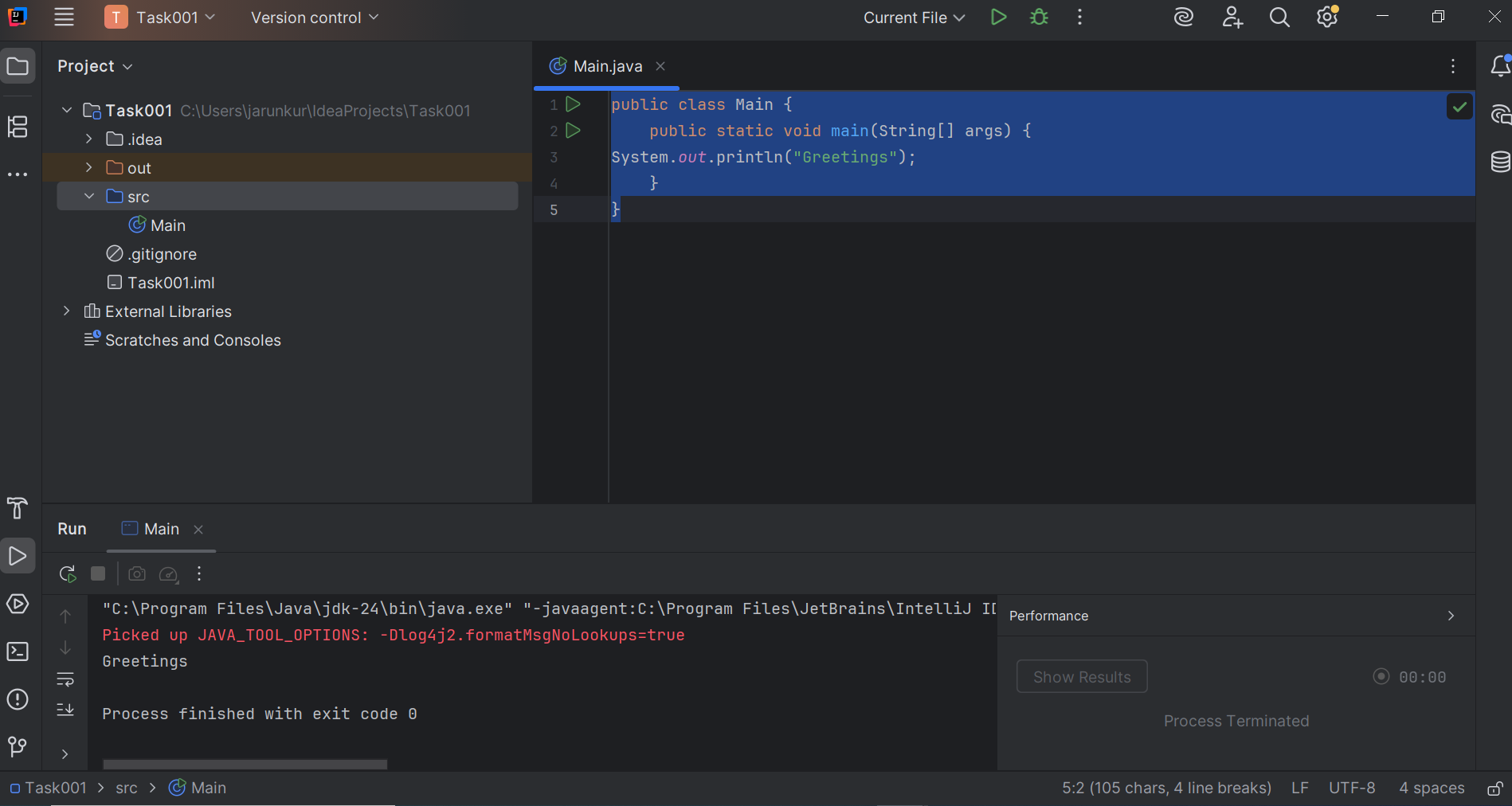
Task001:

Wap to display greetings

public class Main {  
 public static void main(String[] args) {  
System.*out*.println("Greetings");  
 }  
}

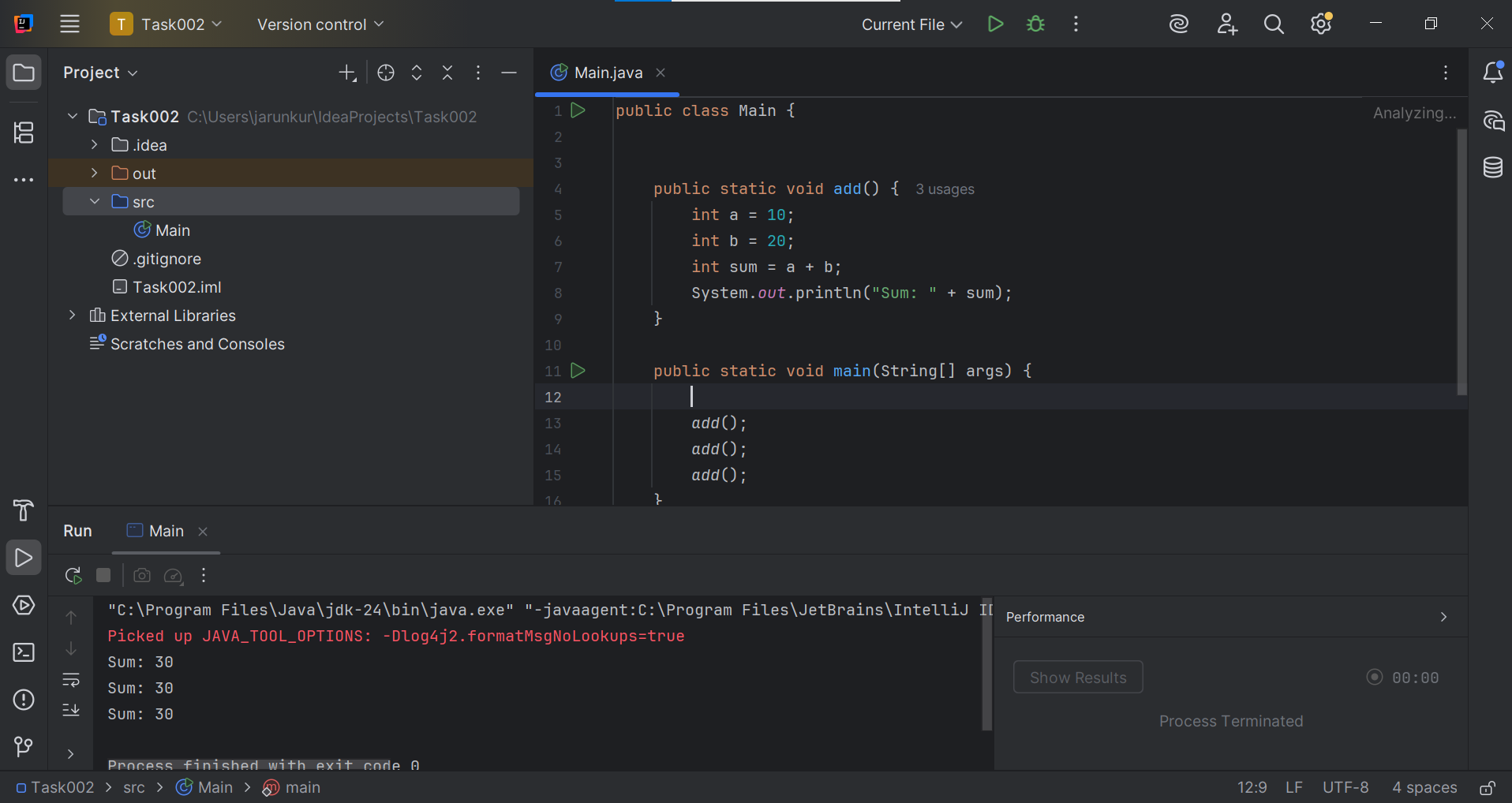


Task002:

Wap to create a add method and call the method 3 times ..

Hint in method add declare variables and display them

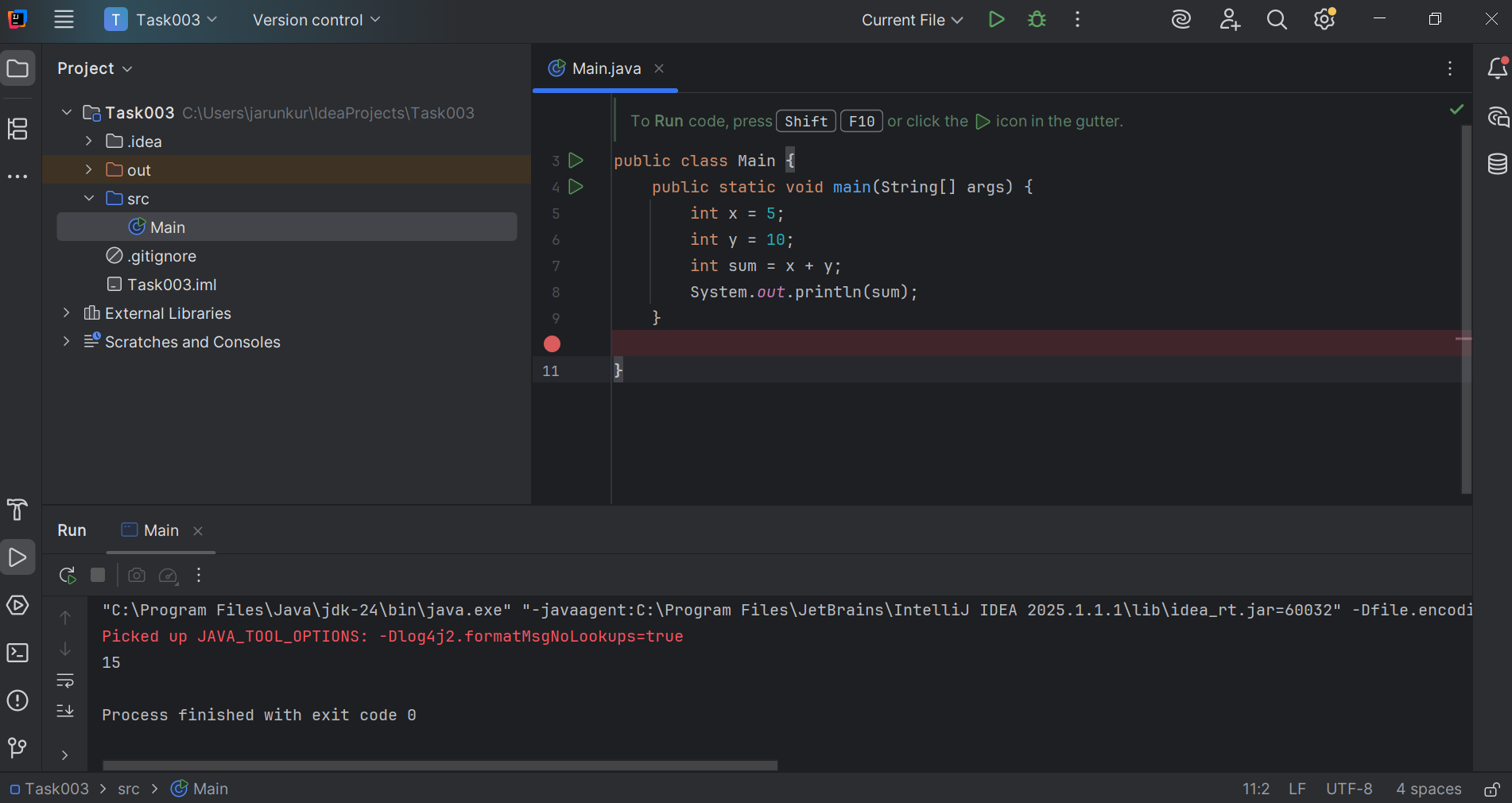
public class Main {  
  
  
 public static void add() {  
 int a = 10;  
 int b = 20;  
 int sum = a + b;  
 System.*out*.println("Sum: " + sum);  
 }  
  
