ROUNAK KATIYAR
Batch: T - 13
Roll no. 47

EXPERIMENT 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.

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"'
            }
        }
    }
}
```

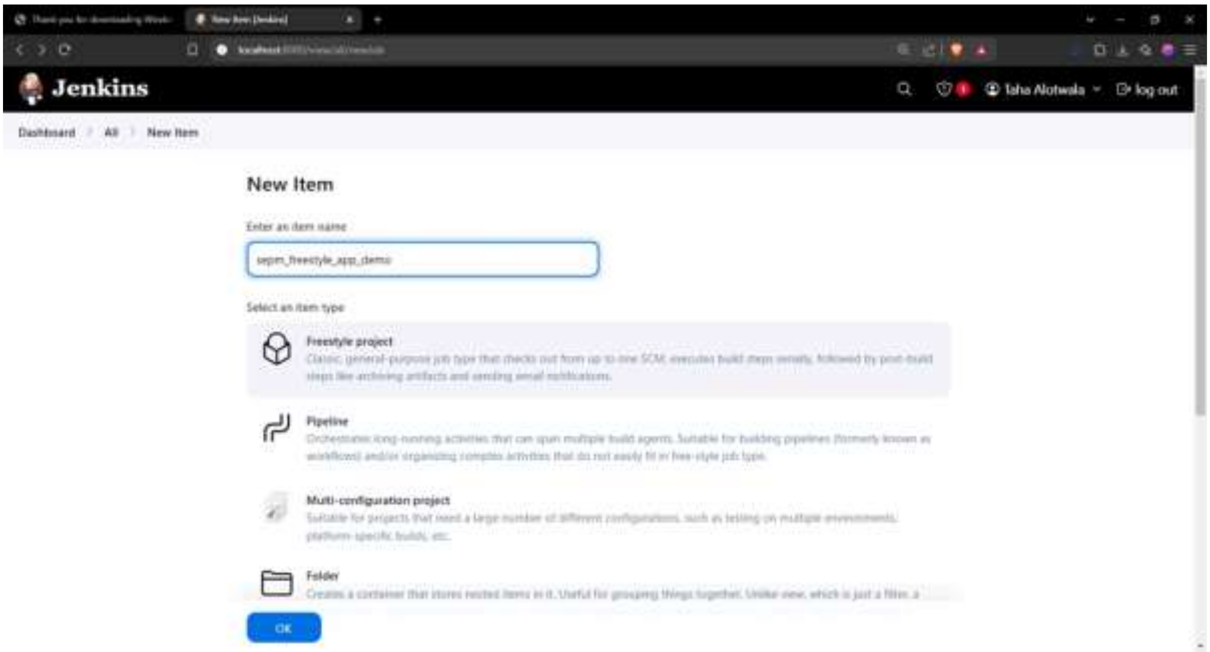
This pipeline defines three stages: Build, Test, and Deploy, each with its own commands.

Benefits of Jenkins in Programming:

- Automates repetitive tasks (builds, tests, deployments).
- Detects bugs early with continuous testing.
- Speeds up the development lifecycle.

DEMONSTRATION:

1. Freestyle App:



New Item

Enter an item name

sepm_freestyle_app_demo

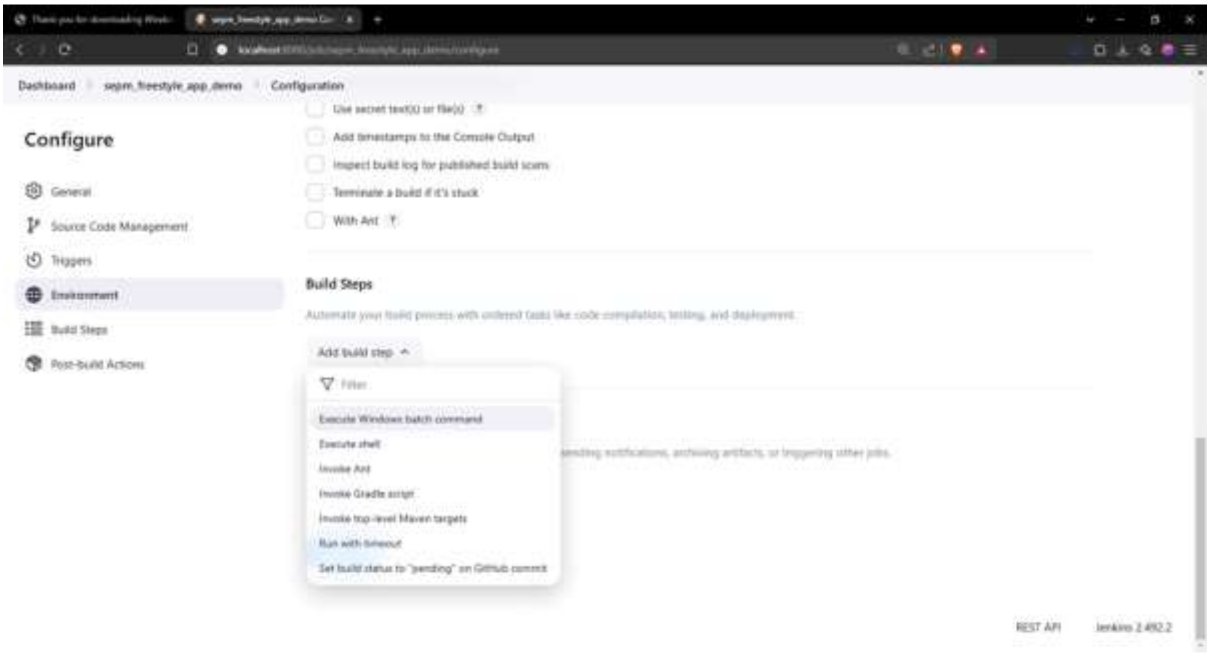
Select an item type

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



☐ Use secret text(s) or file(s)

☐ Add timestamps to the Console Output

☐ Inspect build log for published build scans

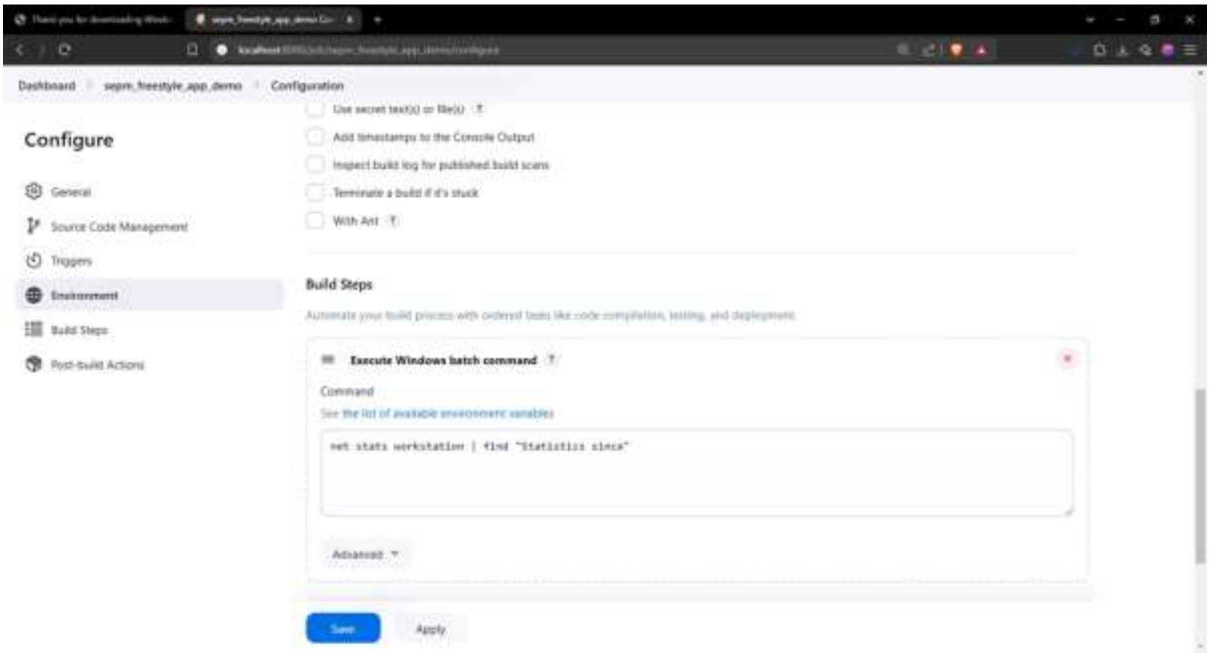
☐ Terminate a build if it's stuck

☐ With Ant

Build Steps

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

...sending notifications, archiving artifacts, or triggering other jobs.



☐ 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

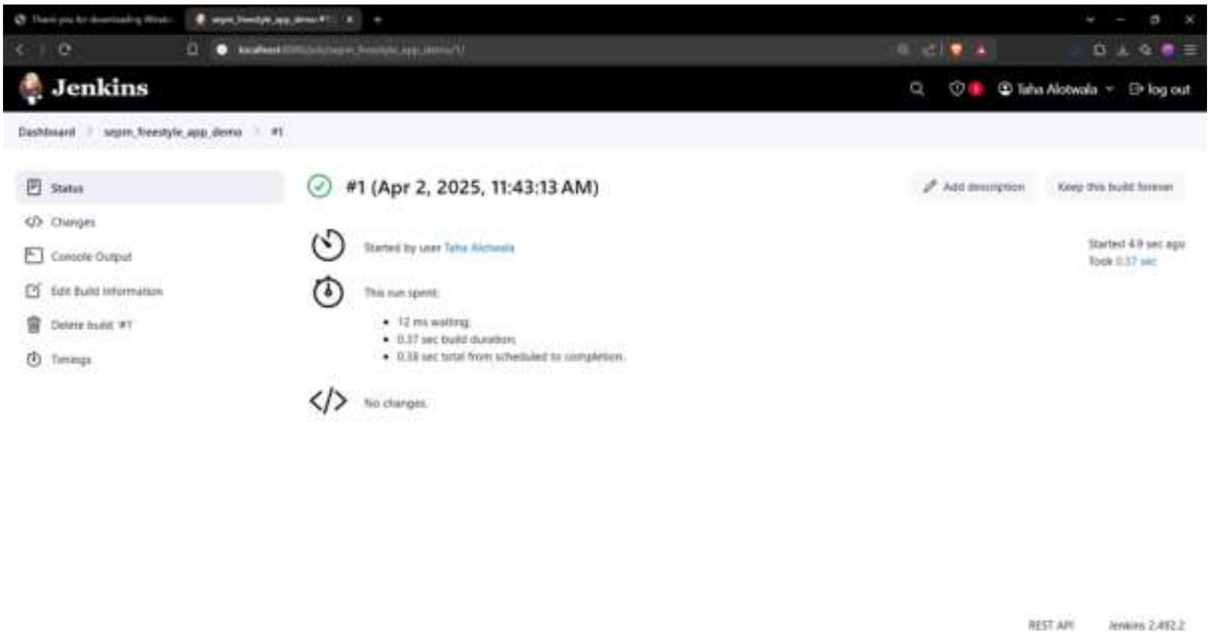
Automate your build process with ordered tasks like code compilation, testing, and deployment.

Execute Windows batch command

Command

See the list of available environment variables

