**Programs on Overloading, Recursion, and Access Specifier**

1. Write a program to calculate the area of rectangle, square, and circle. Use Method Overloading for demonstration. Create three area methods with each describing the geometric figures.

**Logic:**

Step 1: Start

Step 2: package welcome{

Step 3: We have to first of all create a class area\_pr.

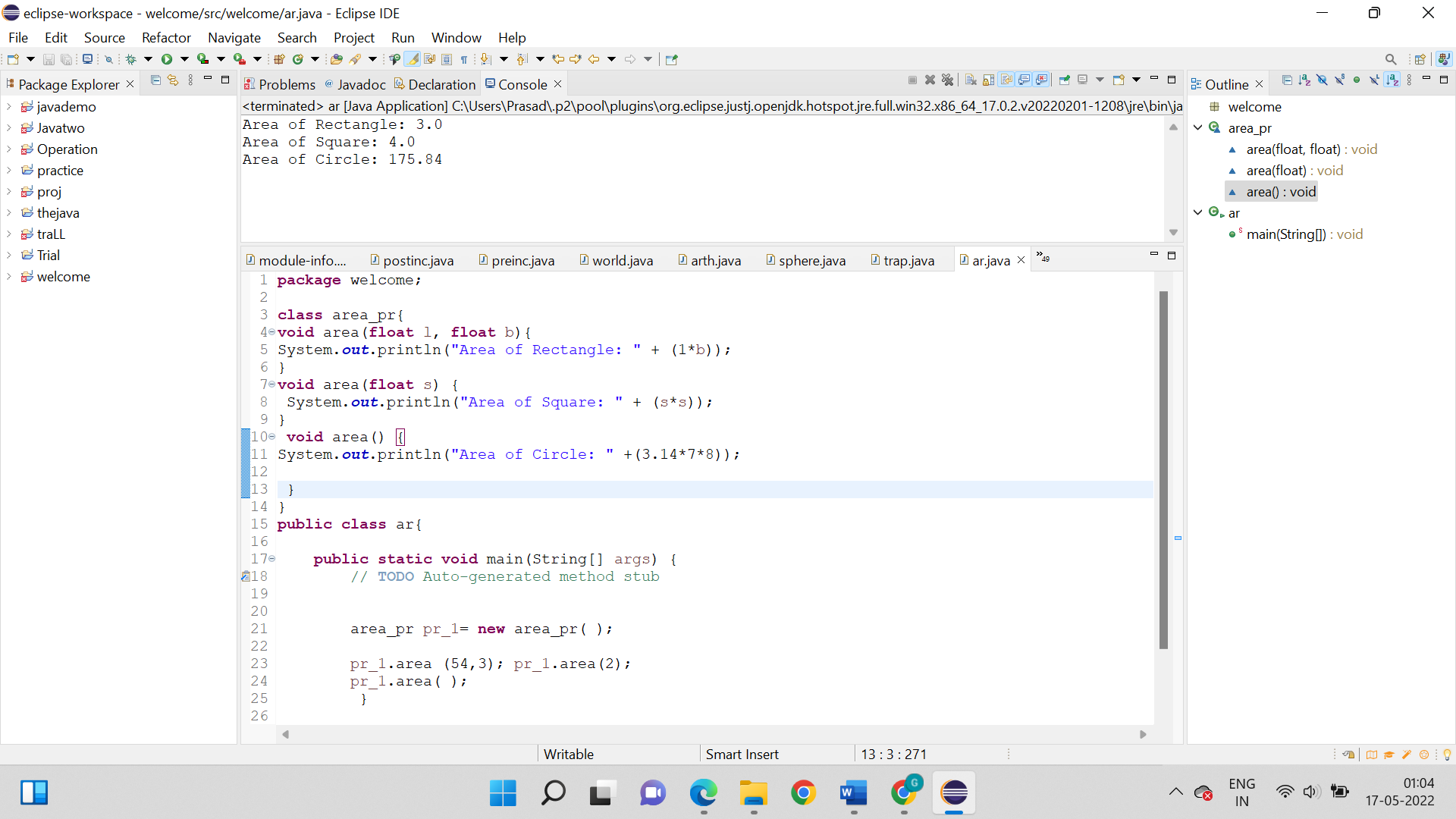
Step 4: Then create different methods for area of different geometric shapes, with same name

Step 5: Also, every class upon calling will return some value or output.

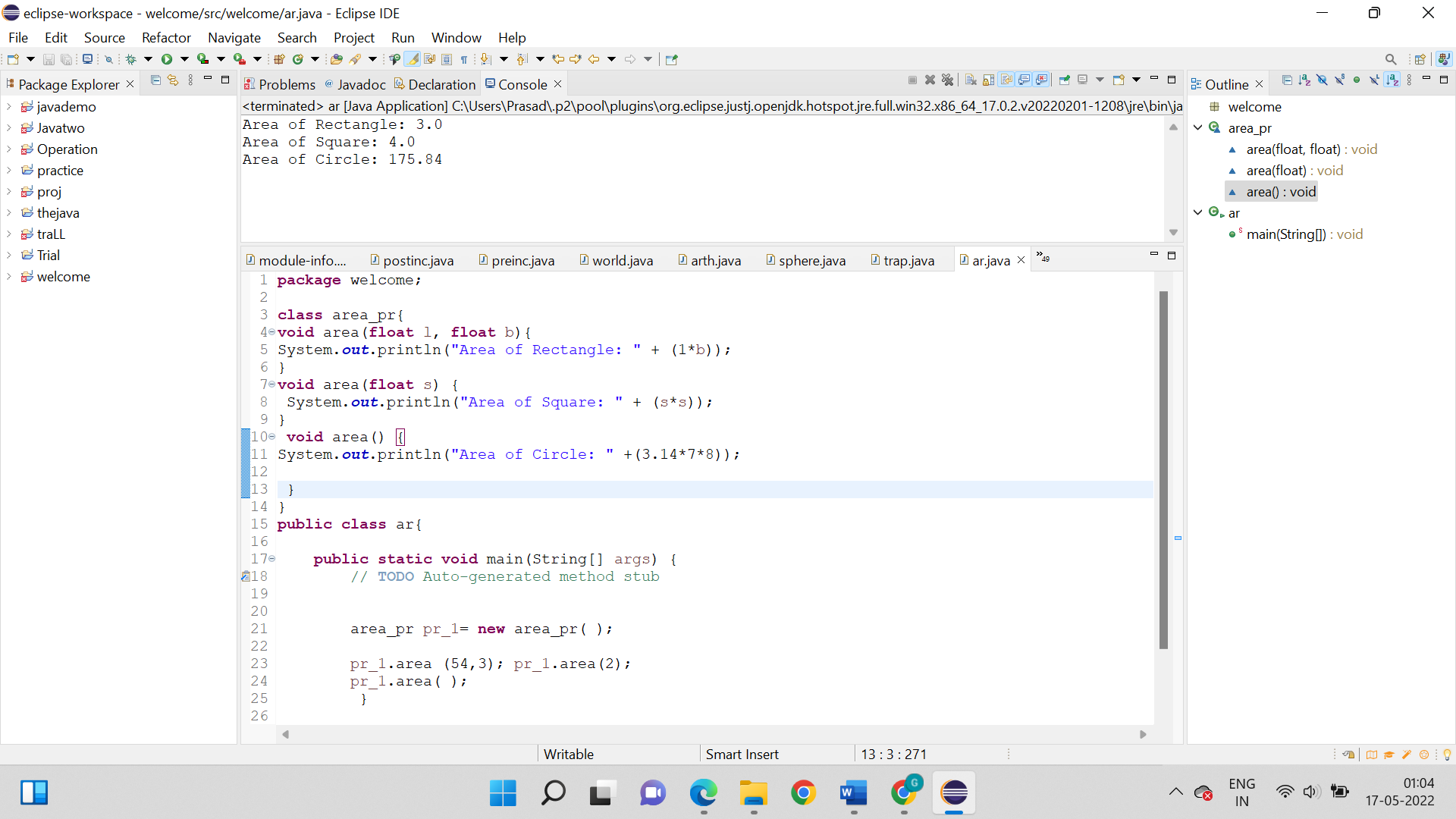
Step 6: Then in the main class call the methods in the class and pass values to the class

Step 7: Stop

Code:



**Output:**



1. Write a program to calculate the hypotenuse and area of a triangle. Use Method Overloading to create two hypotenuse and three area methods for demonstration.