 public static void main(String[] args) {  
   
 *add*();  
 *add*();  
 *add*();  
 }  
}



Task003

 Write a Program in Java to Add two Numbers.

public class Main {  
 public static void main(String[] args) {  
 int x = 5;  
 int y = 10;  
 int sum = x + y;  
 System.*out*.println(sum);  
 }  
  
}



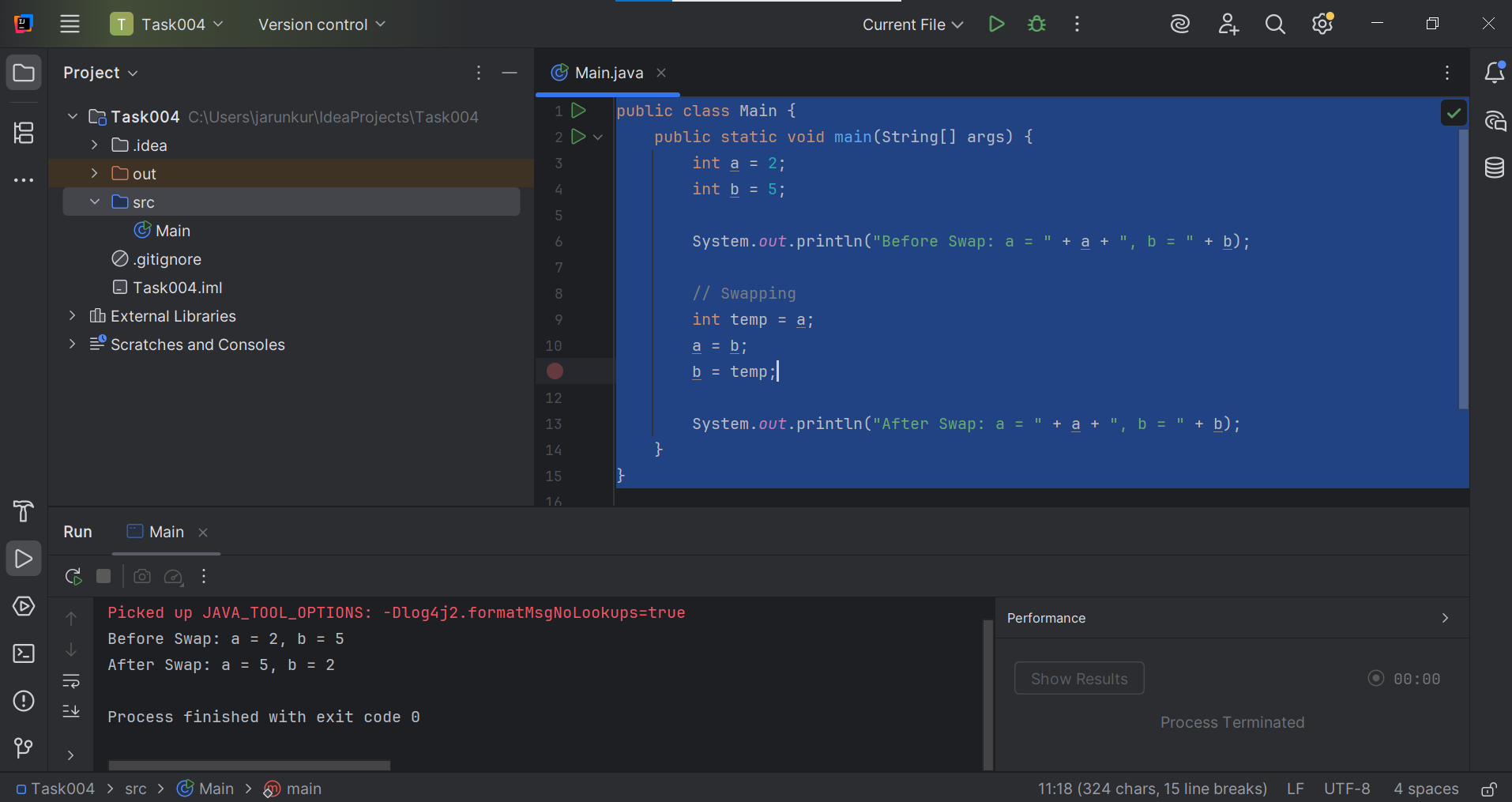
Task004

Write a Program to Swap Two Numbers

Input: a=2  b=5

Output: a=5  b=2

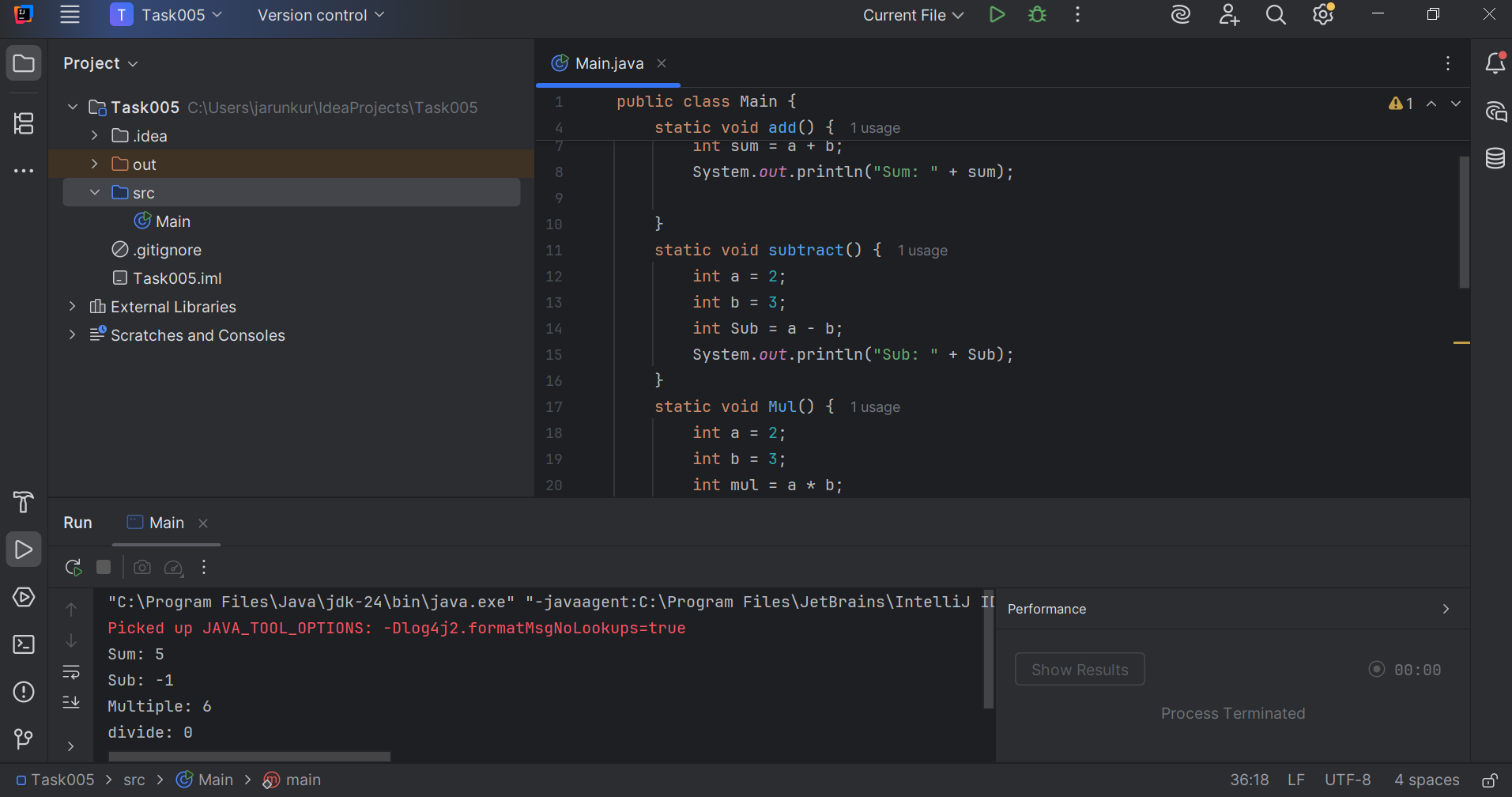
public class Main {  
 public static void main(String[] args) {  
 int a = 2;  
 int b = 5;  
  
 System.*out*.println("Before Swap: a = " + a + ", b = " + b);  
  
 // Swapping  
 int temp = a;  
 a = b;  
 b = temp;  
  
 System.*out*.println("After Swap: a = " + a + ", b = " + b);  
 }  
}



Task005

Create a code in which you have 4 methods add, subtract, multiply and divide (return type int) with a main [method..to](http://method..to) call all the other methods