```
net stats workstation | find "Statistics since"
```



#1 (Apr 2, 2025, 11:43:13 AM)

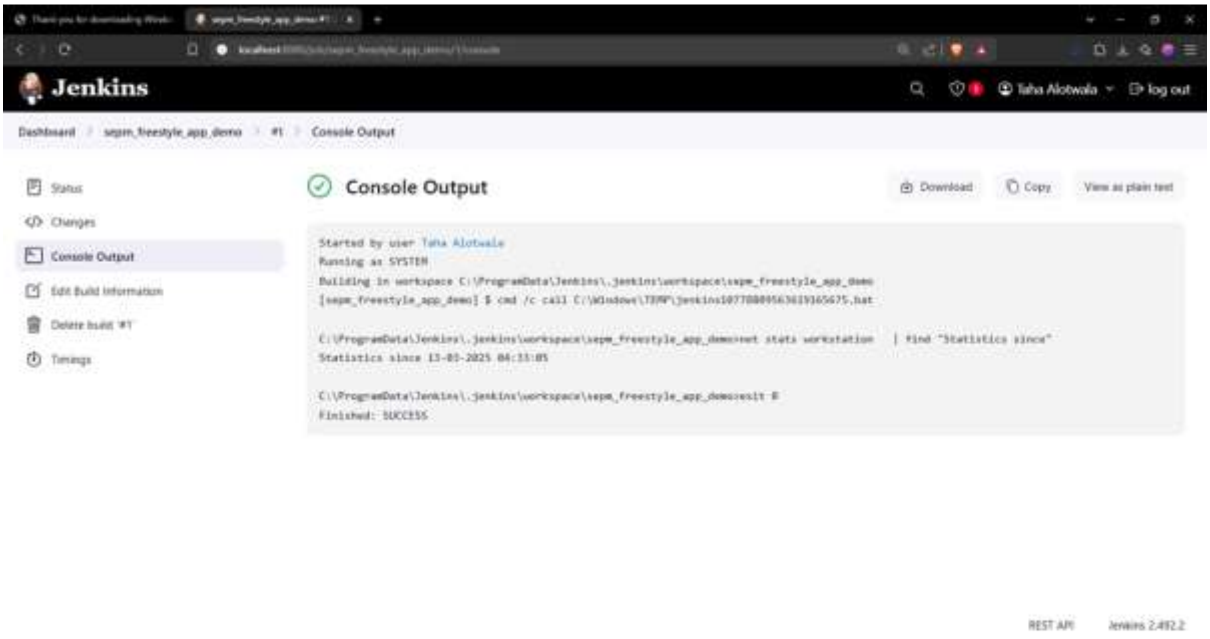
Started by user Taha Alotwala

Started 4.9 sec ago
Took 0.37 sec

This run spent:

- 12 ms waiting;
- 0.37 sec build duration;
- 0.38 sec total from scheduled to completion.

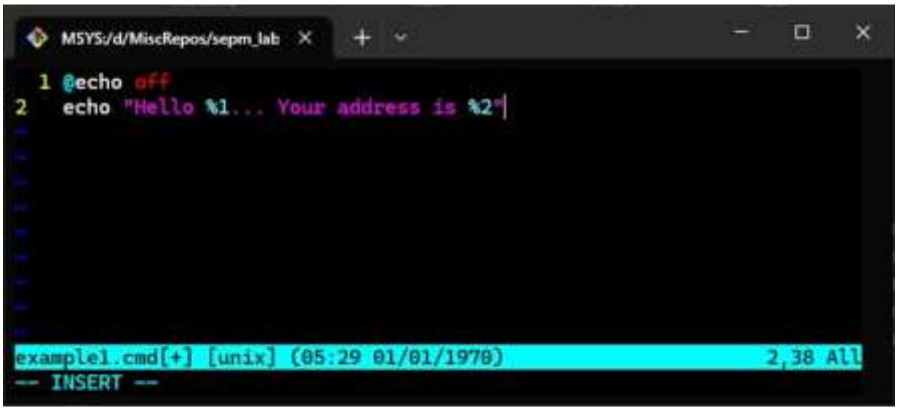
No changes.



Console Output

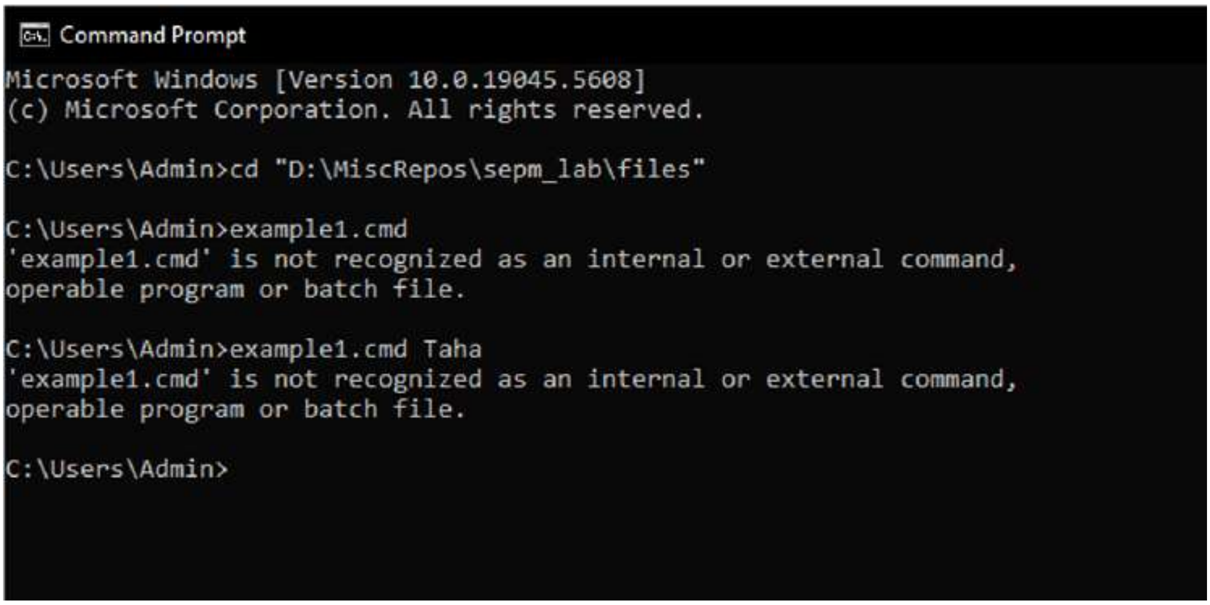
```
Started by user Taha Alotwala
Running as SYSTEM
Building in workspace C:\ProgramData\Jenkins\.jenkins\workspace\sepm_freestyle_app_demo
[sepm_freestyle_app_demo] $ cmd /c call C:\Windows\TEMP\jenkins107708009563619165675.bat

