ 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

☐ With Ant ?

Build Steps

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

Add build step ^

Filter

Execute Windows batch command

Execute shell

Invoke Ant

Invoke Gradle script

Invoke top-level Maven targets

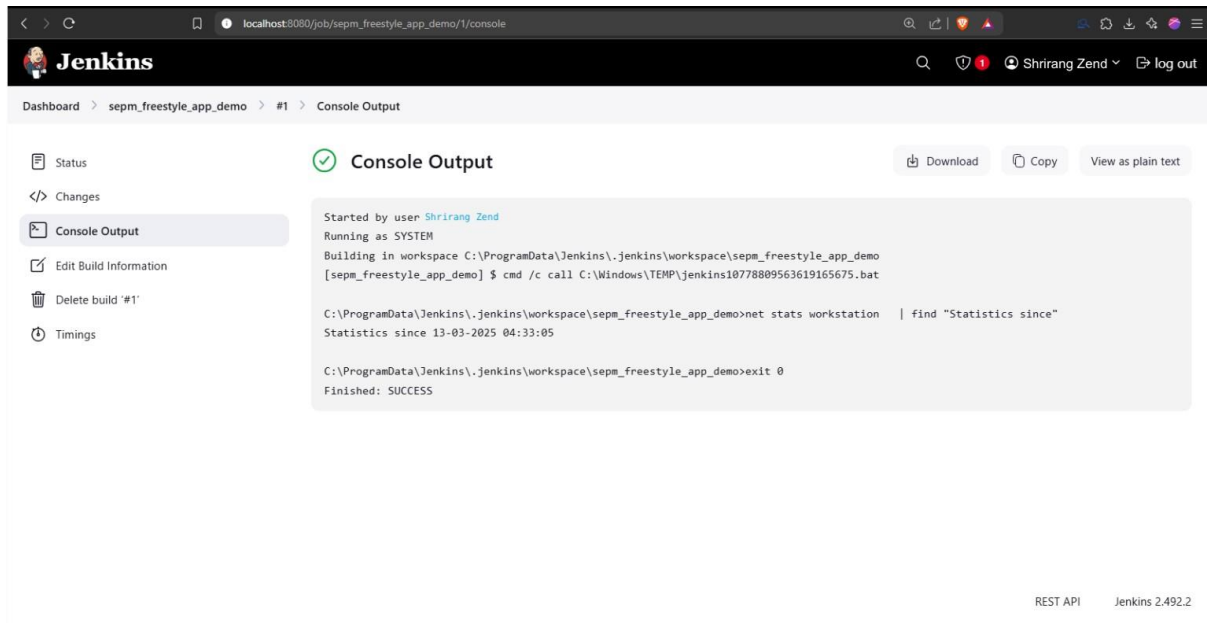
Run with timeout

Set build status to "pending" on GitHub commit

REST API Jenkins 2.492.2

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 categories: General, Source Code Management, Triggers, Environment (selected), Build Steps, and Post-build Actions. The main content area is titled 'Configuration' and includes several checkboxes for general settings: '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'. Below these is the 'Build Steps' section, which describes automating the build process. A single build step is configured as 'Execute Windows batch command', with the command 'net stats workstation | find "Statistics since"' entered in the text area. An 'Advanced' dropdown is visible below the command field. At the bottom, there are 'Save' and 'Apply' buttons.

The screenshot shows the Jenkins Build History page for the same job. The top header includes the Jenkins logo and the job name 'sepm_freestyle_app_demo' followed by '#1'. The left sidebar lists options: 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)'. It includes buttons for 'Add description' and 'Keep this build forever'. The build details show it was 'Started by user Shrirang Zend' and provides timing information: 'Started 4.9 sec ago' and '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, a code icon indicates 'No changes.'. The bottom right corner shows 'REST API' and 'Jenkins 2.492.2'.



