Java PROGRAMMING LAB

BCA-DS-402

Manav Rachna International Institute of Research and Studies

School of Computer Applications

Department of Computer Applications

Submitted By	
Student Name	Kritika Chopra
Roll No	23/SCA/BCA(AI&ML)/027
Program	Bachelor of Computer Applications
Semester	4
Section/Group	4C
Department	Computer Applications
Batch	2023-26
Submitted To	
Faculty Name	Mrs. Priyanka Sharma

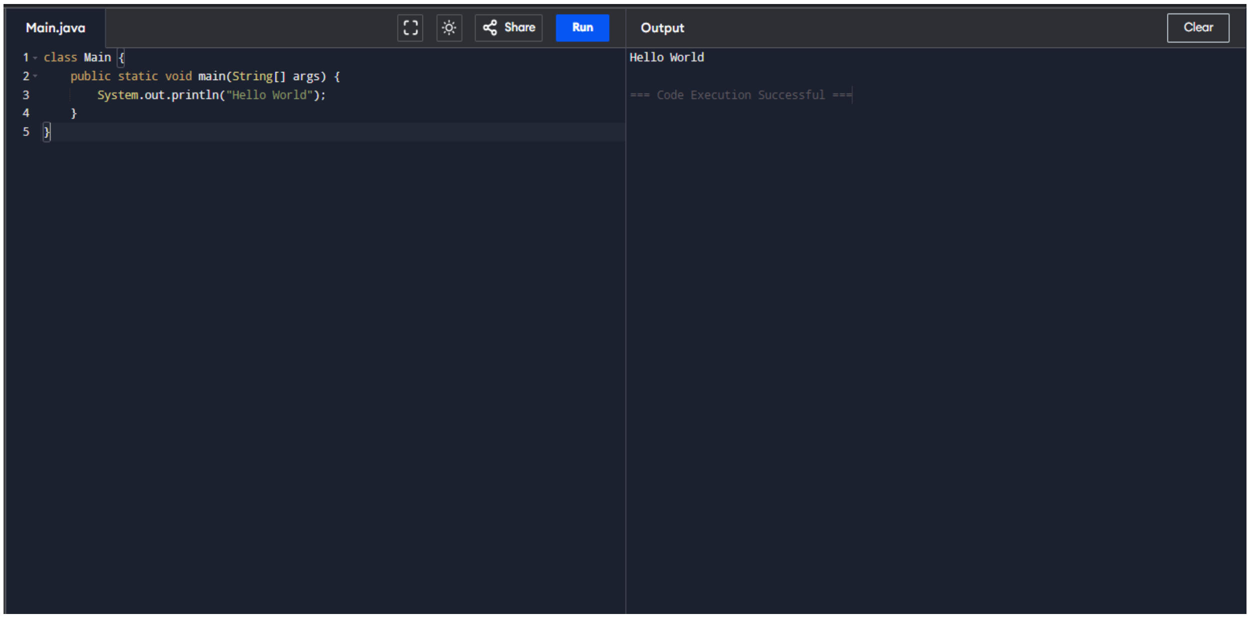
MANAV RACHNA
||vidyanatariksha||
Manav Rachna International
Institute of Research and Studies
Deemed to be University under section
3 of the UGC Act, 1956

SCHOOL OF COMPUTER APPLICATIONS

S.No	Program name	date
	LAB SESSION 1	
1.	Write java program to print hello world	
2.	Java Program to take input from user and print the sum of two numbers	
3.	Create a java program to check whether a number entered by user is even or odd	
4.	Create a java program to print the average and sum of 5 numbers entered by user.	
5.	Program to calculate the factorial of a number	
6.	Program to print Fibonacci series up to n terms	
	LAB SESSION 2	
1.	Program to reverse a number	
2.	Write a program to find the average and sum of the N numbers using Command line argument.	
3.	Program to check if a number is a palindrome	
4.	Program for a simple calculator	
5.	Program to check if a number is prime	
6.	Program to check if a number is an Armstrong number	
	LAB SESSION 3	
1.	Print Multiplication Table	
2.	Calculate Sum and Average of Array Elements	
3.	Reverse a String	
4.	Find Factorial of a Number Using Recursion	
5.	Sort an Array in Ascending Order	
	LAB SESSION 4	
1.	Check Palindrome for a String	
2.	Count Vowels and Consonants in a String	
3.	Implement a Simple Banking System	
4.	Write a program to demonstrate type casting.	
5.	Write a program to generate prime numbers between 1 & given number	
	LAB SESSION 5	
1	Program to Demonstrate a Simple Class with Methods	
2	Program for Class with Parameterized Constructor	
3	Program to Find the Area of a Rectangle Using Methods	
4	Program for Bank Account Class with Deposit and Withdraw Methods	
5	Program to Demonstrate Method Overloading	
	LAB SESSION 6	
1	Program to Demonstrate Static Methods	
2	Program to Demonstrate Method Overriding	
3	Program to Demonstrate Getters and Setters	

4	Program to Demonstrate a Class with Multiple Methods	
5	Program to Demonstrate Object Passing in Methods	
6	Write a program to create a simple class to find out the area and perimeter of rectangle using super and this keyword.	
7	Write a program to count the number of objects created for a class using static member function	

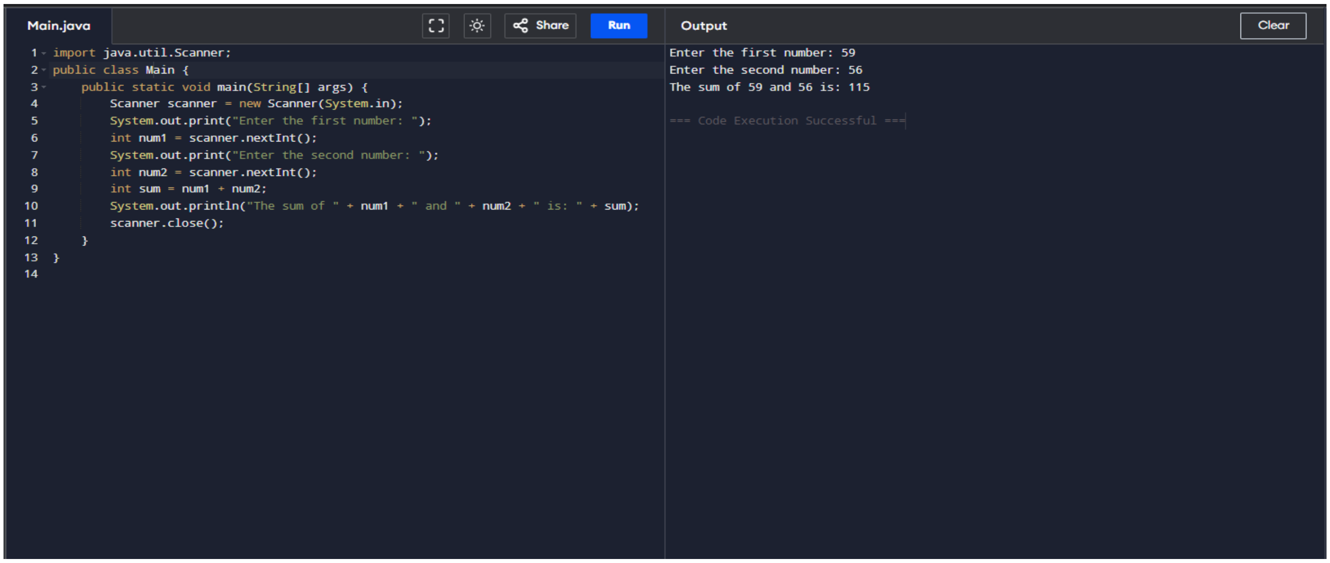
- **Write java program to print hello world**



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

```
Hello World

=== Code Execution Successful ===
```

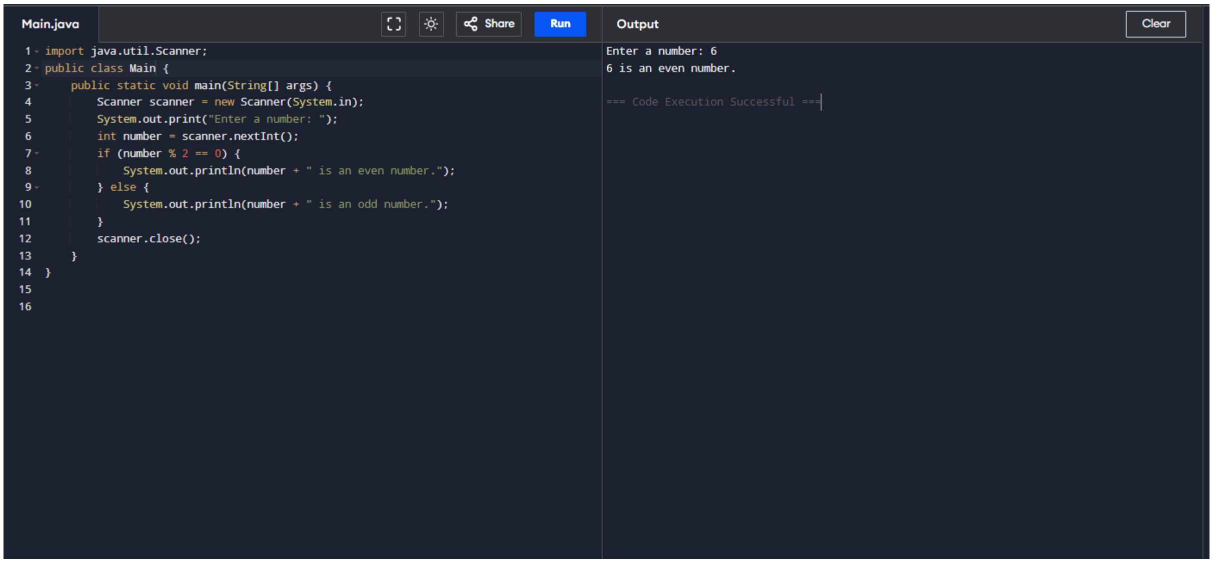
- **Java Program to take input from user and print the sum of two numbers**



```java
import java.util.Scanner;
public class Main {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter the first number: ");
        int num1 = scanner.nextInt();
        System.out.print("Enter the second number: ");
        int num2 = scanner.nextInt();
        int sum = num1 + num2;
        System.out.println("The sum of " + num1 + " and " + num2 + " is: " + sum);
        scanner.close();
    }
}
```

```
Enter the first number: 59
Enter the second number: 56
The sum of 59 and 56 is: 115

=== Code Execution Successful ===
```

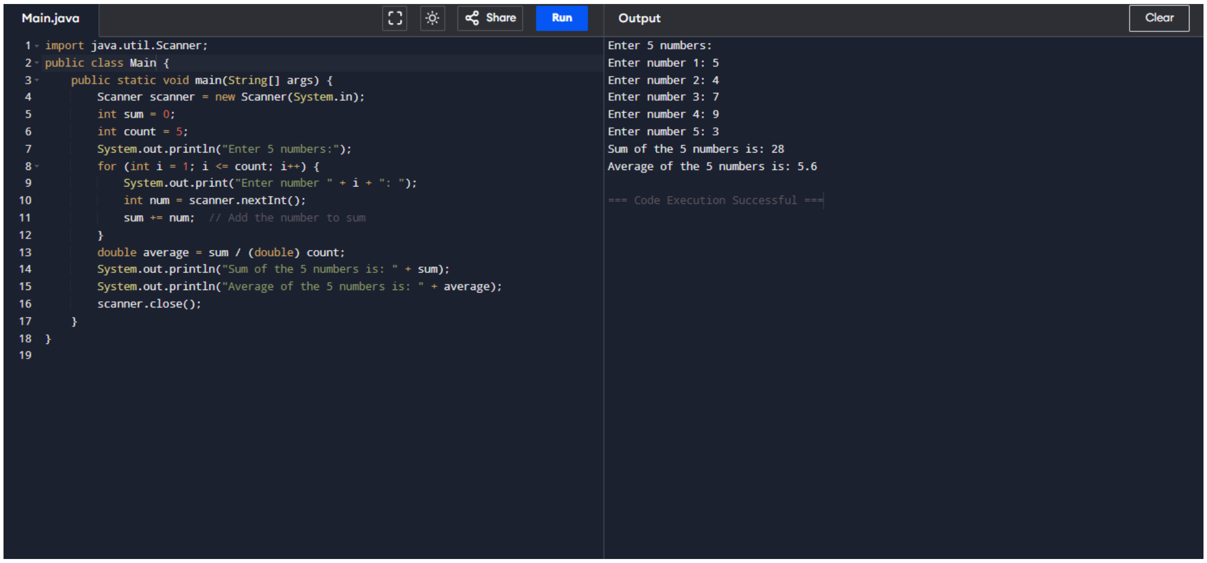
- **Create a java program to check whether a number entered by user is even or Odd**



```java
import java.util.Scanner;
public class Main {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter a number: ");
        int number = scanner.nextInt();
        if (number % 2 == 0) {
            System.out.println(number + " is an even number.");
        } else {
            System.out.println(number + " is an odd number.");
        }
        scanner.close();
    }
}
```

```
Enter a number: 6
6 is an even number.

=== Code Execution Successful ===
```