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
```

2. Taking Parameters:
   Script:



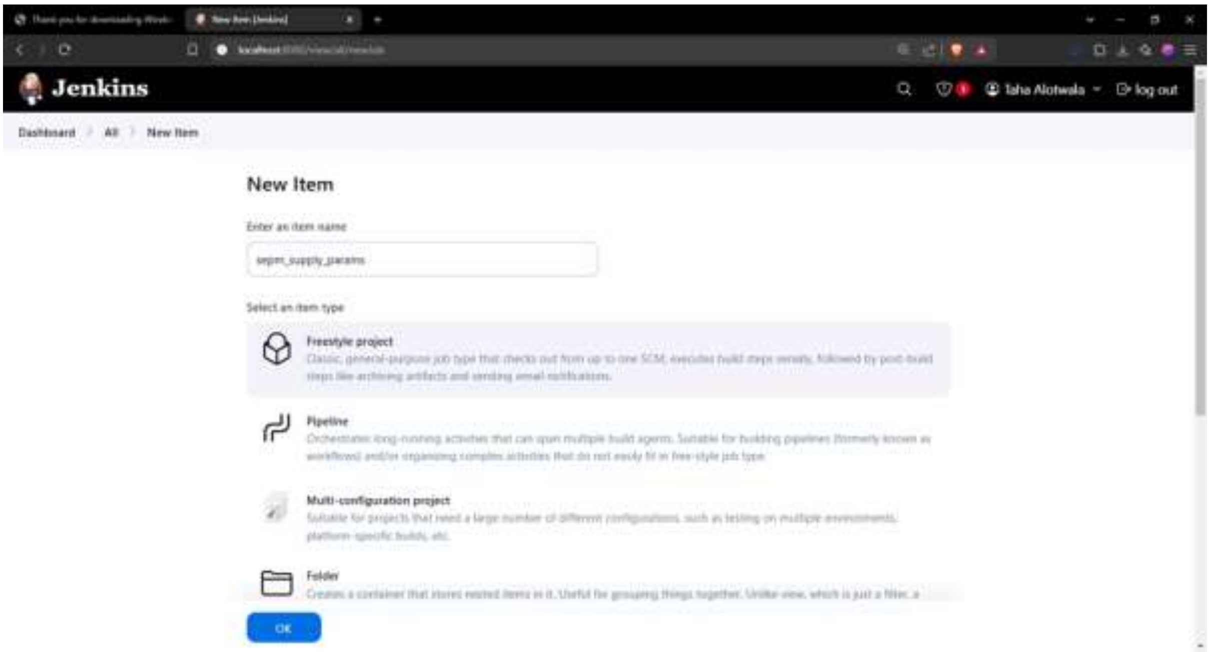
```
@echo off
echo "Hello %1... Your address is %2"
~
~
~
~
~
~
~
example1.cmd[+] [unix] (05:29 01/01/1970)                    2,38 All
-- INSERT --
```

Executing the script (unsuccessful):



```
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>
```

Configure Jenkins:



New Item

Enter an item name

sepm_supply_params

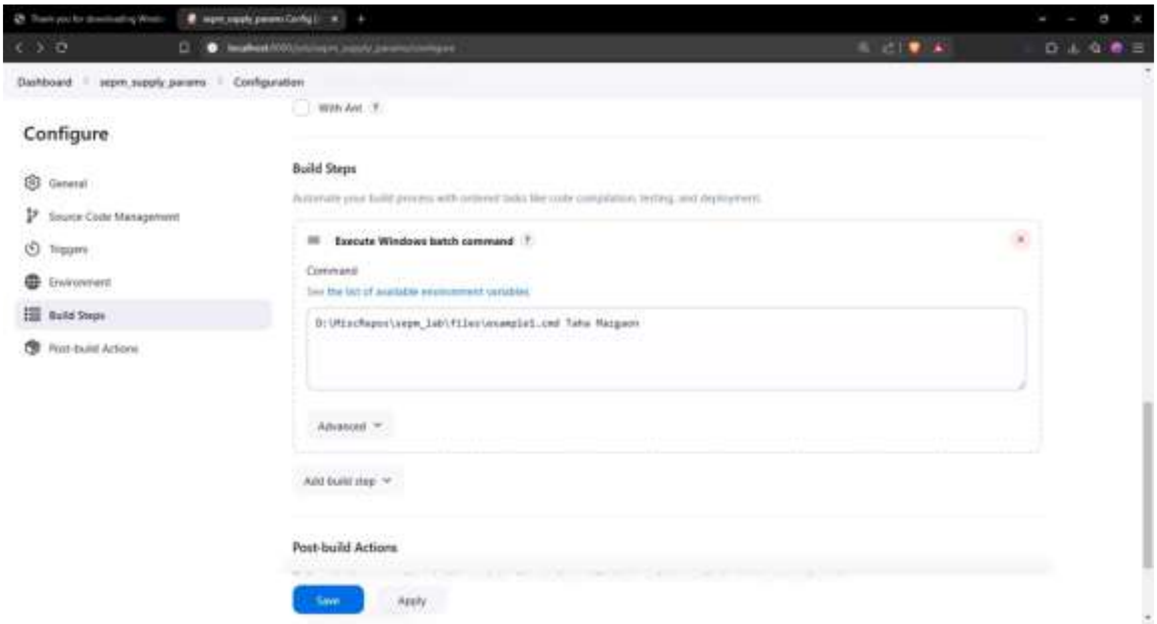
Select an item type

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



Build Steps

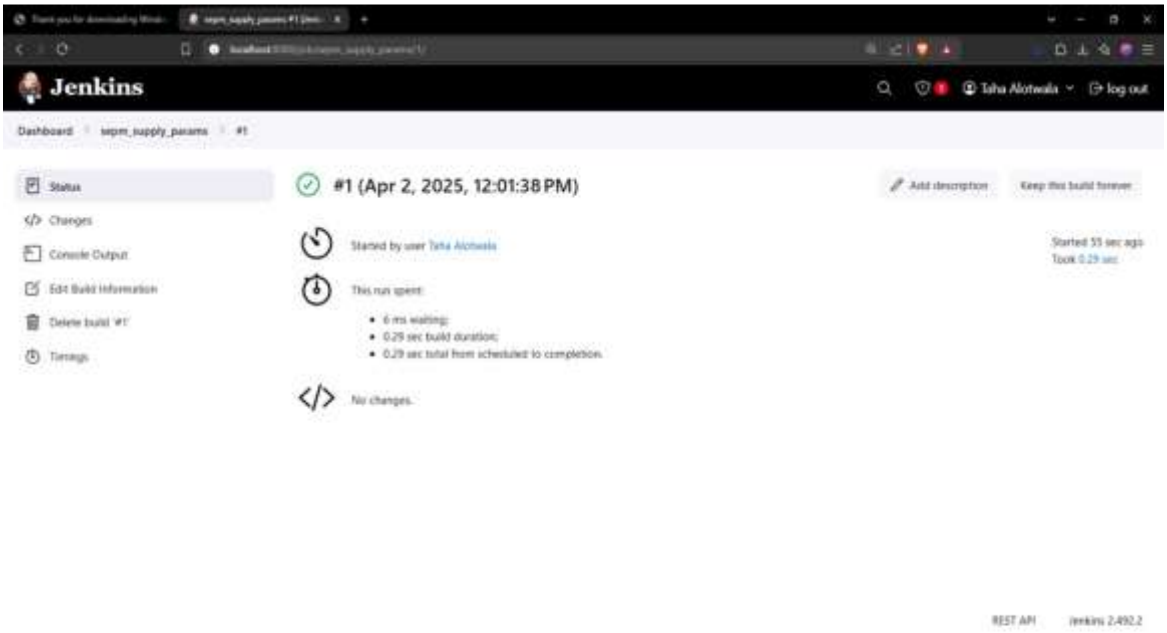
Automate your build process with ordered tasks like code compilation, testing, and deployment.

Execute Windows batch command

Command

See the list of available environment variables

```
D:\MiscRepos\sepm_lab\files\example1.cmd Taha Margaon
```



#1 (Apr 2, 2025, 12:01:38 PM)

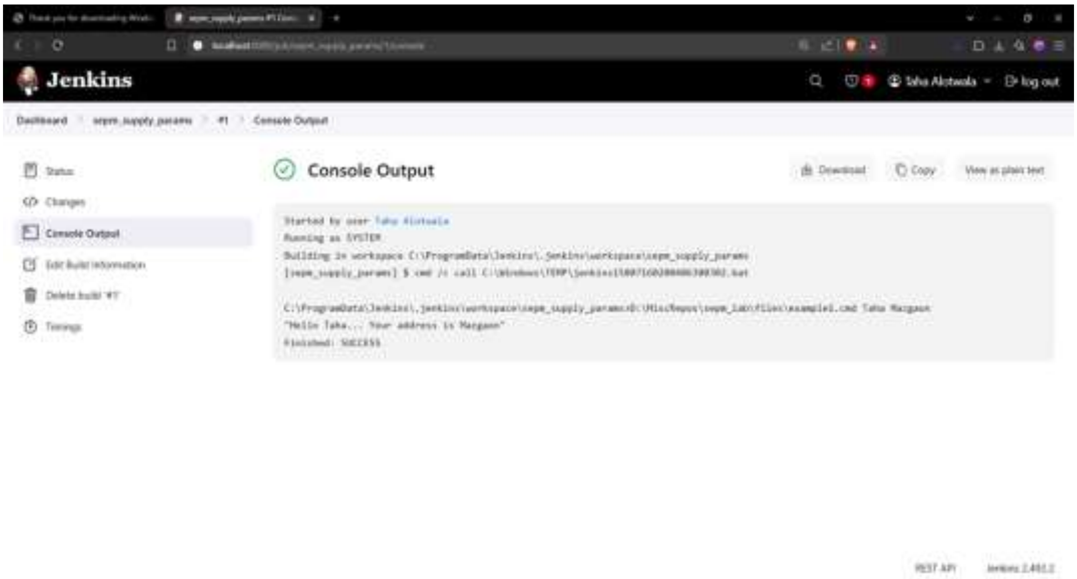
Started by user Taha Alotwala

Started 55 sec ago
Took 0.29 sec

This run spent:

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

No changes.

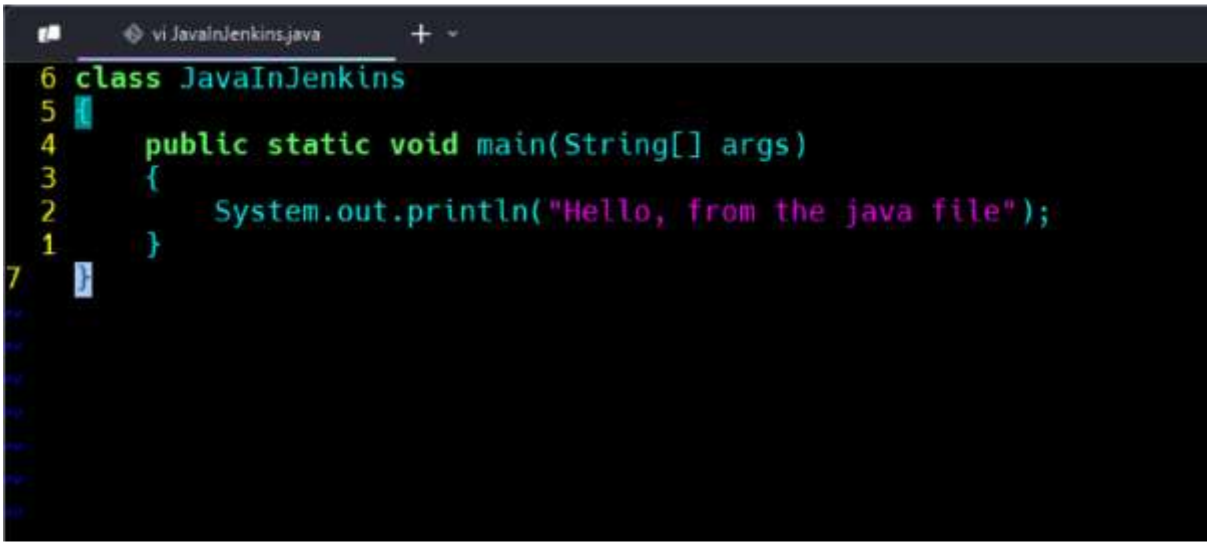


Console Output

