

Assignment – 5

Aim: To build Java program using Jenkins.

LO Mapped: LO1, LO3

Theory:

Create java Programs:

Program-1

```
public class addnum {  
    public static void main(String[] args){  
        System.out.print("The sum of these numbers is: ");  
        int n1 = 6;  
        int n2 = 5;  
        int n3 = 7;  
        int sum = n1+n2+n3;  
        System.out.println(sum);  
    }  
}
```

Program-2

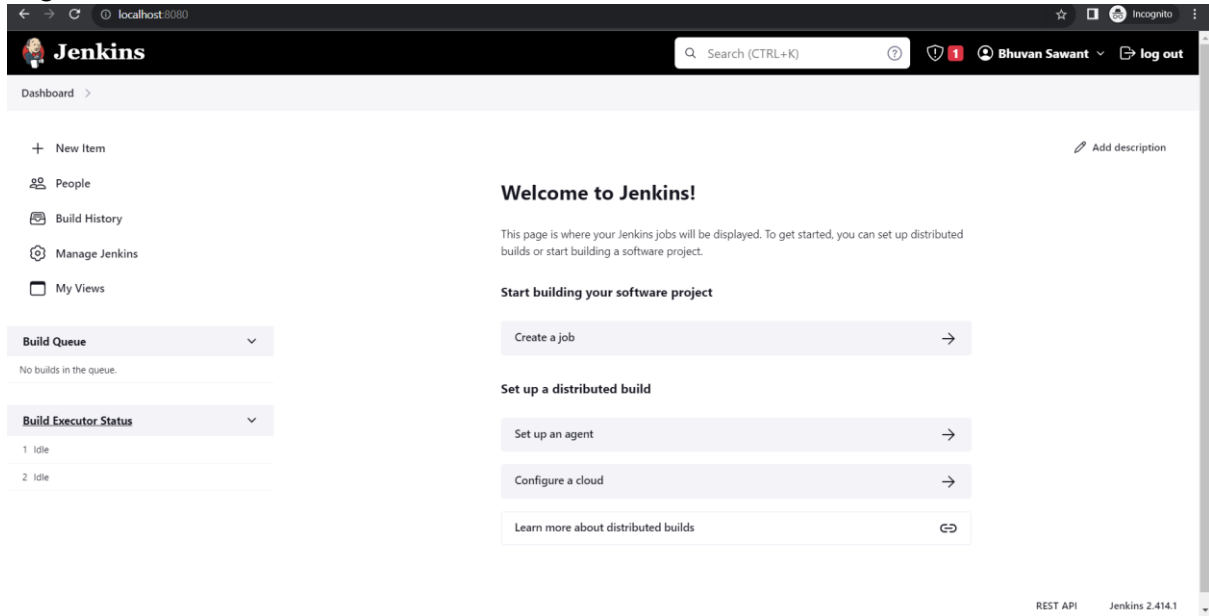
```
public class Matrix_multiplication {  
    public static void main(String args[]) {  
        int n = 3;  
        int[][] a = { {5, 2, 3}, {2, 6, 3}, {6, 9, 1} };  
        int[][] b = { {2, 7, 5}, {1, 4, 3}, {1, 2, 1} };  
        int[][] c = new int[n][n];  
        System.out.println("Matrix A:");  
        for (int i = 0; i < n; i++) {
```

```

        for (int j = 0; j < n; j++) {
            System.out.print(a[i][j] + " ");
        }
        System.out.println();
    }
    System.out.println("Matrix B:");
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            System.out.print(b[i][j] + " ");
        }
        System.out.println();
    }
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            for (int k = 0; k < n; k++) {
                c[i][j] = c[i][j] + a[i][k] * b[k][j];
            }
        }
    }
    System.out.println("The product of two matrices is:");
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            System.out.print(c[i][j] + " ");
        }
        System.out.println();
    }
}

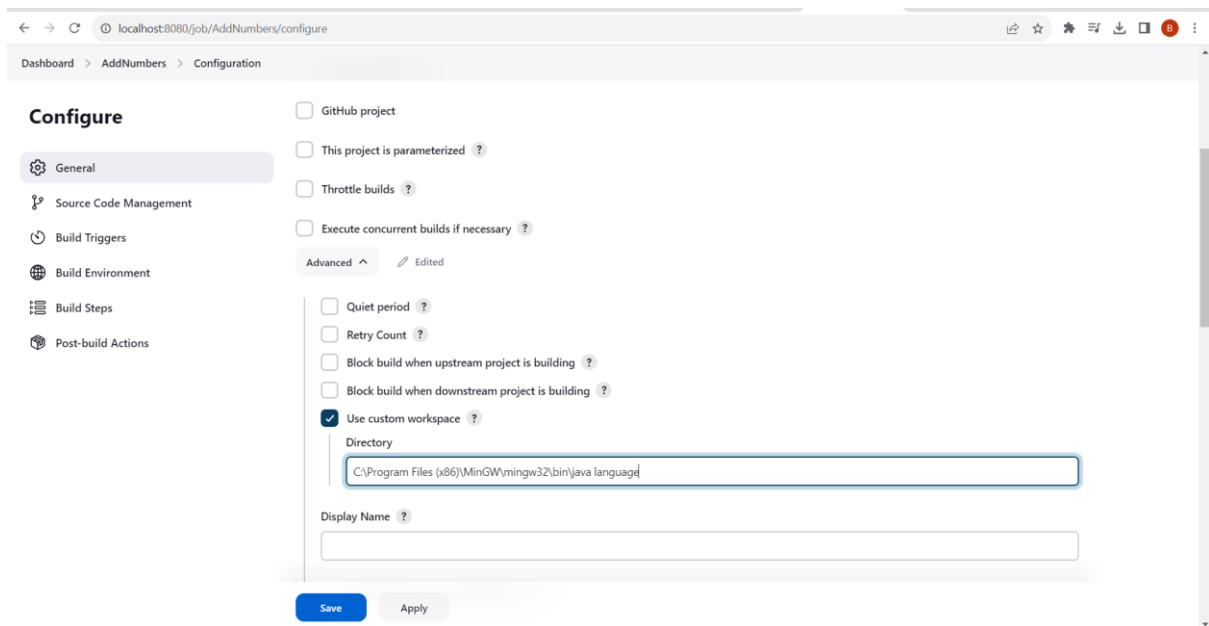
```