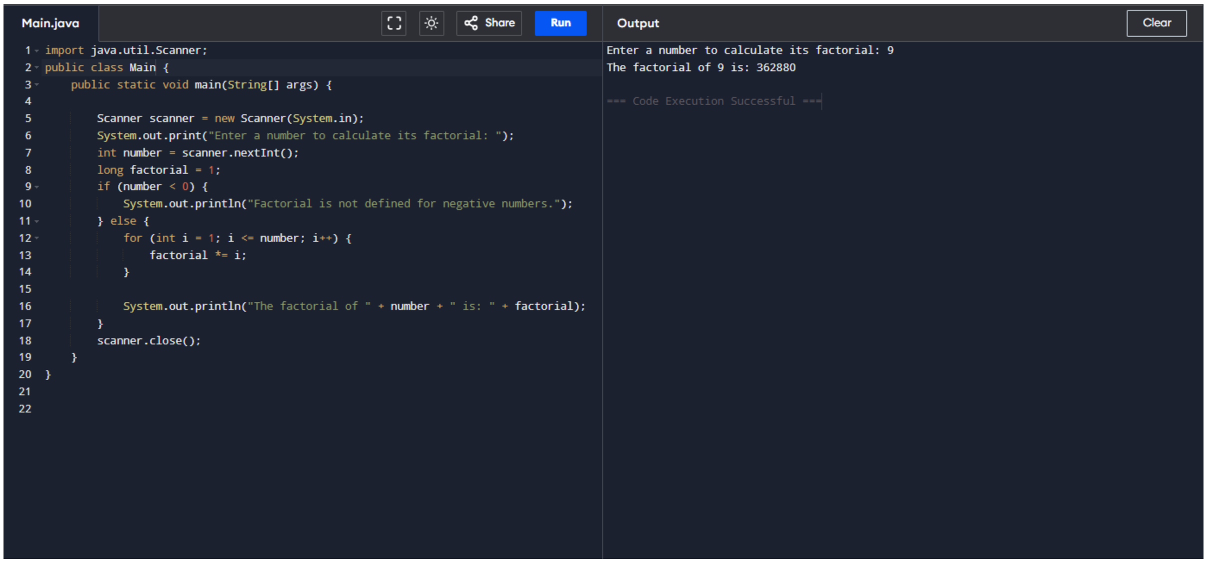
- **Create a java program to print the average and sum of 5 numbers entered by user.**



```java
import java.util.Scanner;
public class Main {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        int sum = 0;
        int count = 5;
        System.out.println("Enter 5 numbers:");
        for (int i = 1; i <= count; i++) {
            System.out.print("Enter number " + i + ": ");
            int num = scanner.nextInt();
            sum += num;  // Add the number to sum
        }
        double average = sum / (double) count;
        System.out.println("Sum of the 5 numbers is: " + sum);
        System.out.println("Average of the 5 numbers is: " + average);
        scanner.close();
    }
}
```

```
Enter 5 numbers:
Enter number 1: 5
Enter number 2: 4
Enter number 3: 7
Enter number 4: 9
Enter number 5: 3
Sum of the 5 numbers is: 28
Average of the 5 numbers is: 5.6

=== Code Execution Successful ===
```

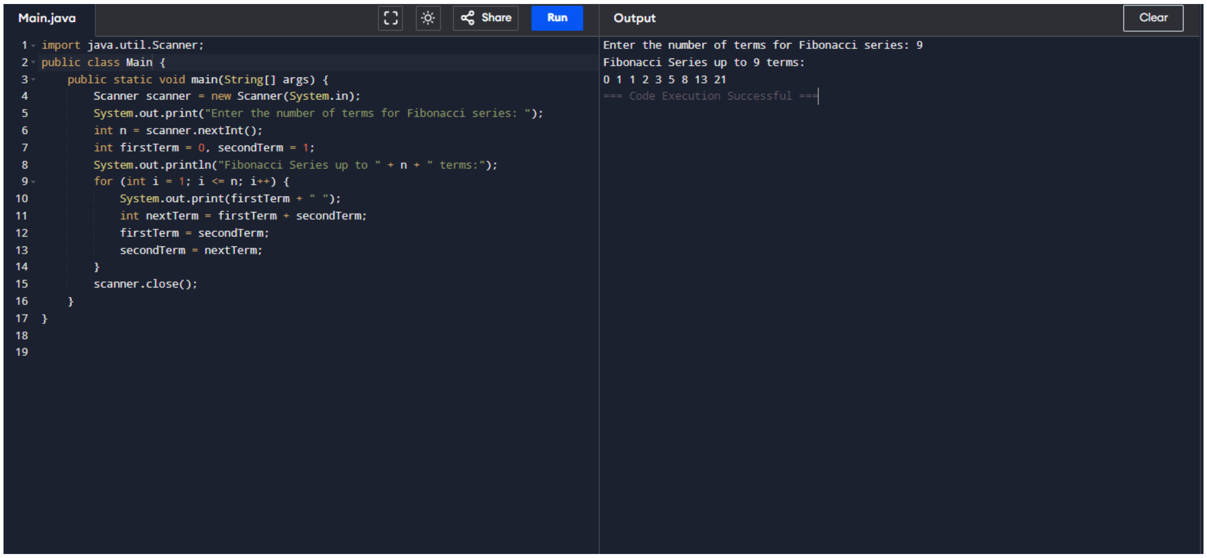
- **Program to calculate the factorial of a number**



```java
import java.util.Scanner;
public class Main {
    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter a number to calculate its factorial: ");
        int number = scanner.nextInt();
        long factorial = 1;
        if (number < 0) {
            System.out.println("Factorial is not defined for negative numbers.");
        } else {
            for (int i = 1; i <= number; i++) {
                factorial *= i;
            }

            System.out.println("The factorial of " + number + " is: " + factorial);
        }
        scanner.close();
    }
}
```

```
Enter a number to calculate its factorial: 9
The factorial of 9 is: 362880

=== Code Execution Successful ===
```

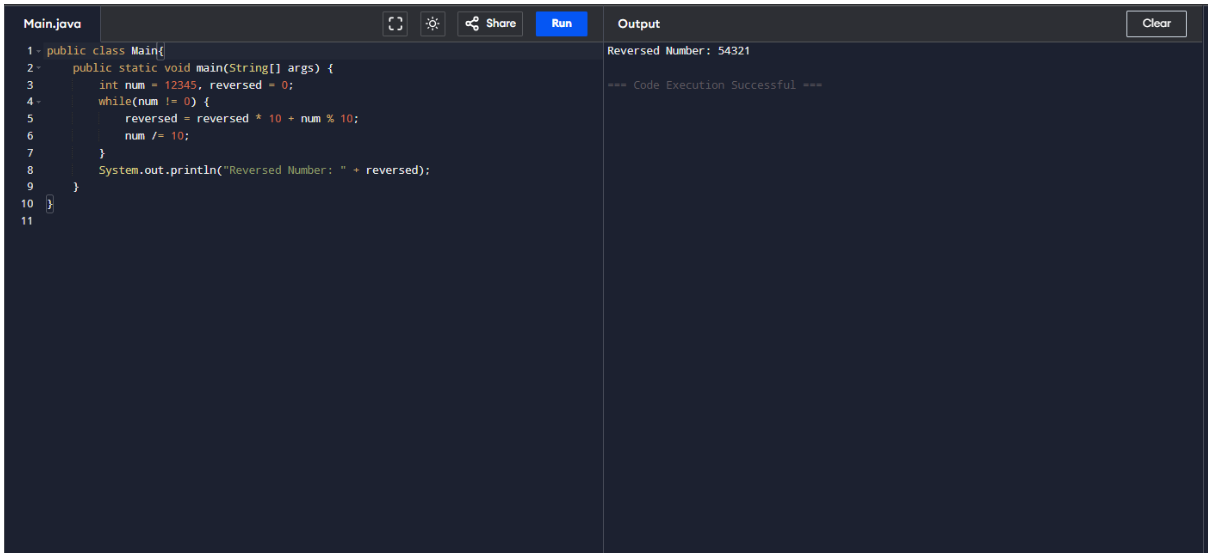
- **Program to print Fibonacci series up to n terms**



```java
import java.util.Scanner;
public class Main {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter the number of terms for Fibonacci series: ");
        int n = scanner.nextInt();
        int firstTerm = 0, secondTerm = 1;
        System.out.println("Fibonacci Series up to " + n + " terms:");
        for (int i = 1; i <= n; i++) {
            System.out.print(firstTerm + " ");
            int nextTerm = firstTerm + secondTerm;
            firstTerm = secondTerm;
            secondTerm = nextTerm;
        }
        scanner.close();
    }
}
```

```
Enter the number of terms for Fibonacci series: 9
Fibonacci Series up to 9 terms:
0 1 1 2 3 5 8 13 21
=== Code Execution Successful ===
```

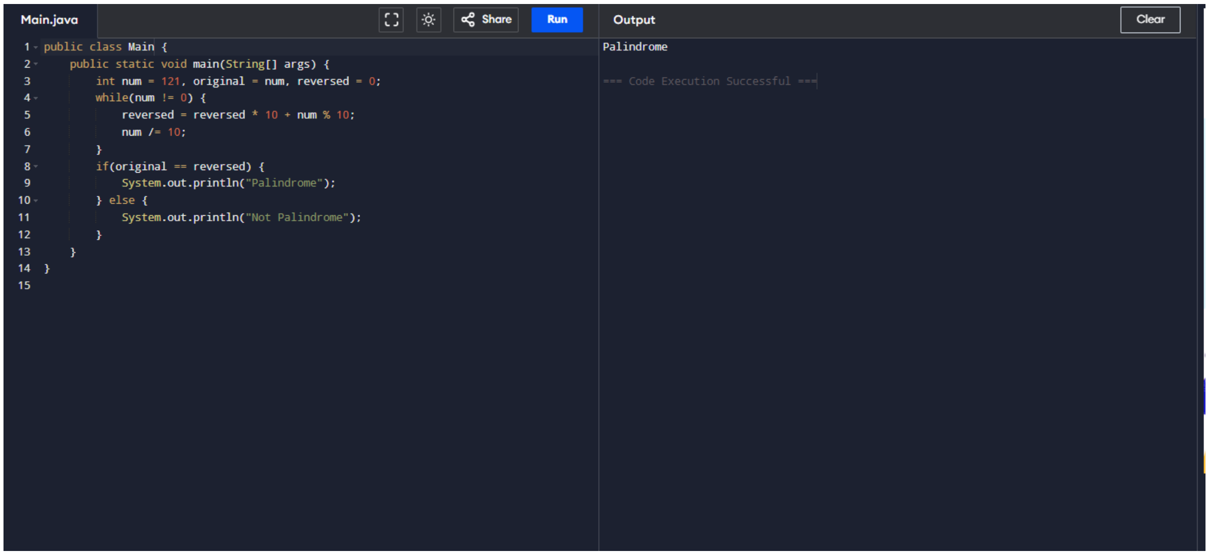
- ## Program to reverse a number



```java
public class Main{
    public static void main(String[] args) {
        int num = 12345, reversed = 0;
        while(num != 0) {
            reversed = reversed * 10 + num % 10;
            num /= 10;
        }
        System.out.println("Reversed Number: " + reversed);
    }
}
```

```
Reversed Number: 54321

=== Code Execution Successful ===
```