```
Started by user Taha Alotwala
Running as SYSTEM
Building in workspace C:\ProgramData\Jenkins\.jenkins\workspace\sepm_supply_params
[sepm_supply_params] $ cmd /c call C:\Windows\TEMP\jenkins[illegible].bat

C:\ProgramData\Jenkins\.jenkins\workspace\sepm_supply_params>D:\MiscRepos\sepm_lab\files\example1.cmd Taha Margaon
"Hello Taha... Your address is Margaon"
Finished: SUCCESS
```

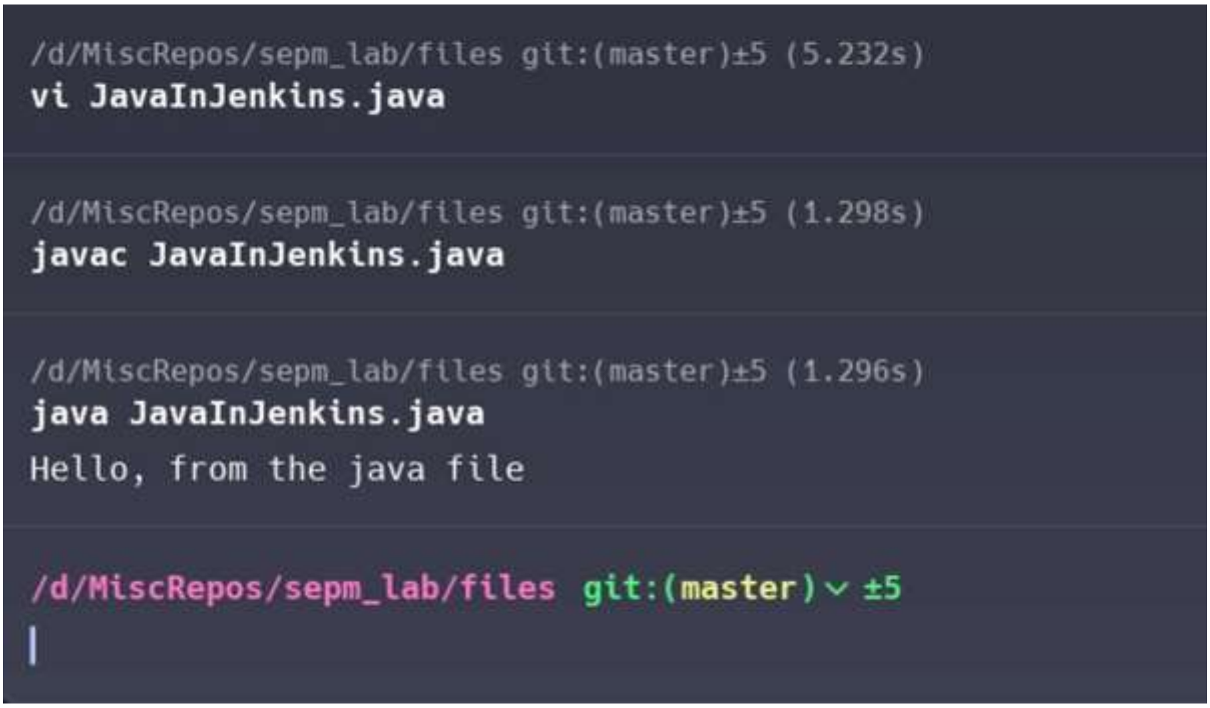
3. Running a Java code:

CODE:



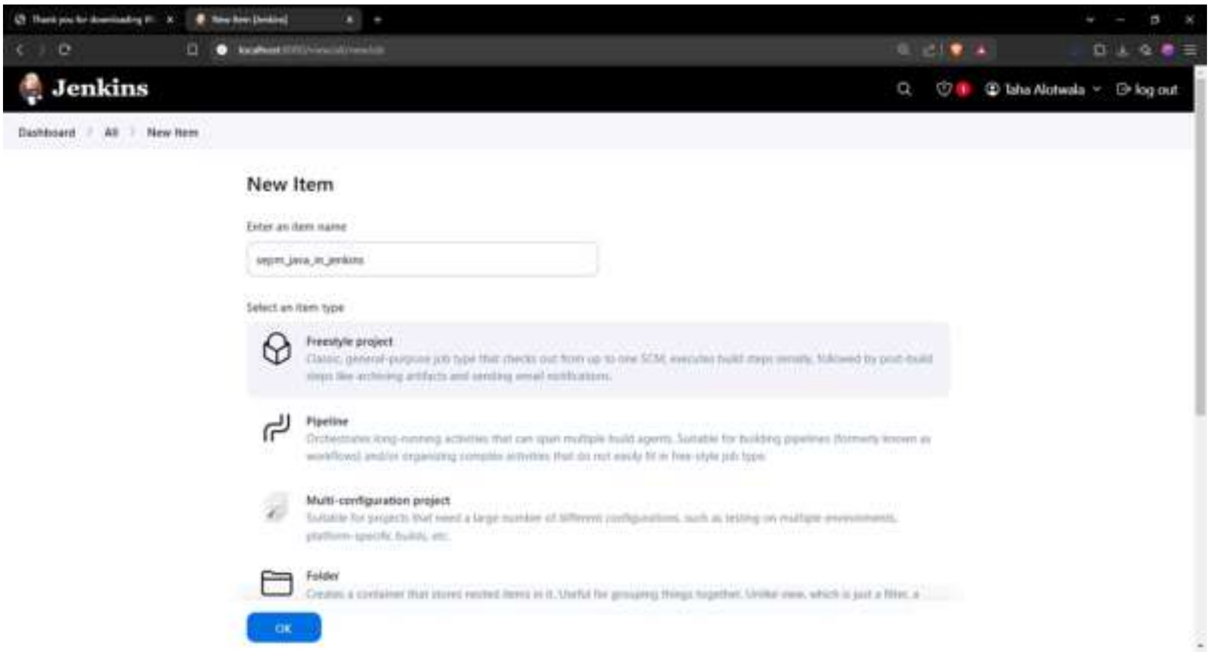
```
class JavaInJenkins
{
    public static void main(String[] args)
    {
        System.out.println("Hello, from the java file");
    }
}
```

Running the file through the terminal:



```
/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
```

Configure Jenkins:



Jenkins — Taha Alotwala — log out

Dashboard > All > New Item

New Item

Enter an item name

sepm_java_in_jenkins

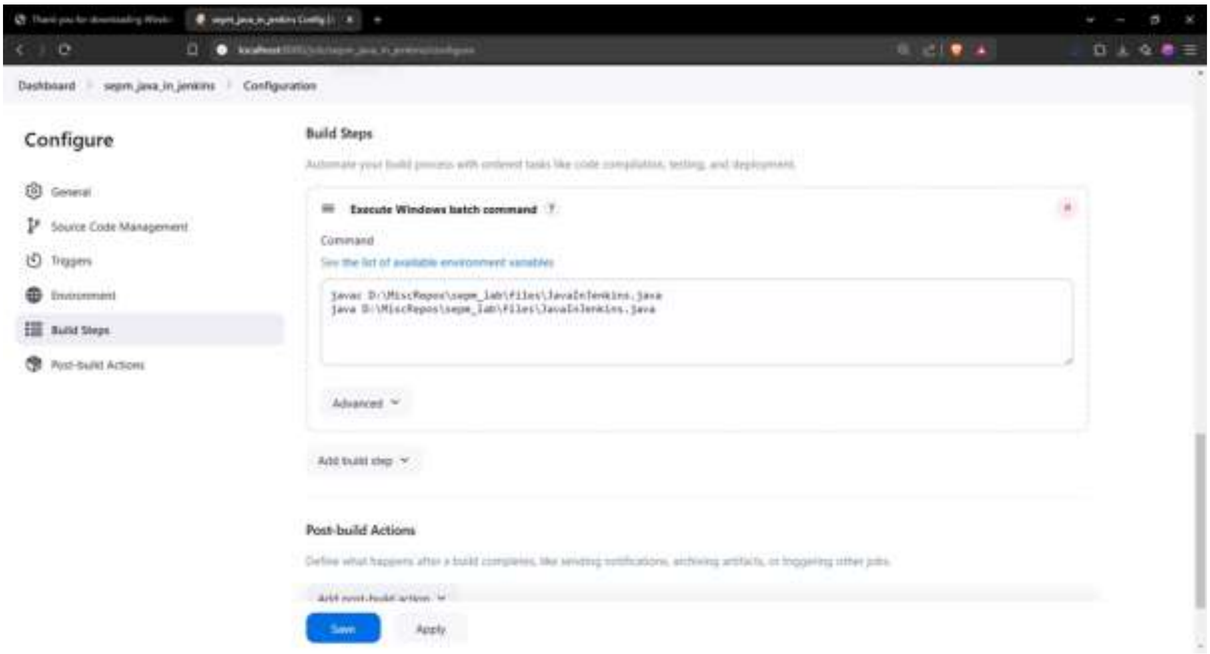
Select an item type

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



Dashboard > sepm_java_in_jenkins > Configuration

Build Steps

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

Execute Windows batch command

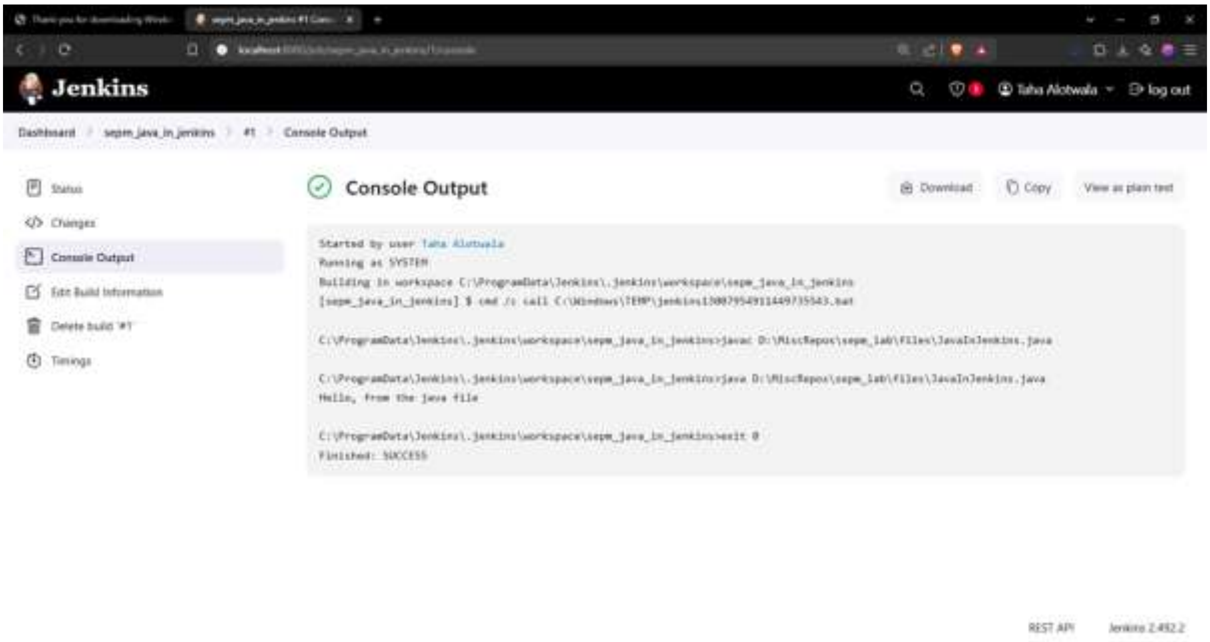
Command

See the list of available environment variables

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

Post-build Actions

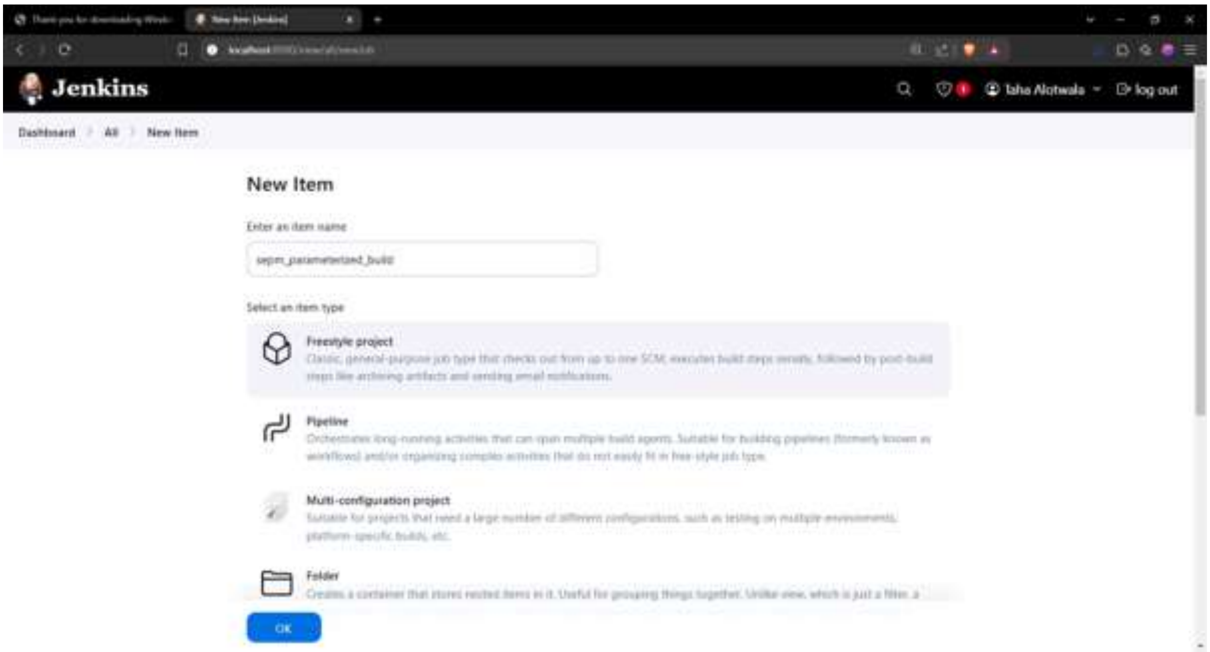
Define what happens after a build completes, like sending notifications, archiving artifacts, or triggering other jobs.



Console Output

```
Started by user Taha Alotwala
Running as SYSTEM
Building in workspace C:\ProgramData\Jenkins\.jenkins\workspace\sepm_java_in_jenkins
[sepm_java_in_jenkins] $ cmd /c call C:\Windows\TEMP\jenkins13007954911449735543.bat

C:\ProgramData\Jenkins\.jenkins\workspace\sepm_java_in_jenkins>javac D:\MiscRepos\sepm_lab\Files\JavaInJenkins.java

C:\ProgramData\Jenkins\.jenkins\workspace\sepm_java_in_jenkins>java D:\MiscRepos\sepm_lab\Files\JavaInJenkins.java
Hello, from the java file