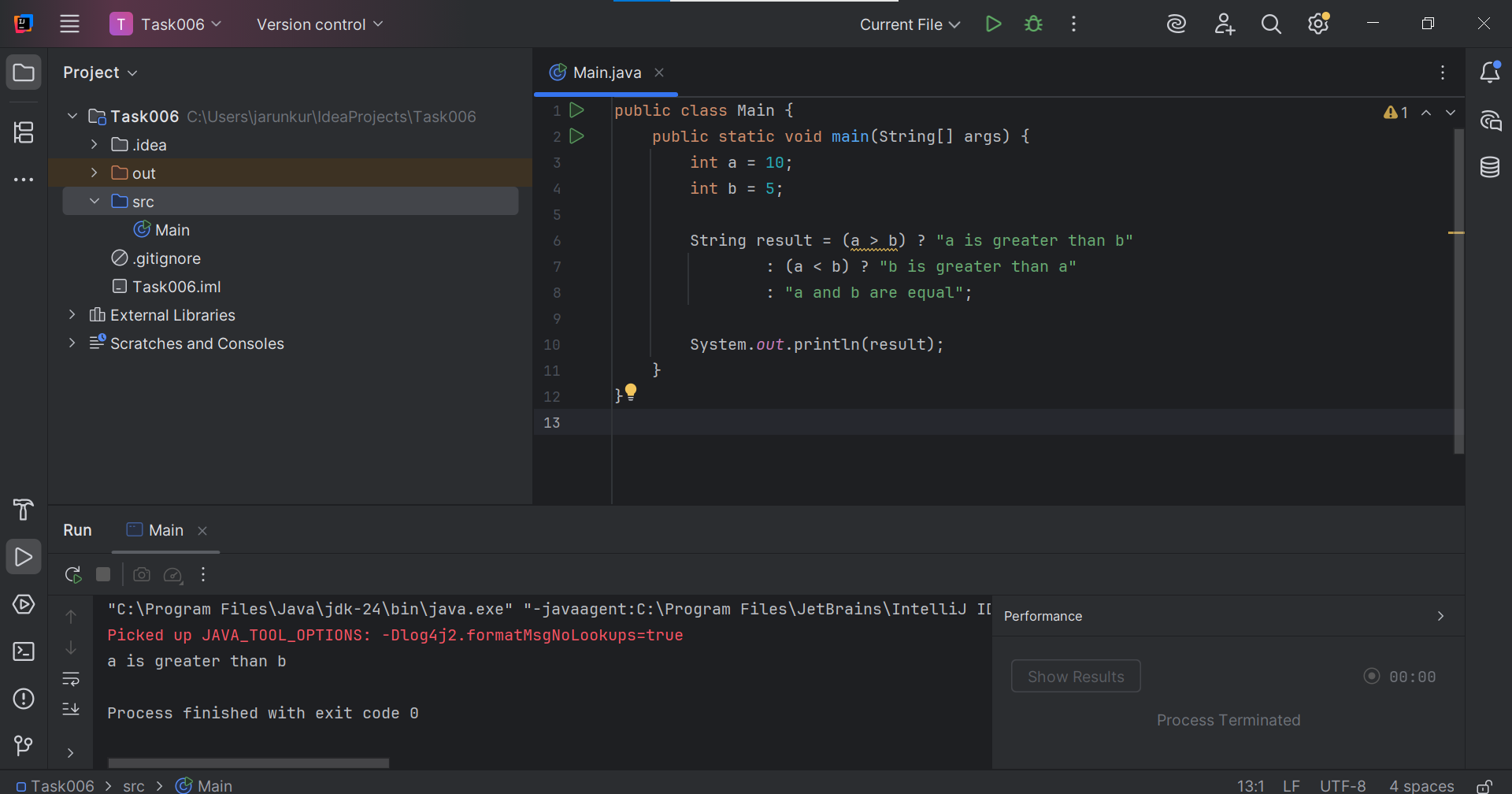
public class Main {  
  
   
 static void add() {  
 int a = 2;  
 int b = 3;  
 int sum = a + b;  
 System.*out*.println("Sum: " + sum);  
  
 }  
 static void subtract() {  
 int a = 2;  
 int b = 3;  
 int Sub = a - b;  
 System.*out*.println("Sub: " + Sub);  
 }  
 static void Mul() {  
 int a = 2;  
 int b = 3;  
 int mul = a \* b;  
 System.*out*.println("Multiple: " + mul);  
 }  
  
 static void divide() {  
 int a = 2;  
 int b = 3;  
 int div = a / b;  
 System.*out*.println("divide: " + div);  
 }  
  
 public static void main(String[] args) {  
  
 *add*();  
 *subtract*();  
 *Mul*();  
 *divide*();  
  
  
 }  
}



Task006

Write a program to check if a is greater or b.. Use ternary op

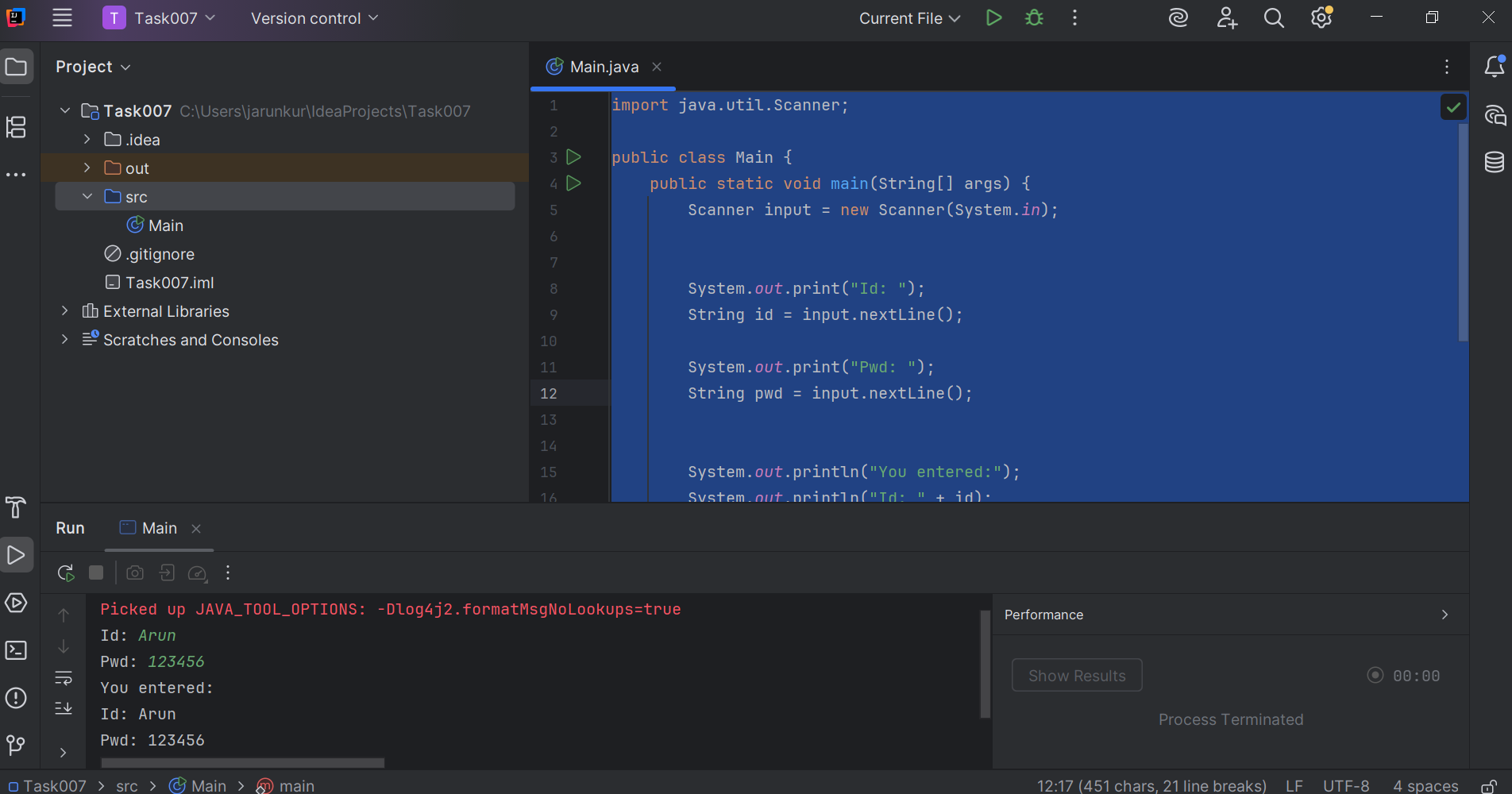
public class Main {  
 public static void main(String[] args) {  
 int a = 10;  
 int b = 5;  
  
 String result = (a > b) ? "a is greater than b"  
 : (a < b) ? "b is greater than a"  
 : "a and b are equal";  
  
 System.*out*.println(result);  
 }  
}



Task007

Write a program to take input from the user and display it to the user

import java.util.Scanner;  
  
public class Main {  
 public static void main(String[] args) {  
 Scanner input = new Scanner(System.*in*);  
  
  
 System.*out*.print("Id: ");  
 String id = input.nextLine();  
  
 System.*out*.print("Pwd: ");  
 String pwd = input.nextLine();  
  
