

**Installation of Java and Simple Java Programs Aim:**

To install Java Development Kit (JDK), configure the environment, and write simple Java programs including Hello World.

**PRE LAB EXERCISE****QUESTIONS**

1. What is JDK and why is it required?

**JDK (Java Development Kit)** is a software package used to **develop Java programs**.

2. Difference between JDK, JRE, and JVM.

Term	Full Form	Purpose
<b>JDK</b>	Java Development Kit	Used to <b>develop and run</b> Java programs
<b>JRE</b>	Java Runtime Environment	Used to <b>run</b> Java programs
<b>JVM</b>	Java Virtual Machine	Executes Java <b>bytecode</b> and makes Java platform-independent

3. What is the purpose of the main() method in Java?

The **main()** method is the **starting point** of execution of a Java program.

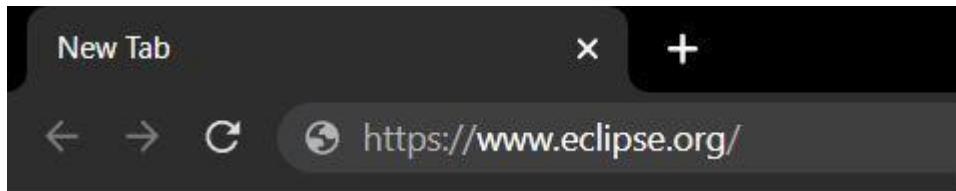
- >The JVM starts running the program from main().
- >Without main(), the program will not run.

**IN LAB EXERCISE Objective:**

To verify Java installation and execute a basic Java program.

**INSTALLATION STEPS:****STEP 1: Open Browser**

- Open your browser and go to the official [URL](https://www.eclipse.org/) Eclipse Downloads page.



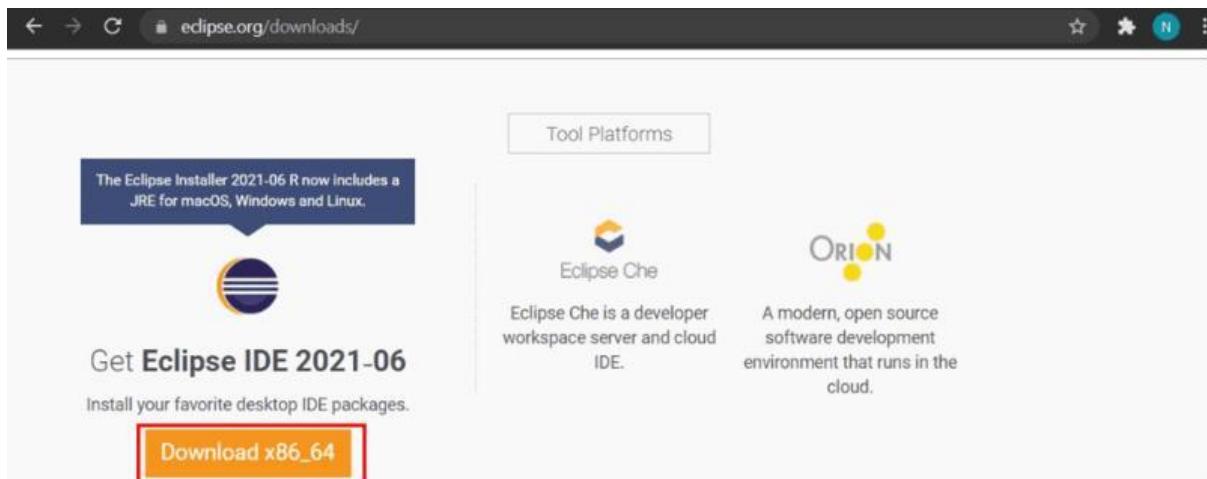
## STEP 2: Download Eclipse Installer

- Then, click on the "Download" button to download Eclipse IDE.



## STEP 3: Download EXE

- Now, click on the "Download x86\_64" button.