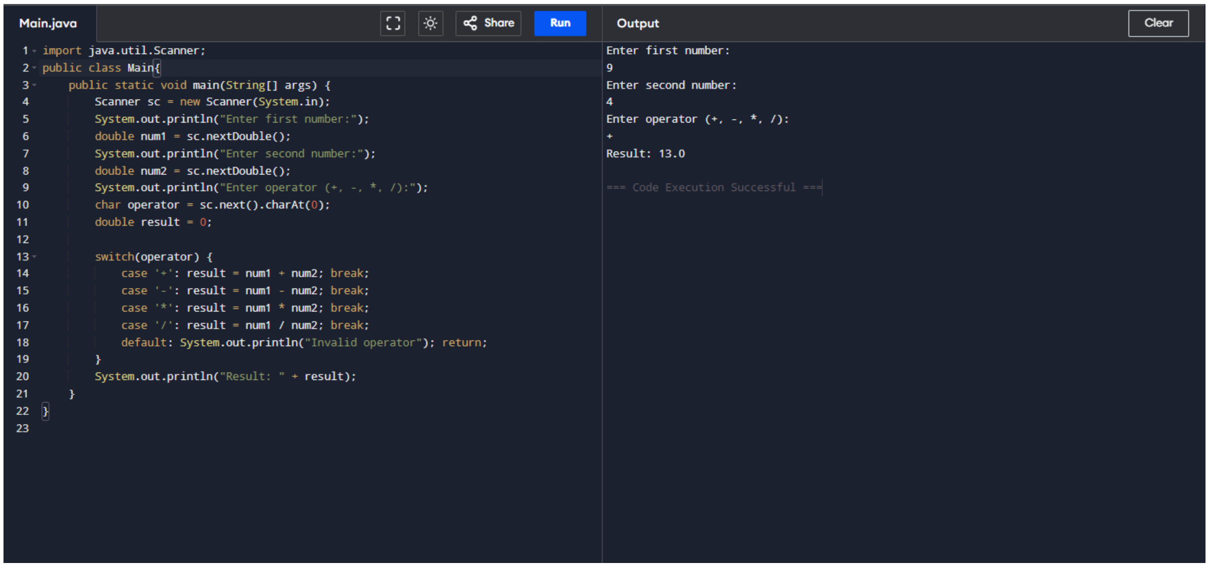
- ## Program to check if a number is a palindrome



```java
public class Main {
    public static void main(String[] args) {
        int num = 121, original = num, reversed = 0;
        while(num != 0) {
            reversed = reversed * 10 + num % 10;
            num /= 10;
        }
        if(original == reversed) {
            System.out.println("Palindrome");
        } else {
            System.out.println("Not Palindrome");
        }
    }
}
```

```
Palindrome

=== Code Execution Successful ===
```

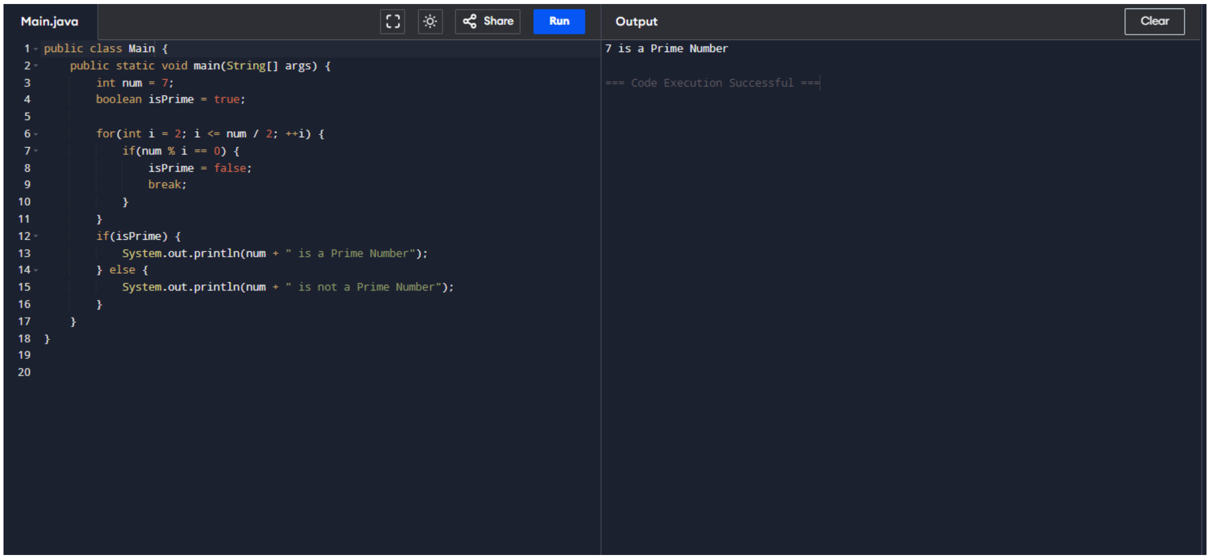
- ## Program for a simple calculator



```java
import java.util.Scanner;
public class Main{
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter first number:");
        double num1 = sc.nextDouble();
        System.out.println("Enter second number:");
        double num2 = sc.nextDouble();
        System.out.println("Enter operator (+, -, *, /):");
        char operator = sc.next().charAt(0);
        double result = 0;

        switch(operator) {
            case '+': result = num1 + num2; break;
            case '-': result = num1 - num2; break;
            case '*': result = num1 * num2; break;
            case '/': result = num1 / num2; break;
            default: System.out.println("Invalid operator"); return;
        }
        System.out.println("Result: " + result);
    }
}
```

```
Enter first number:
9
Enter second number:
4
Enter operator (+, -, *, /):
+
Result: 13.0

=== Code Execution Successful ===
```

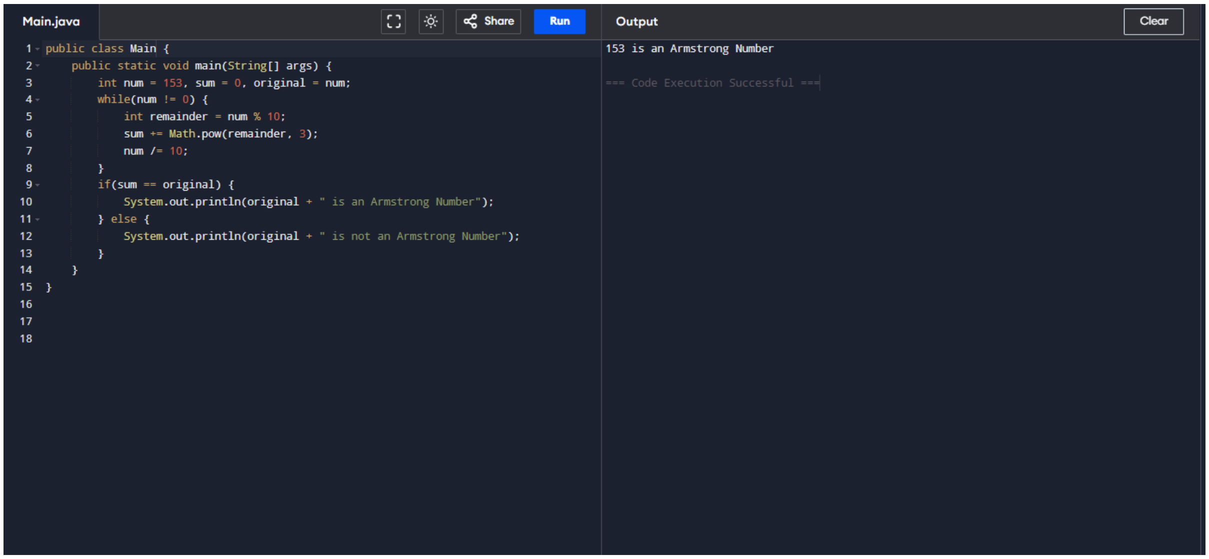
- ## Program to check if a number is prime



```java
public class Main {
    public static void main(String[] args) {
        int num = 7;
        boolean isPrime = true;

        for(int i = 2; i <= num / 2; ++i) {
            if(num % i == 0) {
                isPrime = false;
                break;
            }
        }
        if(isPrime) {
            System.out.println(num + " is a Prime Number");
        } else {
            System.out.println(num + " is not a Prime Number");
        }
    }
}
```

```
7 is a Prime Number

=== Code Execution Successful ===
```

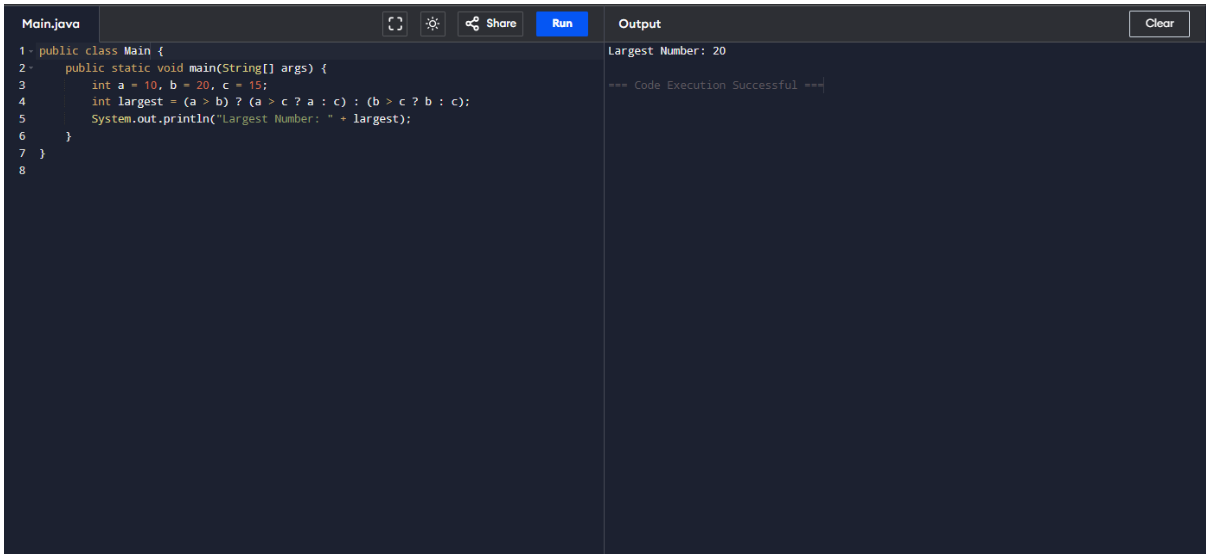
- **Program to check if a number is an Armstrong number**



```java
public class Main {
    public static void main(String[] args) {
        int num = 153, sum = 0, original = num;
        while(num != 0) {
            int remainder = num % 10;
            sum += Math.pow(remainder, 3);
            num /= 10;
        }
        if(sum == original) {
            System.out.println(original + " is an Armstrong Number");
        } else {
            System.out.println(original + " is not an Armstrong Number");
        }
    }
}
```

```
153 is an Armstrong Number

=== Code Execution Successful ===
```

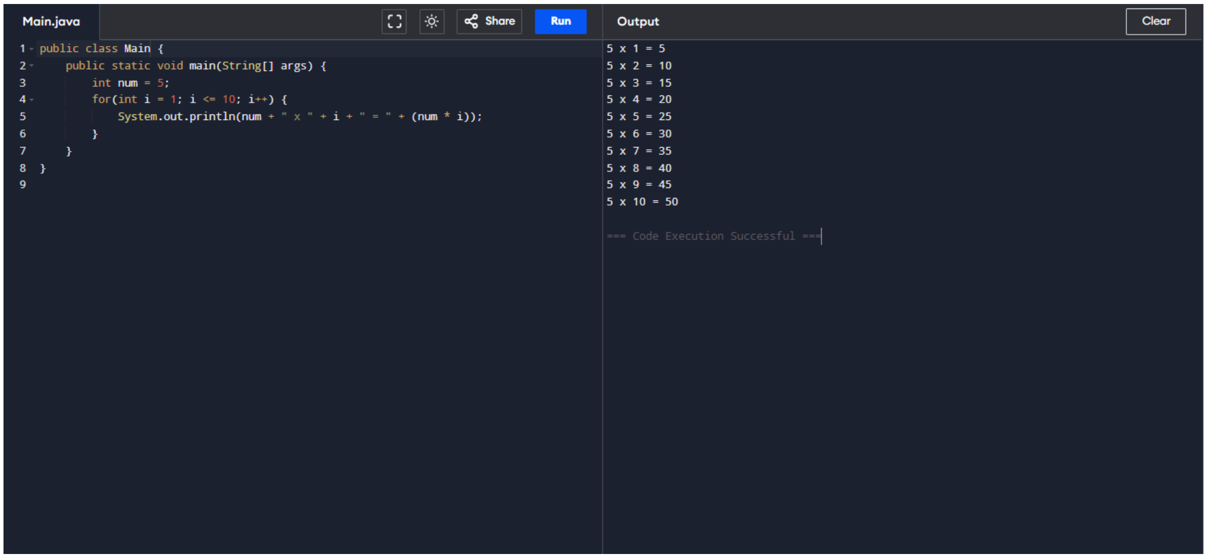
- **Find the Largest of Three Numbers using ternary operator**



```java
public class Main {
    public static void main(String[] args) {
        int a = 10, b = 20, c = 15;
        int largest = (a > b) ? (a > c ? a : c) : (b > c ? b : c);
        System.out.println("Largest Number: " + largest);
    }
}
```

```
Largest Number: 20

=== Code Execution Successful ===
```