  
 System.*out*.println("You entered:");  
 System.*out*.println("Id: " + id);  
 System.*out*.println("Pwd: " + pwd);  
  
 input.close();  
 }  
}

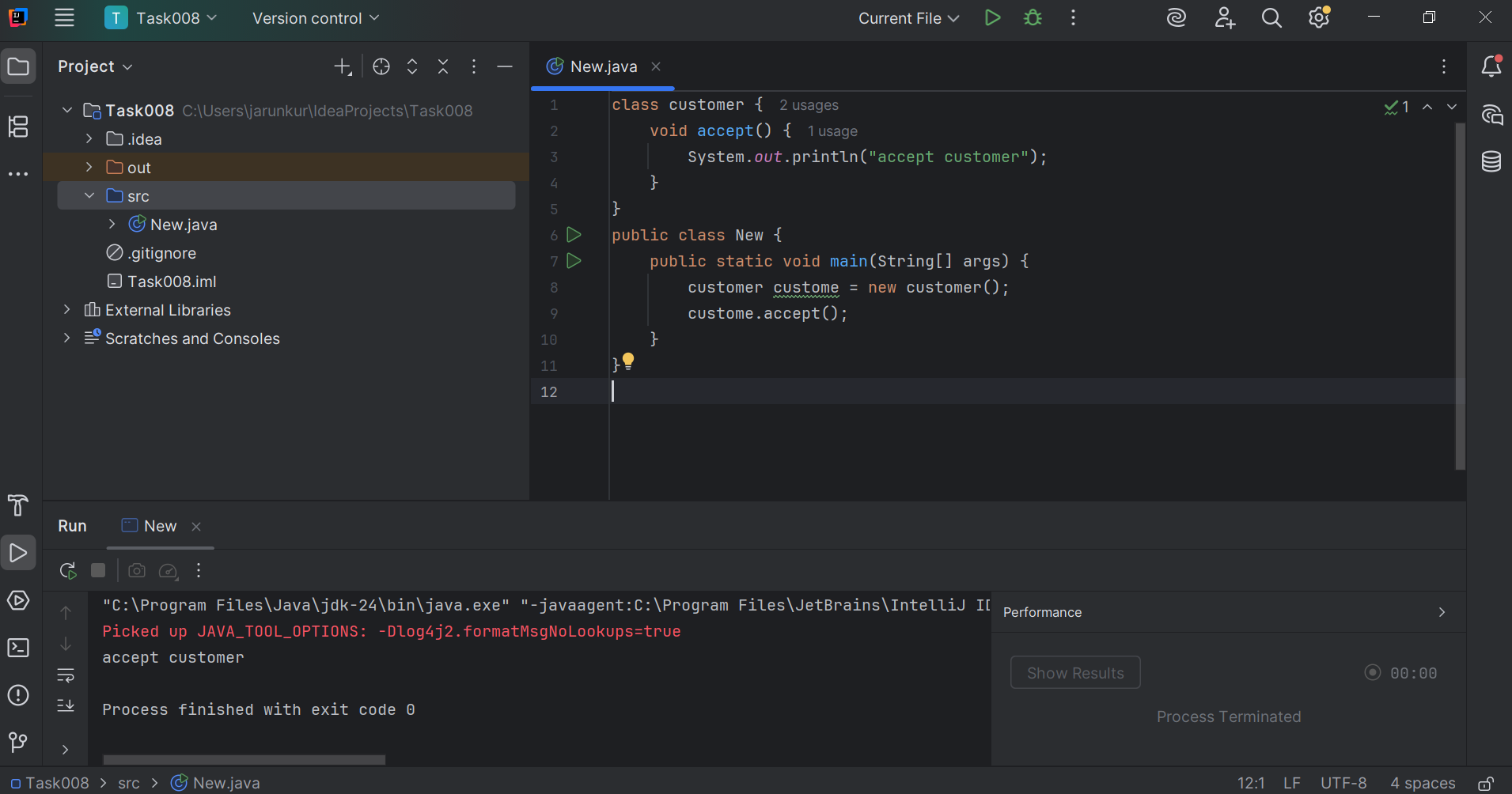


Task008

Write a program to create a class named Customer

Call the customer class in Task008 class using an object.

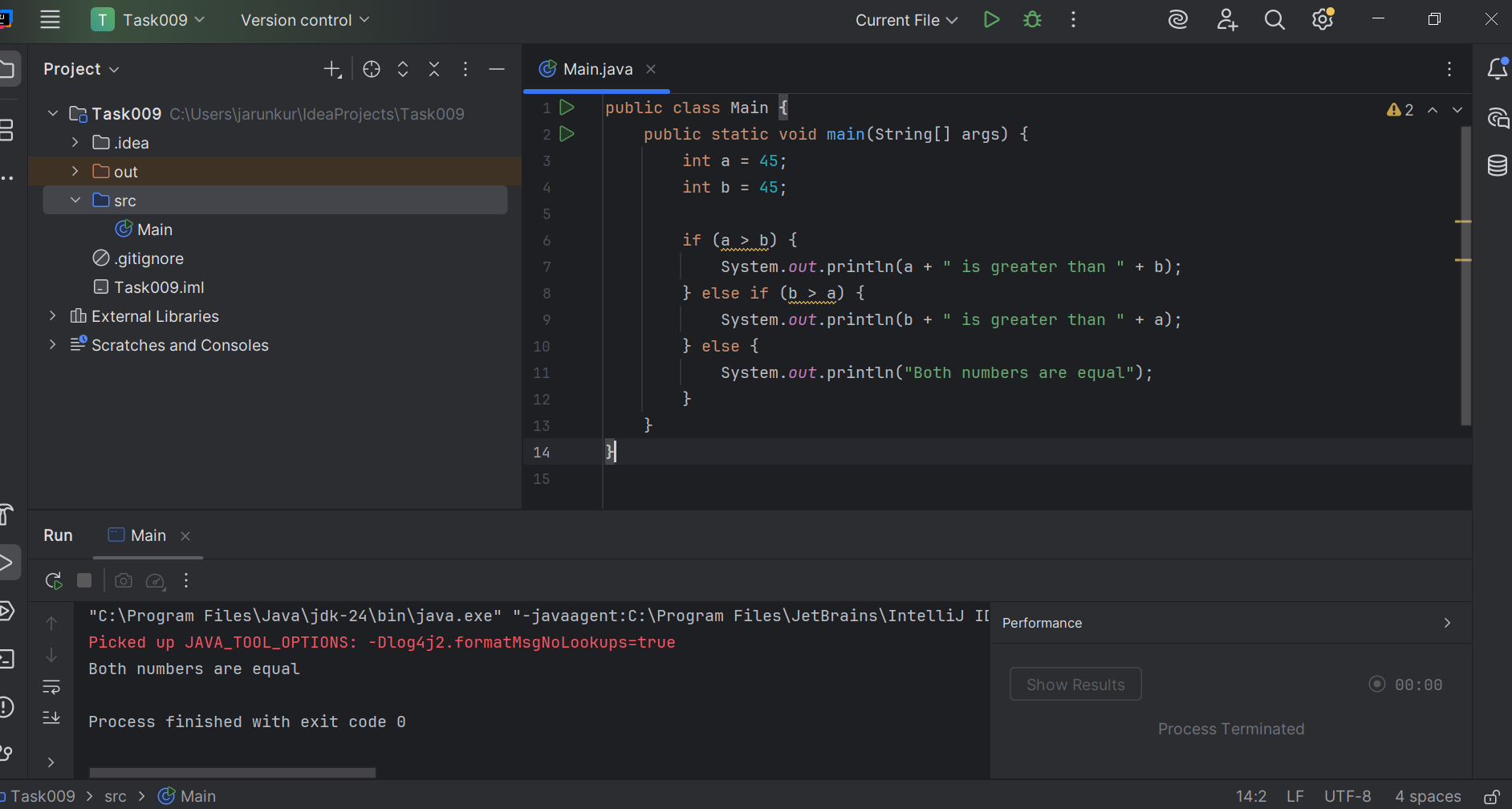
class customer {  
 void accept() {  
 System.*out*.println("accept customer");  
 }  
}  
public class New {  
 public static void main(String[] args) {  
 customer custome = new customer();  
 custome.accept();  
 }  
}



Task009:

Wap to check the greater of 2 numbers

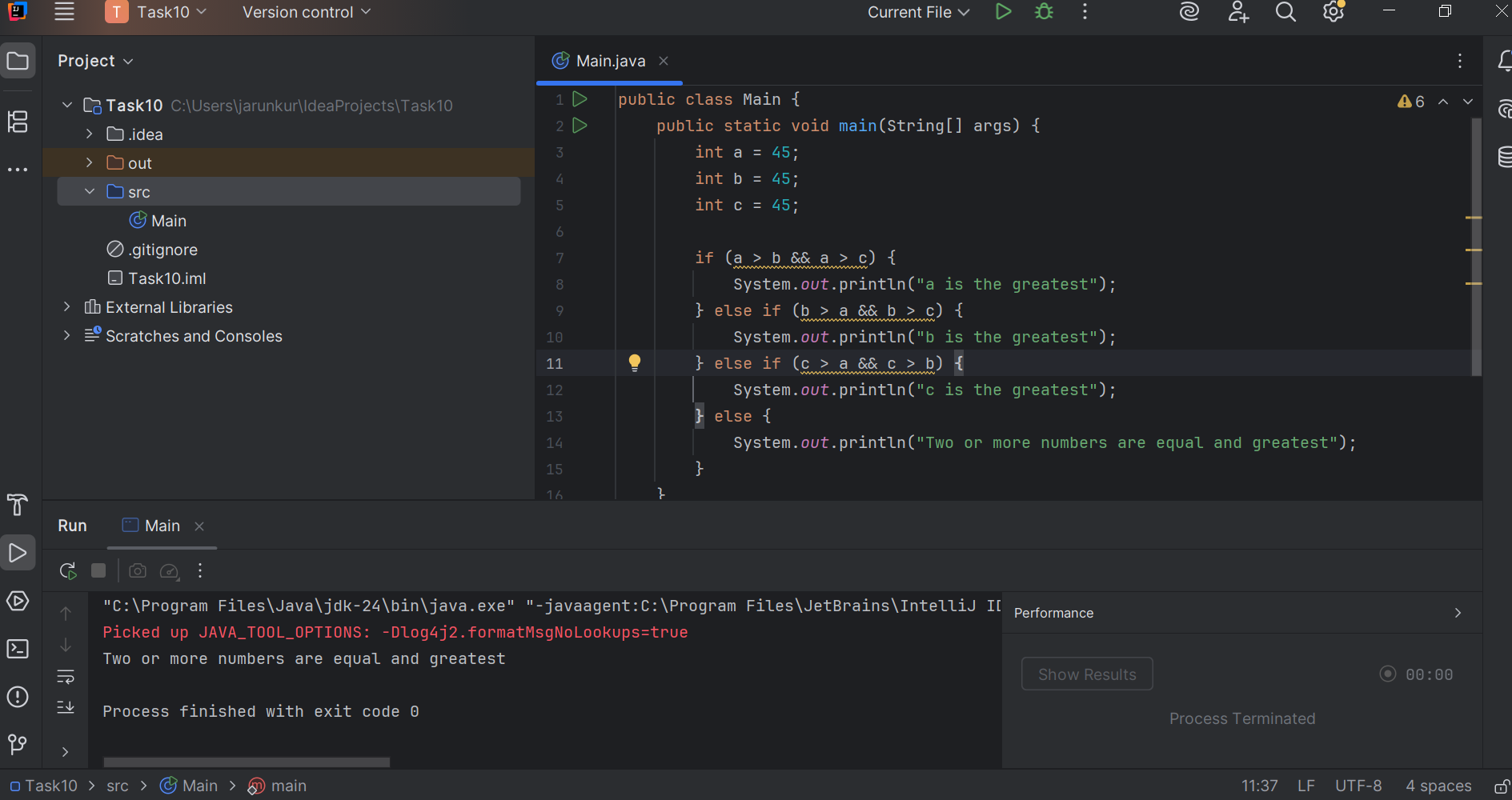
public class Main {  
 public static void main(String[] args) {  
 int a = 45;  
 int b = 45;  
  