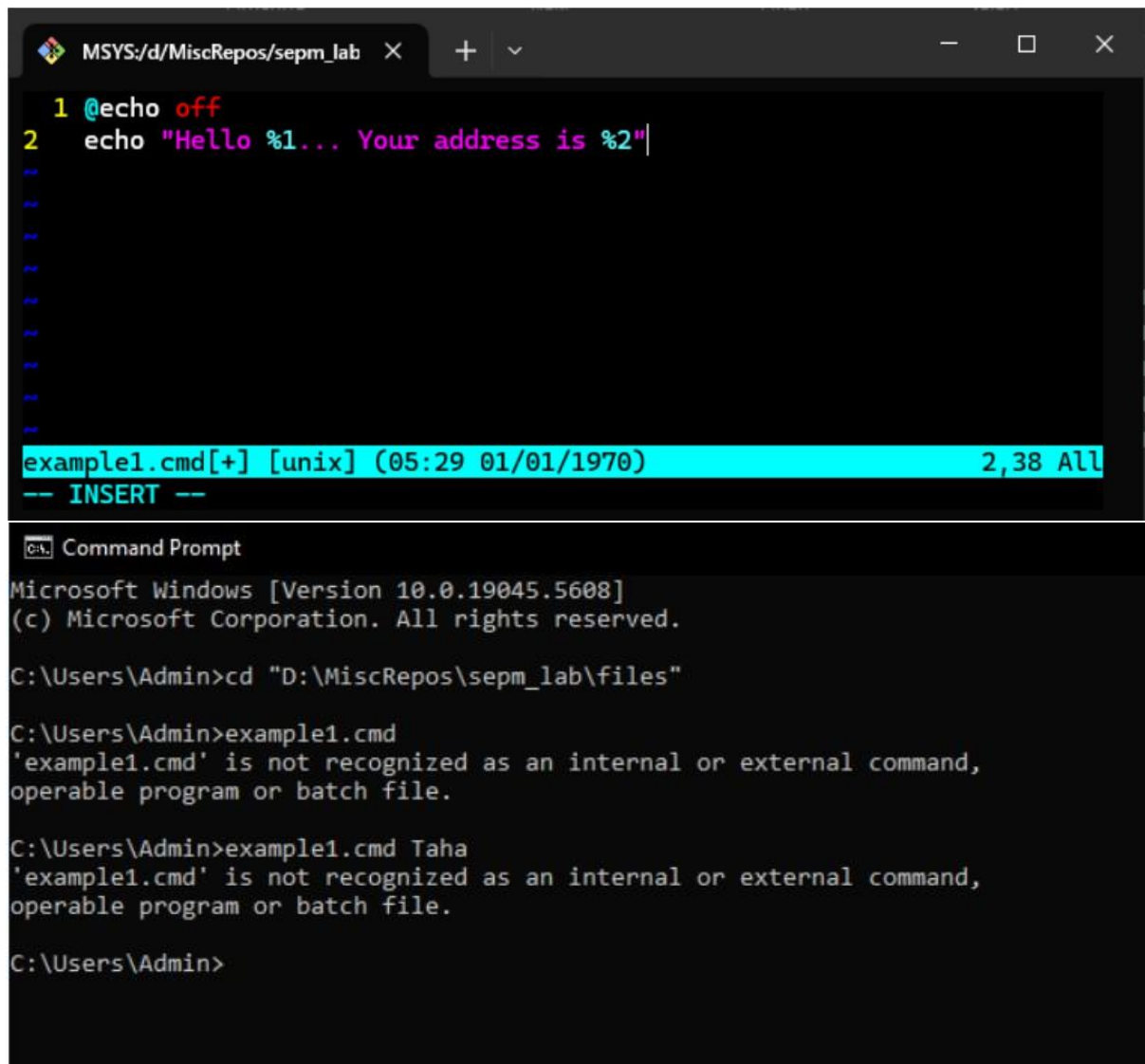
The screenshot shows the Jenkins web interface. The breadcrumb navigation is Dashboard > sepm_freestyle_app_demo > #1 > Console Output. On the left sidebar, the 'Console Output' tab is selected. The main area displays the console output for build #1, which was started by user 'Shrirang Zend' and is running as 'SYSTEM'. The output shows the build directory, a command to call a batch file, and the execution of 'net stats workstation' which finds 'Statistics since' information. The build finished successfully.

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

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

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

REST API Jenkins 2.492.2



The screenshot shows a Windows Command Prompt window. The title bar indicates the path 'MSYS\>d\MiscRepos\sepm_lab'. The prompt shows the execution of a batch file 'example1.cmd' with the following commands:

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

The output of the batch file is displayed in a cyan box:

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

Below the Command Prompt, a screenshot of a Windows Command Prompt window is shown. The title bar indicates the path 'C:\>'. The prompt shows the execution of the batch file 'example1.cmd' with the following commands:

```
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>
```

Dashboard > sepm_supply_params > Configuration

☐ With Ant ?

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](#)

D:\MiscRepos\sepm_lab\files\example1.cmd Taha Mazgaon

Advanced ▾

Add build step ▾

Post-build Actions

Save Apply

Dashboard > sepm_supply_params > #1

Status ✓ #1 (Apr 2, 2025, 12:01:38 PM) Add description Keep this build forever

- Changes
- Console Output
- Edit Build Information
- Delete build '#1'
- Timings

This run spent:

- 6 ms waiting;
- 0.29 sec build duration;
- 0.29 sec total from scheduled to completion.

No changes.

Started 55 sec ago
Took 0.29 sec

```
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) v ±5
```

Dashboard > All > New Item

New Item

Enter an item name

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

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 > All > New Item

New Item

Enter an item name

sepm_parameterized_build

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.