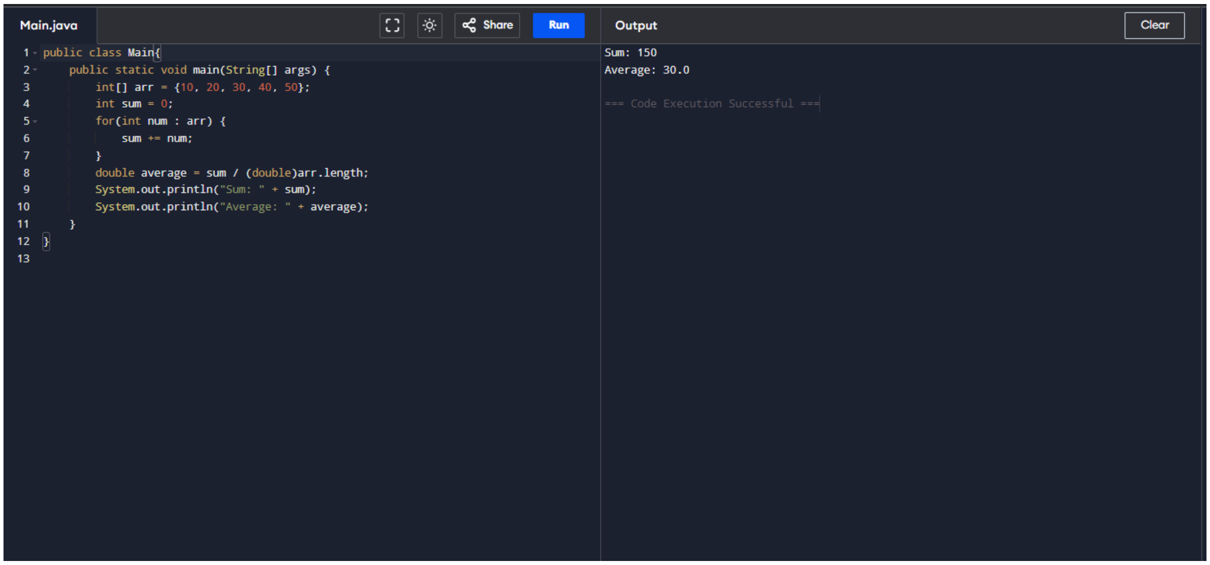
• Print Multiplication Table



```java
public class Main {
    public static void main(String[] args) {
        int num = 5;
        for(int i = 1; i <= 10; i++) {
            System.out.println(num + " x " + i + " = " + (num * i));
        }
    }
}
```

```
5 x 1 = 5
5 x 2 = 10
5 x 3 = 15
5 x 4 = 20
5 x 5 = 25
5 x 6 = 30
5 x 7 = 35
5 x 8 = 40
5 x 9 = 45
5 x 10 = 50

=== Code Execution Successful ===
```

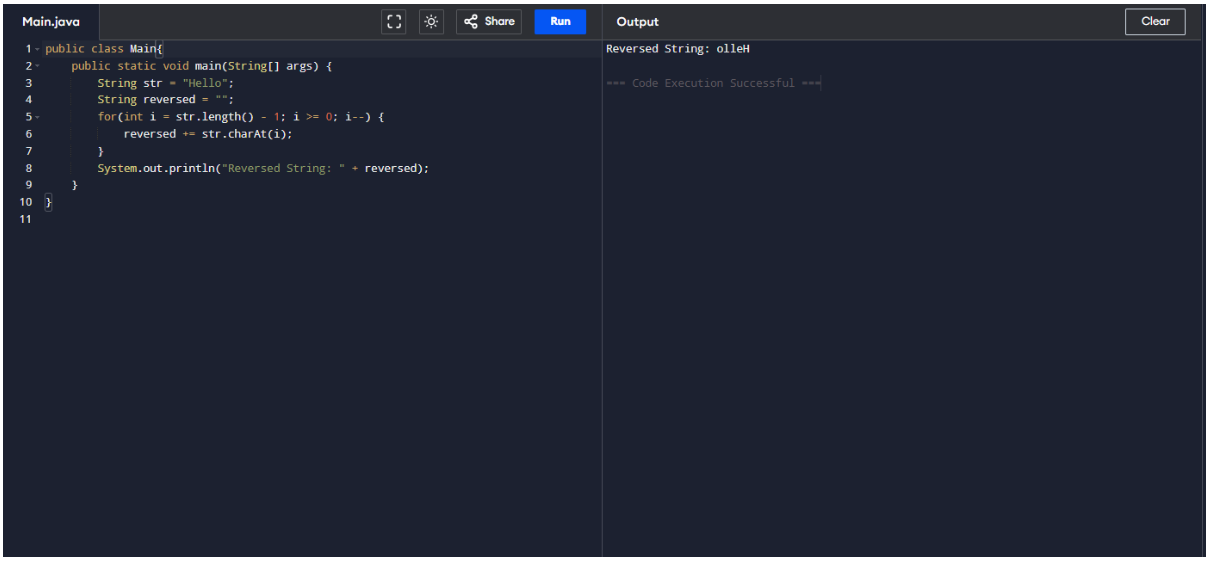
• Calculate Sum and Average of Array Elements



```java
public class Main{
    public static void main(String[] args) {
        int[] arr = {10, 20, 30, 40, 50};
        int sum = 0;
        for(int num : arr) {
            sum += num;
        }
        double average = sum / (double)arr.length;
        System.out.println("Sum: " + sum);
        System.out.println("Average: " + average);
    }
}
```

```
Sum: 150
Average: 30.0

=== Code Execution Successful ===
```

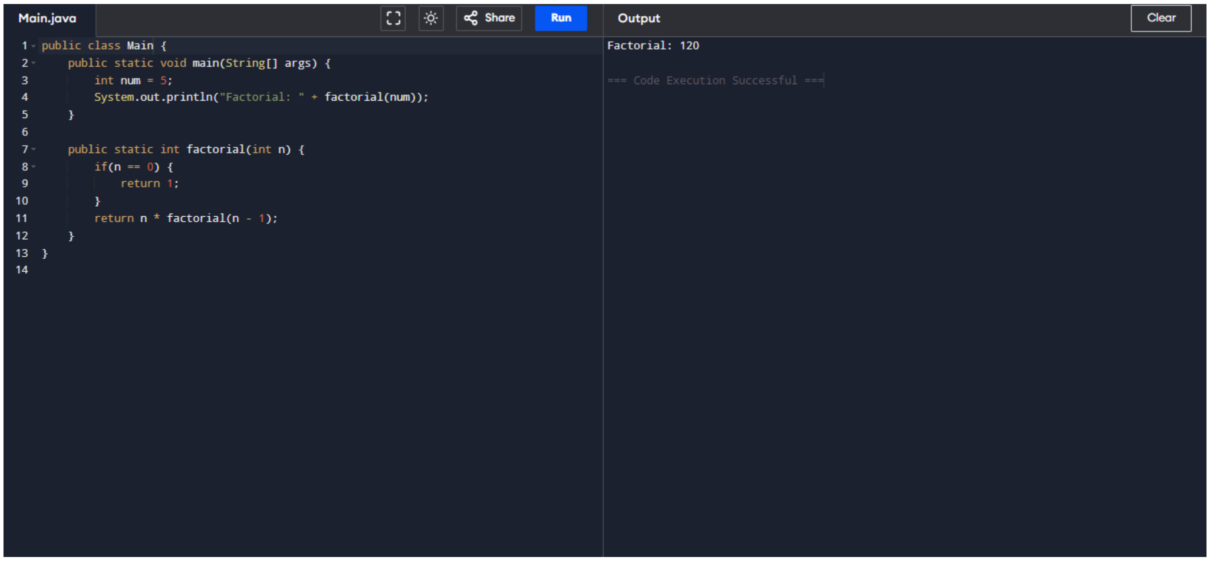
- **Reverse a String**



```
public class Main{
    public static void main(String[] args) {
        String str = "Hello";
        String reversed = "";
        for(int i = str.length() - 1; i >= 0; i--) {
            reversed += str.charAt(i);
        }
        System.out.println("Reversed String: " + reversed);
    }
}
```

```
Reversed String: olleH

=== Code Execution Successful ===
```

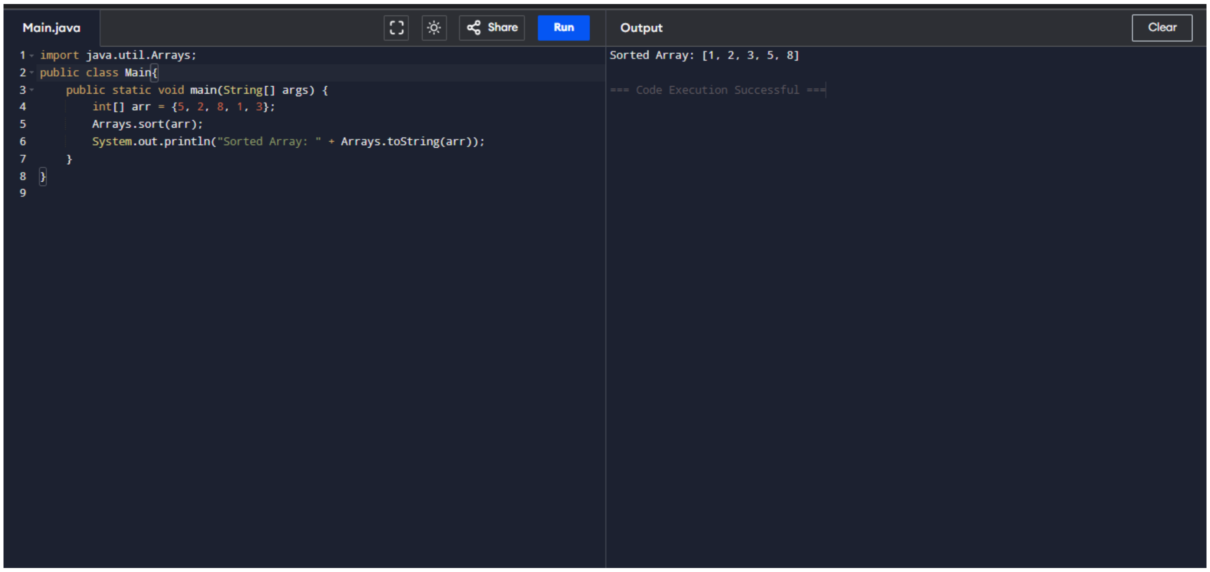
- **Find Factorial of a Number Using Recursion**



```
public class Main {
    public static void main(String[] args) {
        int num = 5;
        System.out.println("Factorial: " + factorial(num));
    }

    public static int factorial(int n) {
        if(n == 0) {
            return 1;
        }
        return n * factorial(n - 1);
    }
}
```

```
Factorial: 120

=== Code Execution Successful ===
```

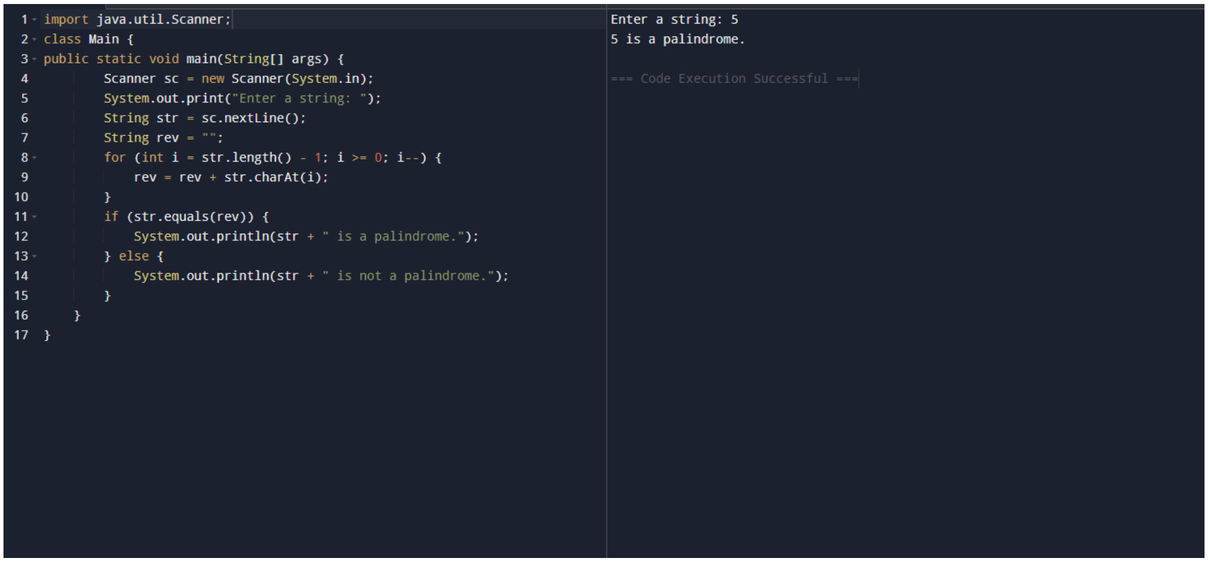
- ## Sort an Array in Ascending Order:



```java
import java.util.Arrays;
public class Main{
    public static void main(String[] args) {
        int[] arr = {5, 2, 8, 1, 3};
        Arrays.sort(arr);
        System.out.println("Sorted Array: " + Arrays.toString(arr));
    }
}
```

```
Sorted Array: [1, 2, 3, 5, 8]

=== Code Execution Successful ===
```