 if (a > b) {  
 System.*out*.println(a + " is greater than " + b);  
 } else if (b > a) {  
 System.*out*.println(b + " is greater than " + a);  
 } else {  
 System.*out*.println("Both numbers are equal");  
 }  
 }  
}



task 010

Wap to check greater of 3 numbers

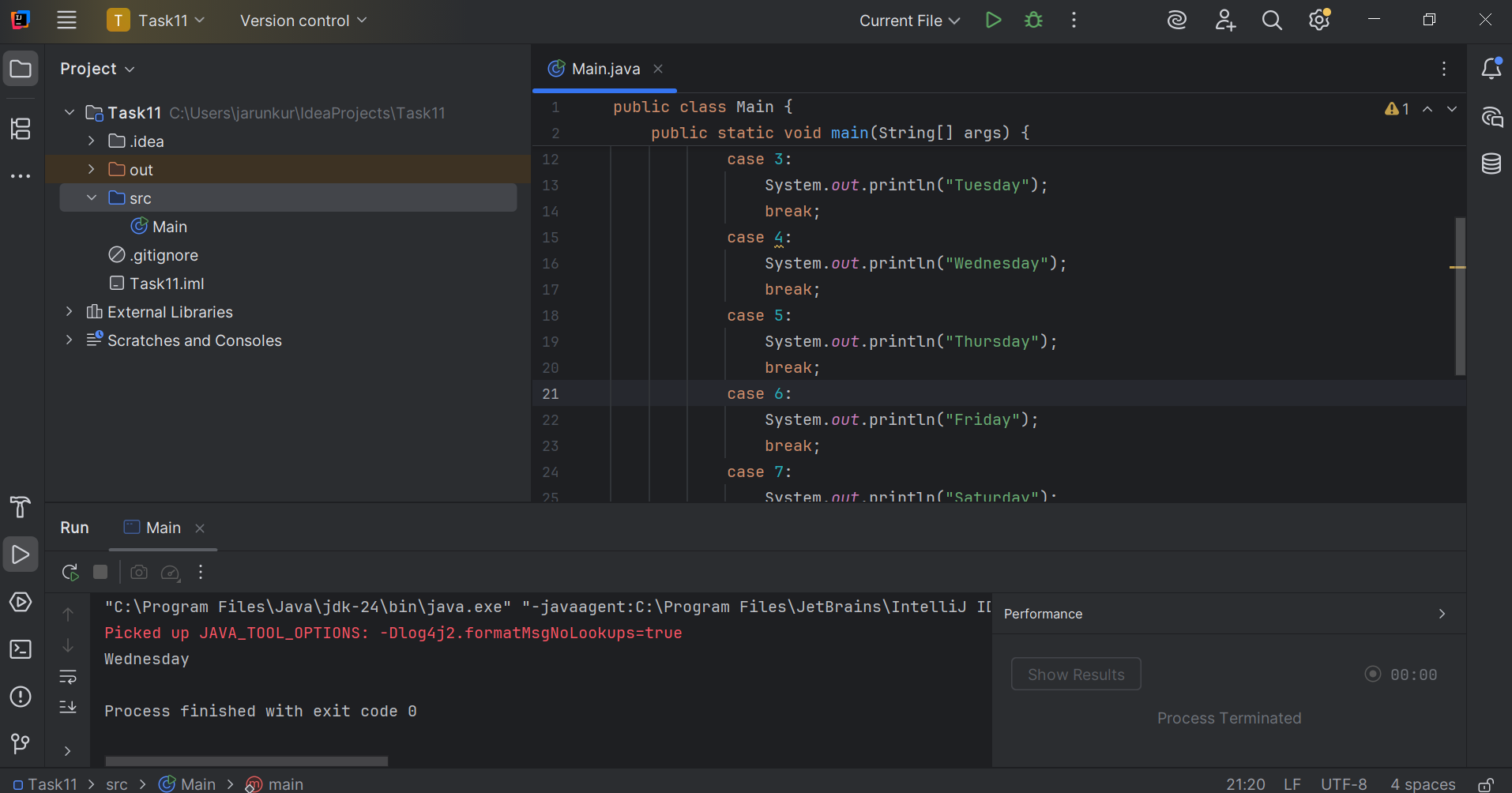
public class Main {  
 public static void main(String[] args) {  
 int a = 45;  
 int b = 45;  
 int c = 45;  
  
 if (a > b && a > c) {  
 System.*out*.println("a is the greatest");  
 } else if (b > a && b > c) {  
 System.*out*.println("b is the greatest");  
 } else if (c > a && c > b) {  
 System.*out*.println("c is the greatest");  
 } else {  
 System.*out*.println("Two or more numbers are equal and greatest");  
 }  
 }  
}



Task11:

Wap to check if  week days

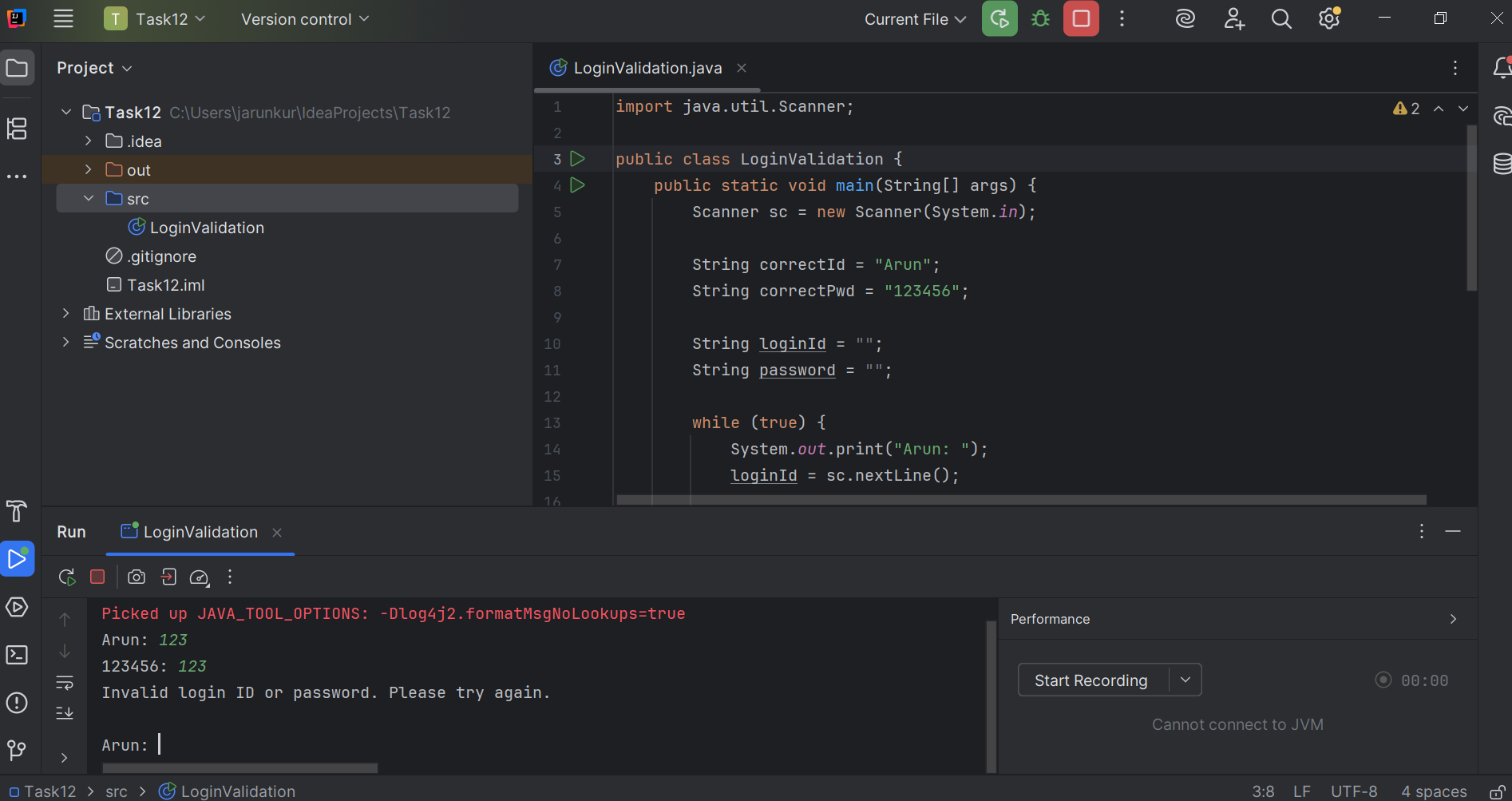
public class Main {  
 public static void main(String[] args) {  
 int day = 4; // change this number to test  
  
 switch(day) {  
 case 1:  
 System.*out*.println("Sunday");  
 break;  
 case 2:  
 System.*out*.println("Monday");  
 break;  
 case 3:  
 System.*out*.println("Tuesday");  
 break;  
 case 4:  
 System.*out*.println("Wednesday");  
 break;  
 case 5:  
 System.*out*.println("Thursday");  
 break;  
 case 6:  
 System.*out*.println("Friday");  
 break;  
 case 7:  
 System.*out*.println("Saturday");  
 break;  
 default:  
 System.*out*.println("Invalid day");  
 }  
 }  
}



