PROGRAM 1: Introduction to Maven and Gradle

1. **Install Eclipse.**
2. **Select appropriate Eclipse IDE for Java.**
3. **Choose Java version and installation folder.**
4. **Install JDK and set environment variables.**
   First make sure JDK current version is installed
   https://www.oracle.com/java/technologies/downloads/?er=221886#jdk23-windows
5. **Install Apache Maven.**
   To install apache maven pls go to link as in below and download zip file of bin
   https://maven.apache.org/download.cgi
6. **Set PATH for Maven and verify with mvn --v.**
   Setup a PATH in environmental settings "Add the bin directory of the created directory apache-maven-3.9.9 to the PATH environment variable
7. **Install Gradle and set up in C:\Gradle.**
8. **Set environment variables and verify with gradle -v.**

PROGRAM 2: Working with Maven

1. **Open Eclipse → File → New → Maven Project.**
2. **Select default workspace.**
3. **Choose archetype: maven-archetype-quickstart.**
4. **Enter Group Id, Artifact Id, and Package name**
   Group Id:com.program2.maven Artifact Id:program2-example-jar
5. **Right-click project → Maven Build → Enter Goal: package.**
6. **Run the App.java file.**

PROGRAM 3: Working with Gradle

1. **In CMD:**
   - **mkdir pgm3**
   - **cd pgm3**
   - **gradle init → Choose application type, Groovy, Java version.**

   After execution of command the screen shows as in below where we opt for build type select as 1
   After that select Groovy
   It will ask for version and project name .After providing version and project name Select application structure as Single application structure and Domain Specific Language as Kotlin
2. **Build script using gradlew run**

PROGRAM 4: Build with Maven and Migrate to Gradle

1. **Create Maven project and run it.**
2. **Then to migrate to gradle Open terminal and press (Ctrl + Alt + Shift + T).**
3. **Run gradle init → select migration from Maven.**
4. **Choose Kotlin DSL.**
5. **Validate API generator prompt.**
6. **Type the command gradle build .Add required build.gradle.kts content.**
7. **Run gradle clean build and gradle run.**

PROGRAM 6: CI with Jenkins

1. **Open Jenkins → Manage Jenkins**
2. **Select Plugins.**
3. **Search for Maven IntegrationPlugin in Available Plugins and Install.**
4. **After Maven Integration Plugin is Installed We able to see Maven Project as New Item**
5. **configure the Location to properly Build and Run Maven Project So again click on Manage Jenkins and select Tools.**
6. **Install Maven Integration Plugin.**
7. **Go to New Item → Freestyle Project.**
8. **Add build step → Invoke top-level Maven targets.**
9. **Enter Goal: clean install.**
10. **Add path to pom.xml.**

11. **Click Build → View console output**

PROGRAM 9:

1. Go to Google chrome and type azure for students https://azure.microsoft.com/en-us/free/students

2. provide your email id at place of create account.

3. Verification code be mailed to the mentioned once kindly type it

4. Go To search and type Azure Devops

5. Click on Get started with Azure After the click u get the screen of Get free need not to do anything just click on Signin You will get screen as in below

6. search for services

7. :Select Azure Devops Organization

8. After u opting for Azure Devops Organizations u get a screen as in below now select My Azure DevOps Organizations

9. After above selection it once again reverifies name and email just click Continue After it U get

10. You will be able to see Organization is Created

PROGRAM 10: Creating Build Pipelines

1. **On creating organization goto Organization settings goto Policy And Allow Public Projects active**
2. **GOTO GITBASH TYPE COMMANDS AS IN BELOW mkdir maventest1 cd maventest1**
3. **to create simple hellow world maven project type command as in below mvn archetype:generate -DgroupId=com.dineshonjava -DartifactId=Javateam -DarchetypeArtifactId=m**
4. **to add files from local to github Follow the procedure a) First create a repository in github as maventestazure b) Then come to gitbash and type git init git add . git commit -m "azure pipeline example" git branch -M main git remote add origin https://github.com/gurumurthy974/maventestazure.git git push -u origin main After completion of above command my repository looks**
5. **: Now goto Azure Devops Organization create Public Project**
6. **SELECT PIPELINE and then click on create pipeline**
7. **After creating Pipeline select type of repo as Github**
8. **It asks for minimum signin verification after that ur screen be as in below select required repository there to run maven project in my case its maventest123**
9. **AFTER REQUIRED REPO is selected click on Approve and Install**
10. **You will be able to see starter pipeline**
11. **After selecting maven it asks for save and run just click on it**
12. **Finally You be able to see tasks running its failed bec we shld mention proper path For pom.xml**
13. **The commits will also be visible in github**