**Logic:**

Step 1: Start

Step 2: package welcome; class funct\_1;

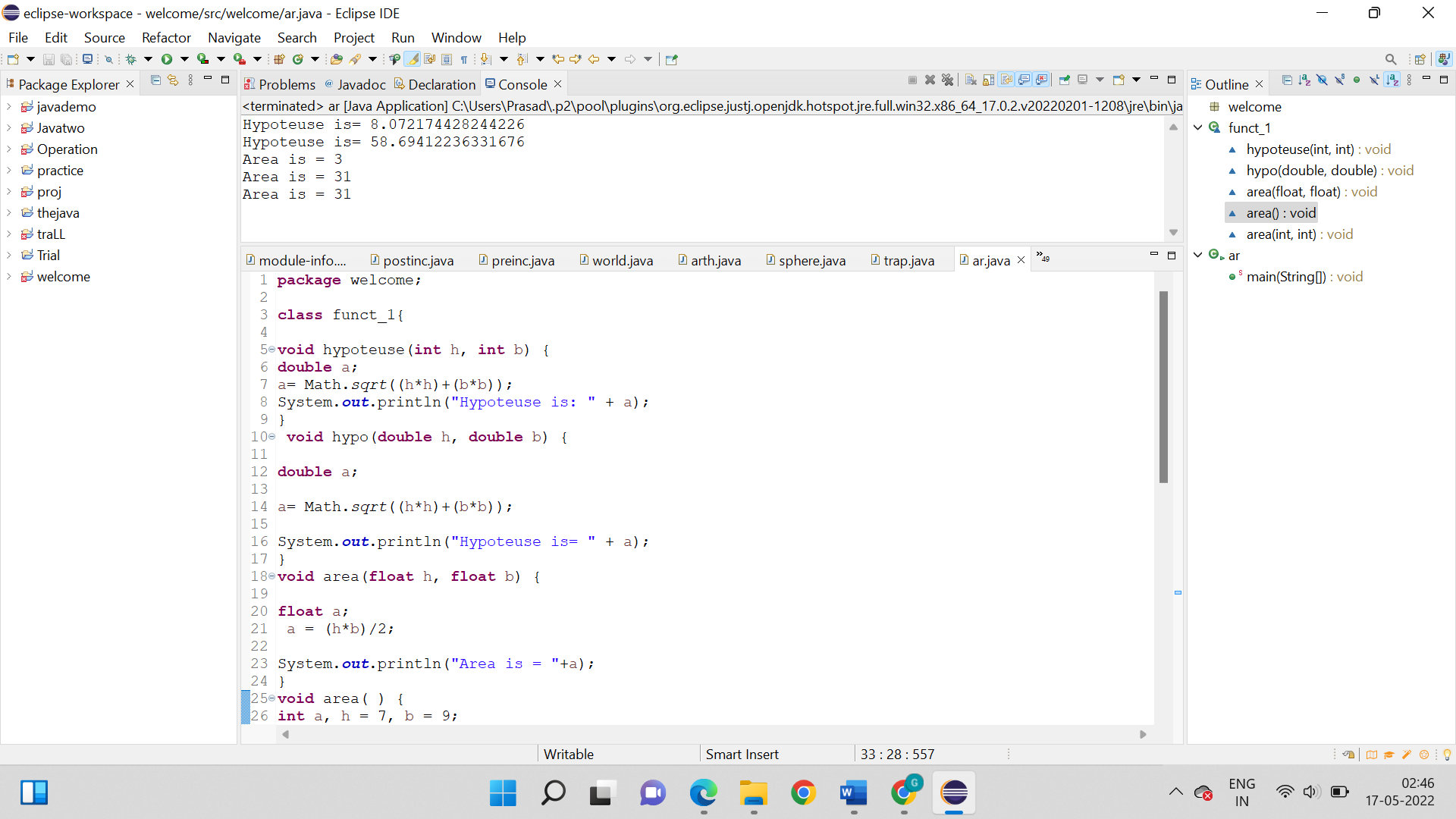
Step 3: Now we have to create a class function and create two hypotenuse and three area methods with same name.

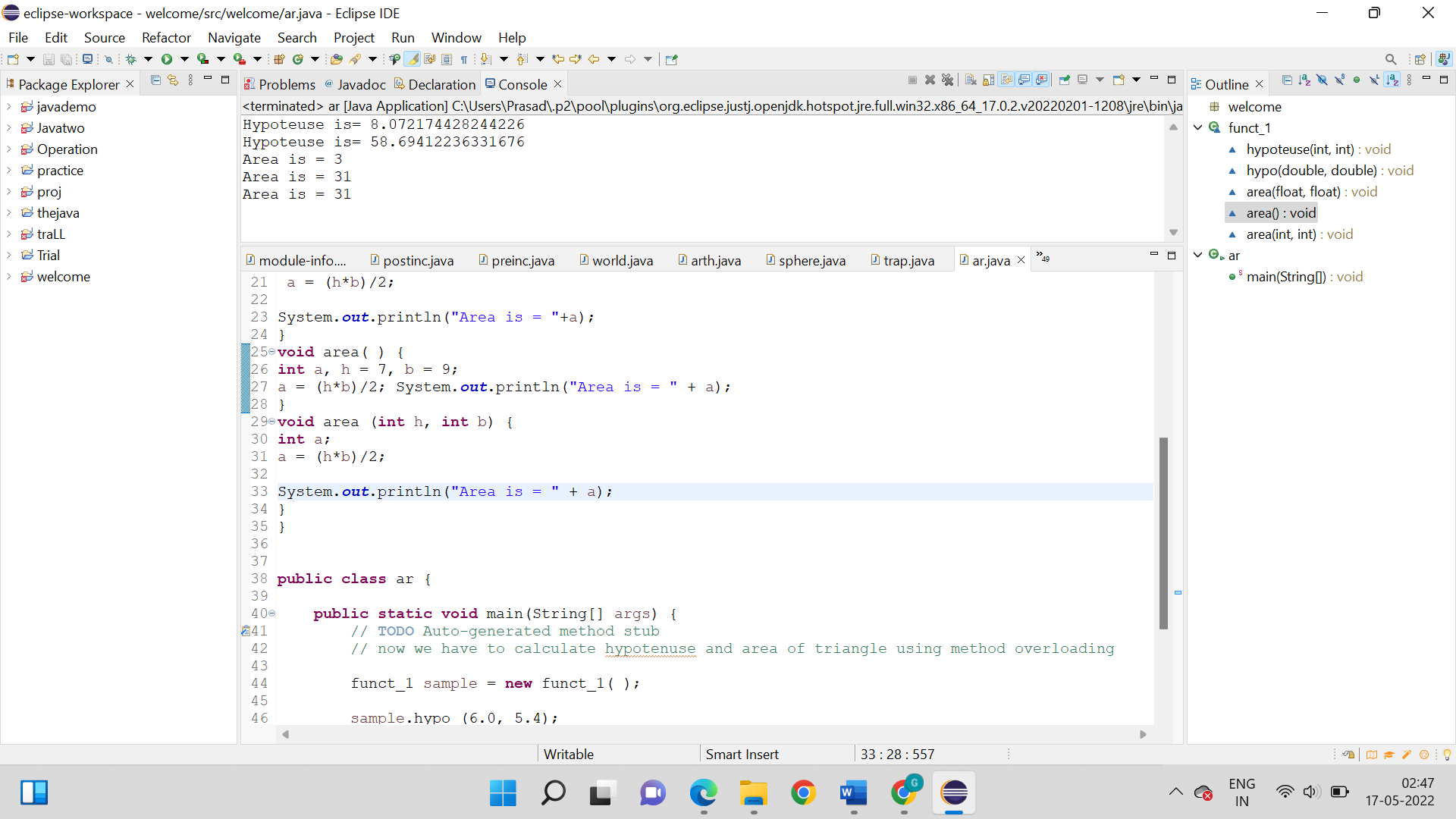
Step 4: We know that every method will contain the respective formula and will print result simultaneously.

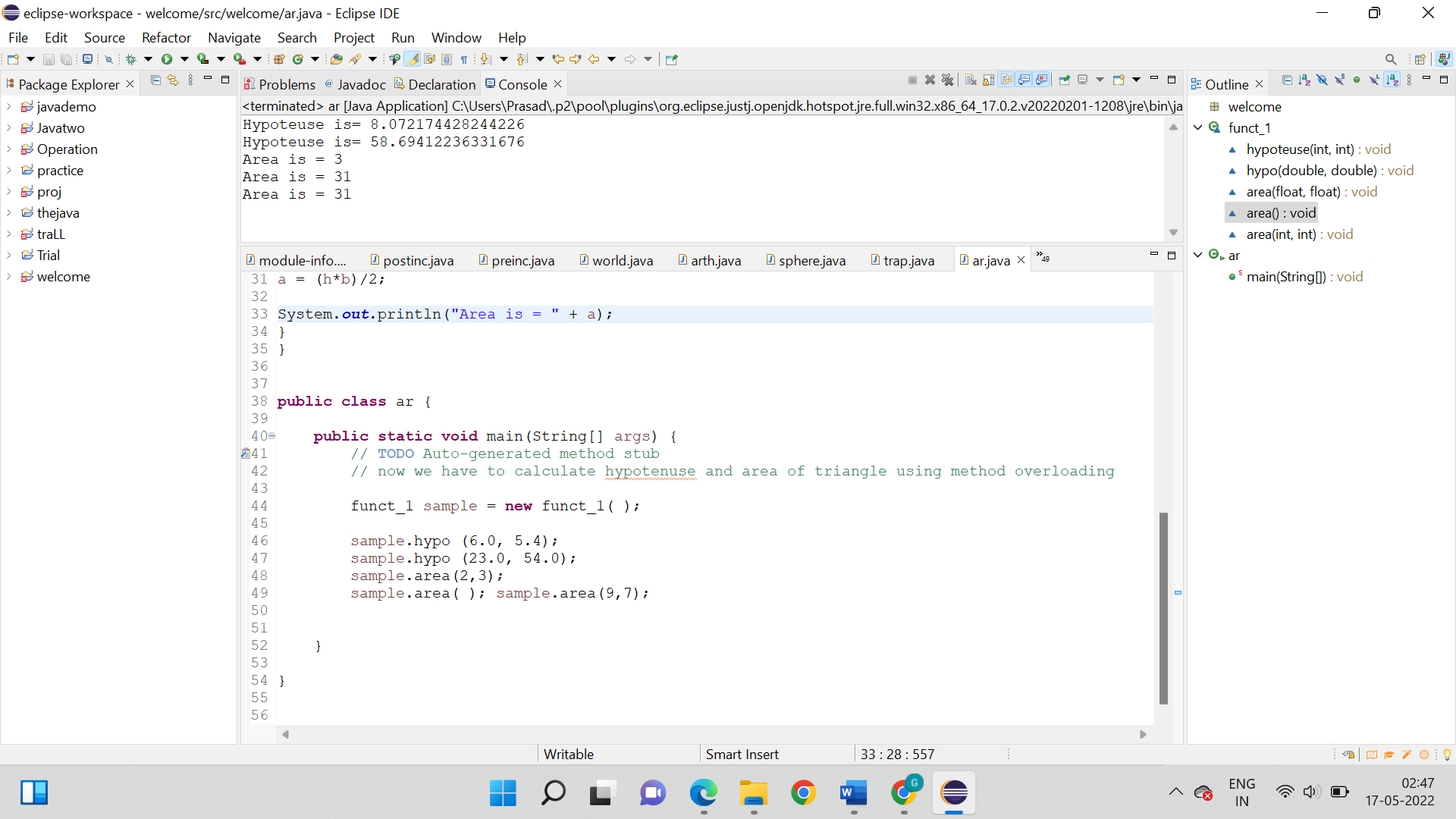
Step 5: So, then in the main class call for the class function and pass the values to the methods in the class

