UNIVERSITY INSTITUTE of COMPUTING
Asia's Fastest Growing University

NAAC GRADE A+ ACCREDITED UNIVERSITY

ExperimentNo.1.3

Student Name: Gaurav Kumar
UID: 22MCC20177
Branch: MCA–CCD
Section/Group: MCD-1/A
Semester: III
Date of Performance: 14th Sept 23
Subject Name: DevOps Process Automation Lab
Subject Code: 22CAP-745

1. Aim/Overview of the practical:

With an understanding of the theory and techniques that Maven uses to manage a project, we can now develop our own Maven project.
In this application, you must create a simple adder that computes the sum of two integers. During this process, you will:

a) Generate a Maven project using the Maven command-line tool.
b) Configure the pom.xml file.
c) Create main and test source code files.
d) Execute a Maven build.
e) Execute the resulting JAR file.

2. Code for practical: (a)

Step 1 : First, to work with maven you must install Java and java install path to **PATH** variable of your system.

Step 2 : After successful java configuration you must download the maven zip file and extract it anywhere in your system. After that add the bin folder path in your system variable named PATH.

Step 3 : Open **Terminal** and use "**mvn archetype:generate**" mvn command to create new maven project.

```
mvn archetype:generate -DgroupId=dev.gauravkumar -DartifactId= ws_1_3 -
DarchetypeArtifactId=maven-archetype-quickstart -DinteractiveMode=false
```

Step 4 : Cd into newly created folder. Verify your maven project using `mvn validate` command.

```
gaurav D: … DevOps ws_1_3 ws_1_3                                  27ms 00:43:25
$ mvn validate
[INFO] Scanning for projects...
[INFO]
[INFO] ----------------------< dev.gauravkumar:ws_1_3 >----------------------
[INFO] Building ws_1_3 1.0-SNAPSHOT
[INFO]   from pom.xml
[INFO] --------------------------------[ jar ]---------------------------------
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  0.063 s
[INFO] Finished at: 2023-09-15T00:43:37+05:30
[INFO] ------------------------------------------------------------------------
```

Code for practical: (b)

Step 1 : Now there is a pom.xml Configuration file present inside maven project. You can edit this file. In our case we add **junit** for testing purposes.

Step 2 : Open project in any **IDE.**

Step 3 : Edit pom.xml to add configuration.

```
21    <dependencies>
22      <dependency>
23        <groupId>junit</groupId>
24        <artifactId>junit</artifactId>
25        <version>4.11</version>
26        <scope>test</scope>
27      </dependency>
28    </dependencies>
```

Code for practical: (c)

Step 1 : Edit App.java write adder function logic.

```
public class App
{
    public static void main( String[] args ) {
        int num1 = Integer.parseInt(args[0]);
        int num2 = Integer.parseInt(args[1]);
        System.out.println(
                "The sum of " + num1 + " and " + num2 + " is " + adder(num1, num2)
        );
    }
    3 usages
    public static int adder(int a, int b) { return a + b; }
}
```

Step 2 : Edit AppTest.java write testing logic to test the adder function.

```
    @Test
    public void fifteenPlusTwentyIsThirtyFive(){
        assertEquals( expected: 35, App.adder( a: 15,  b: 20));
    }

    @Test
    public void twentyPlusTwentyIsForty(){
        assertEquals( expected: 40, App.adder( a: 20,  b: 20));
    }
```

Code for practical: (d)

Step 1 : Use mvn package command to build your project.

Step 2 : This will start the build process. Maven will compile your code, run any tests, and package your code into a JAR file which you can find in the target directory of your project.

Code for practical: (e)

Step 1 : To execute the resulting JAR file, you can use the java -jar command followed by the name of your JAR file.

Step 2 : Open Terminal. Navigate to the directory of your JAR file using the cd command.

Step 3 : Now use java -jar command to run .jar file.

```
java -jar .\target\ws_1_3-1.0-SNAPSHOT.jar <num1> <num2>
```