Task 012

Wap to check loginid and password validation

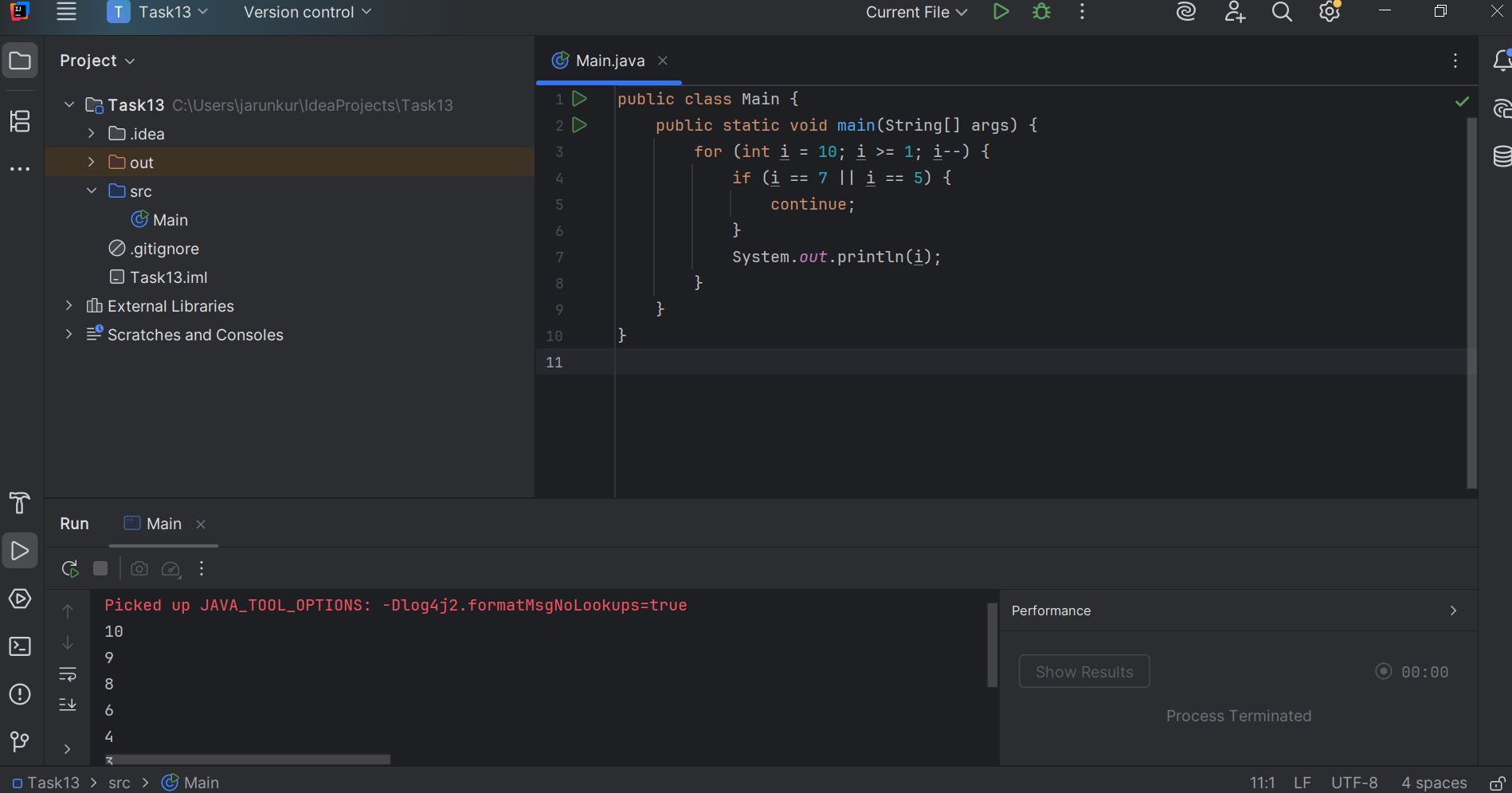
import java.util.Scanner;  
  
public class LoginValidation {  
 public static void main(String[] args) {  
 Scanner sc = new Scanner(System.*in*);  
  
 String correctId = "Arun";  
 String correctPwd = "123456";  
  
 String loginId = "";  
 String password = "";  
  
 while (true) {  
 System.*out*.print("Arun: ");  
 loginId = sc.nextLine();  
  
 System.*out*.print("123456: ");  
 password = sc.nextLine();  
  
 if (loginId.equals(correctId) && password.equals(correctPwd)) {  
 System.*out*.println("Login successful!");  
 break;  
 } else {  
 System.*out*.println("Invalid login ID or password. Please try again.\n");  
 }  
 }  
 sc.close();  
 }  
}



Task 13:

Wap to display numbers from 10 to 1 .. skip 7 and 5.

public class Main {  
 public static void main(String[] args) {  
 for (int i = 10; i >= 1; i--) {  
 if (i == 7 || i == 5) {  
 continue;  
 }  
 System.*out*.println(i);  
 }  
 }  
}



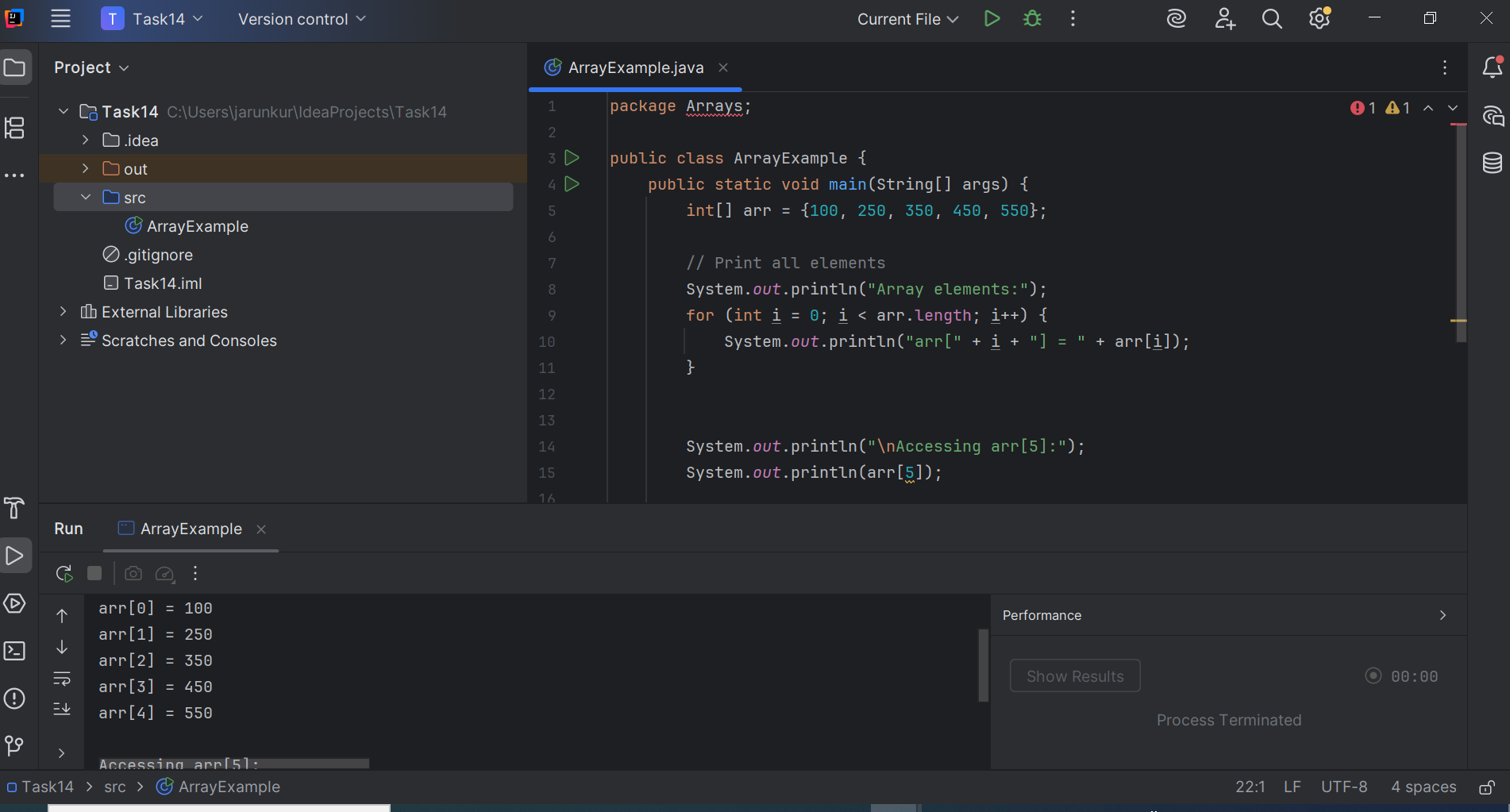
Task 014:

Arrays:

Try the below code and display the output…

Now play with it try to access arr of 5th index and see the output…and try to access arr of -1 index and see the output..

package Arrays;  
  
public class ArrayExample {  
 public static void main(String[] args) {  
 int[] arr = {100, 250, 350, 450, 550};  
  
 // Print all elements  
 System.*out*.println("Array elements:");  
 for (int i = 0; i < arr.length; i++) {  
 System.*out*.println("arr[" + i + "] = " + arr[i]);  
 }  
  
  
 System.*out*.println("\nAccessing arr[5]:");  
 System.*out*.println(arr[5]);  
  
  
 System.*out*.println("\nAccessing arr[-1]:");  
 System.*out*.println(arr[-1]);  
 }  
}