Step 7: Stop

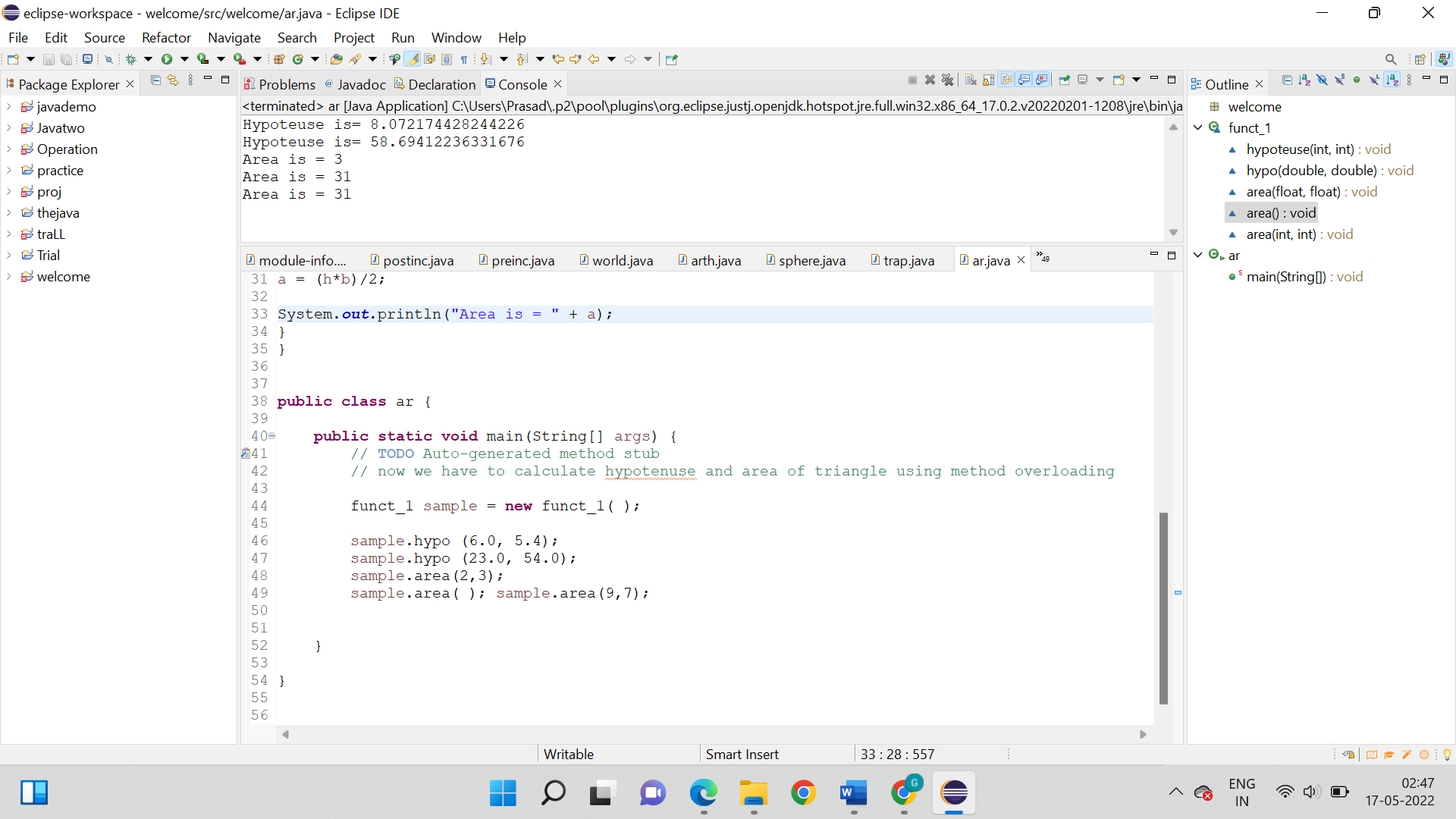
**Code:**







**Output:**



1. Write a program to calculate the cube of a number to demonstrate constructor overloading. Use three constructors for the demonstration.

**Logic:**

Step 1: Start

Step 2: package welcome{

Step 3: Create a class cubee and than declare required variables.

Step 4: Now we have to declare 3 constructors with same name.

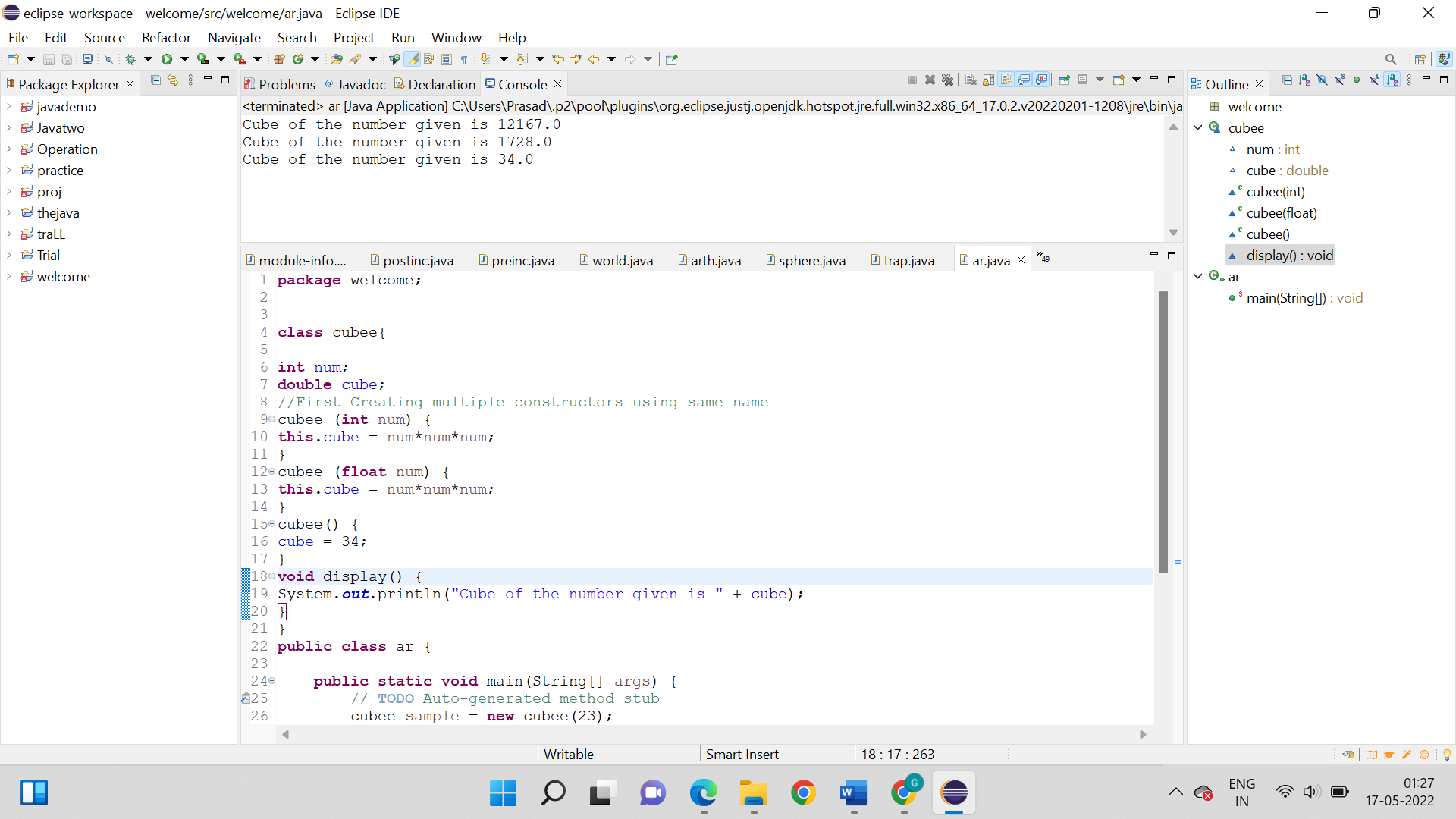
Step 5: Than declare a method that will print result.