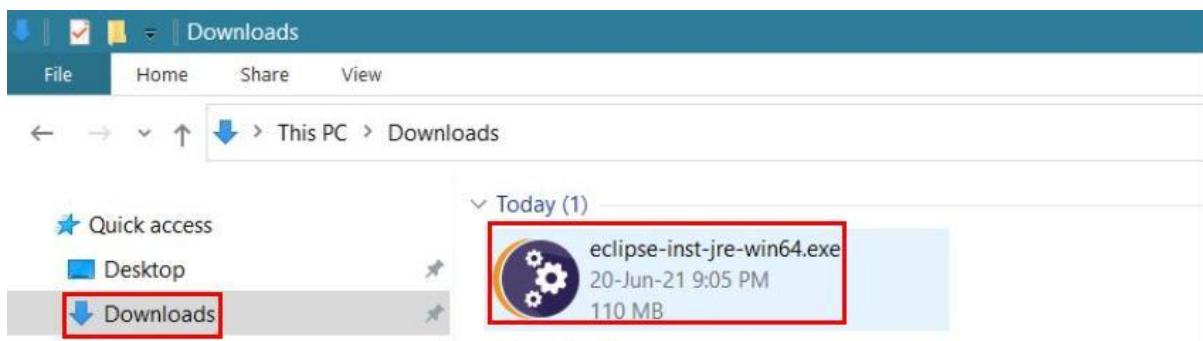


**STEP 4:** Then click on the "Download" button. After clicking on the download button the .exe file for the eclipse will be downloaded.

The screenshot shows the Eclipse Foundation Downloads page. At the top, there's a navigation bar with the Eclipse logo and a menu icon. Below it, a breadcrumb trail reads "Home / Downloads / Eclipse downloads - Select a mirror". A note states: "All downloads are provided under the terms and conditions of the Eclipse Foundation Software User Agreement unless otherwise specified." In the center, there's a large orange "Download" button with a downward arrow icon. Below it, text says "Download from: Japan - Japan Advanced Institute of Science and Technology (https)". Underneath, a box contains the file information: "File: [eclipse-inst-jre-win64.exe](#) SHA-512". At the bottom right of this box is a link "»> Select Another Mirror".

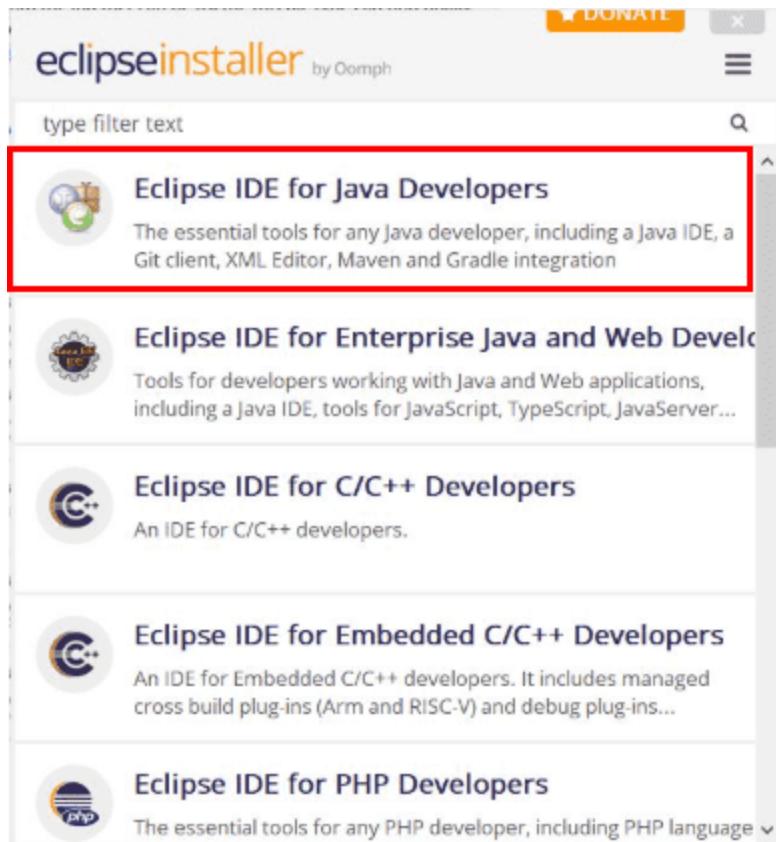
#### STEP 5: Open Download EXE

- Now go to File Explorer and click on "Downloads" after that click on the "*eclipseinst-jre-win64.exe*" file for installing Eclipse IDE.

