Day 7 - 28th May 2025

Employee ID: 201933938

Login ID: iamasif

Name: Shaik Asif

Java

| Java basics | Data Types, Variables, Operators, Control Statements, Loops, Arrays, Classes, Objects, |
| --- | --- |

Open labs

Then check for Java version

In cmd type

Java –version

or

Java -v

17 installed

Plz check for any of the ide’s is installed

Eclipse

Or intellij

Variable:

The variable is a place where we store our data — temporary… –

Name = “Prasunamba”

Age

Phone no

Email id

Data types:

Which type of value you are goinbf to store in a variable

Types

Primitive === > String, cant create an OBJECT, int , float ⇒ basic data types

Non primitive ==== > CAN create an object ⇒ derived data types

Int age = 10; == > declaring a variable and assigning a value to it (assignment operator =)

Int → data types

Age → variable

10 → value

Or

Int age ; —> declaring a variable

Age = 10; → assigning value to a variable

Float – decimal values

Operator:

Which performs operations

Operator operates on operands

X + y ======> x , y are variable also called as operands

= ====> operator

Types of operators 👍

Arithmetic operators ===> + ,-,\*,/, % (reminder)

Logical operators =====> and &&, or || , not !

Comparison opearations ===> ==

Assignment op =====> =

Bitwise ======> &, |, ^, !, >>, <<

Relational op ====> >,<,>=, <= , !=

Increment and decrewment op

Ternary op – : ?, a>b?a:b ======> if a>b — it wil return a or else b

=====================================================================================================================================================================================================================================

Psvm ====> hit tab ====> it slould display

main() function or method

Void ===> return type (void does ot return any value ===> return 0)

Int===> returns int value

Float ===> returns float value

Static ===>

Static X variables (temporary)

main=====> boss ====> entry and exit point off your program.

Before a variable ===> data type

Public ====> access modifiers ===> in java ( c++ ===> access Specifiers)

Access modifiers:

They will modify the access of the variables and the methods

Public === > every one ca access

Protectded =====> only inherited memberzs cab access

Private ====> only that class of method

Default ===> by default access ===> can modify later

Parameters ====>

Method ====> with parameters (String[] args)

and without parameters ()

Before a method ====> return type

Public static void main(String[] args)

=====> Java Virtual Machine or the JVM calls the main() method

===> while calling the main method give some parameters

====> javac File1.java

====> javac is java compiler (translator)

====> after compiling it will create a classf ile (file1.class)

Variables:

Class =====> start with capital letter ===> the class name which consists of main method should be the nam of your program. No spaces allowed

Method ======> always start with small letters No spaces allowed

Package is a folder

Import packages;

class Prog1{

//body of the class

psvm(string[] argos) {

// body of the main method.

Sout ====> tab ===> System.out.println(); || sout || sys

System.out.print(“Welcome onbroad”); ===> keeps the cursor at he same place..

System.out.println(“Hello all good luck!!”); ====> ln moves the cursor to the next line)

Int empid = 1001;

empid = 2001;

System.out.println(“Congratulations to $$$$ “ + empid + “on achieving the milestone”); + ===> concatenation operator

}

}

Compile and then execute ===> running the code

Task 1:

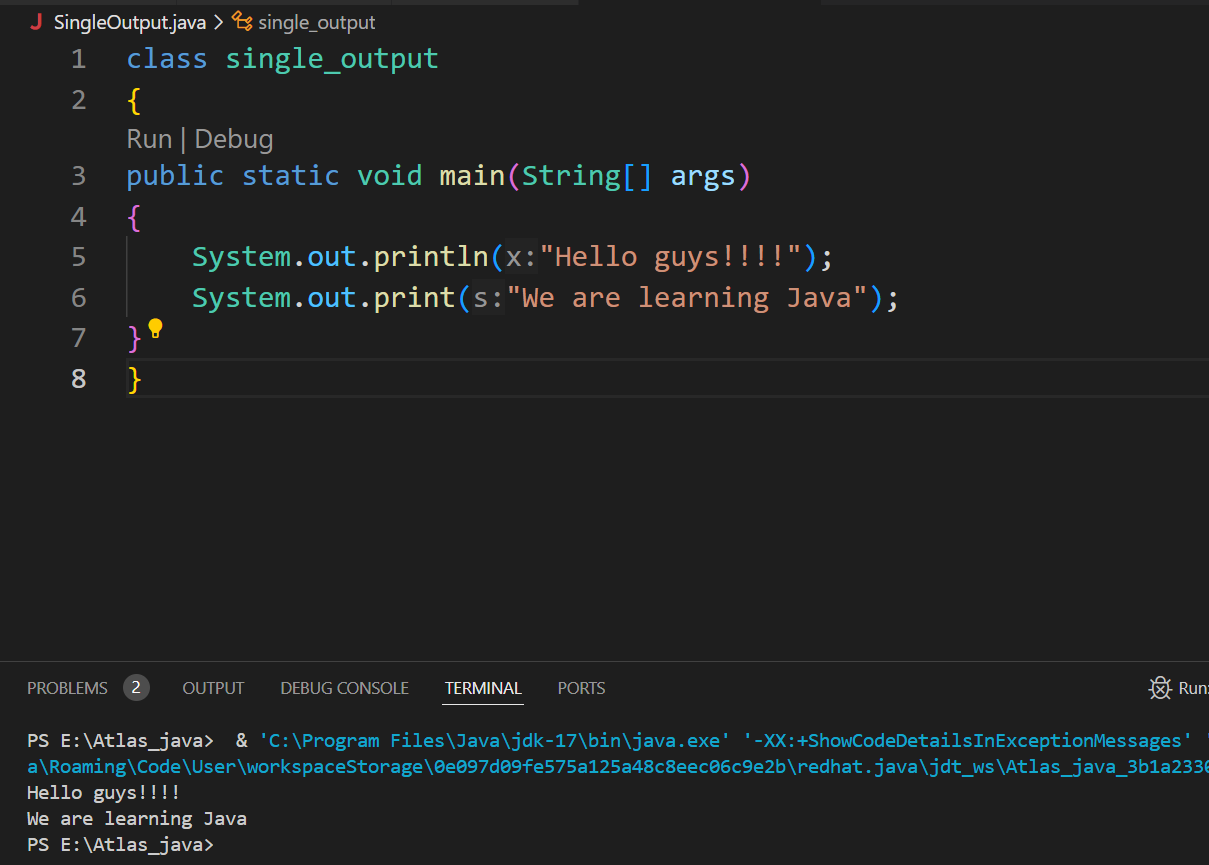
Create a code to display

“Hello guys!!!! “

“We are learning Java”

Using a single output statement…

5 min 11.36 to 11.41



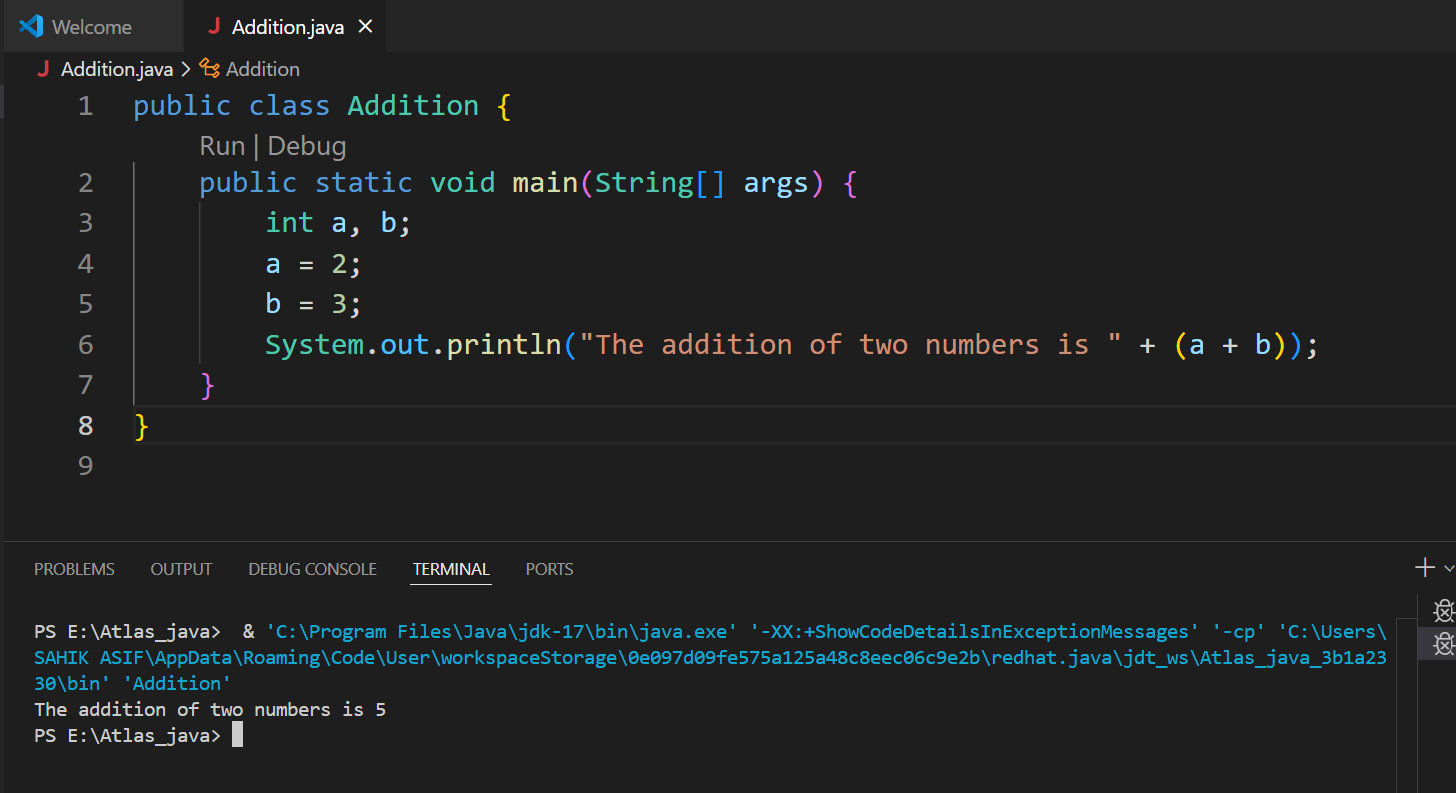
Task 2:

Write a Program in Java to Add two Numbers.