Login to Jenkins



The screenshot shows the Jenkins dashboard in a web browser. The top navigation bar includes the Jenkins logo, a search bar, and a user profile for 'Bhuvan Sawant' with a 'log out' button. The left sidebar contains links for 'New Item', 'People', 'Build History', 'Manage Jenkins', and 'My Views'. The main content area features a 'Welcome to Jenkins!' message, a 'Start building your software project' section with a 'Create a job' button, and a 'Set up a distributed build' section with buttons for 'Set up an agent', 'Configure a cloud', and 'Learn more about distributed builds'. On the left, there are two expandable sections: 'Build Queue' (showing 'No builds in the queue') and 'Build Executor Status' (showing two 'Idle' executors). The bottom right corner displays 'REST API' and 'Jenkins 2.414.1'.

Check use custom workspace and give the path of directory where java file located.




The screenshot shows the 'Configure' page for a Jenkins job named 'AddNumbers'. The left sidebar lists configuration categories: 'General', 'Source Code Management', 'Build Triggers', 'Build Environment', 'Build Steps', and 'Post-build Actions'. The 'General' tab is selected. In the 'Advanced' section, the 'Use custom workspace' checkbox is checked. Below it, the 'Directory' text field contains the path 'C:\Program Files (x86)\MinGW\mingw32\bin\java language'. At the bottom, there are 'Save' and 'Apply' buttons.

Then click on Apply and see the output of java code in console output.

← → ↻

localhost:8080/job/AddNumbers/1/console

🔍 ⭐ ⚙️ 📄 ⬇️ 🏠 🔴

 **Jenkins**

🔍 Search (CTRL+K) ?

🛡️ 1 👤 Bhuvan Sawant 🚪 log out

Dashboard > AddNumbers > #1 > Console Output

📄 Status

</> Changes

📄 Console Output

📄 View as plain text

🔍 Edit Build Information

🗑️ Delete build '#1'

✅ Console Output

Started by user [Bhuvan Sawant](#)
Running as SYSTEM
Building in workspace C:\Program Files (x86)\MinGW\mingw32\bin\java language
[java language] \$ cmd /c call C:\Users\De11\AppData\Local\Temp\jenkins15239848745863819114.bat

C:\Program Files (x86)\MinGW\mingw32\bin\java language>javac addnum.java

C:\Program Files (x86)\MinGW\mingw32\bin\java language>java addnum
The sum of these numbers is: 18


C:\Program Files (x86)\MinGW\mingw32\bin\java language>exit 0
Finished: SUCCESS

REST API Jenkins 2.414.1

← → ↻

localhost:8080/job/Matrix_multiplication/configure

🔍 ⭐ ⚙️ 📄 ⬇️ 🏠 🔴

 **Jenkins**

🔍 Search (CTRL+K) ?