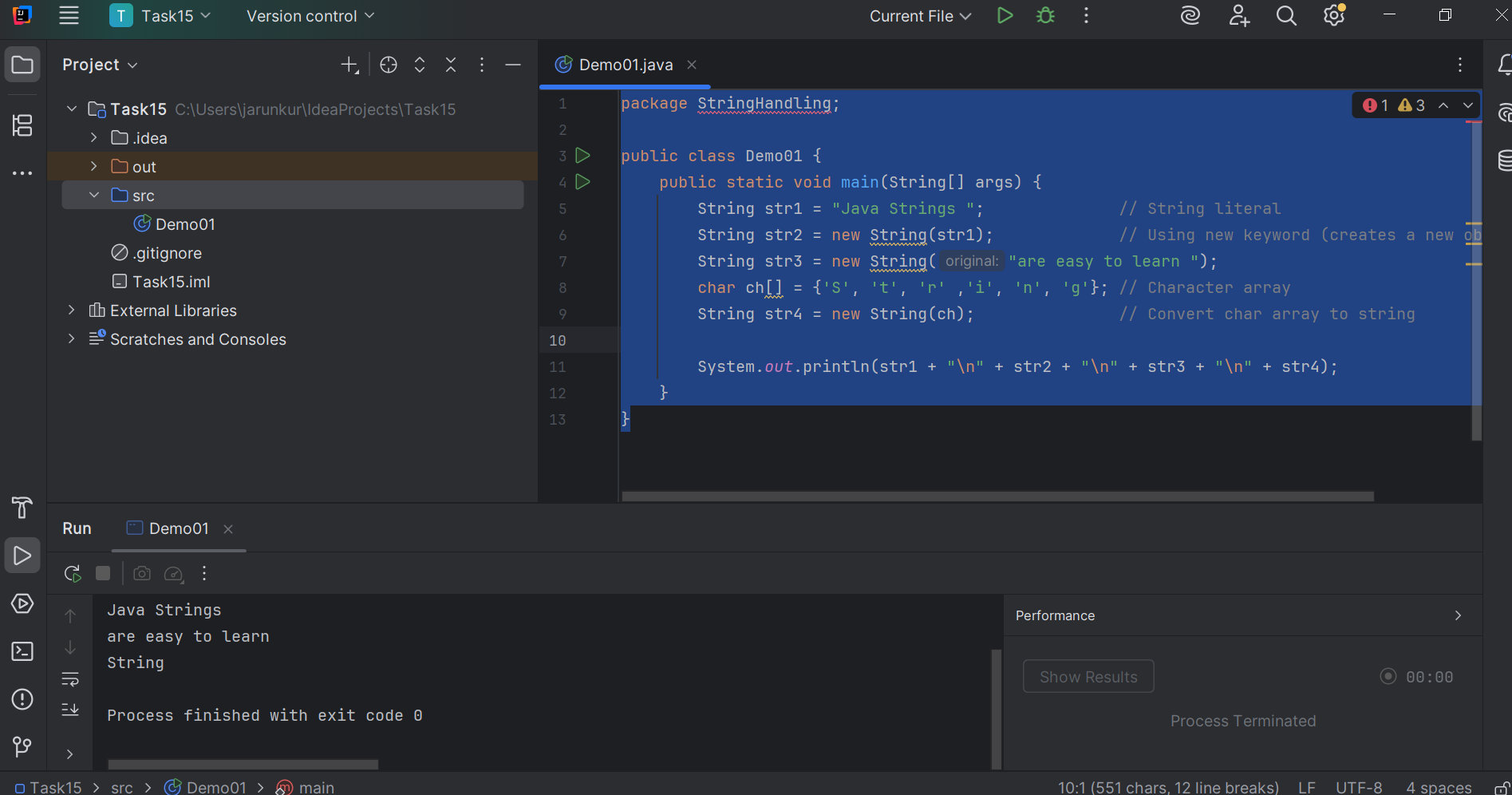


Task 015:

String – non primitive data type —> collection of characters or any value within “ ”

– immutable ⇒ cannot be changed

package StringHandling;  
  
public class Demo01 {  
 public static void main(String[] args) {  
 String str1 = "Java Strings "; // String literal  
 String str2 = new String(str1); // Using new keyword (creates a new object)  
 String str3 = new String("are easy to learn ");  
 char ch[] = {'S', 't', 'r' ,'i', 'n', 'g'}; // Character array  
 String str4 = new String(ch); // Convert char array to string  
  
 System.*out*.println(str1 + "\n" + str2 + "\n" + str3 + "\n" + str4);  
 }  
}

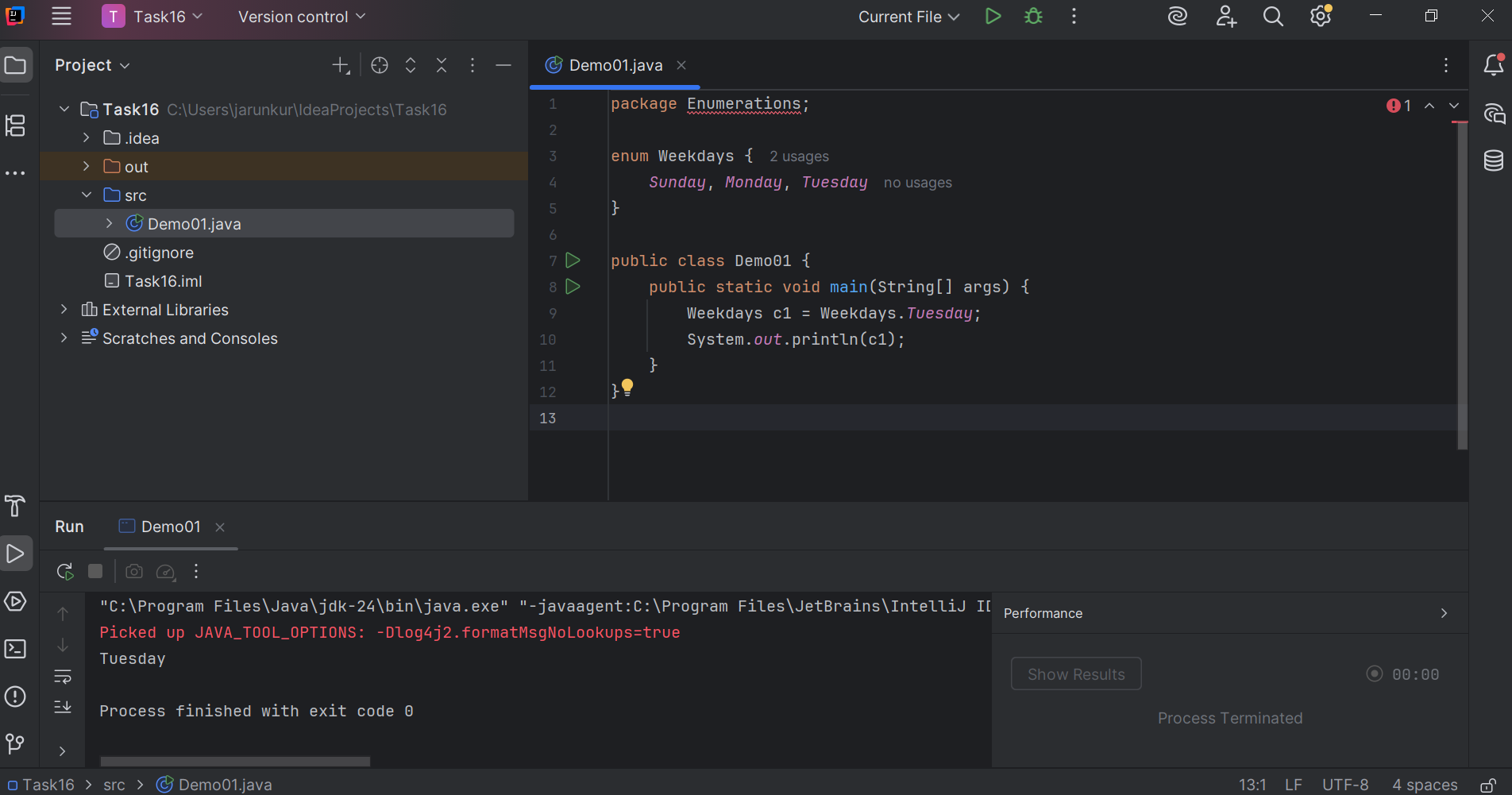


Task 016

Enums or Enumerations   – part of  collection framework

What is the output of the below code snippet

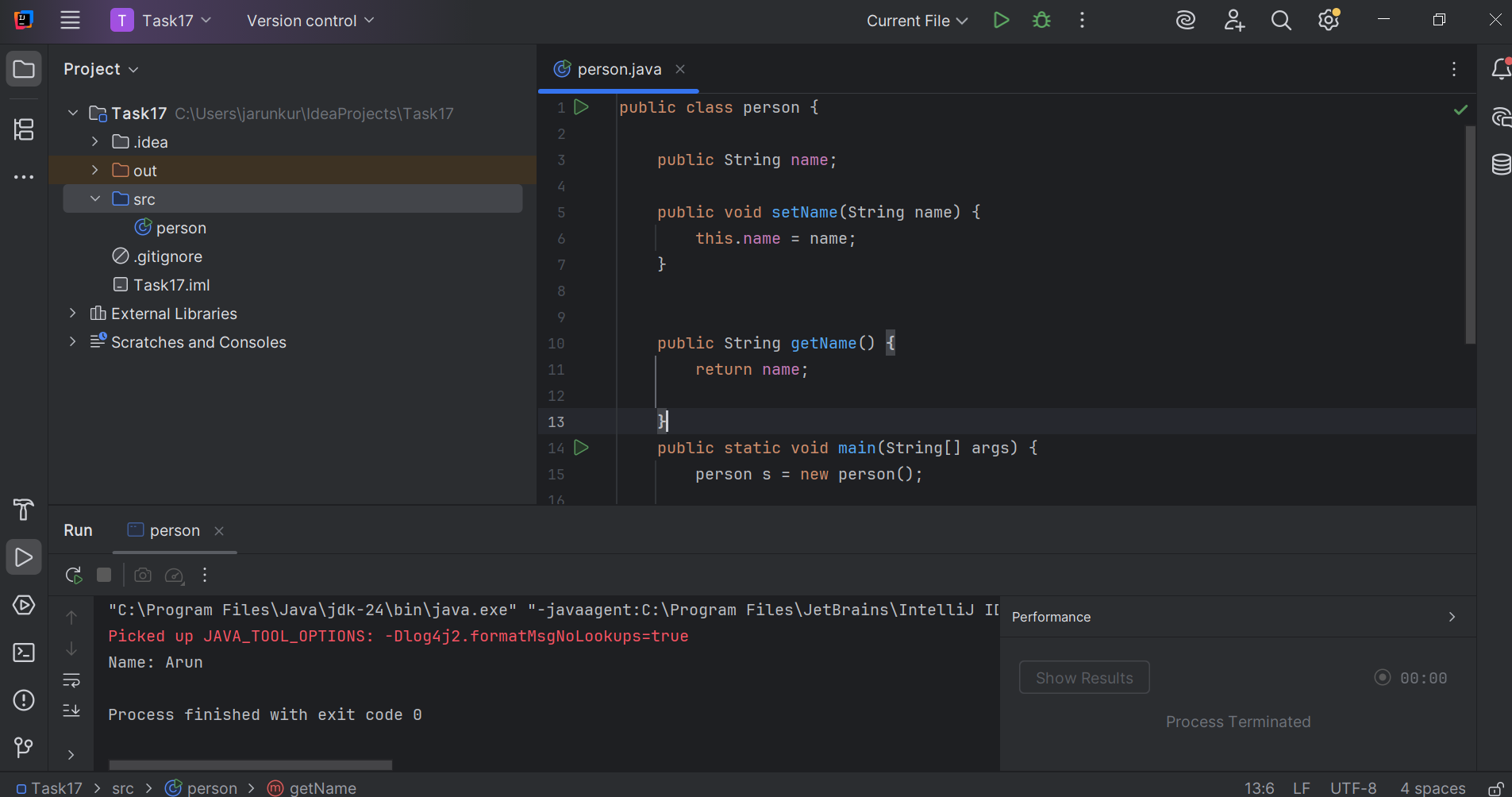
package Enumerations;  
  
enum Weekdays {  
 *Sunday*, *Monday*, *Tuesday*}  
  
public class Demo01 {  
 public static void main(String[] args) {  
 Weekdays c1 = Weekdays.*Tuesday*;  
 System.*out*.println(c1);  
 }  
}