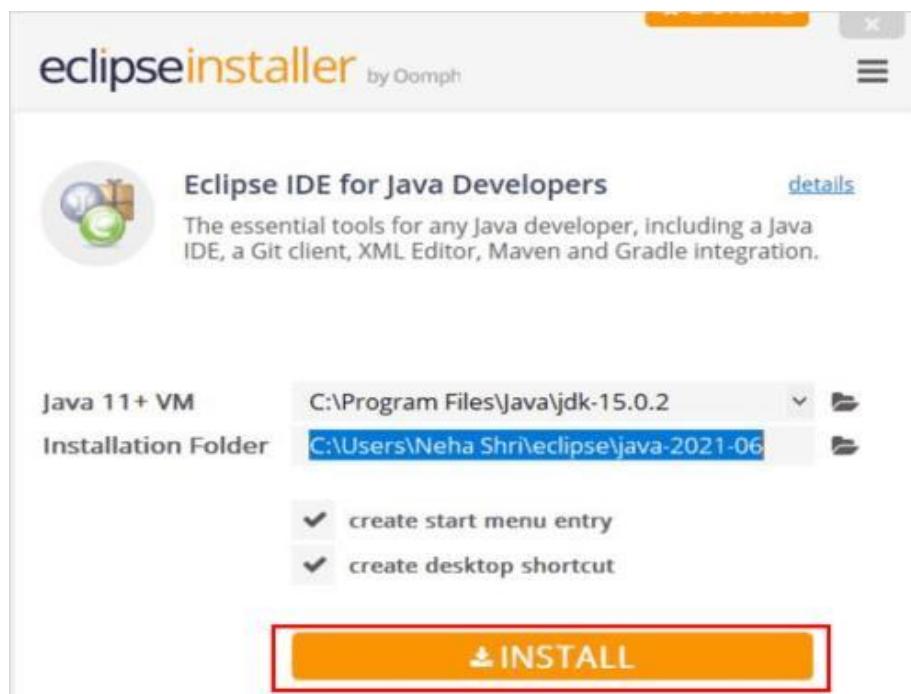


#### STEP 6: Install Eclipse

- Then, click on "Eclipse IDE for Java Developers".

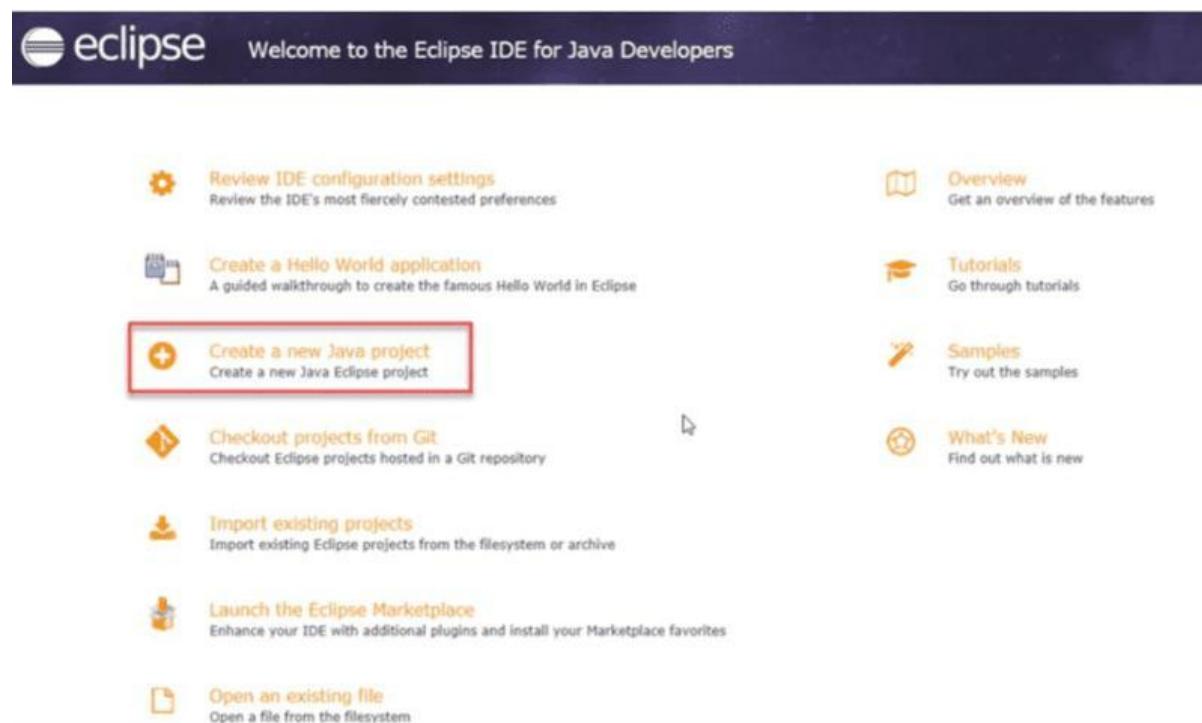


**STEP 7: Then, click on the "Install" button.**



### Step 8: Create New Project

Now click on "Create a new Java project".