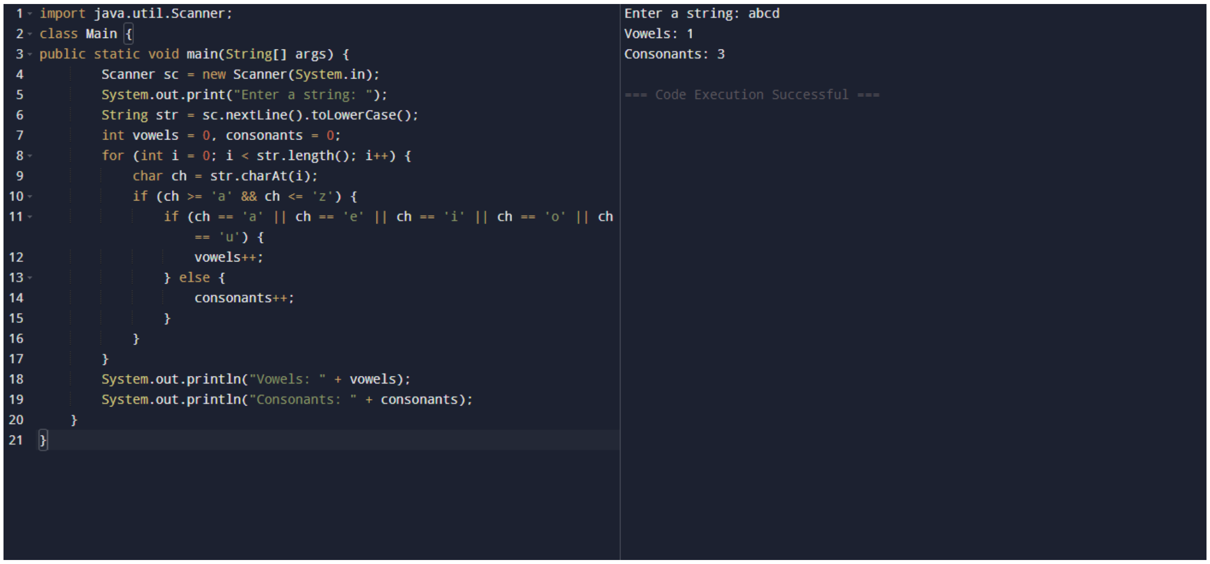
- ## Check Palindrome for a String



```java
import java.util.Scanner;
class Main {
public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a string: ");
        String str = sc.nextLine();
        String rev = "";
        for (int i = str.length() - 1; i >= 0; i--) {
            rev = rev + str.charAt(i);
        }
        if (str.equals(rev)) {
            System.out.println(str + " is a palindrome.");
        } else {
            System.out.println(str + " is not a palindrome.");
        }
    }
}
```

```
Enter a string: 5
5 is a palindrome.

=== Code Execution Successful ===
```

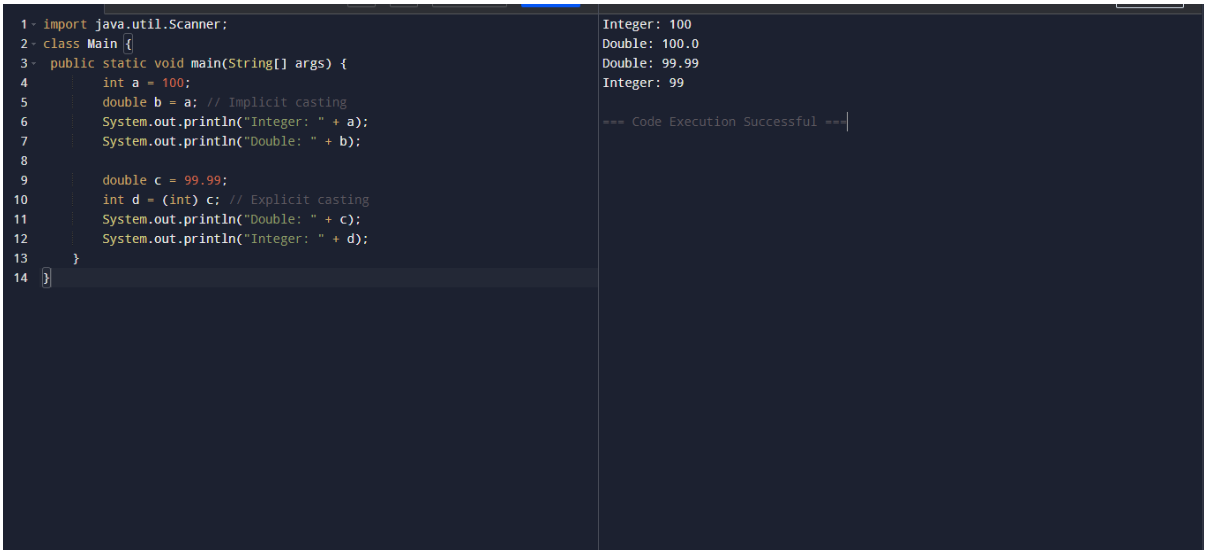
- **Count Vowels and Consonants in a String**



```java
import java.util.Scanner;
class Main {
public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a string: ");
        String str = sc.nextLine().toLowerCase();
        int vowels = 0, consonants = 0;
        for (int i = 0; i < str.length(); i++) {
            char ch = str.charAt(i);
            if (ch >= 'a' && ch <= 'z') {
                if (ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u') {
                    vowels++;
                } else {
                    consonants++;
                }
            }
        }
        System.out.println("Vowels: " + vowels);
        System.out.println("Consonants: " + consonants);
    }
}
```

```
Enter a string: abcd
Vowels: 1
Consonants: 3

=== Code Execution Successful ===
```

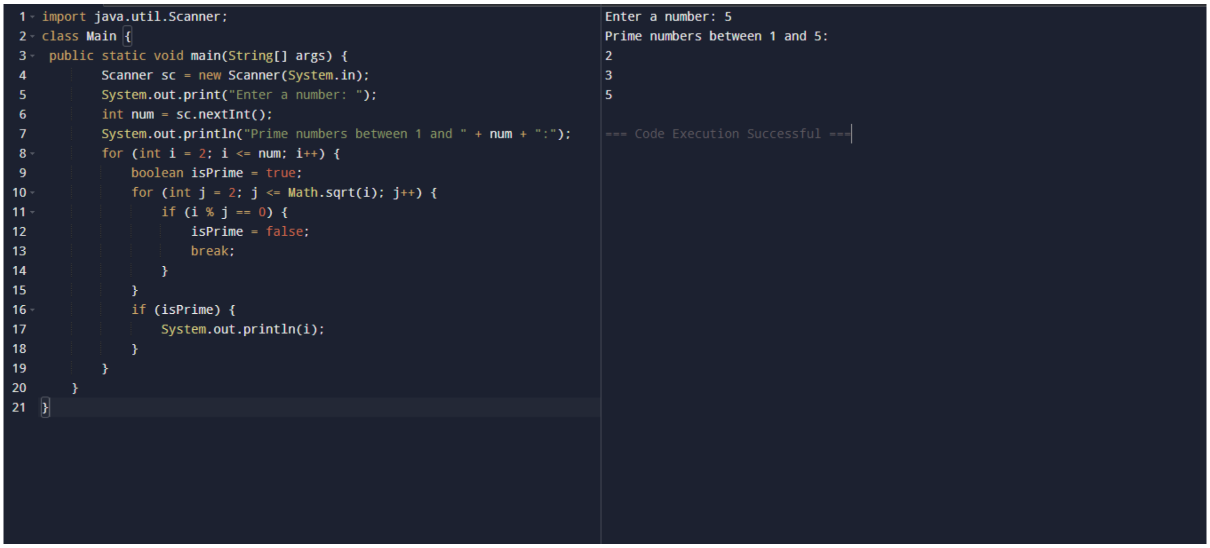
- **Program to Demonstrate Type Casting**



```java
import java.util.Scanner;
class Main {
 public static void main(String[] args) {
        int a = 100;
        double b = a; // Implicit casting
        System.out.println("Integer: " + a);
        System.out.println("Double: " + b);

        double c = 99.99;
        int d = (int) c; // Explicit casting
        System.out.println("Double: " + c);
        System.out.println("Integer: " + d);
    }
}
```

```
Integer: 100
Double: 100.0
Double: 99.99
Integer: 99

=== Code Execution Successful ===
```

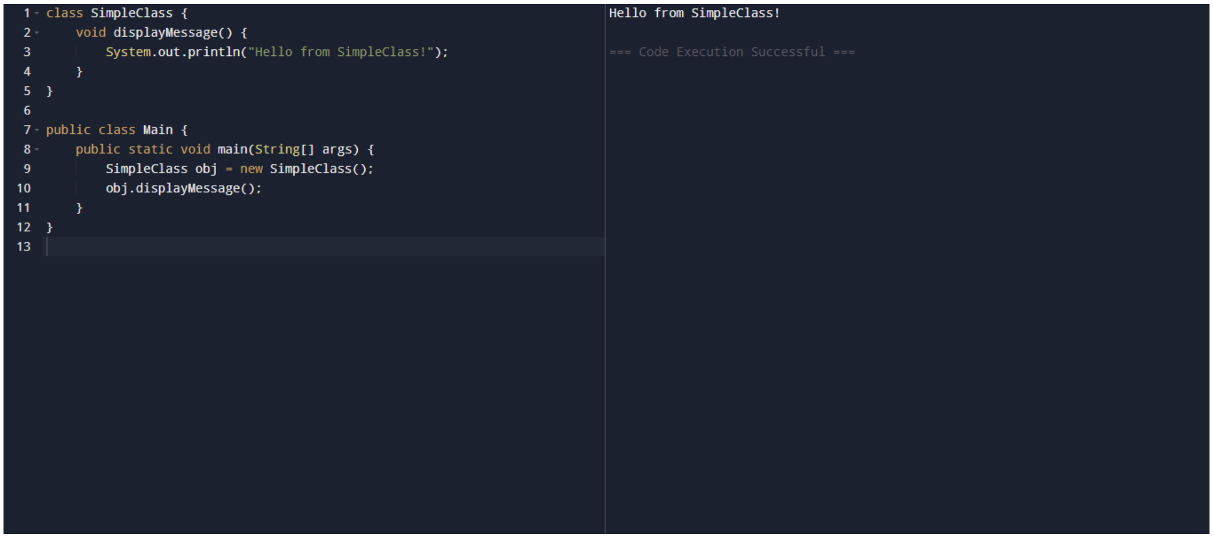
- **Generate Prime Numbers Between 1 and Given Number**



```java
import java.util.Scanner;
class Main {
 public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a number: ");
        int num = sc.nextInt();
        System.out.println("Prime numbers between 1 and " + num + ":");
        for (int i = 2; i <= num; i++) {
            boolean isPrime = true;
            for (int j = 2; j <= Math.sqrt(i); j++) {
                if (i % j == 0) {
                    isPrime = false;
                    break;
                }
            }
            if (isPrime) {
                System.out.println(i);
            }
        }
    }
}
```

```
Enter a number: 5
Prime numbers between 1 and 5:
2
3
5

=== Code Execution Successful ===
```

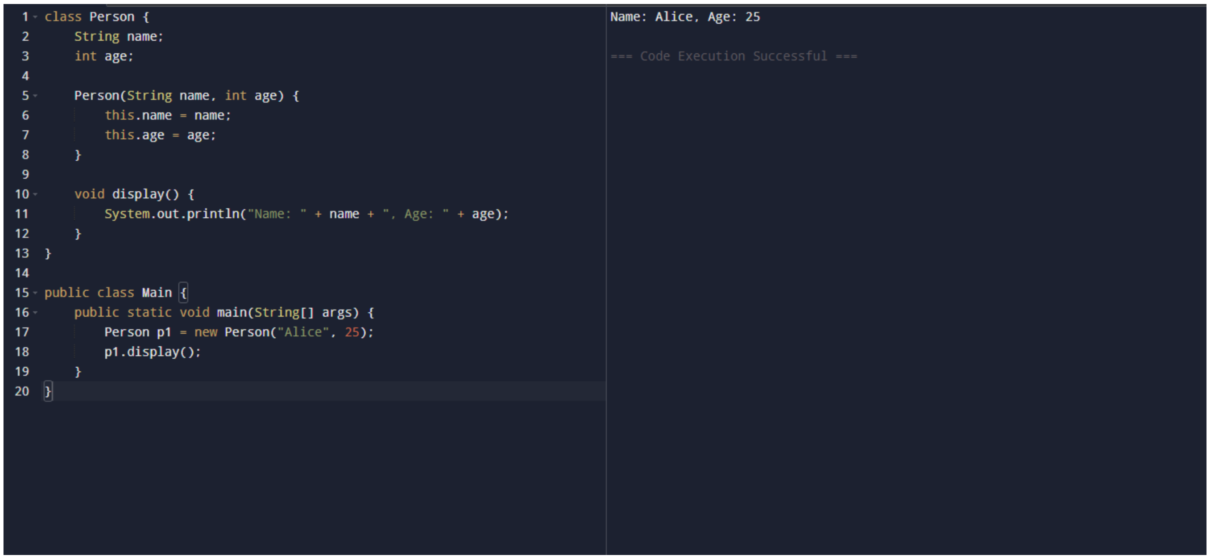
- **Program to Demonstrate a Simple Class with Methods**



```java
class SimpleClass {
    void displayMessage() {
        System.out.println("Hello from SimpleClass!");
    }
}

public class Main {
    public static void main(String[] args) {
        SimpleClass obj = new SimpleClass();
        obj.displayMessage();
    }
}
```

```
Hello from SimpleClass!

=== Code Execution Successful ===
```