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

4. Parameterized Builds



New Item

Enter an item name: sepm_parameterized_build

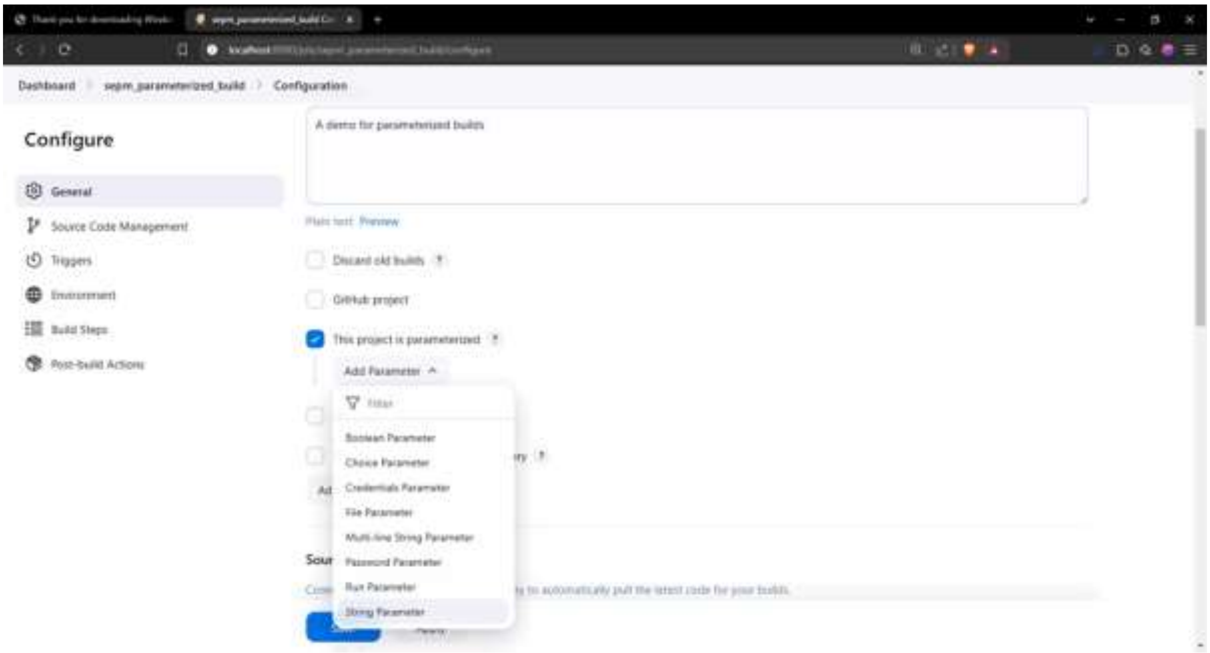
Select an item type

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

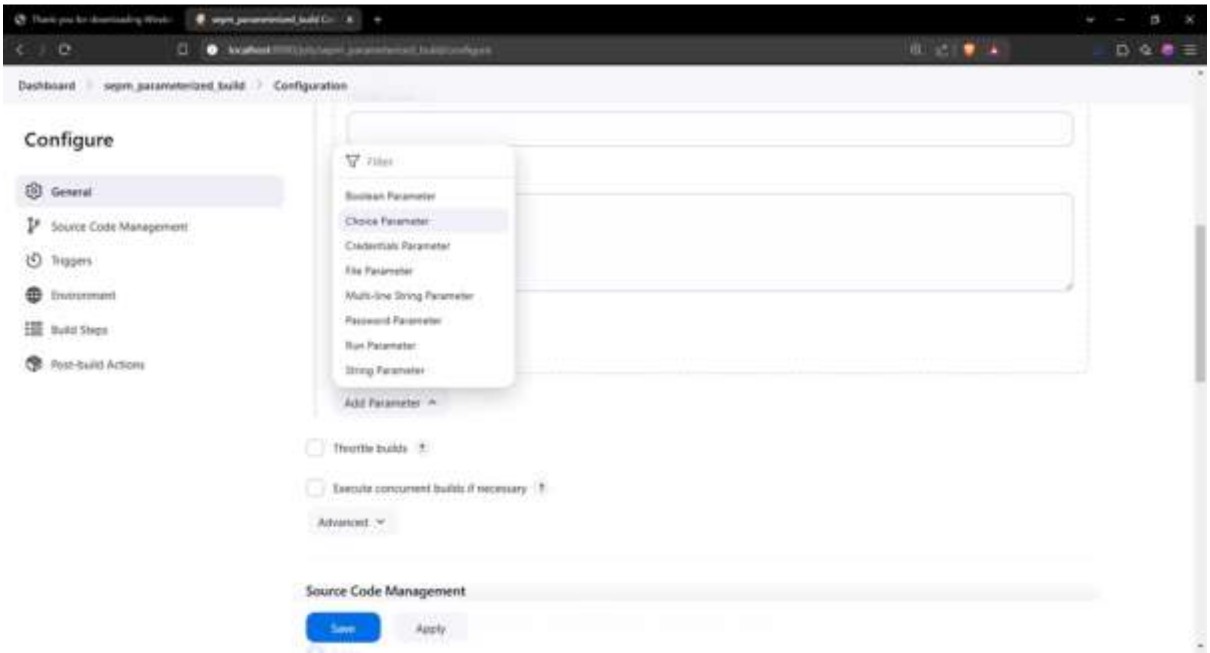


A demo for parameterized builds

□ Discard old builds

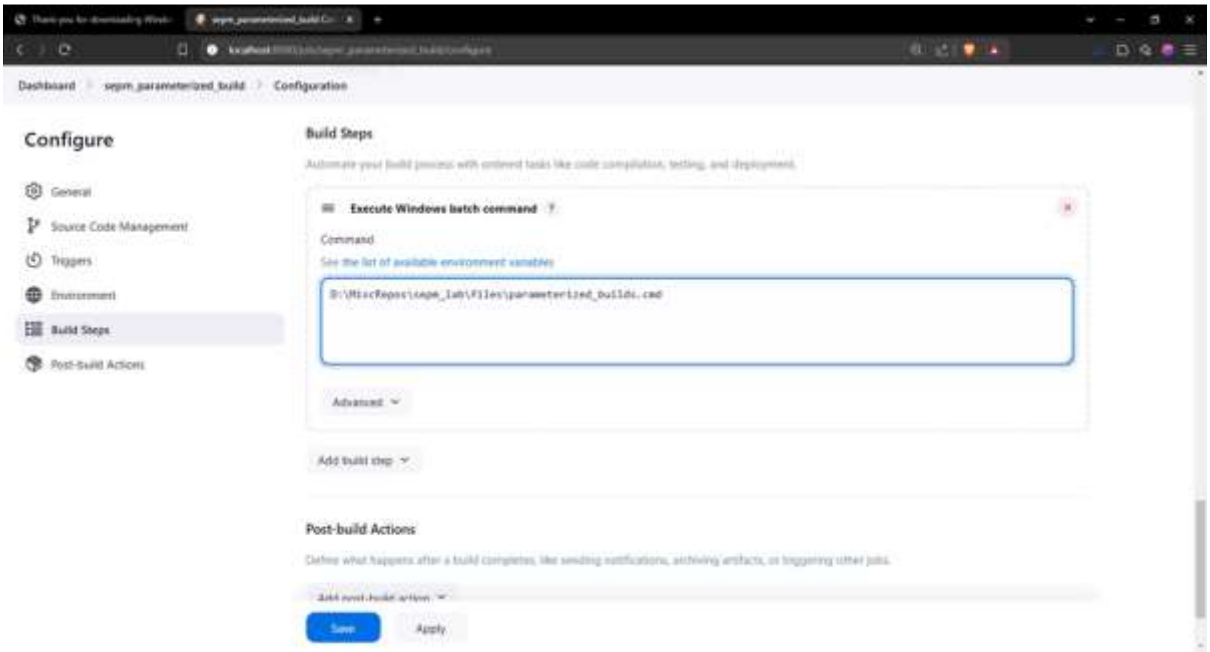
□ GitHub project

☑ This project is parameterized



□ Throttle builds

□ Execute concurrent builds if necessary



Build Steps

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

Execute Windows batch command

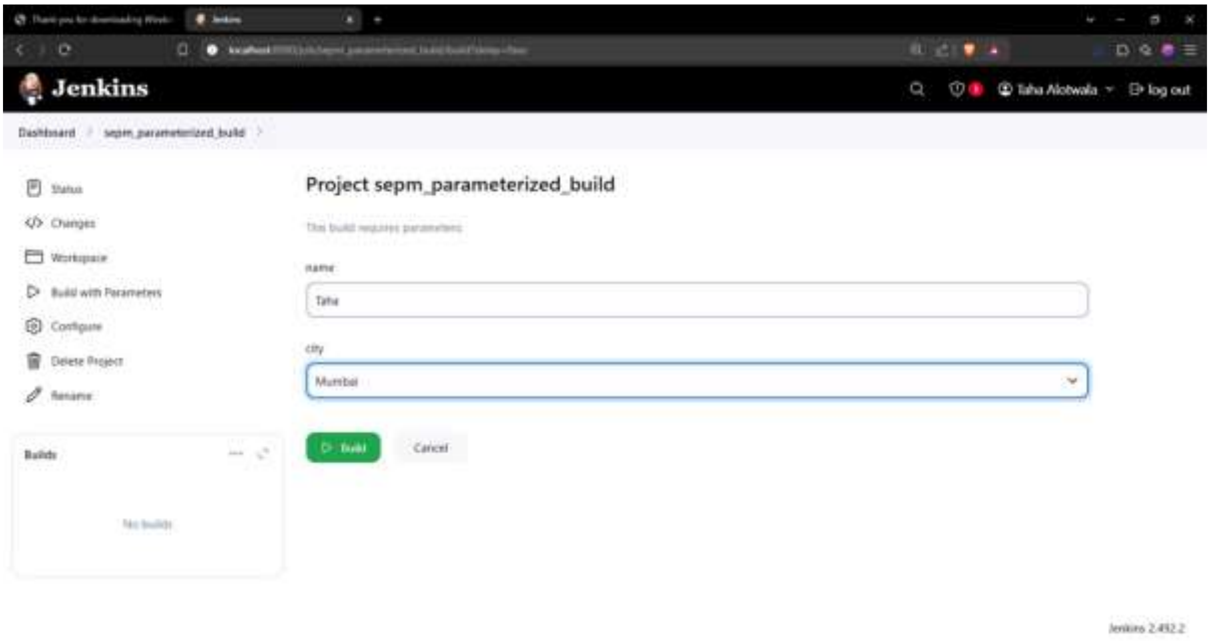
Command

See the list of available environment variables

```
D:\MiscRepos\sepm_lab\Files\parameterized_builds.cmd
```