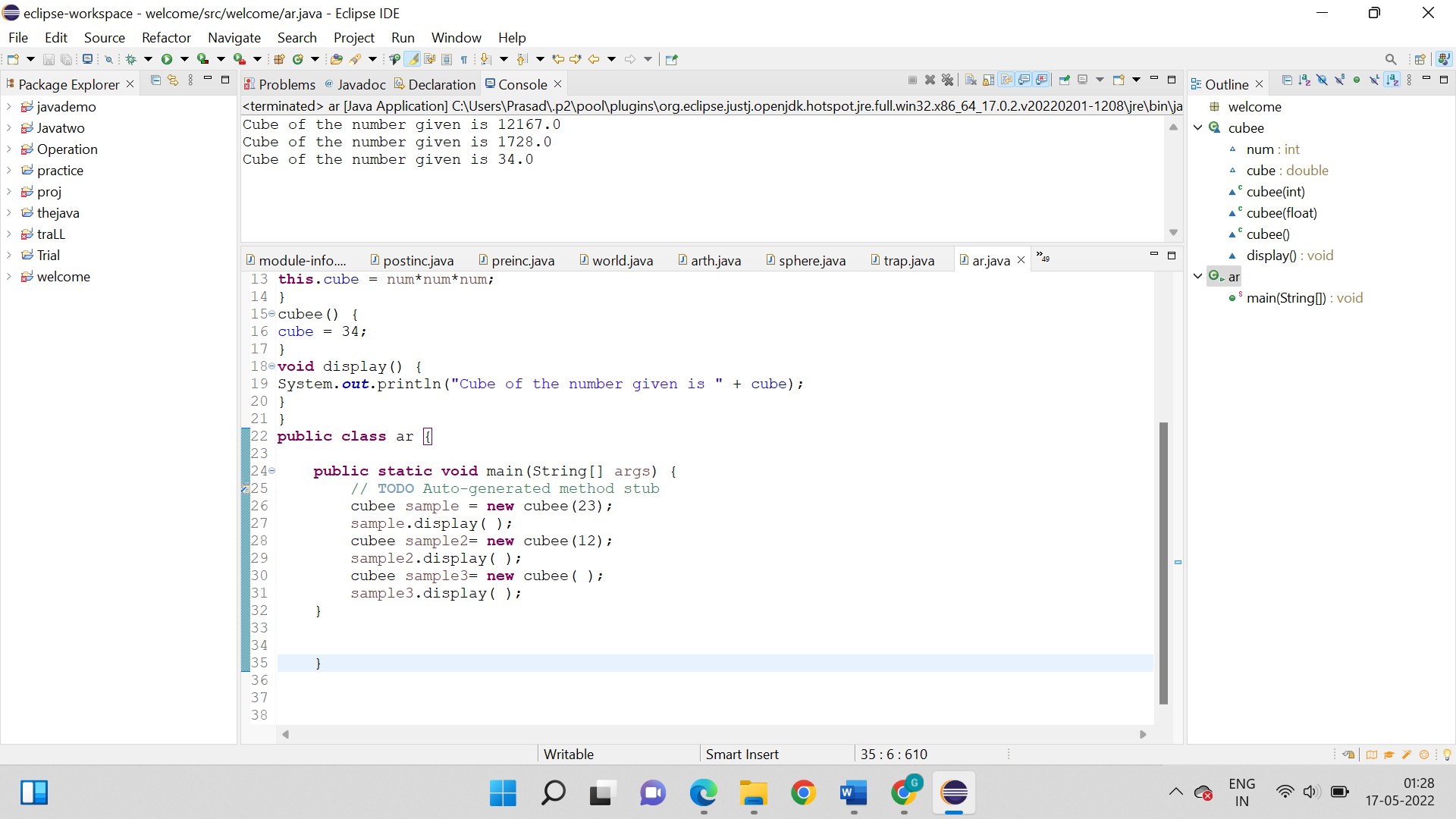
Step 6: We have to now call for the class cube in main and transfer value to the constructor.

Step 7: So, basically the result will be showed, the constructor then will calculate the cube and then calling the display method will print the result.

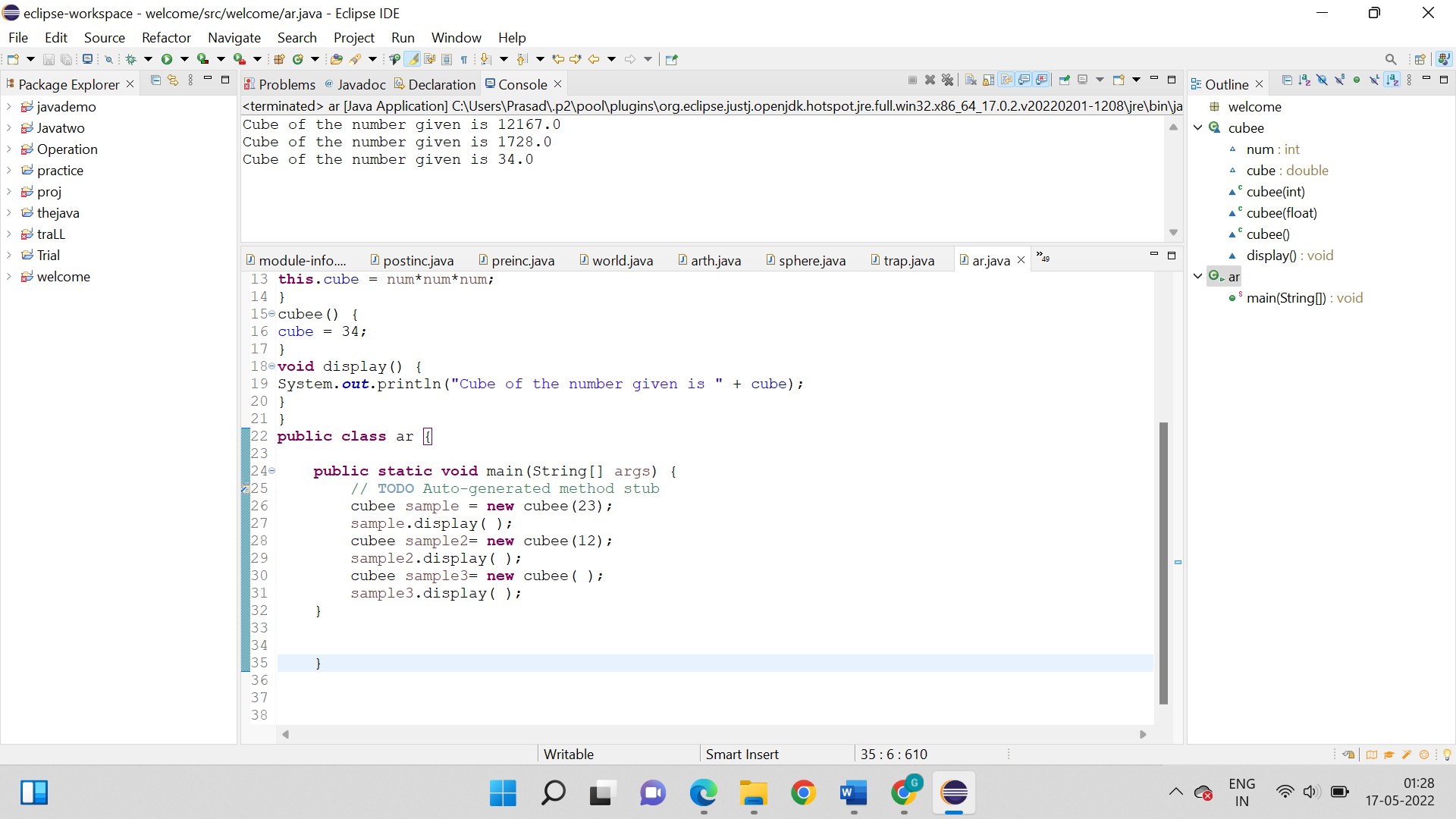
Step 8: Stop

**Code:**





**Output:**



1. Write a program to demonstrate the recursion function, to calculate the addition of a given integer.

**Logic:**

Step 1: Start

Step 2: package welcome; public class ar;

Step 3: First we have to declare a public class and inside it declare two methods.

Step 4: Now we have to first method will be static recursion method that will receive the number to put in recursion.

Step 5: So, actually here in the recursion function use if…else operator until the number is 0. And than add the number to itself and pass it again to same function subtracting it by 1 again and again until it becomes 0.

Step 6: Now, as soon as the number is 0 return the number to the main method.