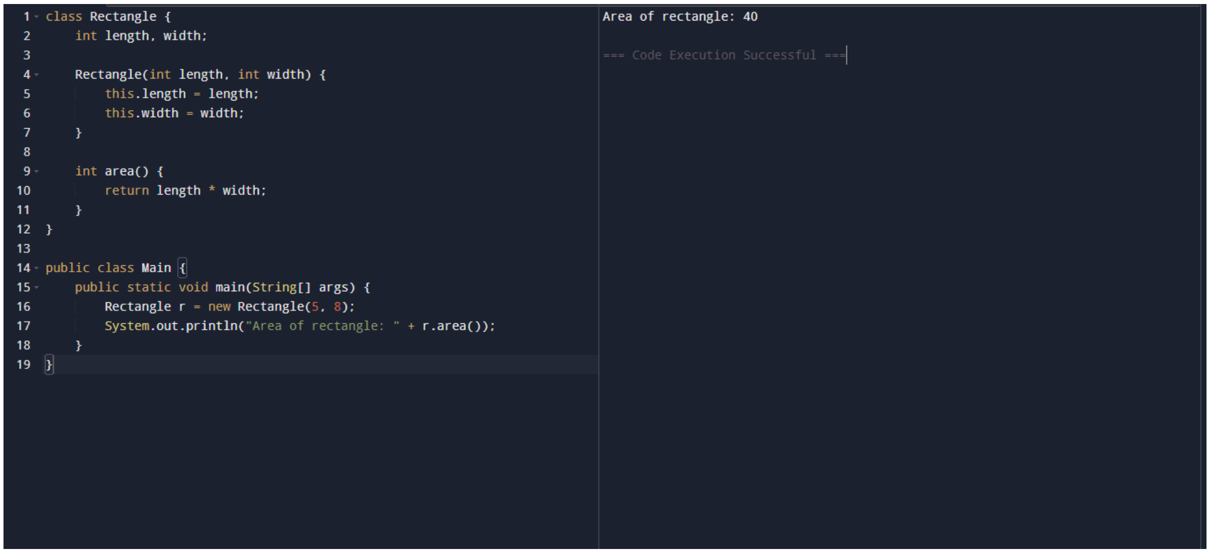
- **Program for Class with Parameterized Constructor**



```java
class Person {
    String name;
    int age;

    Person(String name, int age) {
        this.name = name;
        this.age = age;
    }

    void display() {
        System.out.println("Name: " + name + ", Age: " + age);
    }
}

public class Main {
    public static void main(String[] args) {
        Person p1 = new Person("Alice", 25);
        p1.display();
    }
}
```

```
Name: Alice, Age: 25

=== Code Execution Successful ===
```

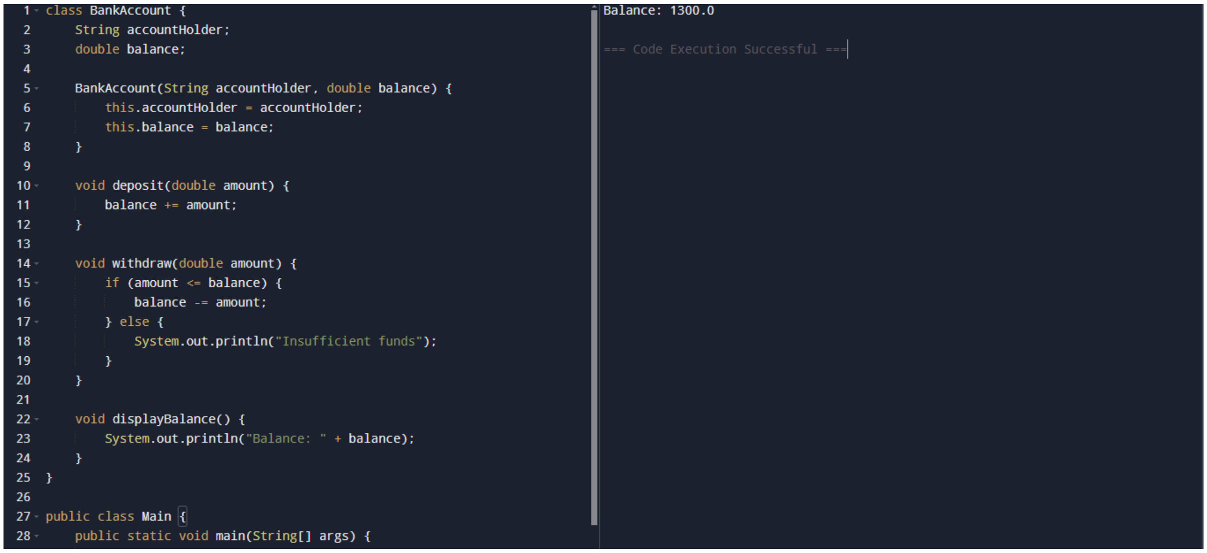
- **Program to Find the Area of a Rectangle Using Methods**



```java
class Rectangle {
    int length, width;

    Rectangle(int length, int width) {
        this.length = length;
        this.width = width;
    }

    int area() {
        return length * width;
    }
}

public class Main {
    public static void main(String[] args) {
        Rectangle r = new Rectangle(5, 8);
        System.out.println("Area of rectangle: " + r.area());
    }
}
```

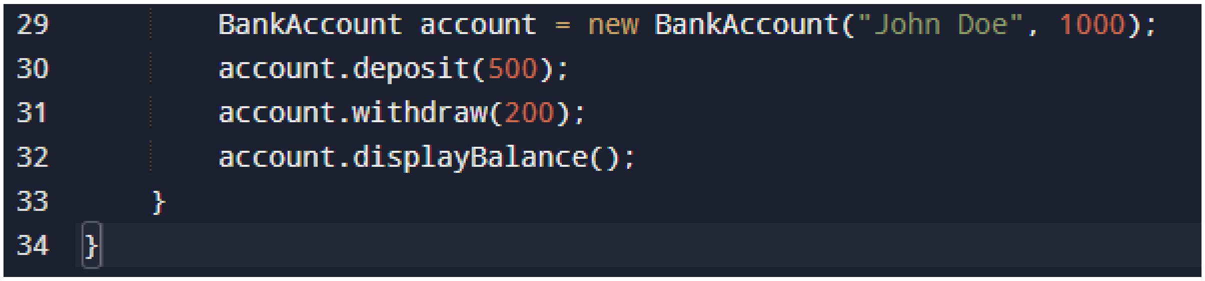
```
Area of rectangle: 40

=== Code Execution Successful ===
```

• Program for Bank Account Class with Deposit and Withdraw Methods



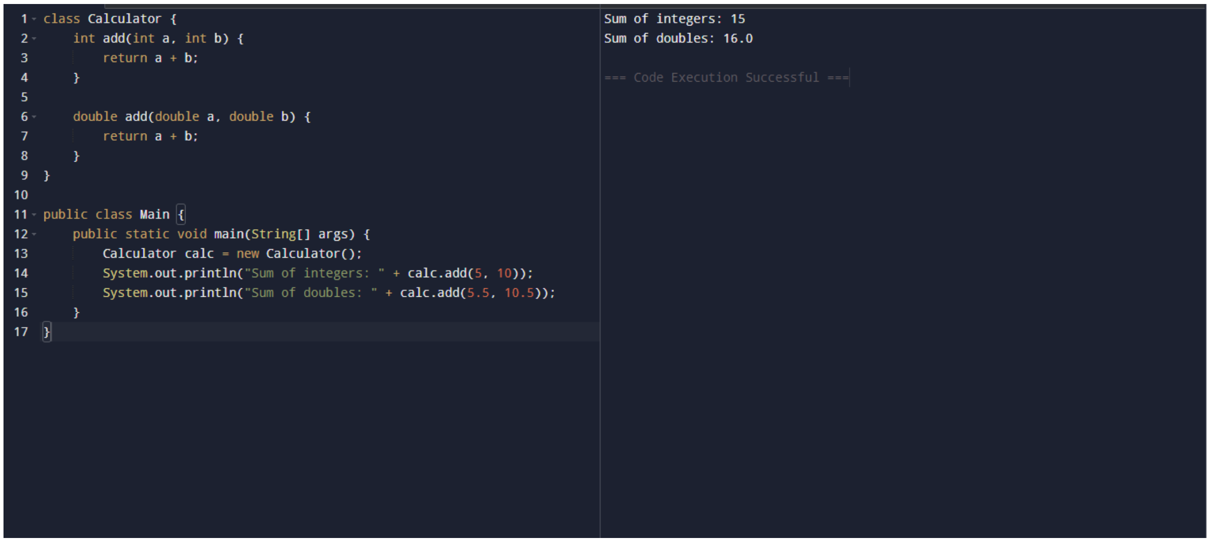
```java
class BankAccount {
    String accountHolder;
    double balance;

    BankAccount(String accountHolder, double balance) {
        this.accountHolder = accountHolder;
        this.balance = balance;
    }

    void deposit(double amount) {
        balance += amount;
    }

    void withdraw(double amount) {
        if (amount <= balance) {
            balance -= amount;
        } else {
            System.out.println("Insufficient funds");
        }
    }

    void displayBalance() {
        System.out.println("Balance: " + balance);
    }
}

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

```
Balance: 1300.0

=== Code Execution Successful ===
```



```java
        BankAccount account = new BankAccount("John Doe", 1000);
        account.deposit(500);
        account.withdraw(200);
        account.displayBalance();
    }
}
```

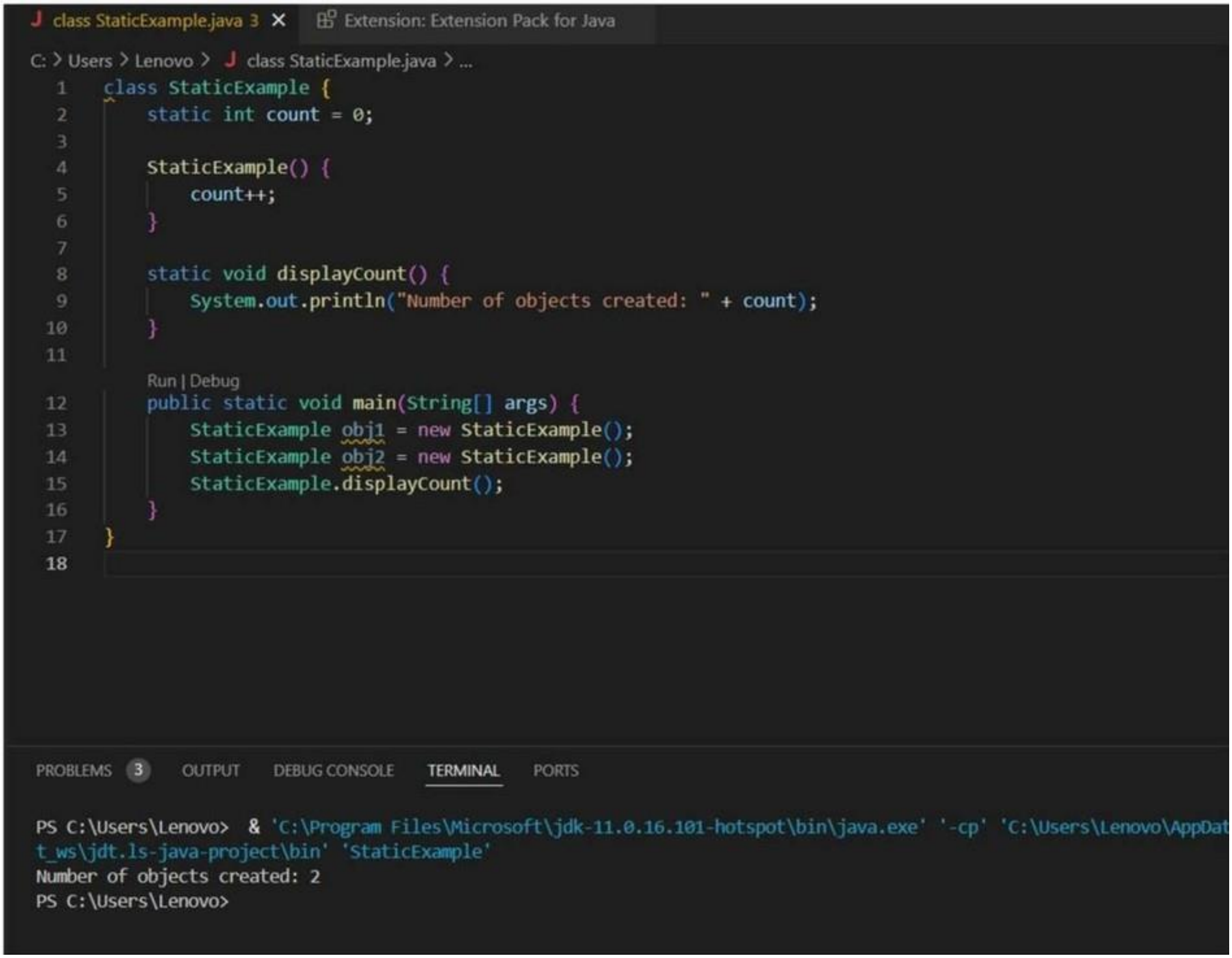
• Program to Demonstrate Method Overloading