🛡️ 1 👤 Bhuvan Sawant 🚪 log out

Dashboard > Matrix_multiplication > Configuration

Configure

⚙️ General

🔑 Source Code Management

🕒 Build Triggers

🌐 Build Environment

📋 Build Steps

🔧 Post-build Actions

General

Enabled ☒

Description

Java code for multiplication of two matrices

Plain text [Preview](#)

☐ Discard old builds ?
☐ GitHub project
☐ This project is parameterized ?
☐ Throttle builds ?
☐ Execute concurrent builds if necessary ?

Advanced ▾ [Edited](#)

Save

Apply

← → ↺

localhost:8080/job/Matrix_multiplication/configure

🔖 ☆ ⚙️ 📄 📥 📂 ⌵

B

Dashboard > Matrix_multiplication > Configuration

Configure

⚙️ General

🔗 Source Code Management

🕒 Build Triggers

🌐 Build Environment

📋 Build Steps

🔧 Post-build Actions

☐ Throttle builds ?

☐ Execute concurrent builds if necessary ?

Advanced ^

✎ Edited

☐ Quiet period ?

☐ Retry Count ?

☐ Block build when upstream project is building ?

☐ Block build when downstream project is building ?

☒ Use custom workspace ?

Directory

C:\Program Files (x86)\MinGW\mingw32\bin\java language

Display Name ?

☐ Keep the build logs of dependencies ?

Source Code Management

Save

Apply

← → ↺

localhost:8080/job/Matrix_multiplication/configure

🔖 ☆ ⚙️ 📄 📥 📂 ⌵

B

Dashboard > Matrix_multiplication > Configuration

Configure

⚙️ General

🔗 Source Code Management

🕒 Build Triggers

🌐 Build Environment

📋 Build Steps

🔧 Post-build Actions

Build Steps

≡ Execute Windows batch command ?

✕

Command

See [the list of available environment variables](#)

javac Matrix_multiplication.java
java Matrix_multiplication

Advanced ▾

Add build step ▾

Post-build Actions

Add post-build action ▾

Save

Apply

The screenshot shows the Jenkins web interface at localhost:8080. The breadcrumb trail is Dashboard > Matrix_multiplication > #1 > Console Output. On the left sidebar, the 'Console Output' option is selected. The main area displays the console output for build #1, which is a successful build. The output text is as follows:

```
Started by user Bhuvan Sawant
Running as SYSTEM
Building in workspace C:\Program Files (x86)\MinGW\mingw32\bin\java language
[java language] $ cmd /c call C:\Users\De11\AppData\Local\Temp\jenkins8001704838850330701.bat

C:\Program Files (x86)\MinGW\mingw32\bin\java language>javac Matrix_multiplication.java

C:\Program Files (x86)\MinGW\mingw32\bin\java language>java Matrix_multiplication
Matrix A:
5 2 3
2 6 3
6 9 1
Matrix B:
2 7 5
1 4 3
1 2 1
The product of two matrices is:
15 49 34
13 44 31
22 80 58

C:\Program Files (x86)\MinGW\mingw32\bin\java language>exit 0
Finished: SUCCESS
```

After executing code, you can see your files on dashboard.

The screenshot shows the Jenkins Dashboard at localhost:8080. The left sidebar contains links for New Item, People, Build History, Manage Jenkins, and My Views. The main area displays a table of builds. The table has columns for status (S), warnings (W), name, last success, last failure, and last duration. Two builds are listed: 'AddNumbers' and 'Matrix_multiplication'. Both builds are successful and have no warnings.

S	W	Name	Last Success	Last Failure	Last Duration
✓	☀	AddNumbers	12 min #1	N/A	2.2 sec
✓	☀	Matrix_multiplication	5 min 0 sec #1	N/A	1.8 sec

Below the table, there is an 'Icon legend' section with links for 'Atom feed for all', 'Atom feed for failures', and 'Atom feed for just latest builds'.

Conclusion: In this experiment We had executed java programs on Jenkins.