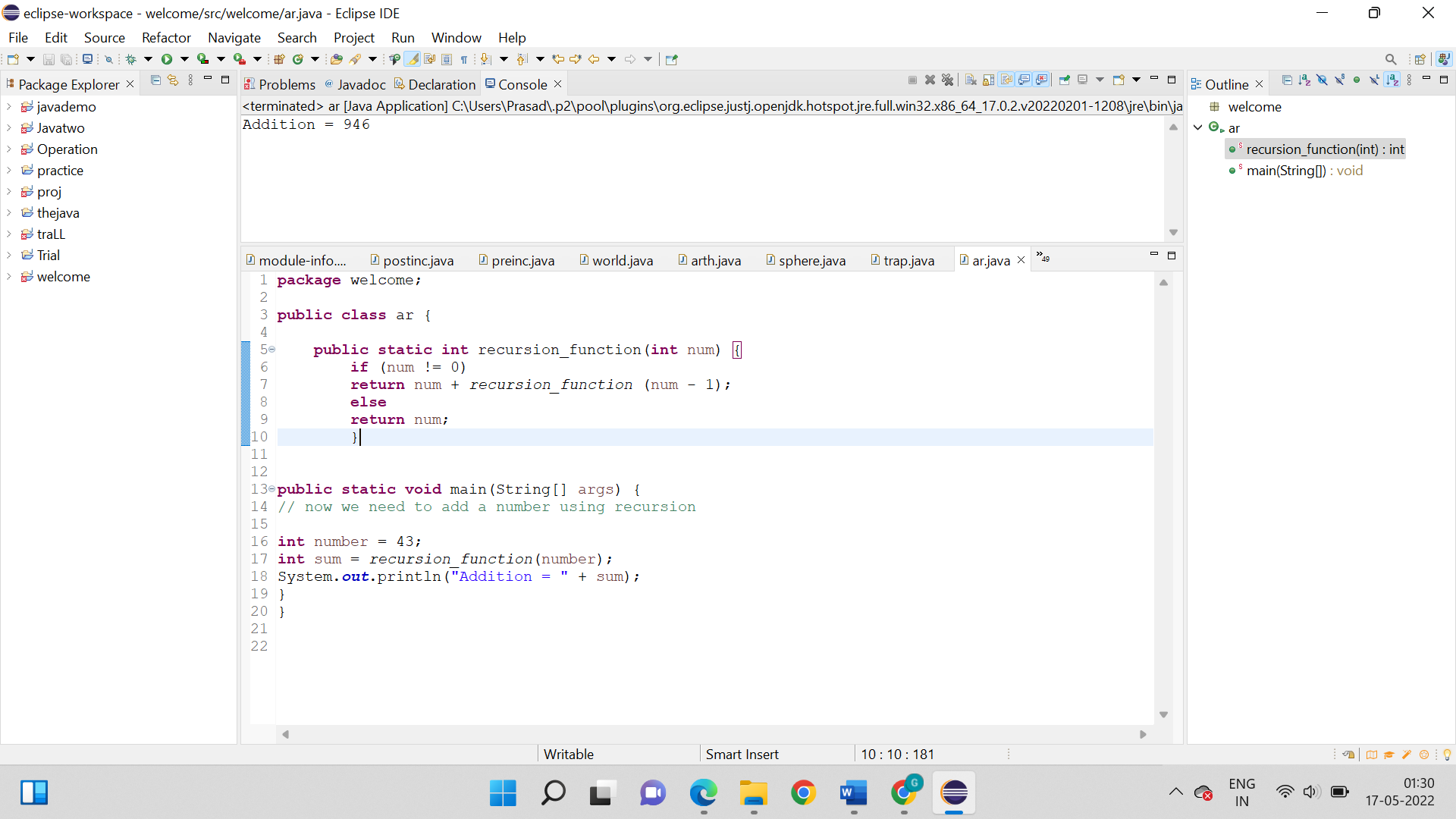
Step 7: Now just in the main method declare a number.

Step 8: Now we have to declare another variable that will transfer the number to the recursion function.

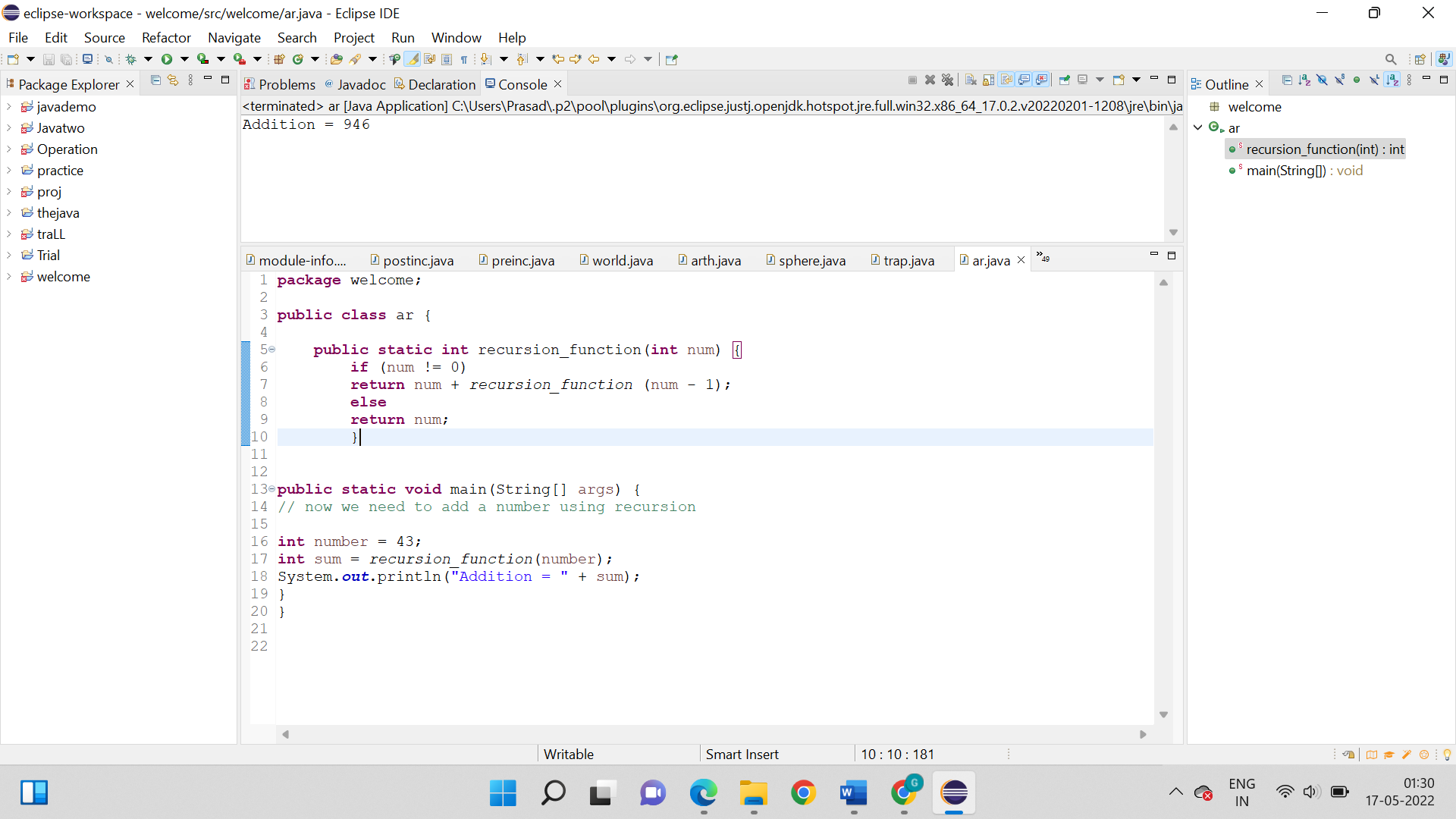
Step 9: Print the result

Step 10: Stop

**Code:**



**Output:**



1. Write a program to use the recursion function, to calculate the Fibonacci series of a given number.

**Logic:**

Step 1: Start

Step 2: package welcome; public class ar{

Step 3: First we have to declare a java class and inside it declare two methods.

Step 4: Now just declare all required variables to solve the program.

Step 5: We have to then declare method for counting the Fibonacci number upto a given number.

Step 6: We know that this method will receive a number, so then inside the method using if operator which

will run until the number is greater than 0, all the required equations are written respectively.

Step 7: We know that before closing the if operator, the number will be deducted and passed into

the function until the number is not 0.