```java
class Calculator {
    int add(int a, int b) {
        return a + b;
    }

    double add(double a, double b) {
        return a + b;
    }
}

public class Main {
    public static void main(String[] args) {
        Calculator calc = new Calculator();
        System.out.println("Sum of integers: " + calc.add(5, 10));
        System.out.println("Sum of doubles: " + calc.add(5.5, 10.5));
    }
}
```

```
Sum of integers: 15
Sum of doubles: 16.0

=== Code Execution Successful ===
```

- **Program to Demonstrate static methods**

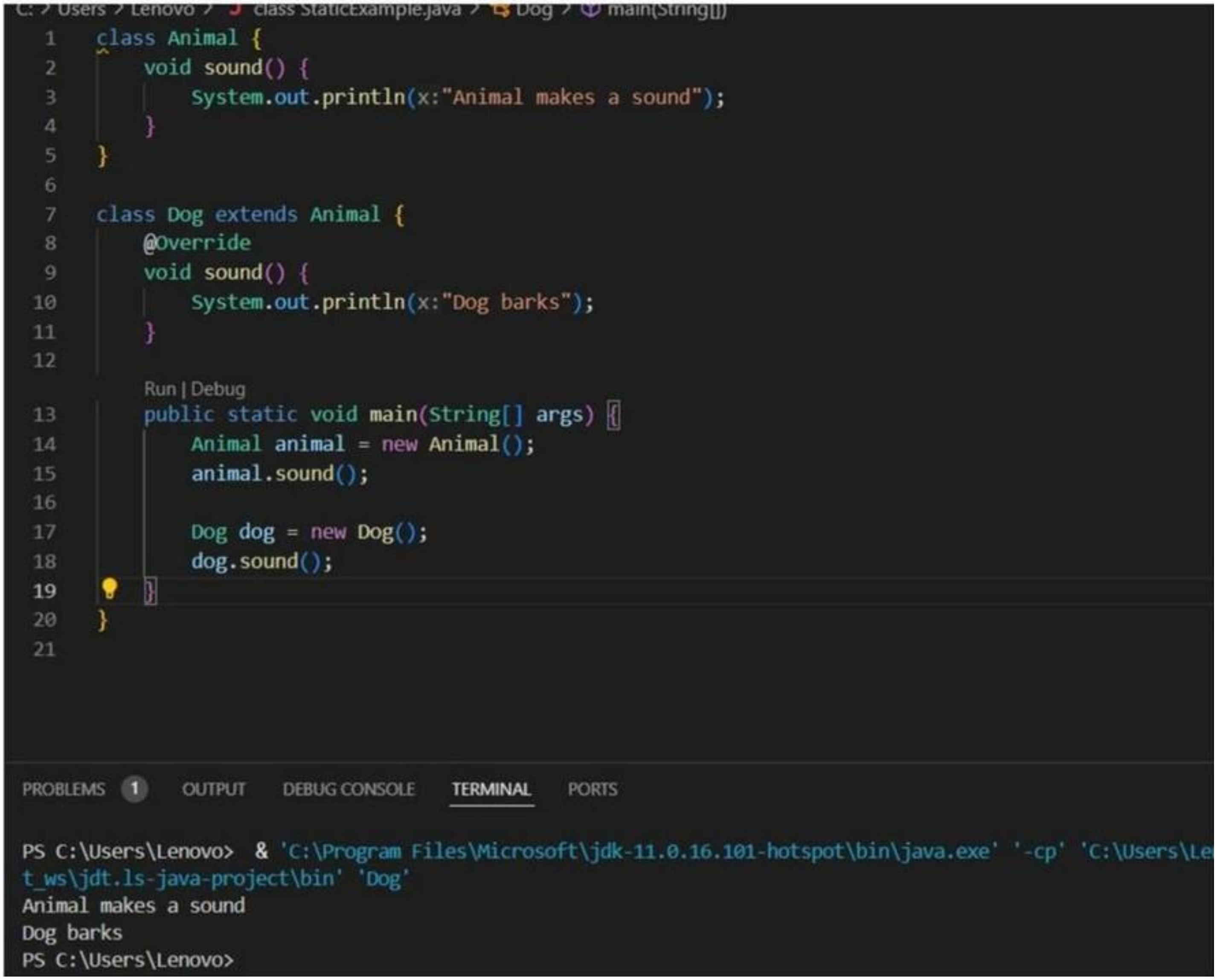


class StaticExample.java 3 ×
Extension: Extension Pack for Java
C: > Users > Lenovo > class StaticExample.java > ...
Run | Debug
PROBLEMS 3
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS

```java
class StaticExample {
    static int count = 0;

    StaticExample() {
        count++;
    }

    static void displayCount() {
        System.out.println("Number of objects created: " + count);
    }

    public static void main(String[] args) {
        StaticExample obj1 = new StaticExample();
        StaticExample obj2 = new StaticExample();
        StaticExample.displayCount();
    }
}

```

```
PS C:\Users\Lenovo>  & 'C:\Program Files\Microsoft\jdk-11.0.16.101-hotspot\bin\java.exe' '-cp' 'C:\Users\Lenovo\AppDat
t_ws\jdt.ls-java-project\bin' 'StaticExample'
Number of objects created: 2
PS C:\Users\Lenovo>
```

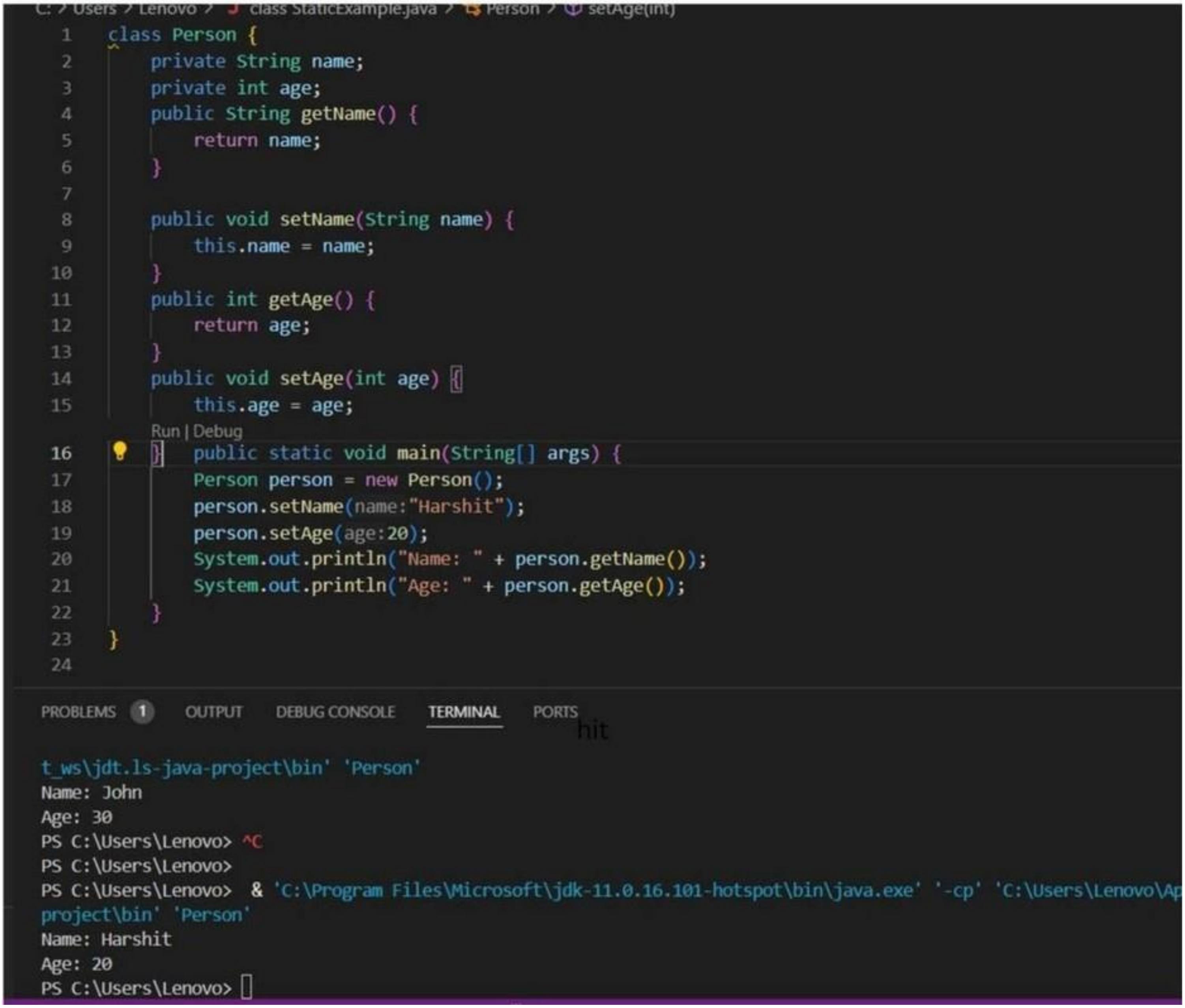
- **Program To Demonstrate Method Overriding**


C: > Users > Lenovo > class StaticExample.java > Dog > main(String[])
Run | Debug
PROBLEMS 1
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS

```java
class Animal {
    void sound() {
        System.out.println(x:"Animal makes a sound");
    }
}

class Dog extends Animal {
    @Override
    void sound() {
        System.out.println(x:"Dog barks");
    }

    public static void main(String[] args) {
        Animal animal = new Animal();
        animal.sound();