Task 017:

Getter and setter

public class person {  
  
 public String name;  
  
 public void setName(String name) {  
 this.name = name;  
 }  
  
  
 public String getName() {  
 return name;  
  
 }  
 public static void main(String[] args) {  
 person s = new person();  
  
 s.setName("Arun");  
  
 System.*out*.println("Name: " + s.getName());  
 }  
}



Task 018

Now create one more program named Task018.java

public class Main {

  public static void main(String[] args) {

    Person myObj = new Person();

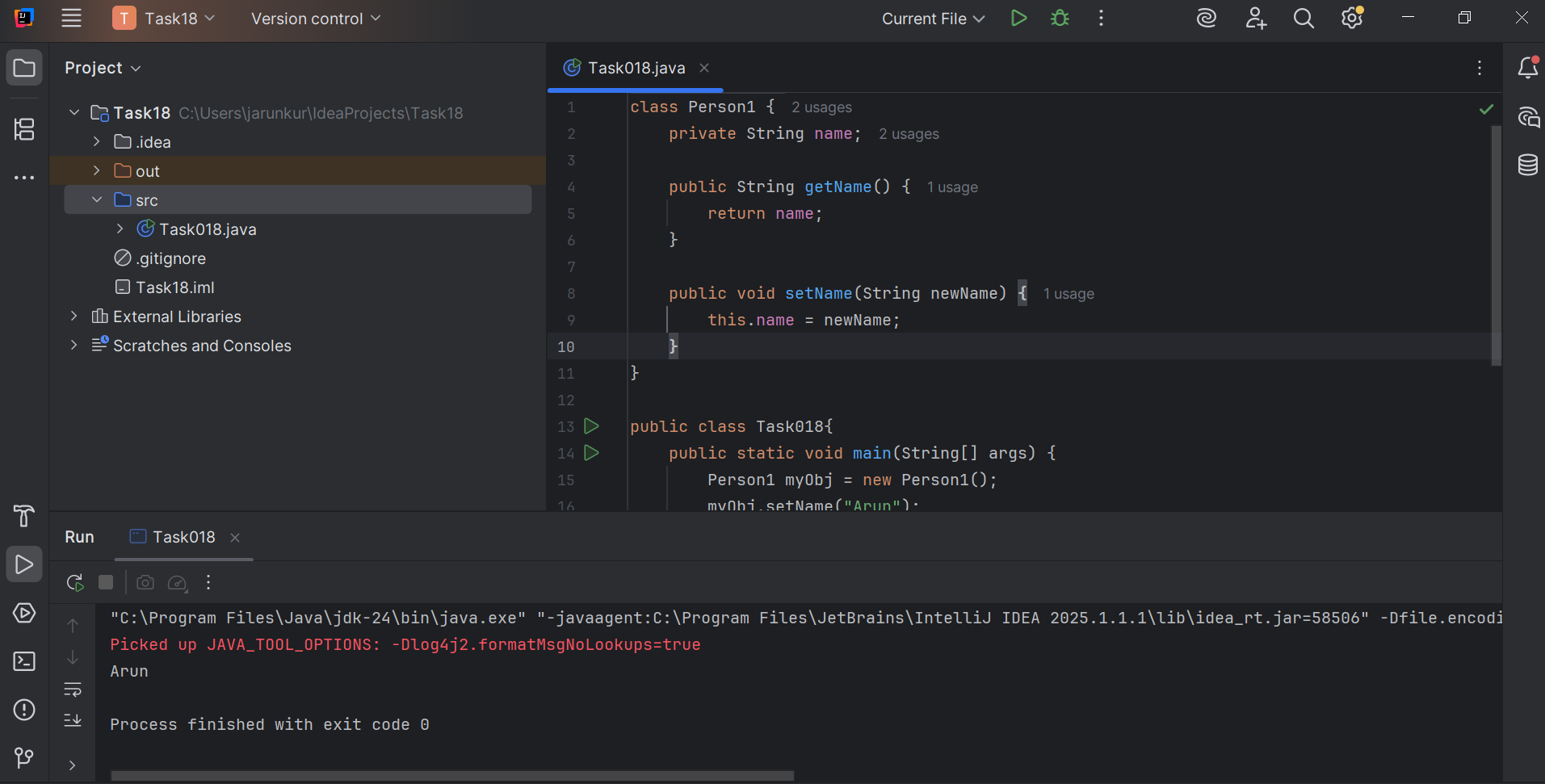
    myObj.setName("John");

    System.out.println(myObj.getName());

  }

}

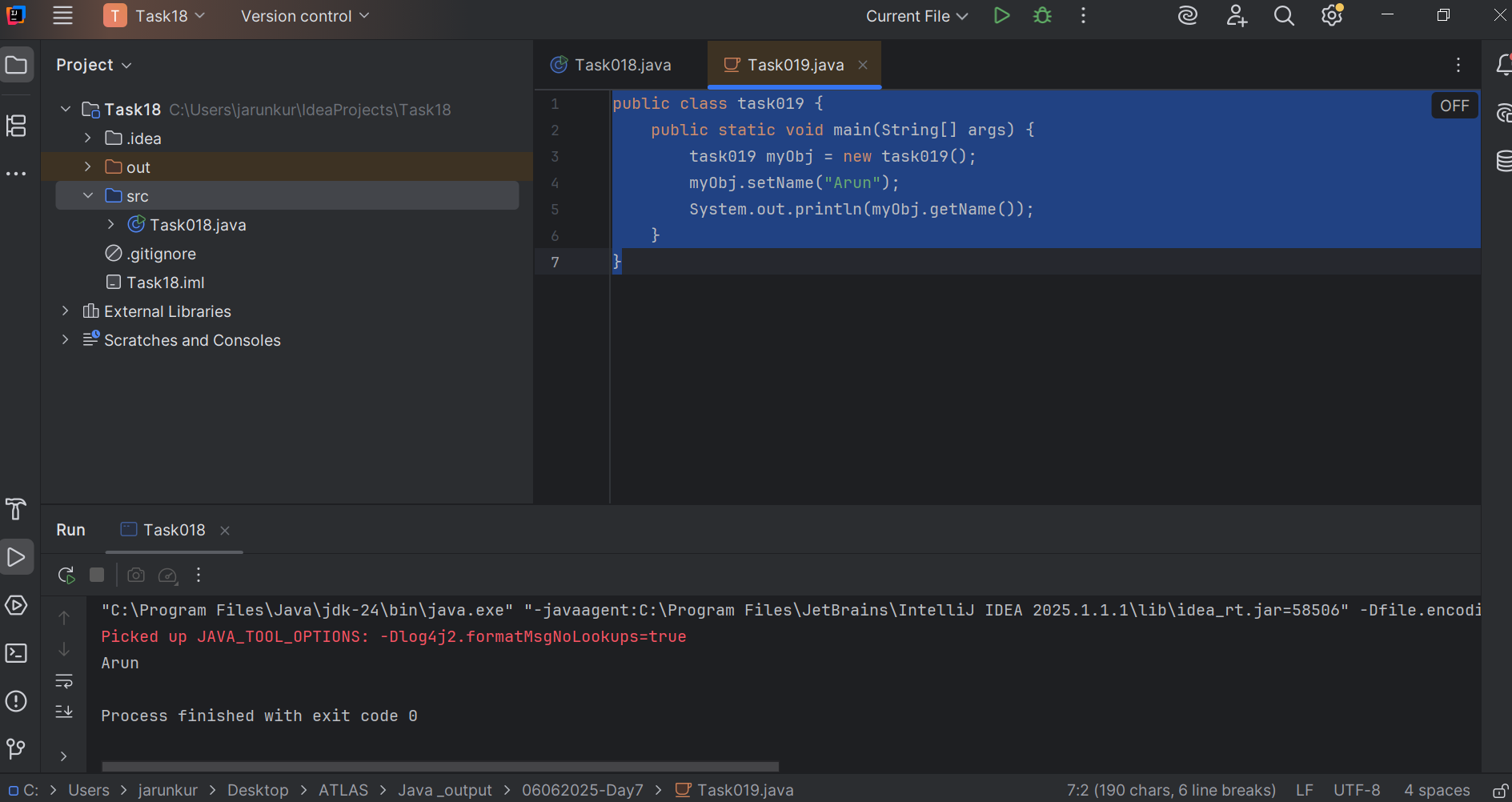
class Person1 {  
 private String name;  
  
 public String getName() {  
 return name;  
 }  
  
 public void setName(String newName) {  
 this.name = newName;  
 }  
}  
  
public class Task018{  
 public static void main(String[] args) {  
 Person1 myObj = new Person1();  
 myObj.setName("Arun");  
 System.*out*.println(myObj.getName());  
  
 }  
}



Task019.java

Wap to display the content of the above enum from the class Task016\_1 in this program.. (main  needs to be added)

public class task019 {  
 public static void main(String[] args) {  
 task019 myObj = new task019();  
 myObj.setName("Arun");  
 System.out.println(myObj.getName());  
 }  
}



Task 020:

Create an array of your name

public class NameArray {  
 public static void main(String[] args) {  
 // Initialize array with characters of the name  
 char[] Name = {'A', 'r', 'u', 'n'};  
  
 // Printing the array reference (not the contents)  
 System.*out*.println(Name); // This will print the name because char[] prints as string  
  
 int n = Name.length;  
 System.*out*.println("There are " + n + " letters in my name");  
  
 // Traversing array using for loop  
 for (int i = 0; i < n; i++) {  
 System.*out*.print(Name[i] + " ");  
 }  
 System.*out*.println(); // newline  
  
 // SHALLOW COPY  
 char[] shallowCopy = Name; // both arrays refer to the same object  
 shallowCopy[0] = 'X'; // modifying shallowCopy modifies Name too  
 System.*out*.print("After shallow copy change, Name: ");  
 for (int i = 0; i < n; i++) {  
 System.*out*.print(Name[i] + " ");  
 }  
 System.*out*.println();  
  
 // DEEP COPY  
 char[] deepCopy = new char[n];  
 for (int i = 0; i < n; i++) {  
 deepCopy[i] = Name[i];  
 }  
 deepCopy[0] = 'P'; // revert change only in deepCopy  
 System.*out*.print("Deep copy array: ");  
 for (int i = 0; i < n; i++) {  
 System.*out*.print(deepCopy[i] + " ");  
 }  
 System.*out*.println();  
 }  
}