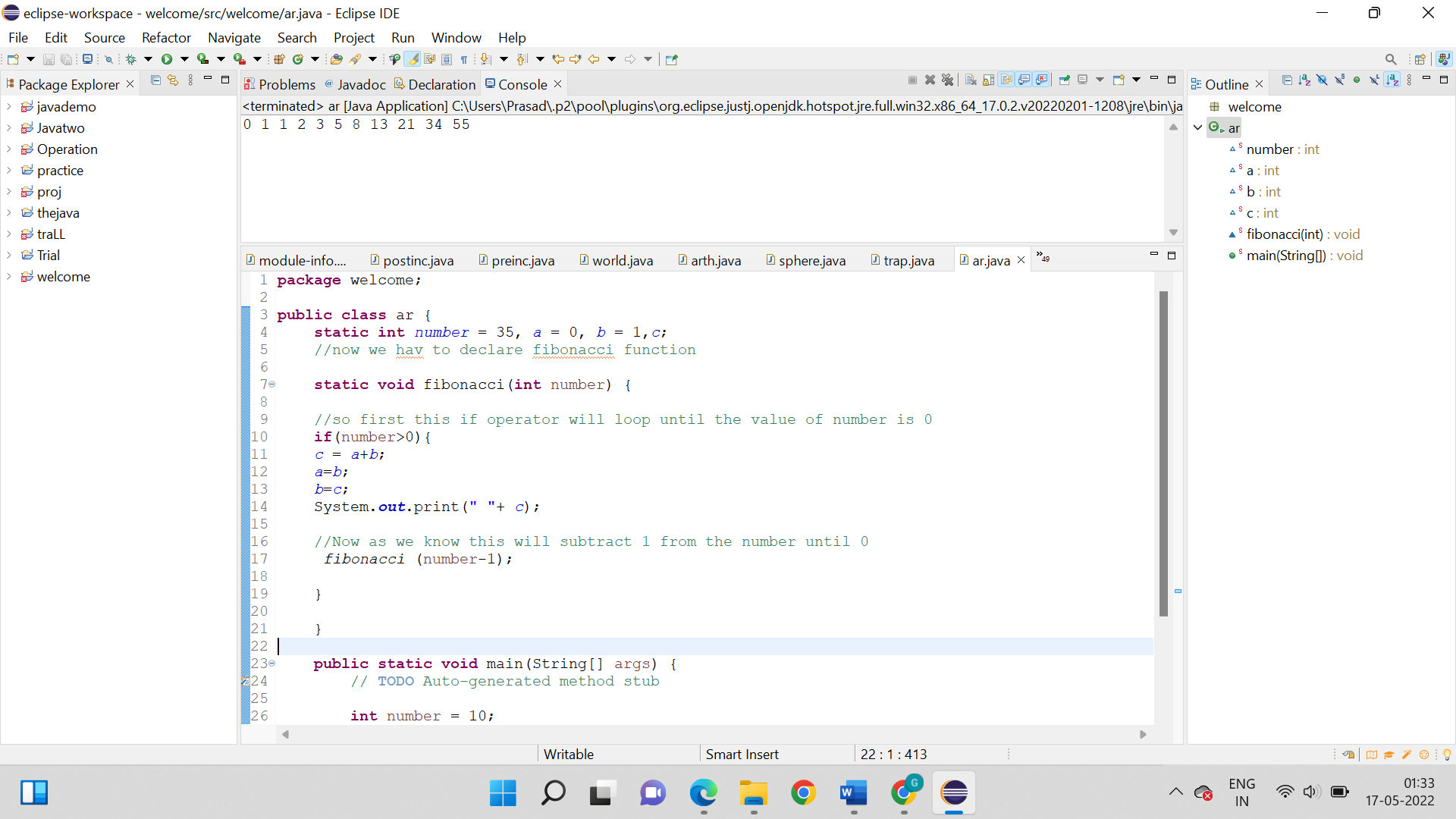
Step 8: So, as we know that this will method will also print the result simultaneously after every run.

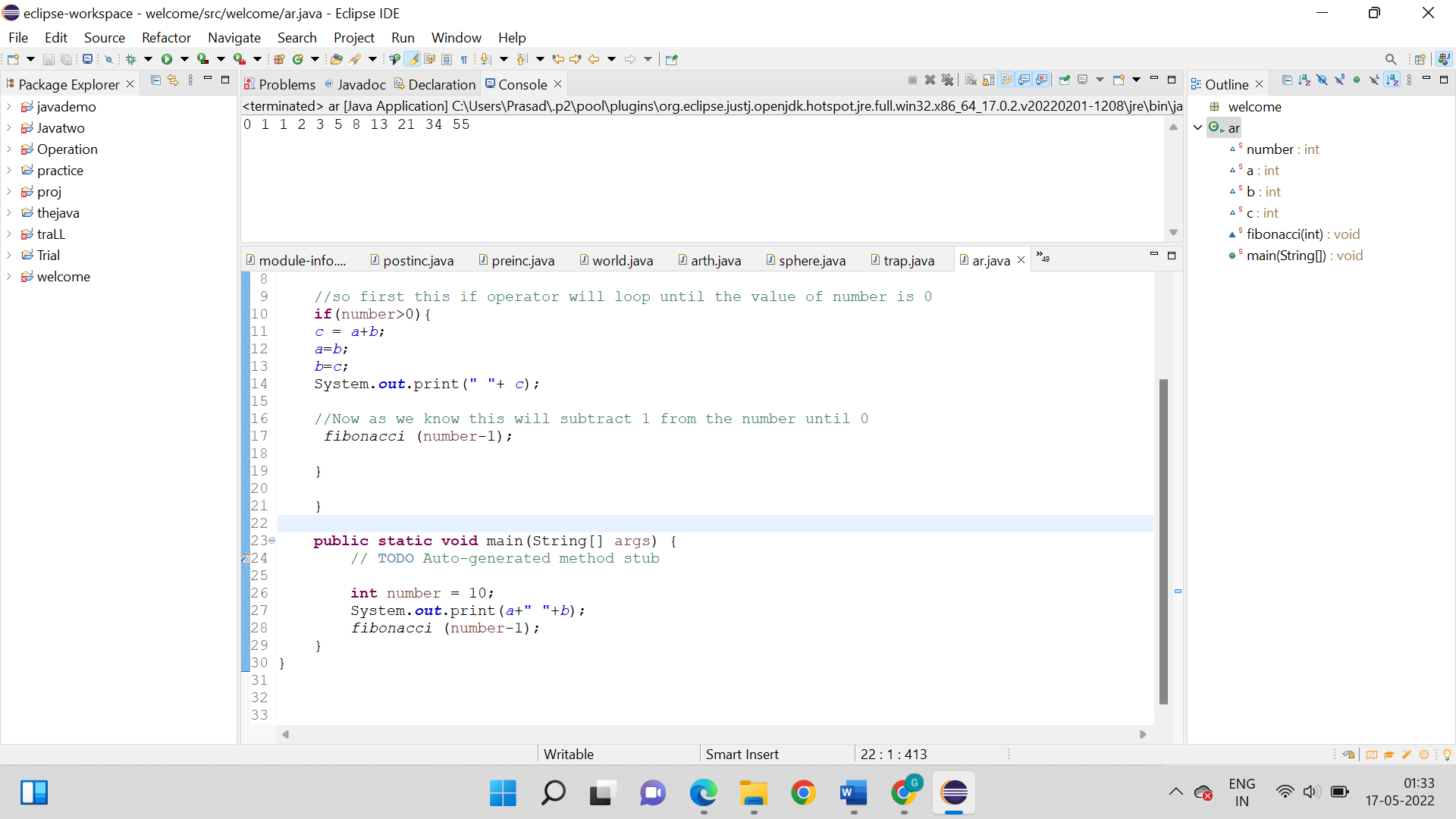
Step 9: Now, we can finally in the main method we will declare the number, print the first two numbers

and pass the number deducted by one to the function.