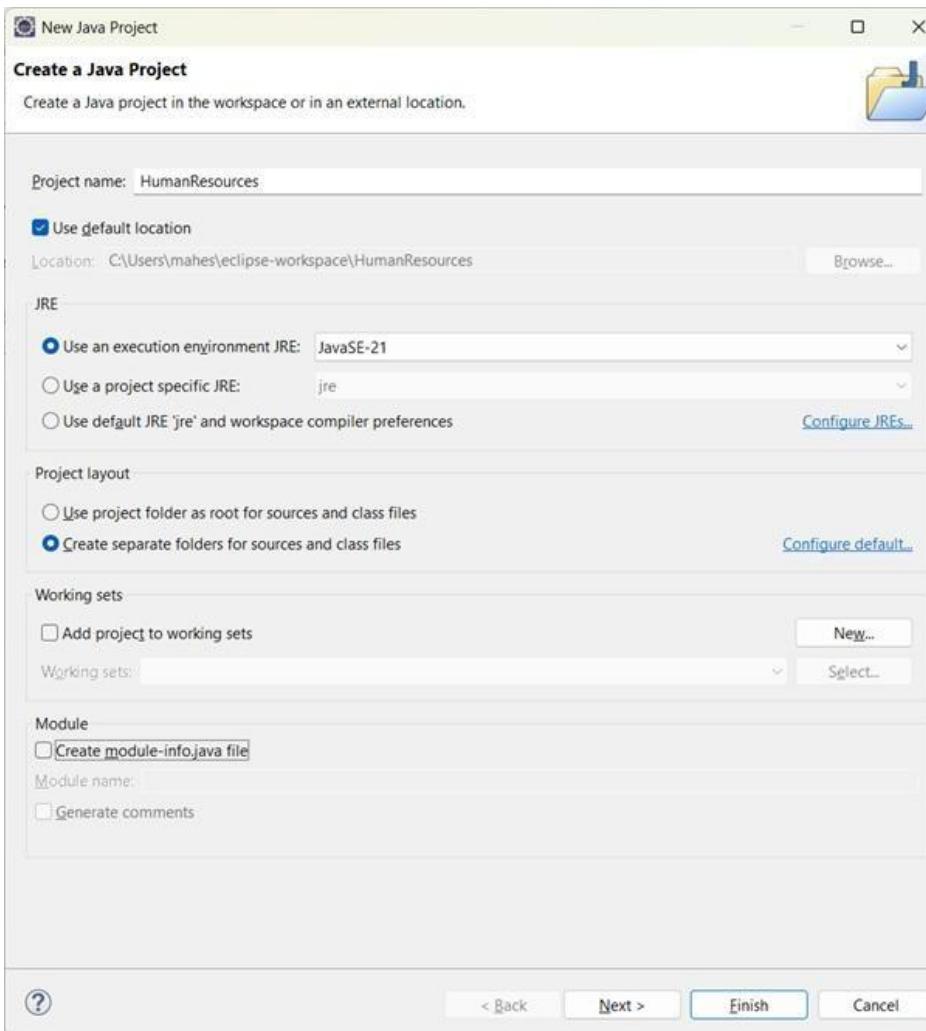
### STEP 9: Create a new java project

- By clicking on the File menu and choosing New → Java Project.
- By right clicking anywhere in the Project Explorer and selecting New → Java Project.

- By clicking on the New button (  ) in the Tool bar and selecting Java Project. **STEP**

### **10: Enter the Project Name**

- Select the Java Runtime Environment (JRE) or leave it at the default
- Select the Project Layout which determines whether there would be a separate folder for the source codes and class files. The recommended option is to create separate folders for sources and class files.

