



**CONTINUOUS INTEGRATION AND CONTINUOUS  
DEPLOYMENT LAB**

**Lab File  
(2023-2024)**

**for**

**5<sup>th</sup> Semester**

**Submitted To**

Dr. Hitesh Kumar Sharma  
CI/CD Professor, Cluster Head  
(Cybernetics)  
School of Computer Science

**Submitted By:**

Arpit Goyal  
B. Tech. CSE DevOps [5<sup>th</sup>  
Semester]  
500094790  
R2142210148  
B-3

## EXPERIMENT 2

### *Creating a Jenkins Pipeline with a Jenkinsfile*

#### Aim

Create a Jenkins pipeline using a Jenkinsfile that builds a simple project, runs tests, and deploys the project to a designated environment

#### Steps

1. Create a maven project
  - a. Create the project

```
[INFO] --- jar:3.1.0:jar (default-jar) @ cicd.lab3 ---
[INFO] Building jar: C:\Users\hp\eclipse-workspace\cicd.lab3\target\myclass.jar
[INFO]
[INFO] --- install:3.1.1:install (default-install) @ cicd.lab3 ---
[INFO] Installing C:\Users\hp\eclipse-workspace\cicd.lab3\pom.xml to C:\Users\hp\.m2\repository\devops\b
[INFO] Installing C:\Users\hp\eclipse-workspace\cicd.lab3\target\myclass.jar to C:\Users\hp\.m2\reposito
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 3.128 s
[INFO] Finished at: 2023-12-01T01:53:36+05:30
[INFO] -----
```

- b. Create a package DevOps inside src/main/java/ and add App.java file

```
GNU nano 7.2 App.java
package DevOps;

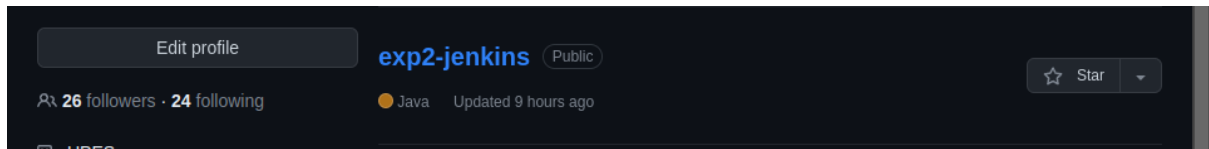
public class App {
    public void greetings() {
        System.out.println("Hello World!");
    }

    public static void main(String[] args) {
        App app = new App();
        app.greetings();
    }
}
```

2. Create a Jenkinsfile and write the steps to be executed in it

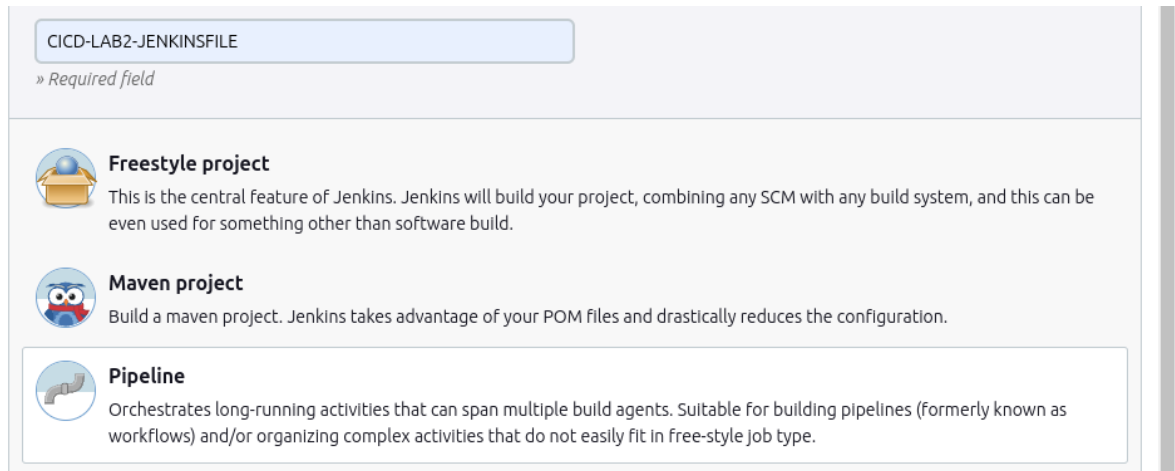
```
GNU nano 7.2 Jenkinsfile
pipeline {
    agent any
    tools {
        maven 'MAVEN_HOME'
    }
    stages {
        stage('Stage 1 : Clean Stage'){
            steps{
                sh 'mvn clean'
            }
        }
        stage('Stage 2 : test Stage'){
            steps{
                sh 'mvn test'
            }
        }
        stage('Stage 3 : Install stage'){
            steps{
                sh 'mvn install'
            }
        }
        stage('Stage Final : Build Sucess'){
            steps{
                echo 'Build Sucessfull'
            }
        }
    }
}
```

### 3. Create a GitHub Repository

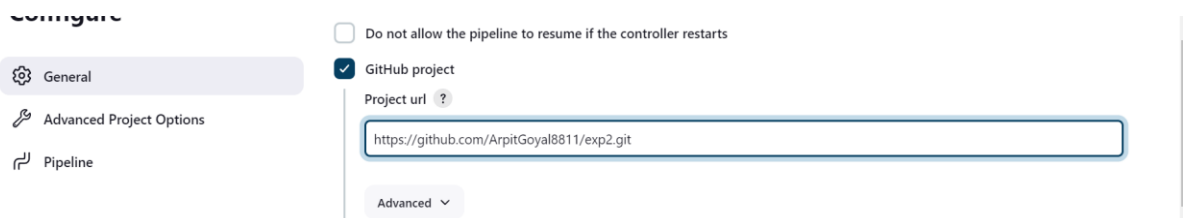


### 4. Create a Jenkins Pipeline

#### a. Create a pipeline



#### b. Add the project url



#### c. Configure the build trigger to poll scm and set it to perform build every minute



#### d. Specify the path to Jenkinsfile in the project

## Configure

General

Advanced Project Options

Pipeline

Definition

Pipeline script from SCM

SCM ?

None

Script Path ?

https://github.com/ArpitGoyal8811/exp2.git

☒ Lightweight checkout ?

Script Path ?

Jenkinsfile

e. Save the pipeline

5. Commit and push the changes of the project on GitHub

```
+ DevOps git:(master) x git add . && git commit -m "Project updated" && git push origin master
[master b8e0680] Project updated
1 file changed, 1 insertion(+), 9 deletions(-)
Username for 'https://github.com': Ayroid
Password for 'https://Ayroid@github.com':
Enumerating objects: 13, done.
```

6. An automatic build is triggered

		Declarative: Checkout SCM	Declarative: Tool Install	Stage 1 : Clean Stage	Stage 2 : test Stage	Stage 3 : Install stage	Stage Final : Build Sucess
Average stage times: (Average full run time: ~7s)		916ms	31ms	1s	2s	1s	55ms
#2	1 Sept 12 20:50 commit	1s	27ms	1s	2s	1s	58ms