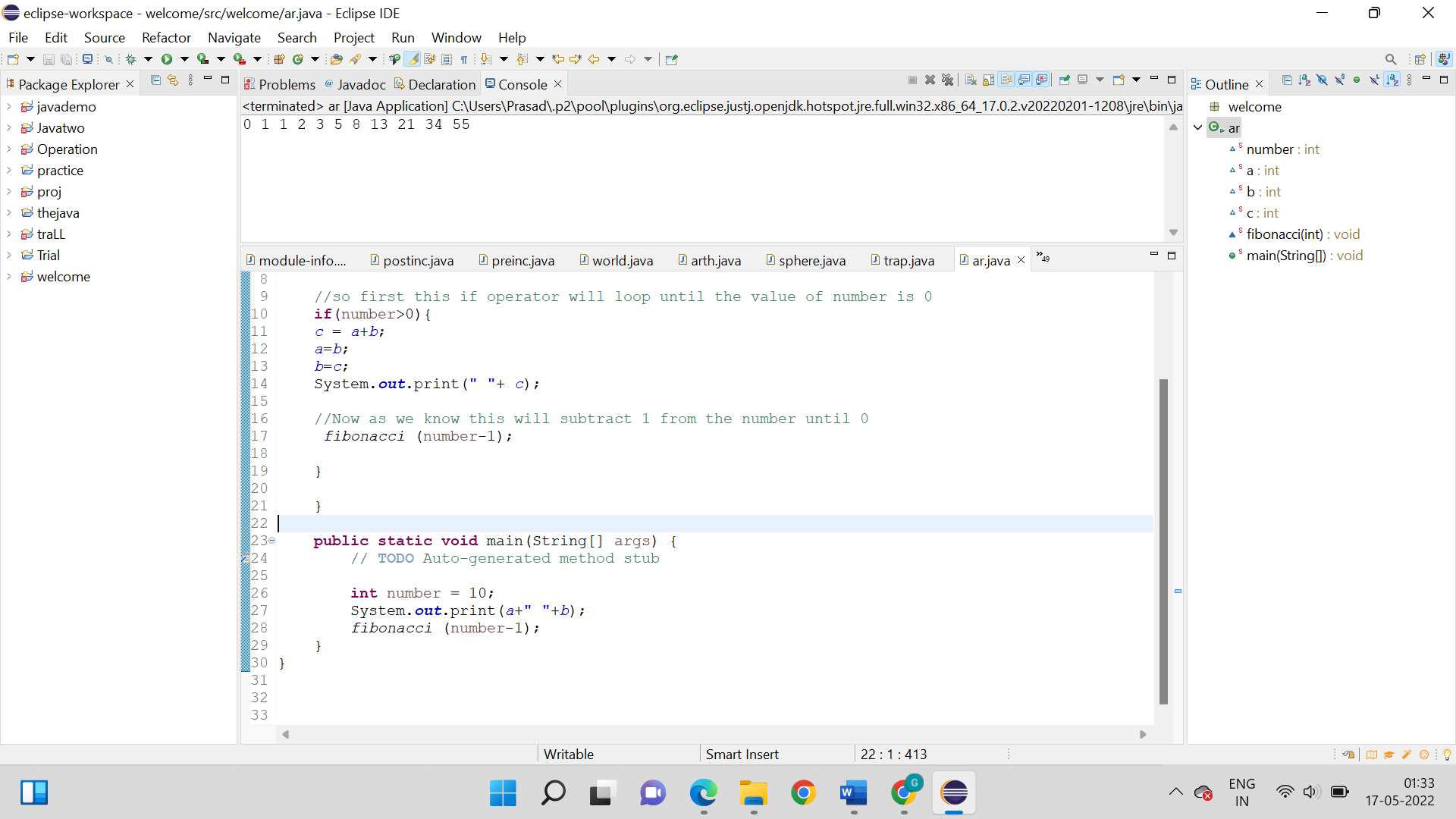
Step 10: Stop

**Code:**





**Output:**



1. Write a program to demonstrate the use of two access specifiers “public and private” to calculate the area and volume of a cylinder.

**Logic:**

Step 1: Start

Step 2: package welcome;

Step 3: class function{

Step 4: So, first we will declare public and private variables. Here pi is public variable and radius and hypotenuse are private variables

Step 5: Just create a method that will hold two variables.

Step 6: But now we have to again, create a method that contains the formula of area of circle and will return the value.

Step 7: So, just create another variable, which has formula for volume and returns the value.

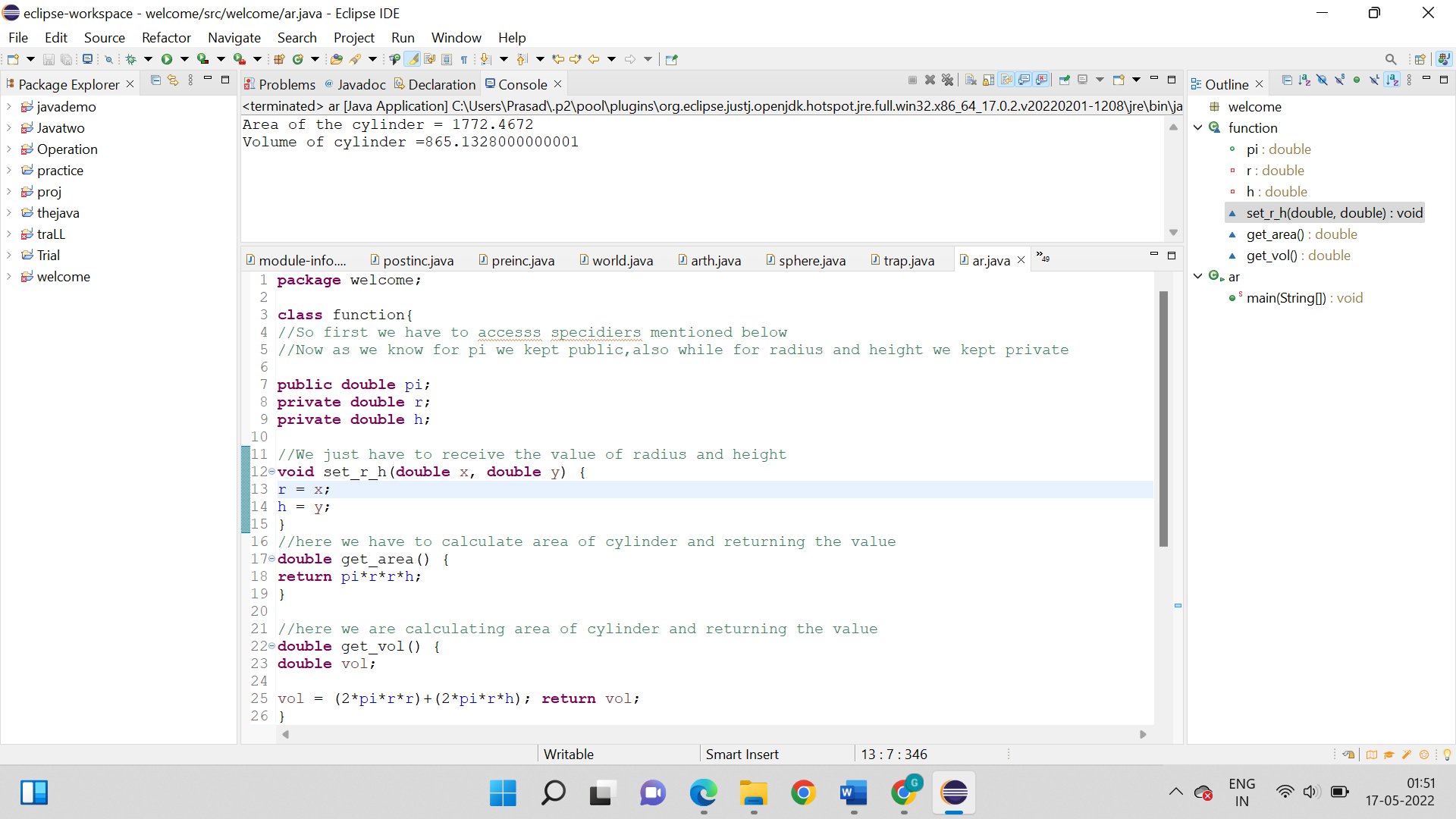
Step 8: We have to then declare a main class where we call for the class function.

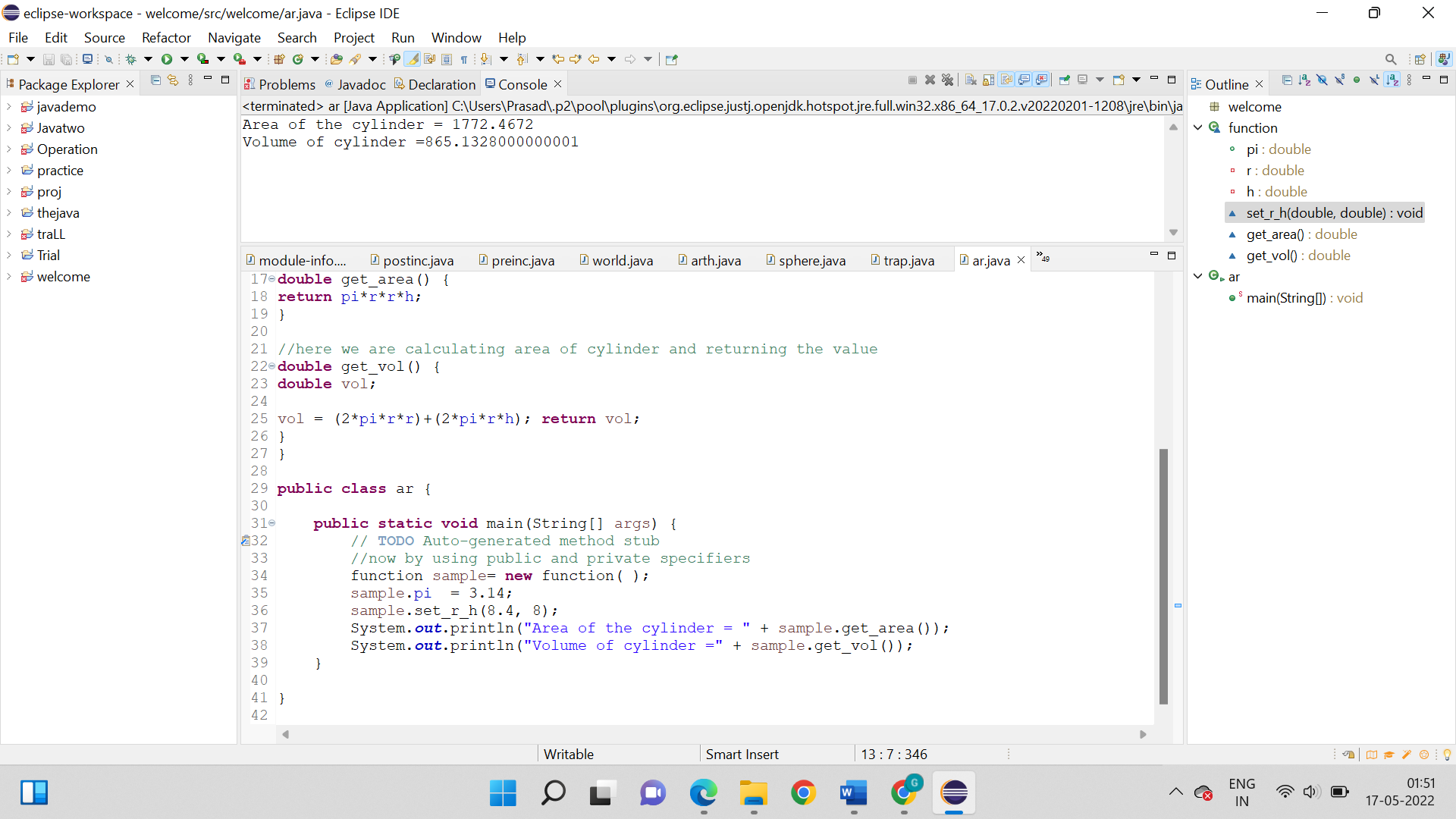
Step 9: Now just one by one we transfer all the values to the method.