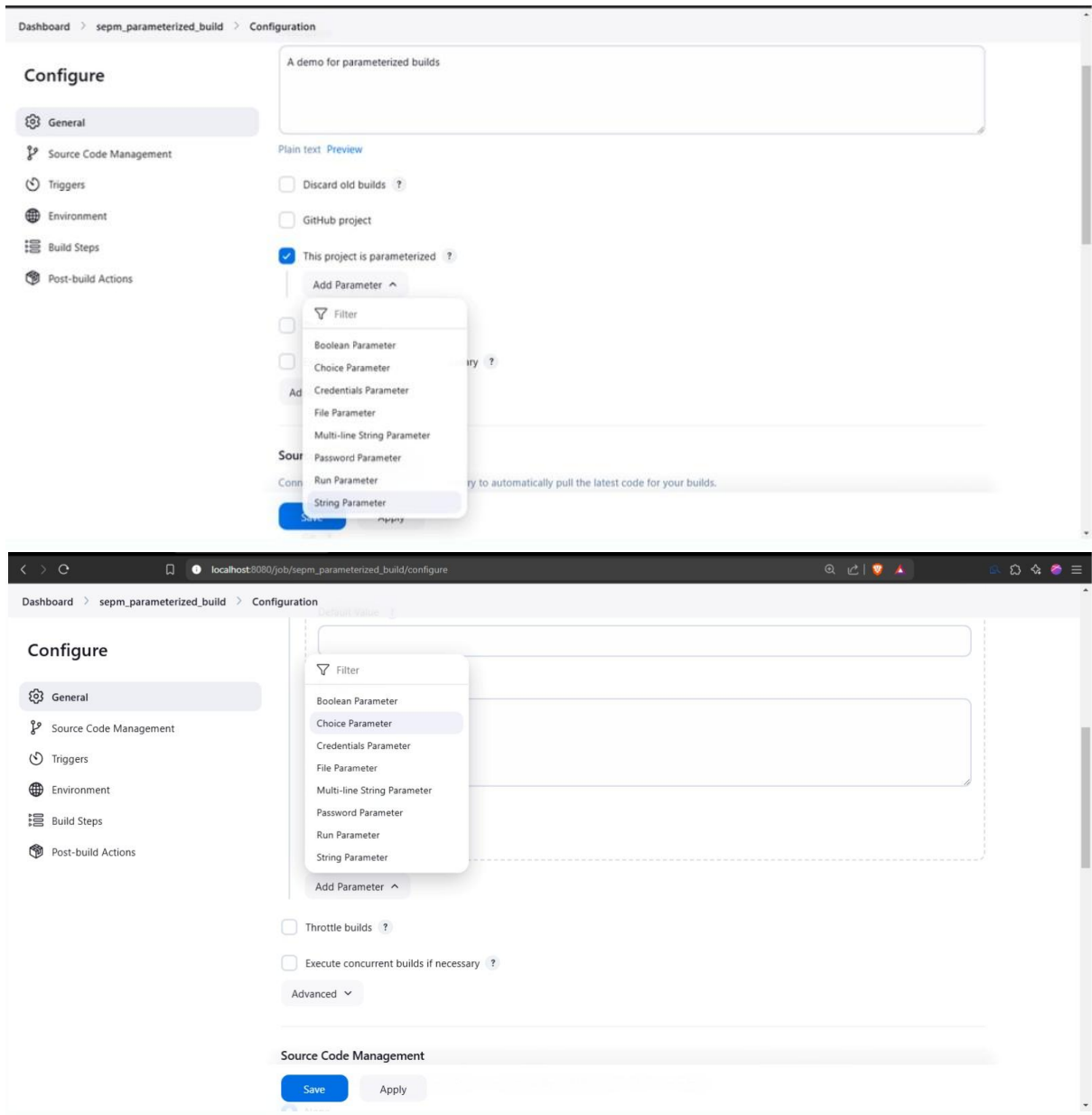
Multi-configuration project

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 folder is a container for items.

OK



The screenshot shows the Jenkins Configuration page for a job named 'sepm_parameterized_build'. The left sidebar contains a 'Configure' section with a 'Build Steps' tab selected. The main area is titled 'Build Steps' and contains a single step named 'Execute Windows batch command'. The command field contains the text 'D:\MiscRepos\sepm_lab\files\parameterized_builds.cmd'. Below the command field is an 'Advanced' dropdown menu. At the bottom of the page, there are 'Save' and 'Apply' buttons.

Dashboard > sepm_parameterized_build > 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

D:\MiscRepos\sepm_lab\files\parameterized_builds.cmd

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

The screenshot shows a Windows command prompt window with a file named 'parameterized_builds.cmd'. The script contains the following commands:

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

The screenshot shows the Jenkins Console Output page for a job named 'sepm_python_in_jenkins'. The left sidebar contains a 'Console Output' tab selected. The main area displays the console output, which includes the following text:

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

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

REST API Jenkins 2.492.2

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