Input: 2 3

Output: 5

5 min 11.44 to 11.49



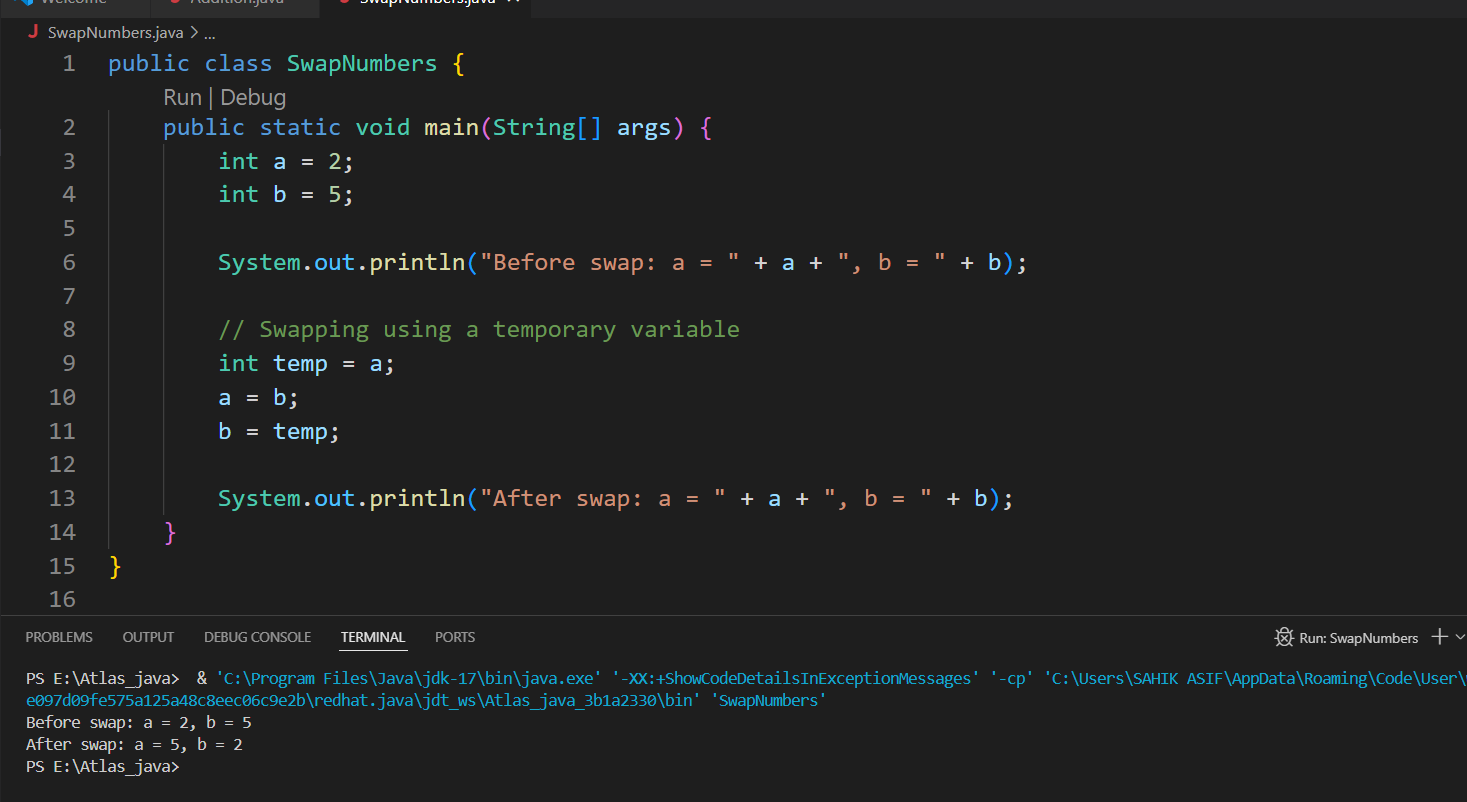
Task 003:

Write a Program to Swap Two Numbers

Input: a=2 b=5

Output: a=5 b=2

5 min 11.50 to 11.54



Task 004:

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) all all the other methods

Out put:

Main started

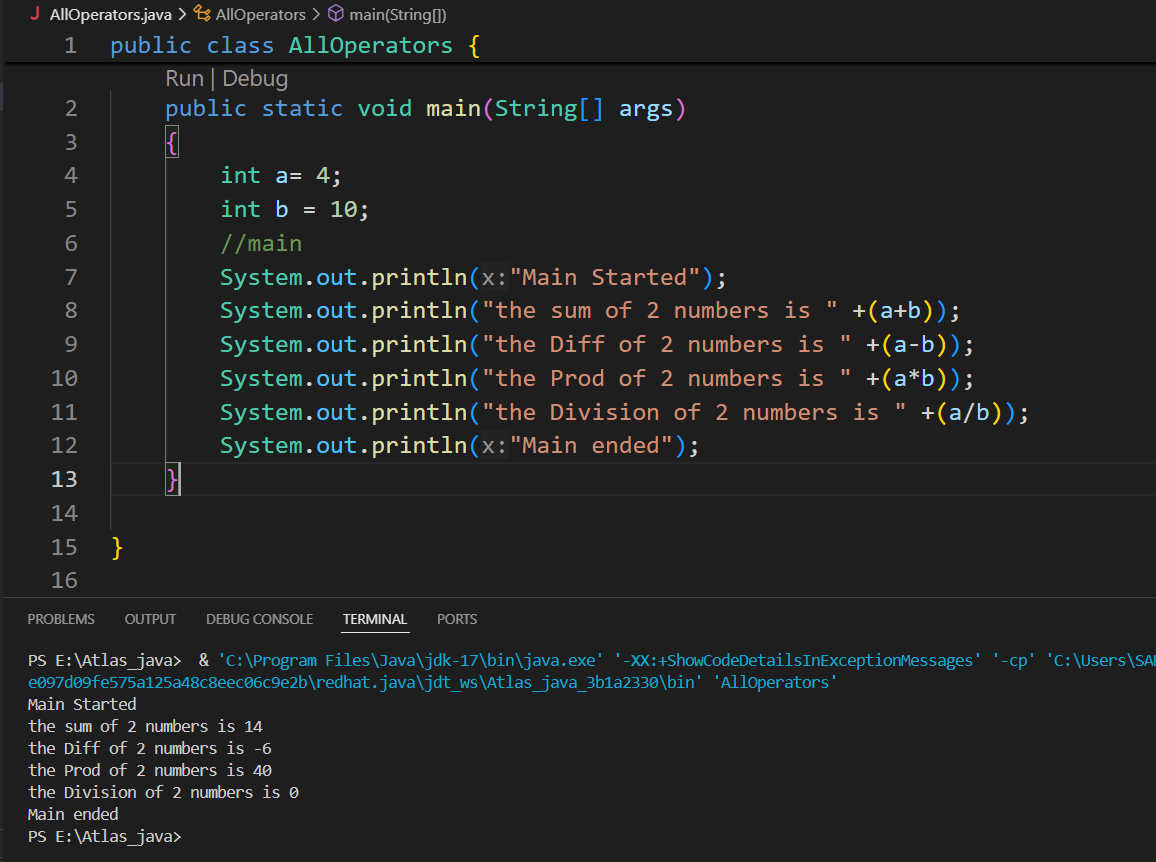
Sum of 2 numbers is …..

Diff of 2 numbers is —-

Product of 2 numbers ….

Division of 2 numbers is ….

Main ended



Task 005:

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

12.14 to 12.18



Task 006:

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

Input:

Id : Prasunamba

Pwd: 123456789

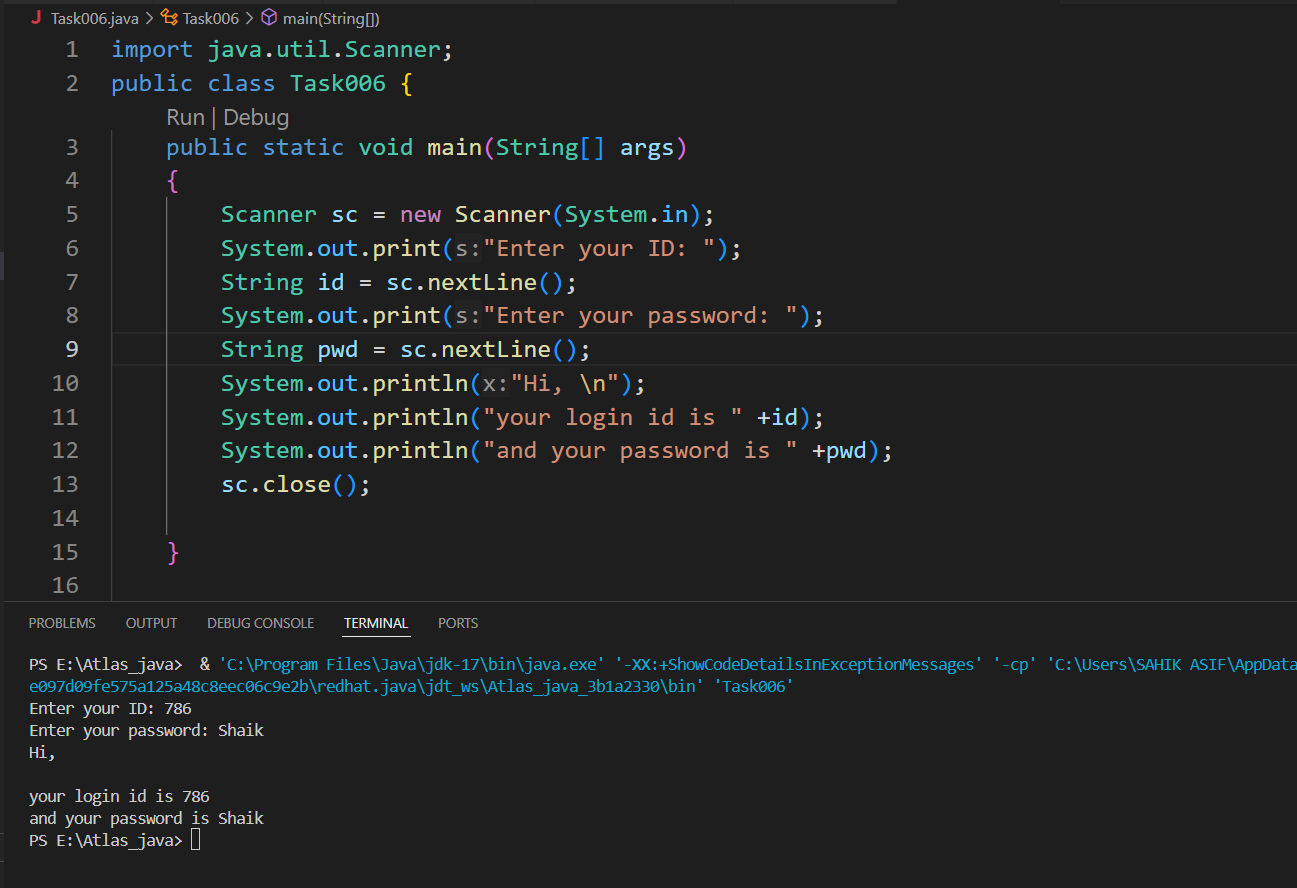
Outpit:

Hi ,

Your login id is Prasuanmba

And your pwd is \*\*\*\*\*\*\*\*\*

12.19 to 12.24



Task 007:

Write a program to create a class named Customer

Call the customer class in Task007 class using an object

12.27 to 12.32

Hint

In the main method

Class Customer{

Void accept(){

sysout(“accept customer called”);

}

Void display(){

sysout(“display customer called”);

}

}

Public class Test007{

psvm(String[] argos){

Customer cobj = new Customer();

cobj.accept();

cobj.display();

}

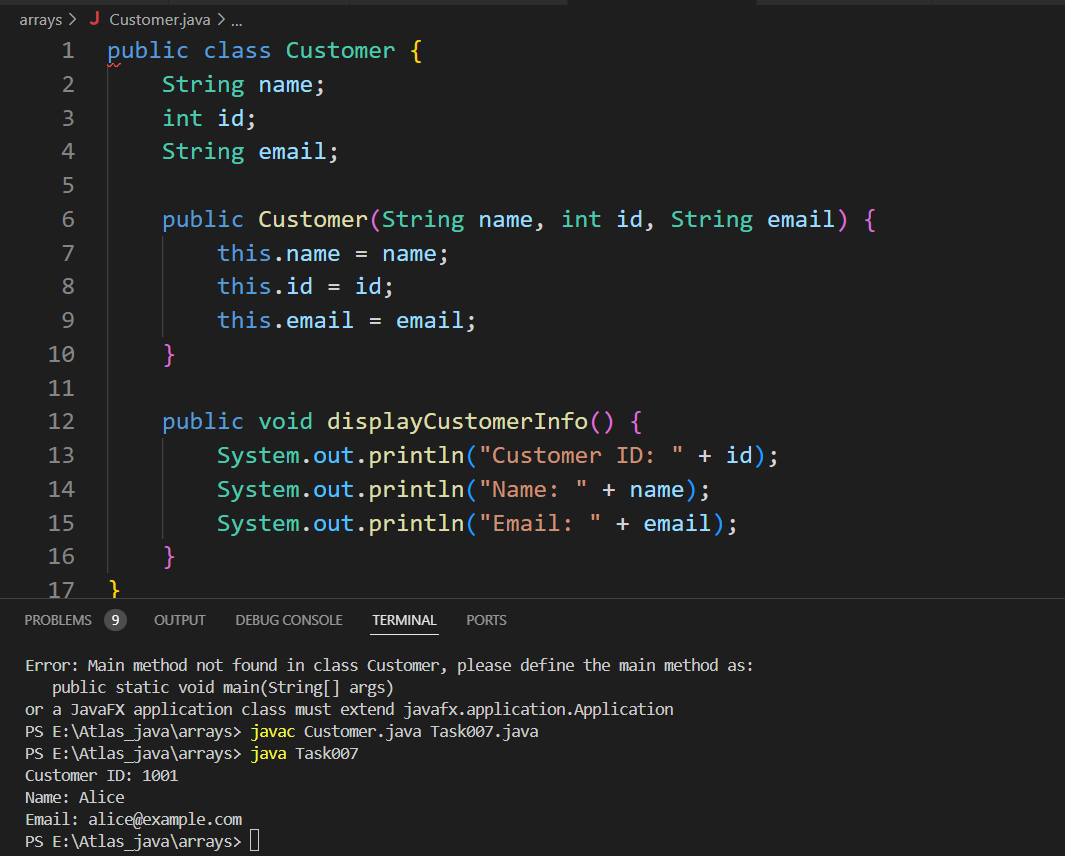
}

12.27 to 12.32

Plz raise ur hands if done till here…@12.35

Done till task 006 – 1 pax

Done till task 007 - 12pax @ 12.57



Task 008:

Wap to check the greater of 2 numbers

Hint:

Use if else

If ( num1 > num2){

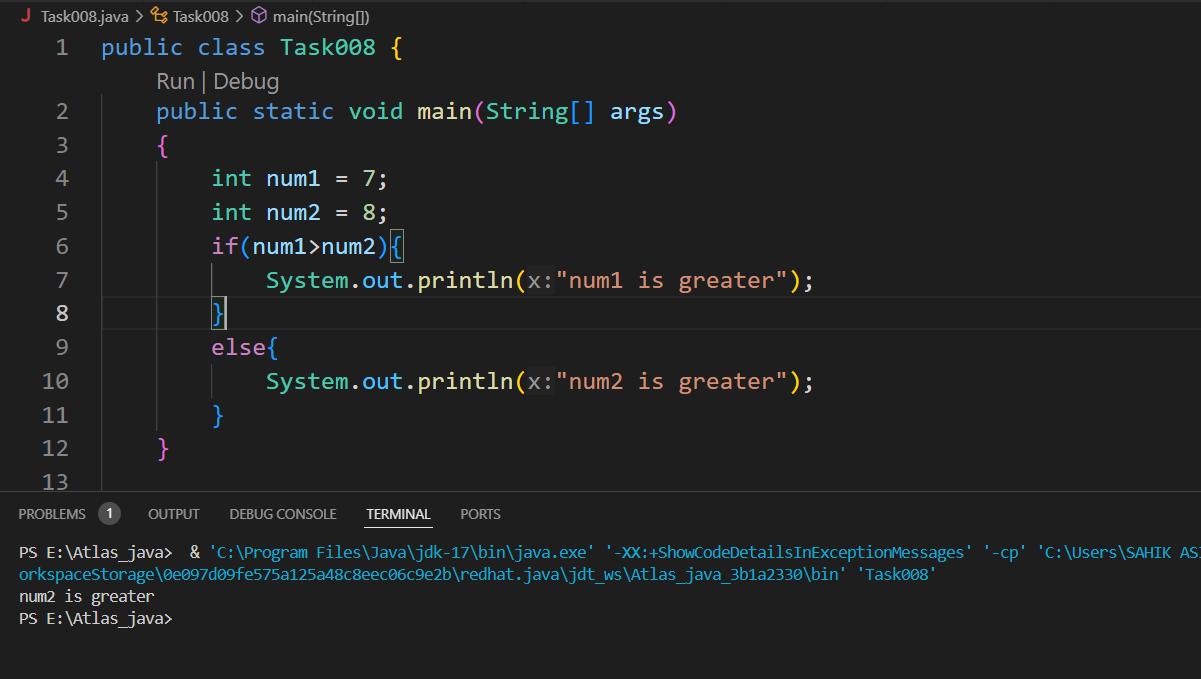
sout(“num1 is greater”);

}

Else {

sout(“num2 is greater”);

