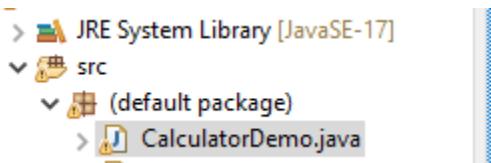


TASK 2

1. Project Structure



2. Source Code

```
import java.util.Scanner;  
  
/*
```

```
* Task 2: Variables, Data Types & Console Input  
*/
```

```
public class CalculatorDemo {  
  
    // Static variable (shared across all objects)  
  
    static String appName = "Console Calculator";
```

```
// Instance variable (belongs to object)  
  
int instanceCounter = 0;
```

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

```
// Local variables (inside method)
```

```
byte b = 10;      // 1 byte
```

```
short s = 100;    // 2 bytes
```

```
int i = 1000;     // 4 bytes
```

```
long l = 100000L; // 8 bytes
```

```
float f = 10.5f;      // 4 bytes  
double d = 99.99;    // 8 bytes  
char c = 'A';       // 2 bytes  
boolean flag = true; // 1 bit (logical)
```

```
System.out.println("==== " + appName + " ===");
```

```
// Scanner for console input  
Scanner sc = new Scanner(System.in);
```

```
// Accepting user input  
System.out.print("Enter first number: ");  
double num1 = sc.nextDouble();
```

```
System.out.print("Enter second number: ");
```

```
double num2 = sc.nextDouble();
```

```
System.out.print("Choose operation (+ - * /): ");
```

```
char operator = sc.next().charAt(0);
```

```
double result = 0;
```

```
boolean valid = true;
```

```
// Arithmetic operations with validation
```

```
switch (operator) {
```

```
case '+':  
    result = num1 + num2;  
break;
```

```
case '-':  
    result = num1 - num2;  
break;
```

```
case '*':  
    result = num1 * num2;  
break;
```

```
case '/':  
    if (num2 != 0) {  
        result = num1 / num2;  
    } else {  
        System.out.println("Error: Division by zero not allowed.");  
        valid = false;  
    }  
break;
```

```
default:  
    System.out.println("Invalid operator selected.");  
    valid = false;  
}
```

```
// Formatted output

if (valid) {

    System.out.printf("Result: %.2f %c %.2f = %.2f\n",
        num1, operator, num2, result);

}

// Type casting examples

int castedInt = (int) d; // Explicit casting

double autoCast = i; // Implicit casting

System.out.println("\nType Casting Examples:");

System.out.println("Double to int: " + castedInt);

System.out.println("Int to double: " + autoCast);

sc.close();

}
```

3.Console Input & Output

The screenshot shows the Eclipse IDE interface with the following details:

- Project Explorer:** Shows multiple Java files: HelloService..., HelloControl..., GoodbyeService..., GoodbyeController..., application..., EurekaService..., HelloJava.java, and CalculatorDemo.java.
- CalculatorDemo.java Content:**

```
26
27     System.out.println("==> " + appName + " ==>");
28
29     // Scanner for console input
30     Scanner sc = new Scanner(System.in);
31
32     // Accepting user input
33     System.out.print("Enter first number: ");
34     double num1 = sc.nextDouble();
35
36     System.out.print("Enter second number: ");
37     double num2 = sc.nextDouble();
38
39     System.out.print("Choose operation (+ - * /): ");
40     char operator = sc.next().charAt(0);
41
42     double result = 0;
43     boolean valid = true;
```
- Console Output:**

```
<terminated> CalculatorDemo [Java Application] C:\Users\Rathod\AppData\Local\Programs\Eclipse Adoptium\jdk-17.0.9.9-hotspot\bin\javaw.exe [16-Jan-2026, 6:26:16 pm - 6:26:29 pm] [pid: 10780]
*** Console Calculator ***
Enter first number: 3
Enter second number: 4
Choose operation (+ - * /): +
Result: 3.00 + 4.00 = 7.00

Type Casting Examples:
Double to int: 99
Int to double: 1000.0
```
- Bottom Status Bar:** Shows "Writable", "Smart Insert", and the current time "43:30:1287".