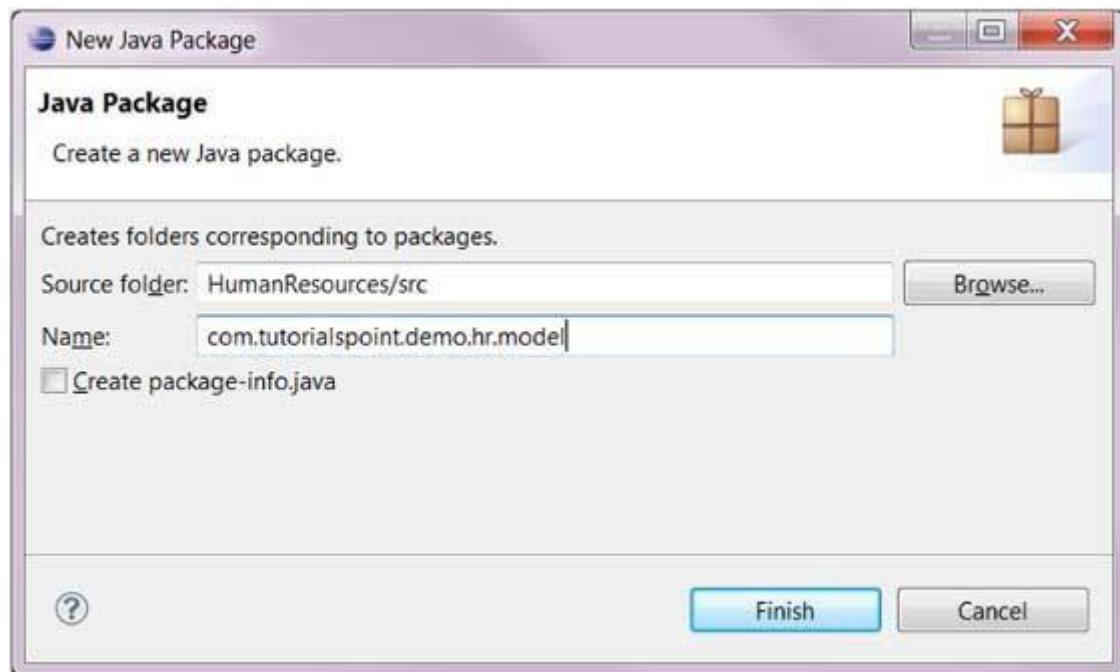


### **STEP 11: Create a new java package**

- By clicking on the File menu and selecting New → Package.
- By right click in the package explorer and selecting New → Package.
- By clicking on the package icon which is in the tool bar(  ).

### **STEP 11:**

- Enter/confirm the source folder name.
- Enter the package name.
- Click on the Finish button.