Step 10: So, print the results by calling out the method.

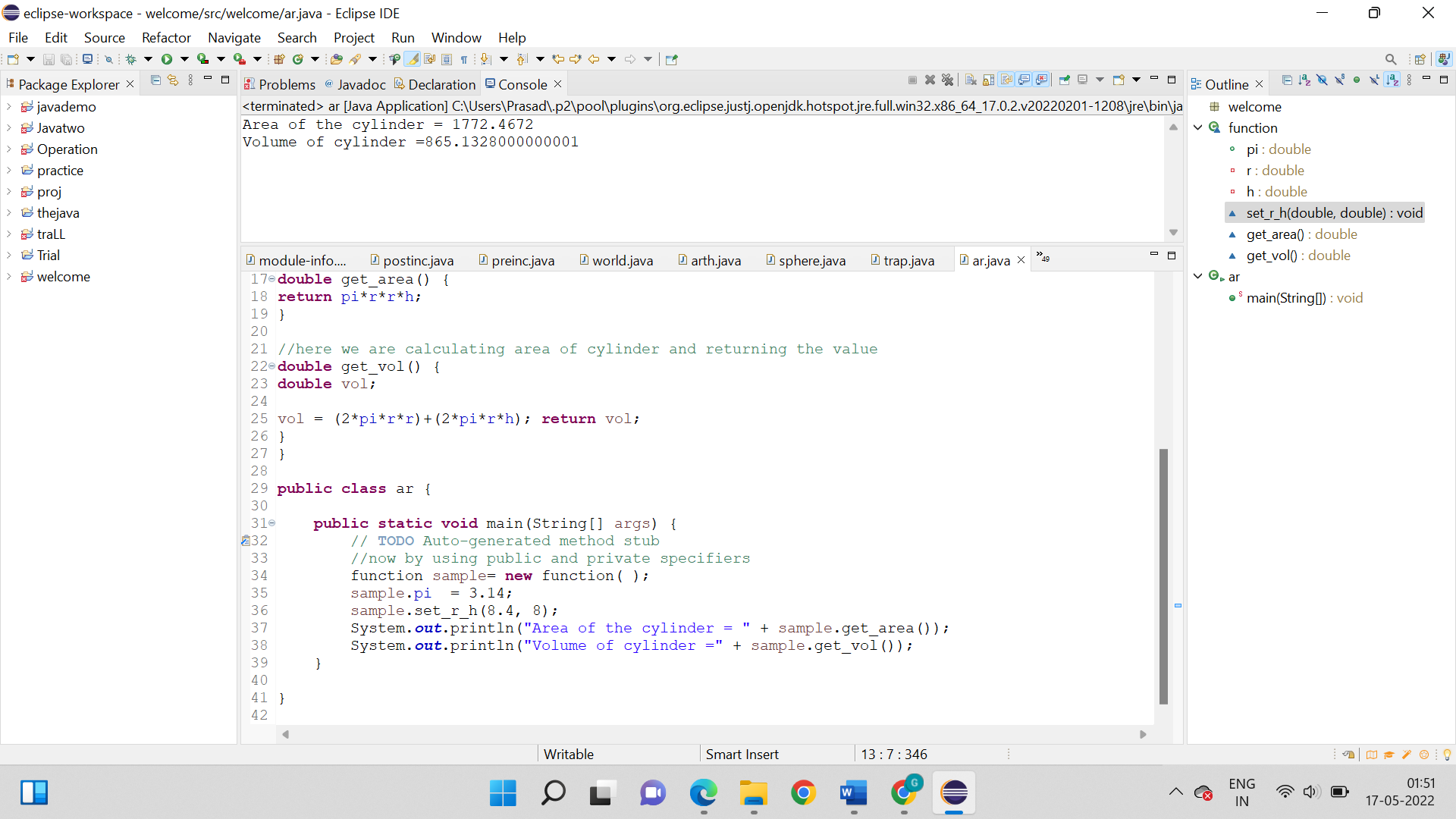
Step 11: Stop

**Code:**





**Output:**



1. Write a program to demonstrate the use of the Static keyword.

**Logic:**

Step 1: Start

Step 2: package welcome;

Step 3: class ar{

Step 4:So, now inside class declare static integers.

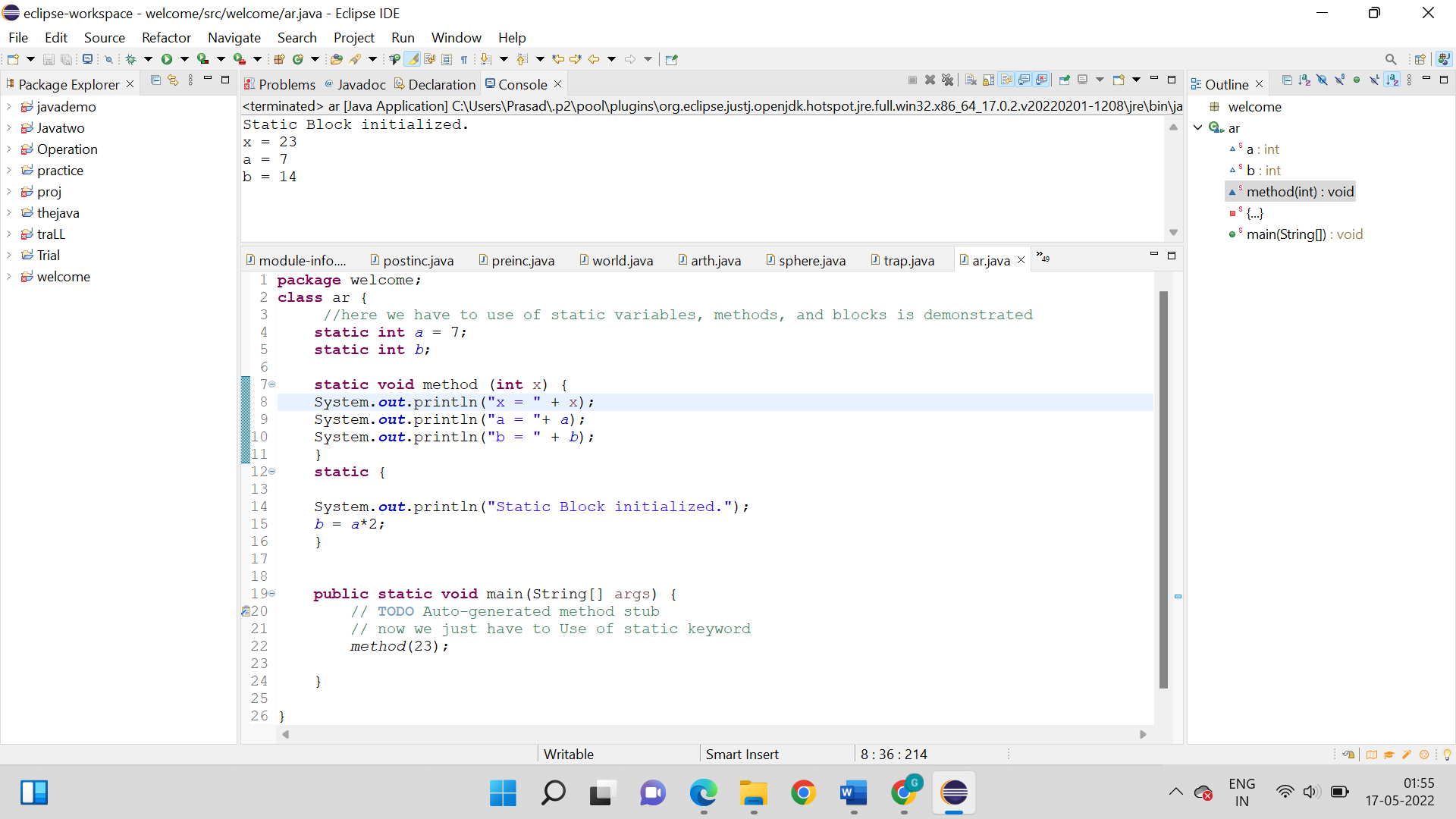
Step 5:So, basically we also have to declare static method that will receive a value from the main class.

Step 6: So after that we have to declare an empty static method that will be the initialization of the code and will multiply the declared variables.

Step 7: So, now finally in the main method, transfer the value to the other method.

Step 8: Stop

**Code:**



**Output:**