Post-build Actions

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

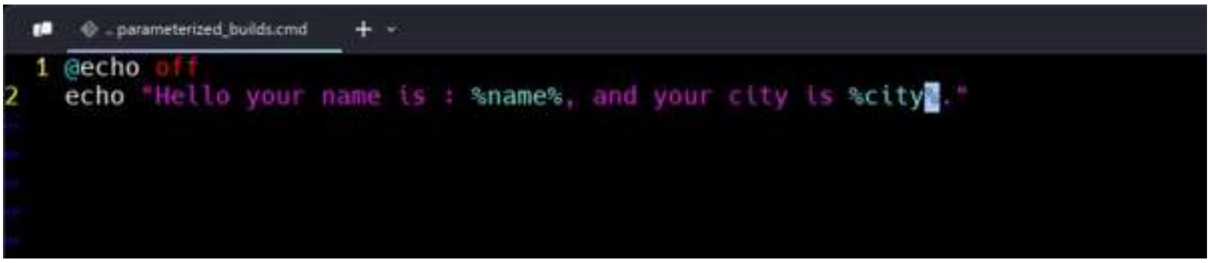


Project sepm_parameterized_build

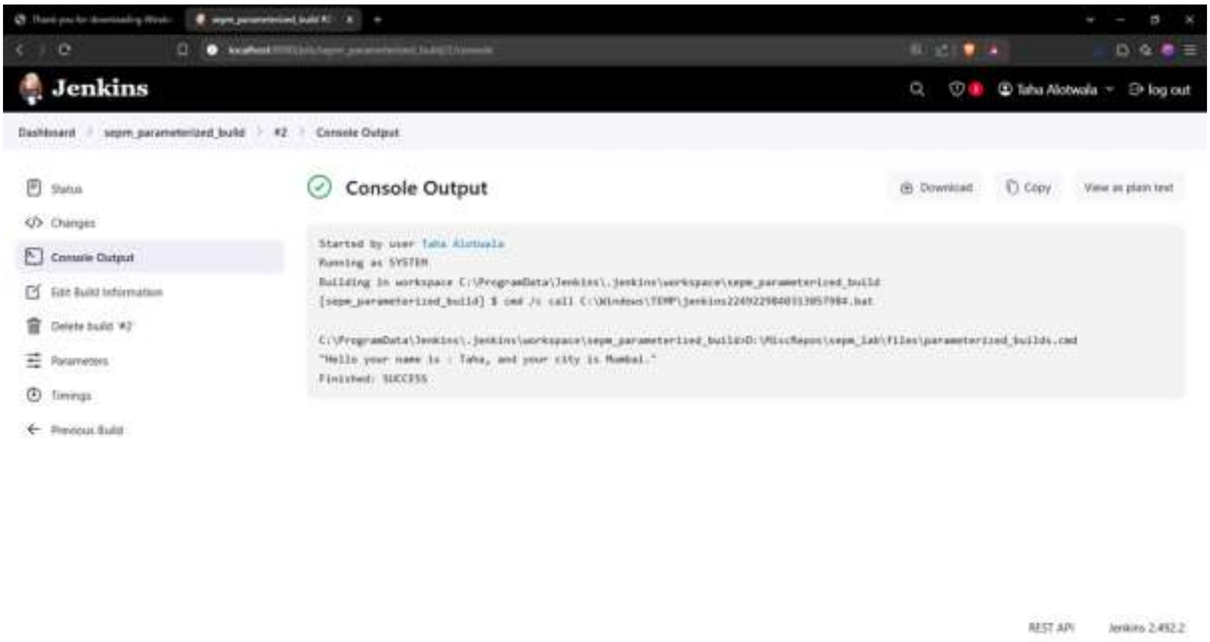
This build requires parameters

name: Taha

city: Mumbai



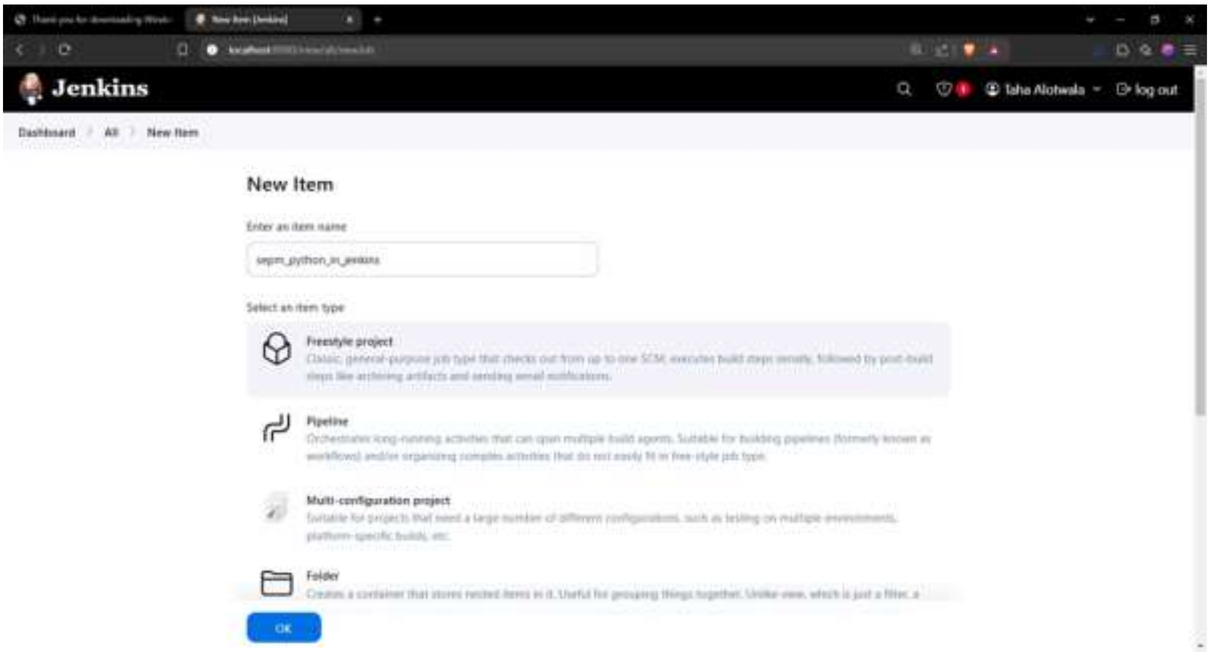
```
@echo off
echo "Hello your name is : %name%, and your city is %city%."
```



Console Output

```
Started by user Taha Alotwala
Running as SYSTEM
Building in workspace C:\ProgramData\Jenkins\.jenkins\workspace\sepm_parameterized_build
[sepm_parameterized_build] $ cmd /c call C:\Windows\TEMP\jenkins2249229840313057984.bat

C:\ProgramData\Jenkins\.jenkins\workspace\sepm_parameterized_build>D:\MiscRepos\sepm_lab\files\parameterized_builds.cmd
"Hello your name is : Taha, and your city is Mumbai."
Finished: SUCCESS
```

5. Running a Python file



New Item

Enter an item name: sepm_python_in_jenkins

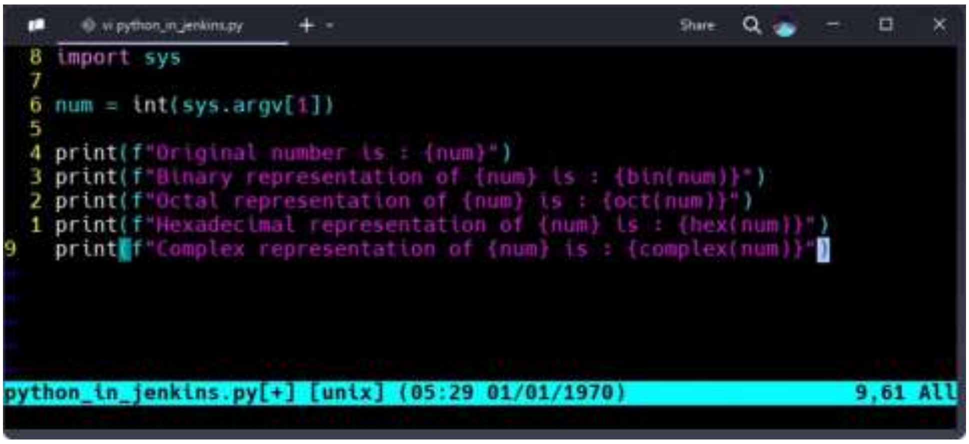
Select an item type

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.

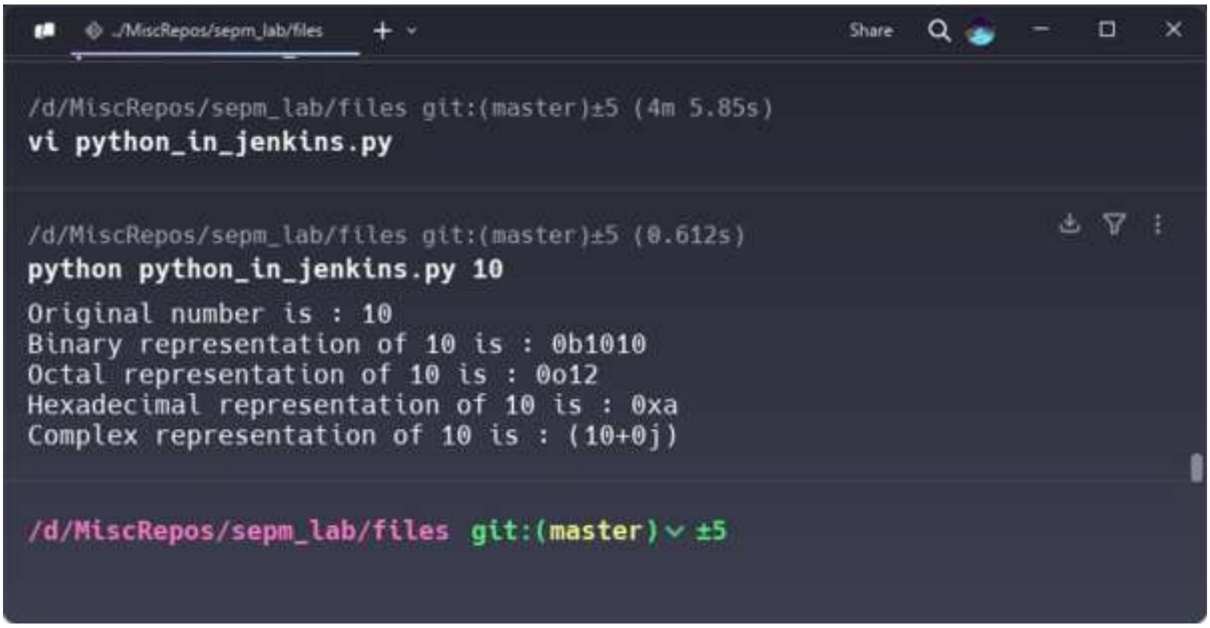
Multi-configuration project
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.