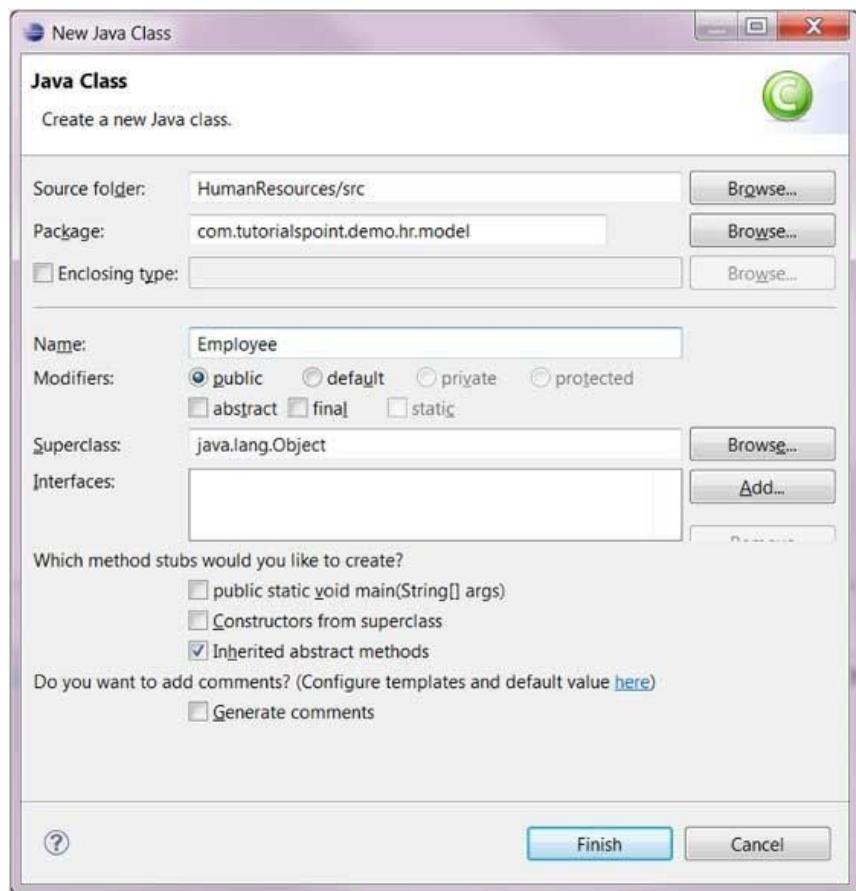


#### **STEP 12: Create a New Java class.**

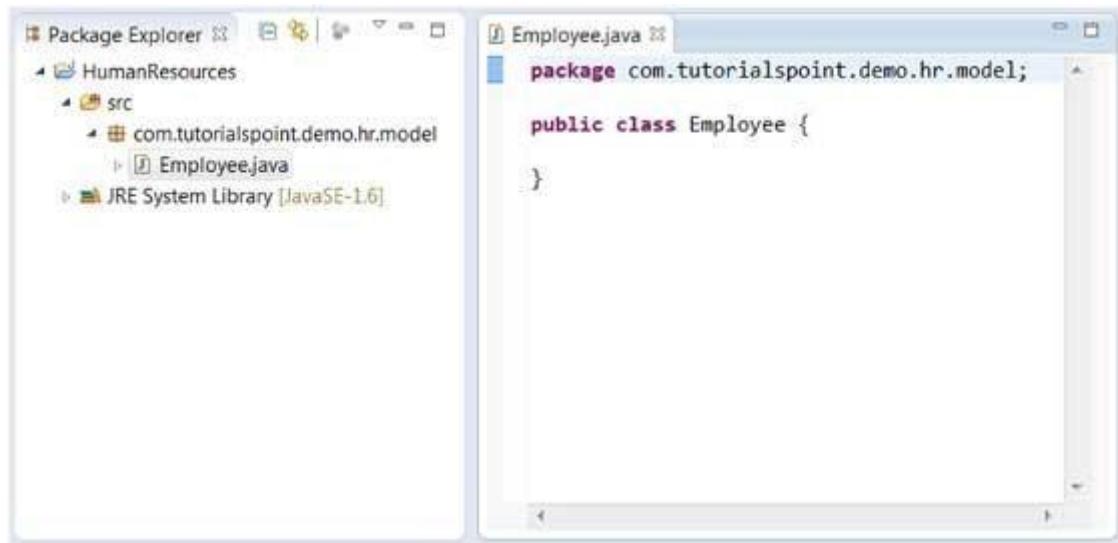
- By clicking on the File menu and selecting New → Class.
- By right clicking in the package explorer and selecting New → Class.
- By clicking on the class drop down button ( ) and selecting class ( ).

#### **STEP 13:**

- Ensure the source folder and package are correct.
- Enter the class name.
- Select the appropriate class modifier.
- Enter the super class name or click on the Browse button to search for an existing class.
- Click on the Add button to select the interfaces implemented by this class.
- Examine and modify the check boxes related to method stubs and comments.



#### STEP 14: Class created successfully.



## BASIC PROGRAMS:

### Program 1: Hello World Program

**Source Code:**

```
class HelloWorld {  
    public static void main(String[] args) {  
        System.out.println("Hello World");  
    }  
}
```

### Output:

Hello World

The screenshot shows a Java code editor with a dark theme. A file named "Untitled-1.java" is open, containing the following code:

```
class HelloWorld {  
    public static void main(String[] args) {  
        System.out.println("Hello World");  
    }  
}
```

Below the code editor is a terminal window titled "PROBLEMS" (with 1 error). The terminal output is:

```
PS C:\Users\ragur> & 'C:\Program Files\Java\jre-preview' '-XX:+ShowCodeDetailsInExceptionMessages'  
Hello World  
PS C:\Users\ragur>
```

### Program 2: Display Personal Details Source

#### Code:

```
class DisplayInfo {  
    public static void main(String[] args) {  
        System.out.println("Name: Anitha");  
        System.out.println("Age: 20");  
    }  
}
```

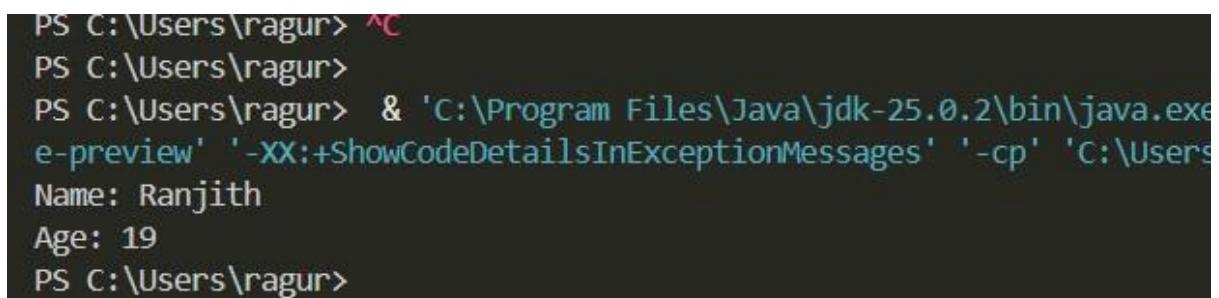
**Output:**

Name: Ranjith B

Age: 19



```
Untitled-1.java 1 X  
1 class DisplayInfo {    Untitled-1.java is a non-project file, only syntax errors are reported  
2     Run | Debug  
3     public static void main(String[] args) {  
4         System.out.println(x: "Name: Anitha");  
5         System.out.println(x: "Age: 20");  
6     }  
7 }
```



```
PS C:\Users\ragur> ^C  
PS C:\Users\ragur>  
PS C:\Users\ragur> & 'C:\Program Files\Java\jdk-25.0.2\bin\java.exe  
e-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users  
Name: Ranjith  
Age: 19  
PS C:\Users\ragur>
```

