Experiment No-5

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

Programming in Jenkins:

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.

Our first job will execute the shell commands. The freestyle project provides enough options and features to build the complex jobs that you will need in your projects.

Maven

- Maven is a popular **Java-based build automation tool**.
- It uses **XML configuration (pom.xml)** to define the project's build lifecycle, dependencies, and plugins.
- Preferred for large, structured enterprise applications.

Gradle

- Gradle is a flexible, powerful build automation tool that uses a Groovy/Kotlin DSL.
- Faster than Maven due to its incremental build support.
- Ideal for both Java and Android projects.

Ant

- Apache Ant is an older build tool that provides fine-grained control using XML files (build.xml).
- It lacks dependency management by default, which Maven and Gradle handle better.

Jenkins Pipeline Basics

A **Jenkins pipeline** is a suite of plugins that supports integrating and implementing continuous delivery pipelines. There are two types of pipelines:

- **Declarative Pipeline**: Simplified, structured format.
- **Scripted Pipeline**: More powerful, written in Groovy.

Pipeline scripts define how Jenkins should build, test, and deploy an application in a step-by-step fashion.

Pipeline Stages

A typical Jenkins pipeline includes the following stages:

- 1. **Checkout** Pull code from version control (e.g., Git)
- 2. **Build** Use Maven/Gradle/Ant to compile the project
- 3. **Test** Run unit/integration tests
- 4. **Package** Create a deployable artifact (.war/.jar)
- 5. **Deploy** Deploy the application to a server (like Tomcat)

5. Jenkins Integration with Build Tools

Using Maven in Jenkins

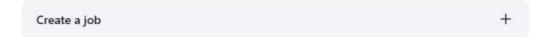
- Install the Maven plugin in Jenkins
- Add Maven in Global Tool Configuration

Example 1

Example 1.1: Deploying a freestyle app in Jenkins Creating a

job:

Start building your software project

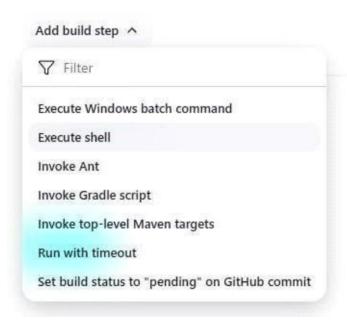


Naming the job and setting it as freestyle:



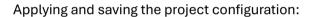
Selecting build type as "Execute shell":

Build Steps



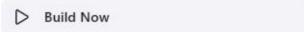
Entering a simple command for the shell execution:



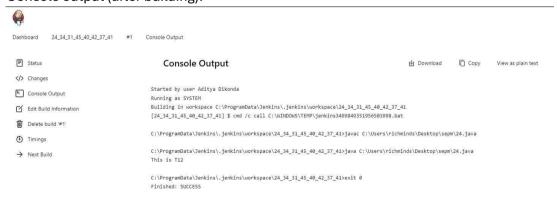




Building the project:



Console output (after building):



REST API Jenkins 2.503



Example 1.2: Taking parameters through files Contents of

script example1.cmd:

Executing script example1.cmd on the terminal:

```
Microsoft Windows [Version 10.0.22621.3296]
(c) Microsoft Corporation. All rights reserved.

C:\Users\AI&DS 202>Microsoft Windows [Version 10.0.22631.3155) (c) Microsoft Corporation. All rights reserved.

'Microsoft' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\AI&DS 202>C:\Admin\Academics\TSEC\Start3\SEPM>example1.cmd

The system cannot find the path specified.

C:\Users\AI&DS 202>"Hello... Your address is "

'"Hello... Your address is "' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\AI&DS 202>C:\Admin\Academics\TSEC\Start3\SEPM>example1.cad Tanishq

The system cannot find the path specified.

C:\Users\AI&DS 202>"Hello Tanishq... Your address is "
'"Hello Tanishq... Your address is "' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\AI&DS 202>C:\Admin\Academics\TSEC\Start3\SEPM>example1.cmd Tanishq Girgaon "Helle Tanishq... Your address is Girgaon"

The system cannot find the path specified.
```

Modifying the Jenkins project to execute the script while supplying required parameters:



Console output after building the modified project:



program under Jenkins

Creating a simple Java program:

Compiling and running the program on the terminal:

```
C:\Users\richminds\Desktop\sepm>javac 24.java
C:\Users\richminds\Desktop\sepm>java 24.java
This is T12
C:\Users\richminds\Desktop\sepm>
```

Creating a new freestyle project:



Configure new project:

Command

See the list of available environment variables

javac C:\Users\richminds\Desktop\sepm\24.java
java C:\Users\richminds\Desktop\sepm\24.java

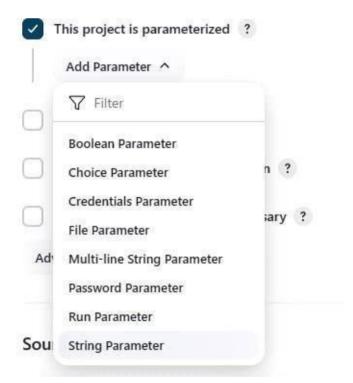
Console output after building:



Example 3

<u>Example 3.1: Parameterise build</u> Creating a new freestyle project:

Enabling parameterisation and adding a String parameter:



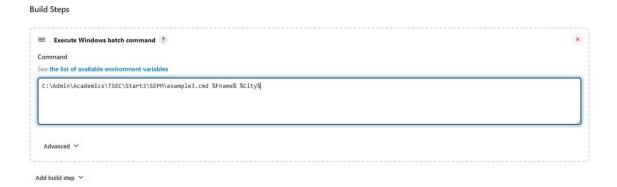
Configuring the string parameter as Fname:



Adding a choice parameter and configuring it as City with the following choices:



Configuring build steps:



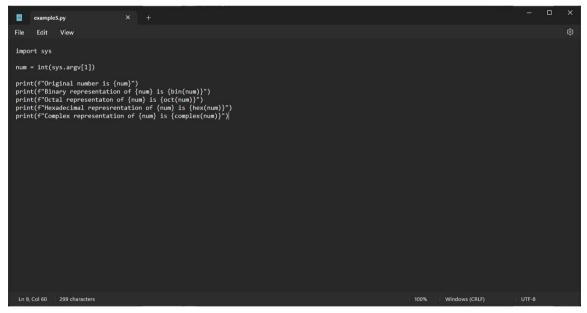
Entering parameters for build:



Console output after building:



Example 5



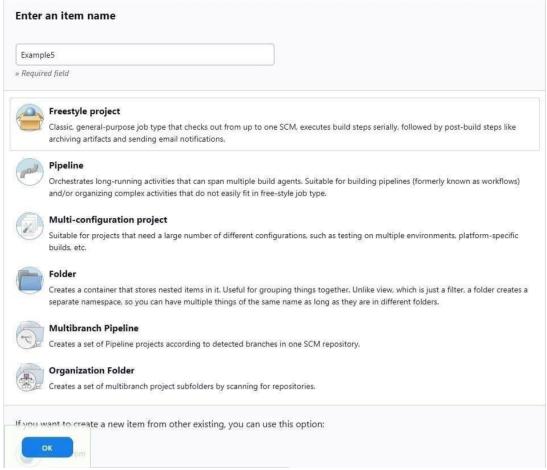
Example 5.1: Running a Python program Creating a

simple Python script:

Running the Python script on the terminal:

```
C:\Users\richminds\Desktop\sepm>python 24.py 10
Number is 10
C:\Users\richminds\Desktop\sepm>_
```

Creating a new freestyle project:



Parameterising the project with a string parameter as follows:



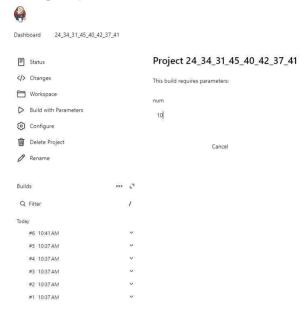
Configuring the build steps:

Command

See the list of available environment variables

python C:\Users\richminds\Desktop\sepm\24.py

Setting the parameter for the build:



Conclusion: Thus, we have successfully studied Continuous Integration and installed, configured, and understood programming with Jenkins.