Folder
Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a



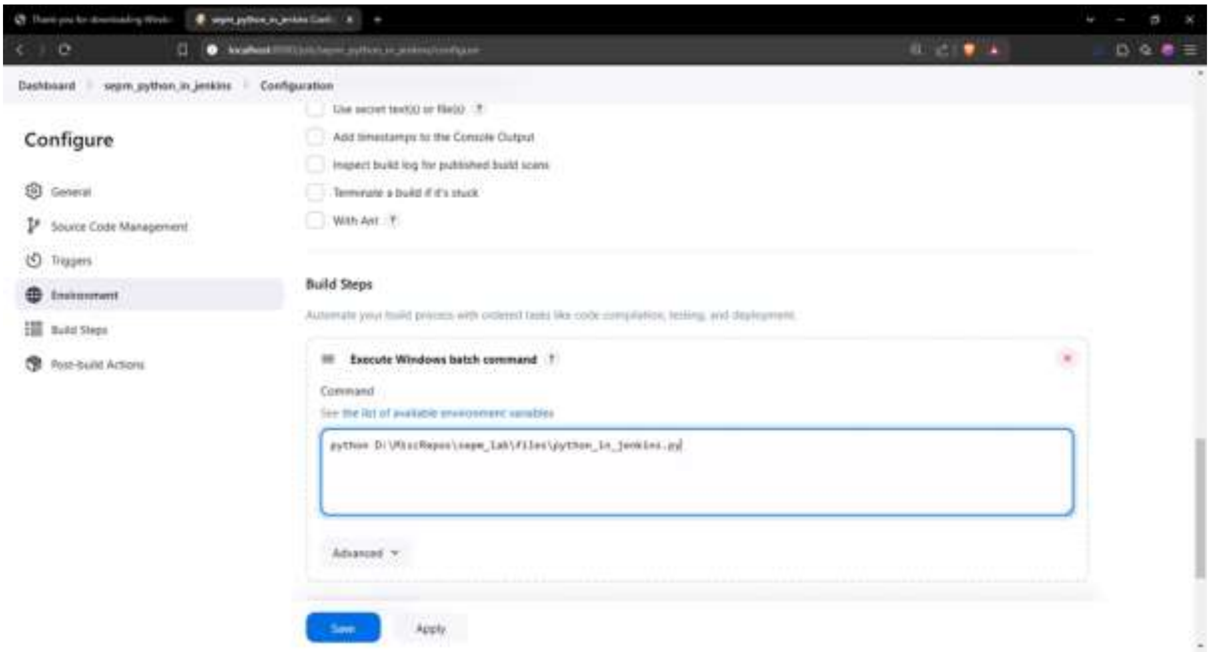
```
import sys

num = int(sys.argv[1])

print(f"Original number is : {num}")
print(f"Binary representation of {num} is : {bin(num)}")
print(f"Octal representation of {num} is : {oct(num)}")
print(f"Hexadecimal representation of {num} is : {hex(num)}")
print(f"Complex representation of {num} is : {complex(num)}")
```

```
python_in_jenkins.py[+] [unix] (05:29 01/01/1970)                9,61 All
```



```
/d/MiscRepos/sepm_lab/files git:(master)±5 (4m 5.85s)
vi python_in_jenkins.py

/d/MiscRepos/sepm_lab/files git:(master)±5 (0.612s)
python 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)

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

Configure Jenkins:



☐ 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

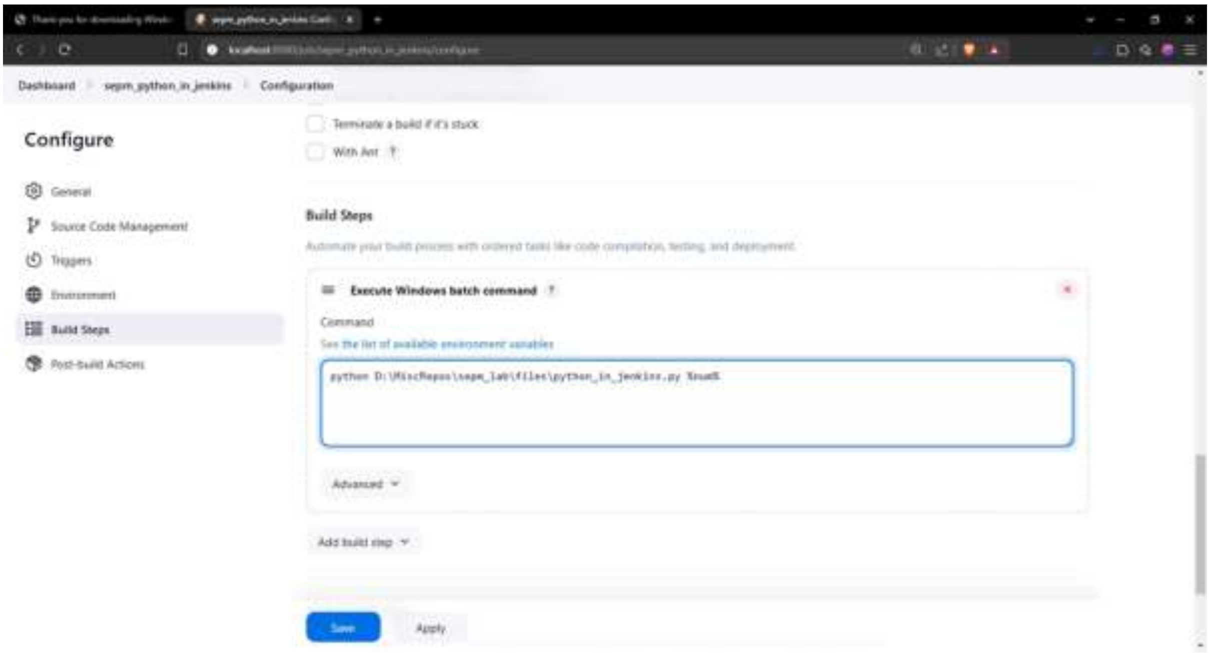
Automate your build process with ordered tasks like code compilation, testing, and deployment.

Execute Windows batch command

Command

See the list of available environment variables

```
python D:\MiscRepos\sepm_lab\files\python_in_jenkins.py
```



□ Terminate a build if it's stuck

□ With Ant

Build Steps

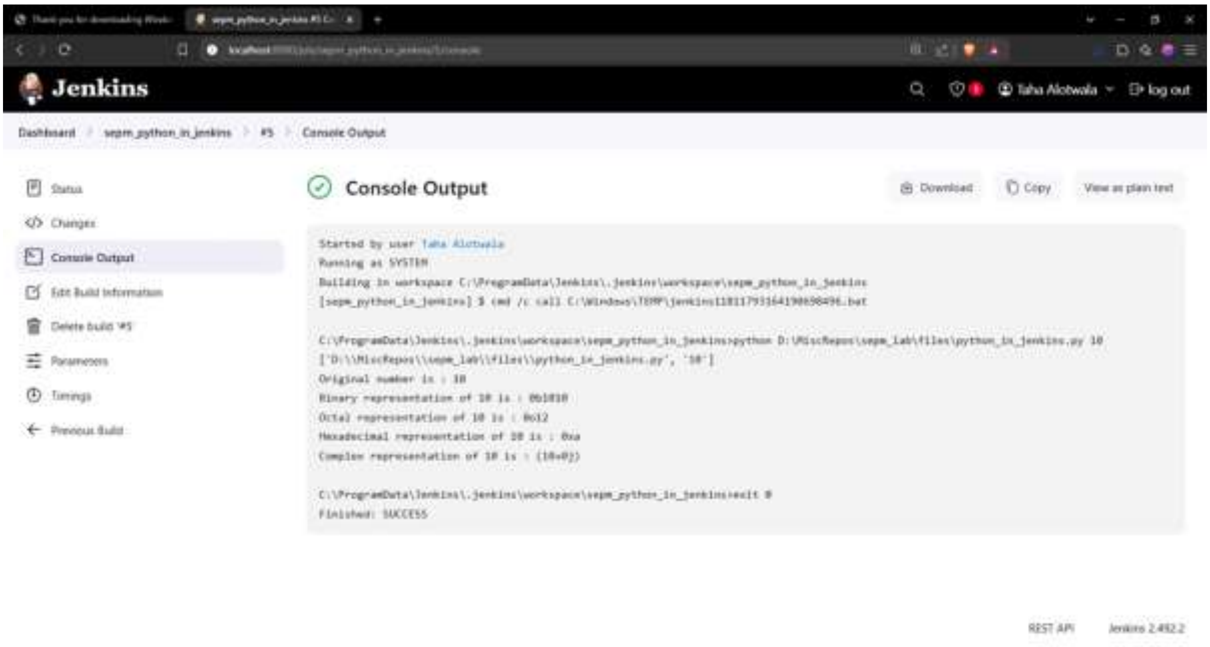
Automate your build process with ordered tasks like code compilation, testing, and deployment.

Execute Windows batch command

Command

See the list of available environment variables

```
python D:\MiscRepos\sepm_lab\files\python_in_jenkins.py %num%
```



Console Output

```
Started by user Taha Alotwala
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
```

CONCLUSION:

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