**Program 3: Addition of Two Numbers Source**

**Code:**

```
class AddTwoNumbers {  
    public static void main(String[] args) { int a  
        = 10, b = 20;  
        System.out.println("Sum = " + (a + b));  
    }  
}
```

**Output:**

Sum = 30

The screenshot shows a Java code editor with the file 'Untitled-1.java' open. The code is identical to the one above. A status bar at the bottom of the editor window indicates 'untitled-1.java is a non-project file, only syntax errors are reported'. Below the editor is a terminal window with the following content:

```
PROBLEMS 1 OUTPUT DEBUG CONSOLE EXPLORER TERMINAL  
PS C:\Users\ragur> & 'C:\Program Files\Java\jdk-25.0.2\bin\java-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp  
Sum = 30  
PS C:\Users\ragur>
```

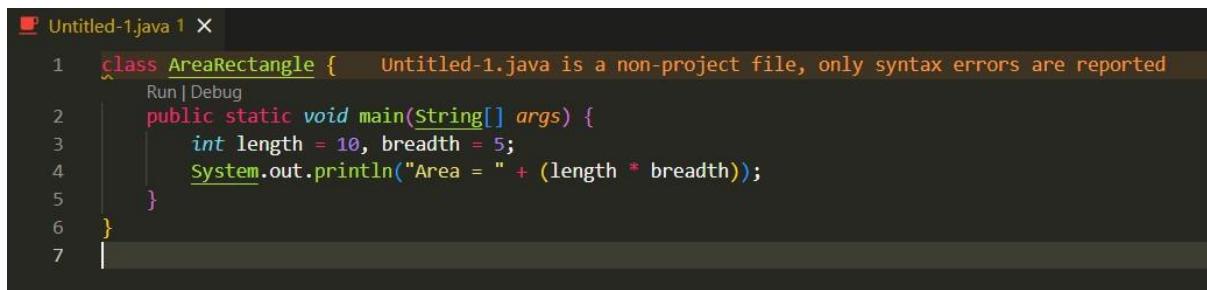
**Program 4: Area of a Rectangle Source**

**Code:**

```
class AreaRectangle { public static void
main(String[] args) { int length = 10,
breadth = 5;
System.out.println("Area = " + (length * breadth));
}
}
```

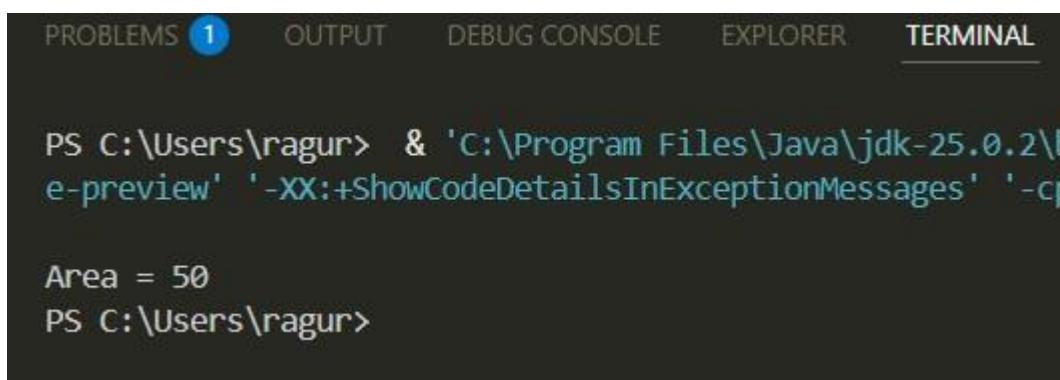
**Output:**

Area = 50



A screenshot of a code editor window titled "Untitled-1.java". The code is a simple Java program that calculates the area of a rectangle with length 10 and breadth 5, printing the result to the console. The code is as follows:

```
1 class AreaRectangle {    Untitled-1.java is a non-project file, only syntax errors are reported
2     Run | Debug
3     public static void main(String[] args) {
4         int length = 10, breadth = 5;
5         System.out.println("Area = " + (length * breadth));
6     }
7 }
```