        Dog dog = new Dog();
        dog.sound();
    }
}
```

```
PS C:\Users\Lenovo>  & 'C:\Program Files\Microsoft\jdk-11.0.16.101-hotspot\bin\java.exe' '-cp' 'C:\Users\Le
t_ws\jdt.ls-java-project\bin' 'Dog'
Animal makes a sound
Dog barks
PS C:\Users\Lenovo>
```

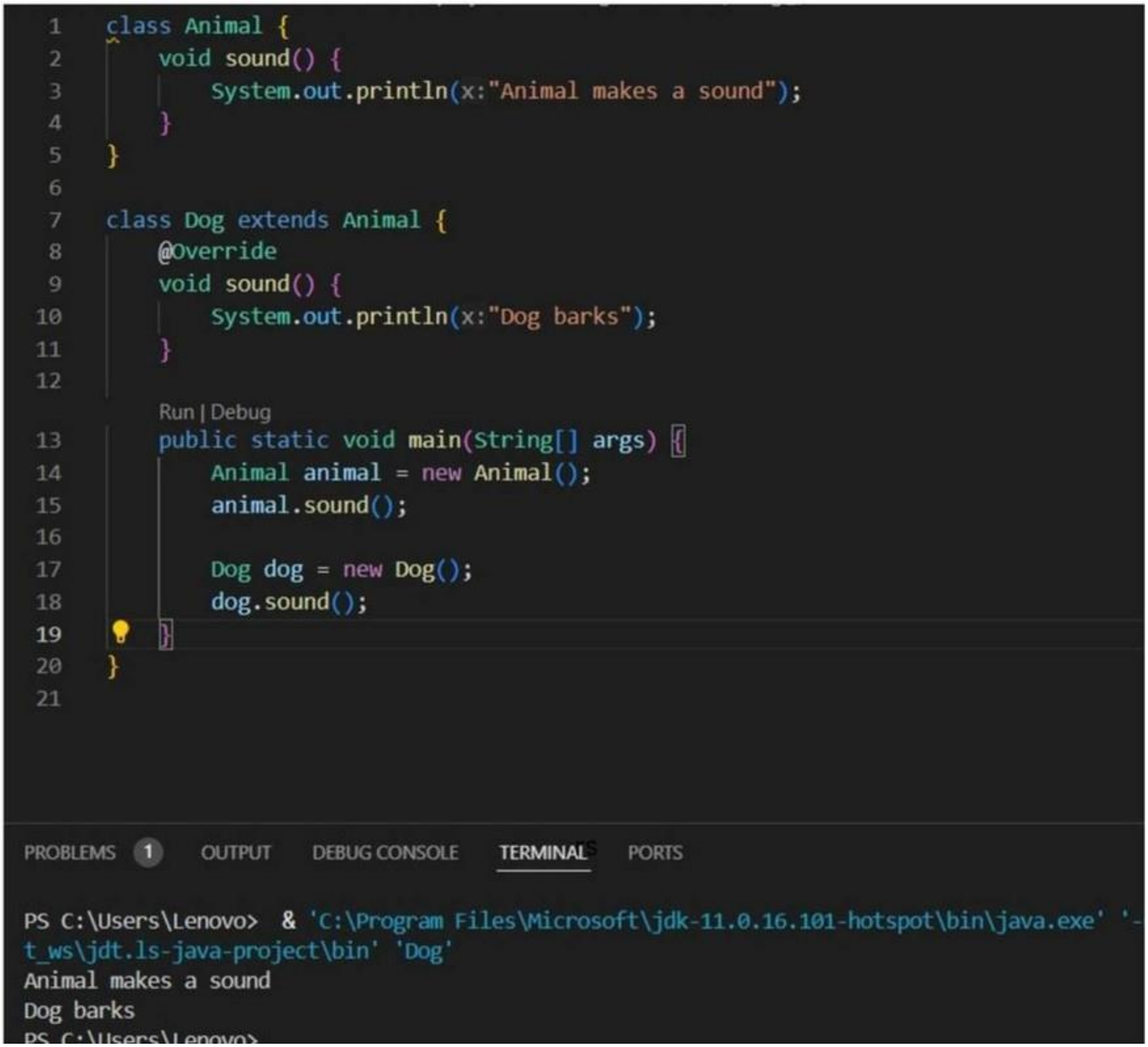
- **Program to Demonstrate Getters and Setters**


C: > Users > Lenovo > class StaticExample.java > Person > setAge(int)
Run | Debug
PROBLEMS 1
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
hit

```java
class Person {
    private String name;
    private int age;
    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }
    public int getAge() {
        return age;
    }
    public void setAge(int age) {
        this.age = age;
    }   public static void main(String[] args) {
        Person person = new Person();
        person.setName(name:"Harshit");
        person.setAge(age:20);
        System.out.println("Name: " + person.getName());
        System.out.println("Age: " + person.getAge());
    }
}
```

```
t_ws\jdt.ls-java-project\bin' 'Person'
Name: John
Age: 30
PS C:\Users\Lenovo> ^C
PS C:\Users\Lenovo>
PS C:\Users\Lenovo>  & 'C:\Program Files\Microsoft\jdk-11.0.16.101-hotspot\bin\java.exe' '-cp' 'C:\Users\Lenovo\Ap
project\bin' 'Person'
Name: Harshit
Age: 20
PS C:\Users\Lenovo> 
```

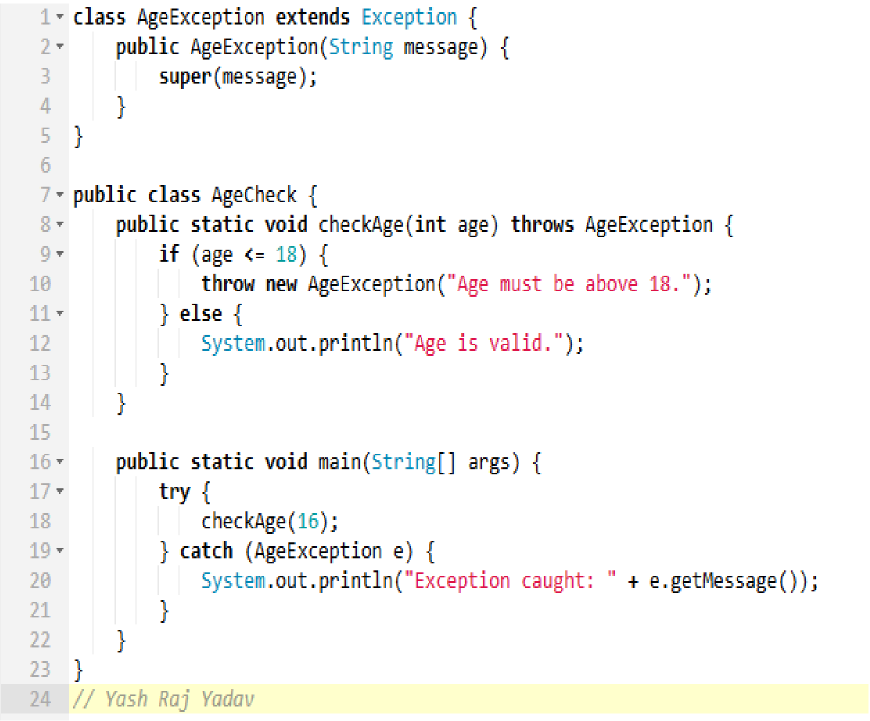
- **Write a Java program demonstrating a class with multiple methods**



```java
class Animal {
    void sound() {
        System.out.println(x:"Animal makes a sound");
    }
}

class Dog extends Animal {
    @Override
    void sound() {
        System.out.println(x:"Dog barks");
    }

    Run | Debug
    public static void main(String[] args) {
        Animal animal = new Animal();
        animal.sound();

        Dog dog = new Dog();
        dog.sound();
    }
}
```

```
PS C:\Users\Lenovo>  & 'C:\Program Files\Microsoft\jdk-11.0.16.101-hotspot\bin\java.exe' '-
t_ws\jdt.ls-java-project\bin' 'Dog'
Animal makes a sound
Dog barks
PS C:\Users\Lenovo>
```

- **Create a custom exception AgeException that checks if a person age is valid (above 18) in java**

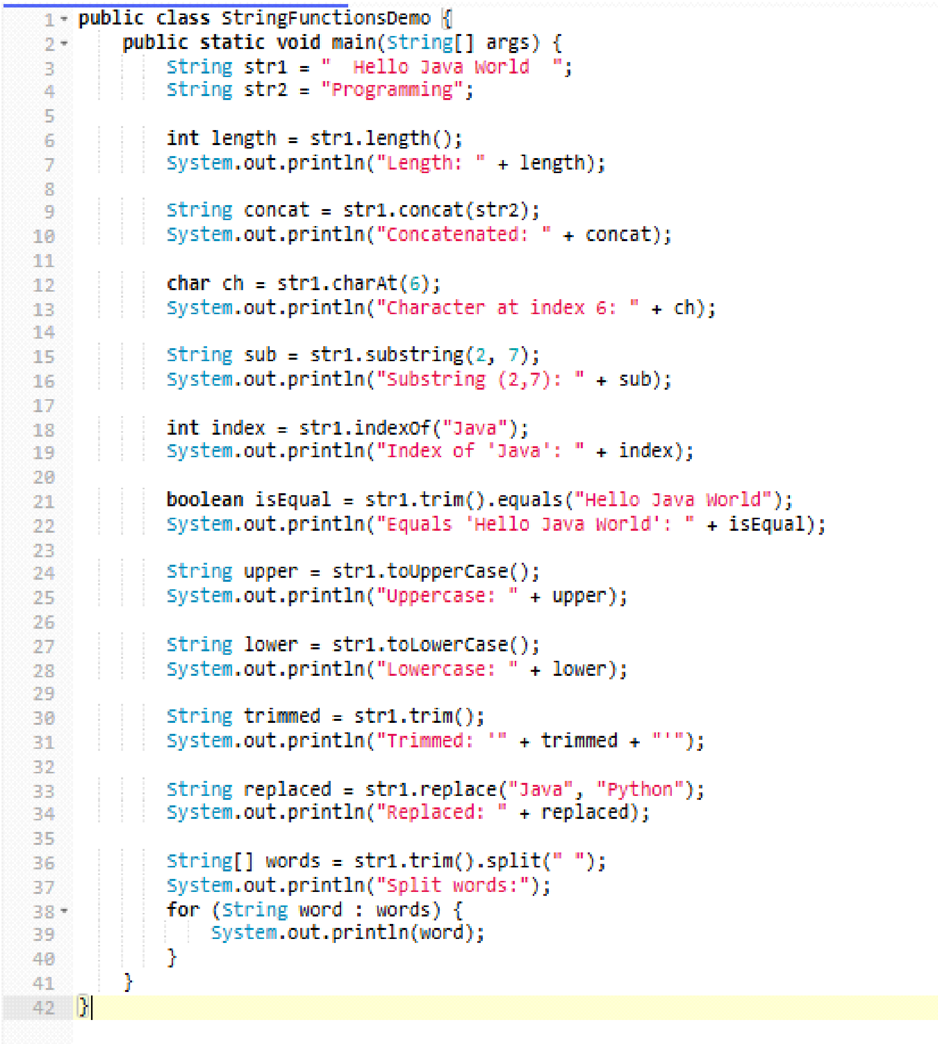


```java
class AgeException extends Exception {
    public AgeException(String message) {
        super(message);
    }
}

public class AgeCheck {
    public static void checkAge(int age) throws AgeException {
        if (age <= 18) {
            throw new AgeException("Age must be above 18.");
        } else {
            System.out.println("Age is valid.");
        }
    }

    public static void main(String[] args) {
        try {
            checkAge(16);
        } catch (AgeException e) {
            System.out.println("Exception caught: " + e.getMessage());
        }
    }
}
// Yash Raj Yadav
```

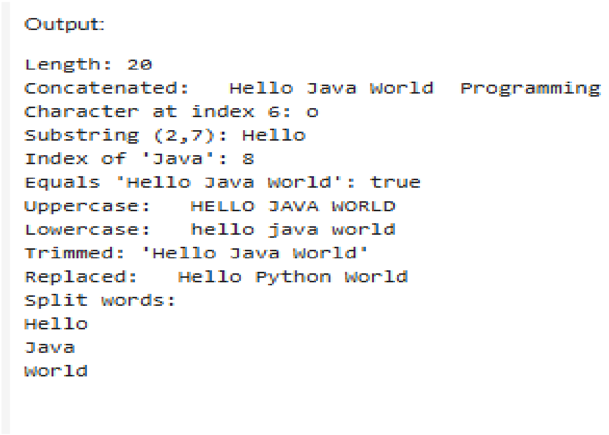
Output:

```
Exception caught: Age must be above 18.
```

- **Create a Java program that demonstrates various string functions and string handling techniques in Java. This program includes common operations like: Length of a string, Concatenation, Character extraction, Substring, Searching, String comparison, Changing case, Trimming, Replacing, Splitting**



```java
public class StringFunctionsDemo {
    public static void main(String[] args) {
        String str1 = "  Hello Java World  ";
        String str2 = "Programming";

        int length = str1.length();
        System.out.println("Length: " + length);

        String concat = str1.concat(str2);
        System.out.println("Concatenated: " + concat);

        char ch = str1.charAt(6);
        System.out.println("Character at index 6: " + ch);

        String sub = str1.substring(2, 7);
        System.out.println("Substring (2,7): " + sub);

        int index = str1.indexOf("Java");
        System.out.println("Index of 'Java': " + index);

        boolean isEqual = str1.trim().equals("Hello Java World");
        System.out.println("Equals 'Hello Java World': " + isEqual);

        String upper = str1.toUpperCase();
        System.out.println("Uppercase: " + upper);

        String lower = str1.toLowerCase();
        System.out.println("Lowercase: " + lower);

        String trimmed = str1.trim();
        System.out.println("Trimmed: '" + trimmed + "'");

        String replaced = str1.replace("Java", "Python");
        System.out.println("Replaced: " + replaced);

        String[] words = str1.trim().split(" ");
        System.out.println("Split words:");
        for (String word : words) {
            System.out.println(word);
        }
    }
}
```



Output:

```
Length: 20
Concatenated:   Hello Java World  Programming
Character at index 6: o
Substring (2,7): Hello
Index of 'Java': 8
Equals 'Hello Java World': true
Uppercase:   HELLO JAVA WORLD
Lowercase:   hello java world
Trimmed: 'Hello Java World'
Replaced:   Hello Python World
Split words:
Hello
Java
World
```