A screenshot of a terminal window showing the execution of the Java program. The terminal shows the command to run the Java compiler and the resulting output.

```
PROBLEMS 1 OUTPUT DEBUG CONSOLE EXPLORER TERMINAL
```

```
PS C:\Users\ragur> & 'C:\Program Files\Java\jdk-25.0.2\bin\java.exe' -XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'src'
```

```
Area = 50
PS C:\Users\ragur>
```

**Program 5: Simple Interest Calculation Source**

**Code:**

```

class SimpleInterest { public static void
main(String[] args) { int p = 1000; int r
= 5; int t = 2;
int si = (p * r * t) / 100;
System.out.println("Simple Interest = " + si);
}
}

```

**Output:**

Simple Interest = 100

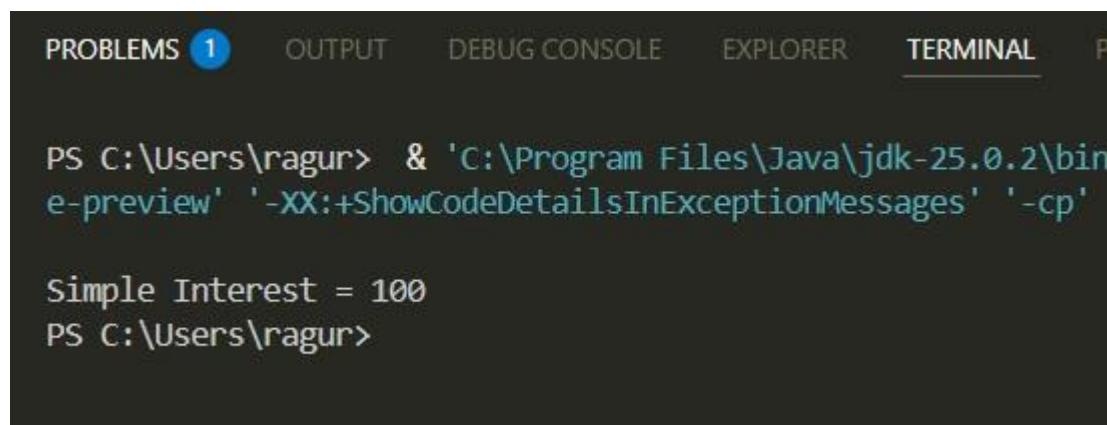


A screenshot of a code editor window titled "Untitled-1.java 1". The code is a Java program named "SimpleInterest" with a main method. It declares variables p, r, and t with values 1000, 5, and 2 respectively. It calculates the simple interest (si) as (p \* r \* t) / 100 and prints it to the console. The code editor has a dark theme and shows line numbers from 1 to 10.

```

1  class SimpleInterest {    Untitled-1.java is a non-project file, only syntax errors are reported
2      Run | Debug
3      public static void main(String[] args) {
4          int p = 1000;
5          int r = 5;
6          int t = 2;
7          int si = (p * r * t) / 100;
8          System.out.println("Simple Interest = " + si);
9      }
10 }

```



A screenshot of a terminal window showing the execution of the Java program. The terminal is running on Windows (PS C:\Users\ragur>) and uses the command "& 'C:\Program Files\Java\jdk-25.0.2\bin\java-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp'". The output shows the calculated simple interest: "Simple Interest = 100".

```

PS C:\Users\ragur> & 'C:\Program Files\Java\jdk-25.0.2\bin\java-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp'
Simple Interest = 100
PS C:\Users\ragur>

```

**POST LAB EXERCISE**

1. Write a Java program to display your name and department.

```

public class Main {
    public static void main(String[] args) {

```

```
        System.out.println("Charaneesh A P");

        System.out.println("AI & DS");

    }

}
```

2. Modify the program to print the output in same line.
- ```
public class Main {
```

```
    public static void main(String[] args) {

        System.out.print("Charaneesh A P ");

        System.out.print("AI & DS");

    }

}
```

3. What happens if main() is written without static?

The **program will not run** because,

- The **Java Virtual Machine** cannot call the main() method.
- Java shows an error: **Main method not found.**

4. Why is Java called platform independent?

- Java uses bytecode.
- Bytecode runs on any system with JVM.

5. Write a program to find the cube of a number.
- ```
public class Main {
```

```
    public static void main(String[] args) {      int num = 3;

        System.out.println(num * num * num);

    }

}
```

**Result:**

**Thus the Java IDE was successfully installed and a simple Java program was executed.**

**ASSESSMENT**

Description	Max Marks	Marks Awarded
Pre Lab Exercise	5	
In Lab Exercise	10	
Post Lab Exercise	5	
Viva	10	
<b>Total</b>	<b>30</b>	
<b>Faculty Signature</b>		