}

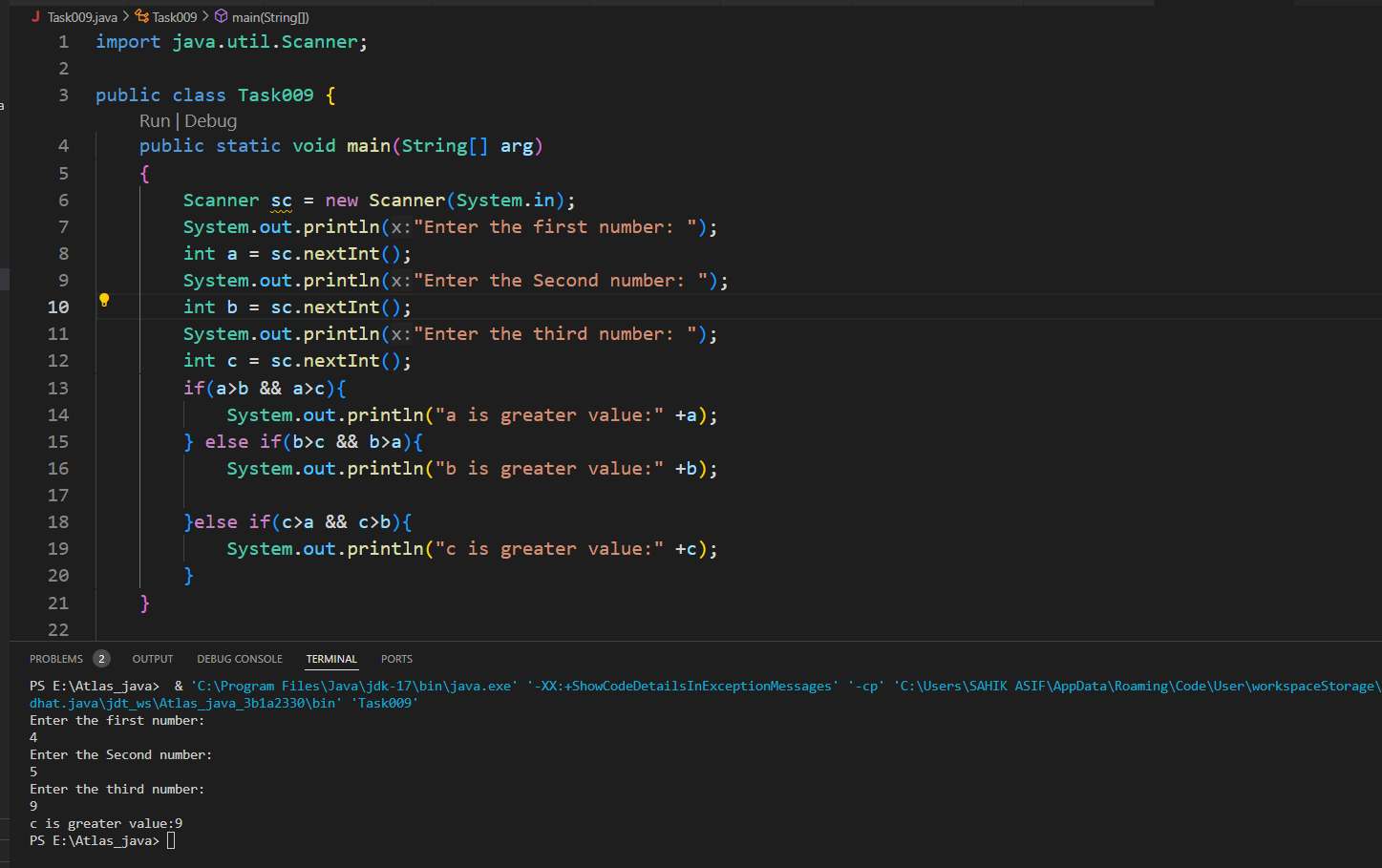


Task 009

Wap to check greater than 3 numbers

Hint 👍

Use elseif



Task 010:

Wap to check if check week days

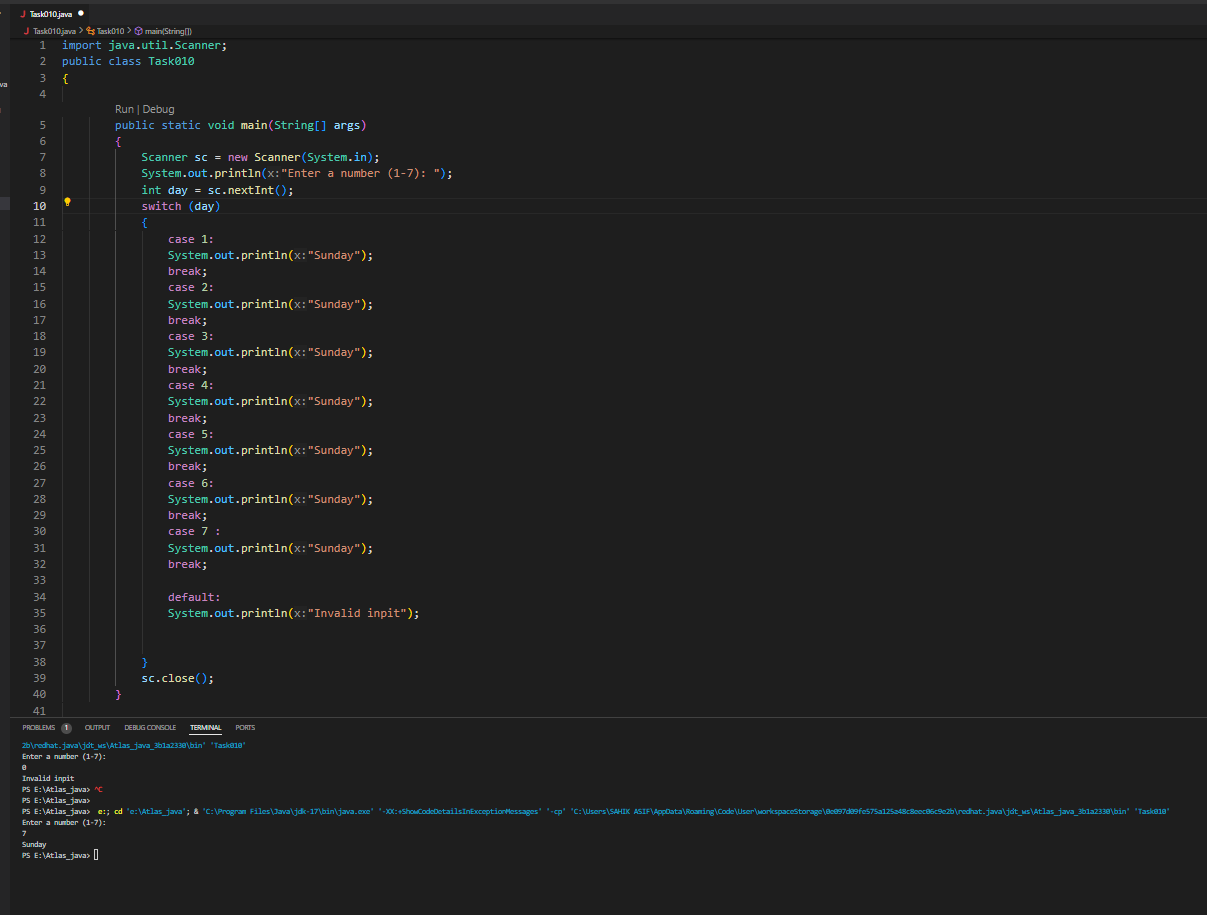
1 ===> sunday

2 ===> monday

So on

8 and above ===> invalid input

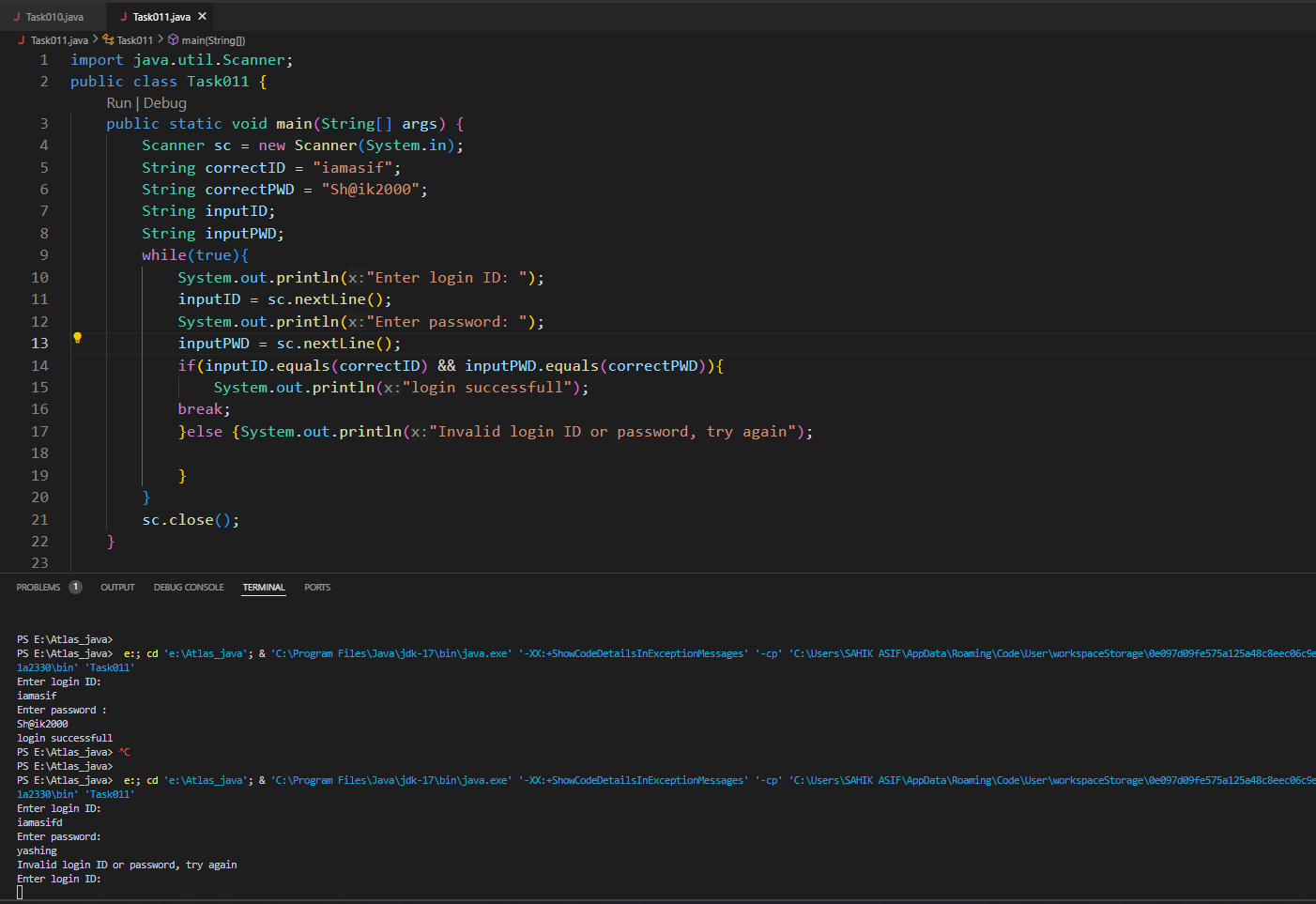
Hint : use Switch case



Task 011:

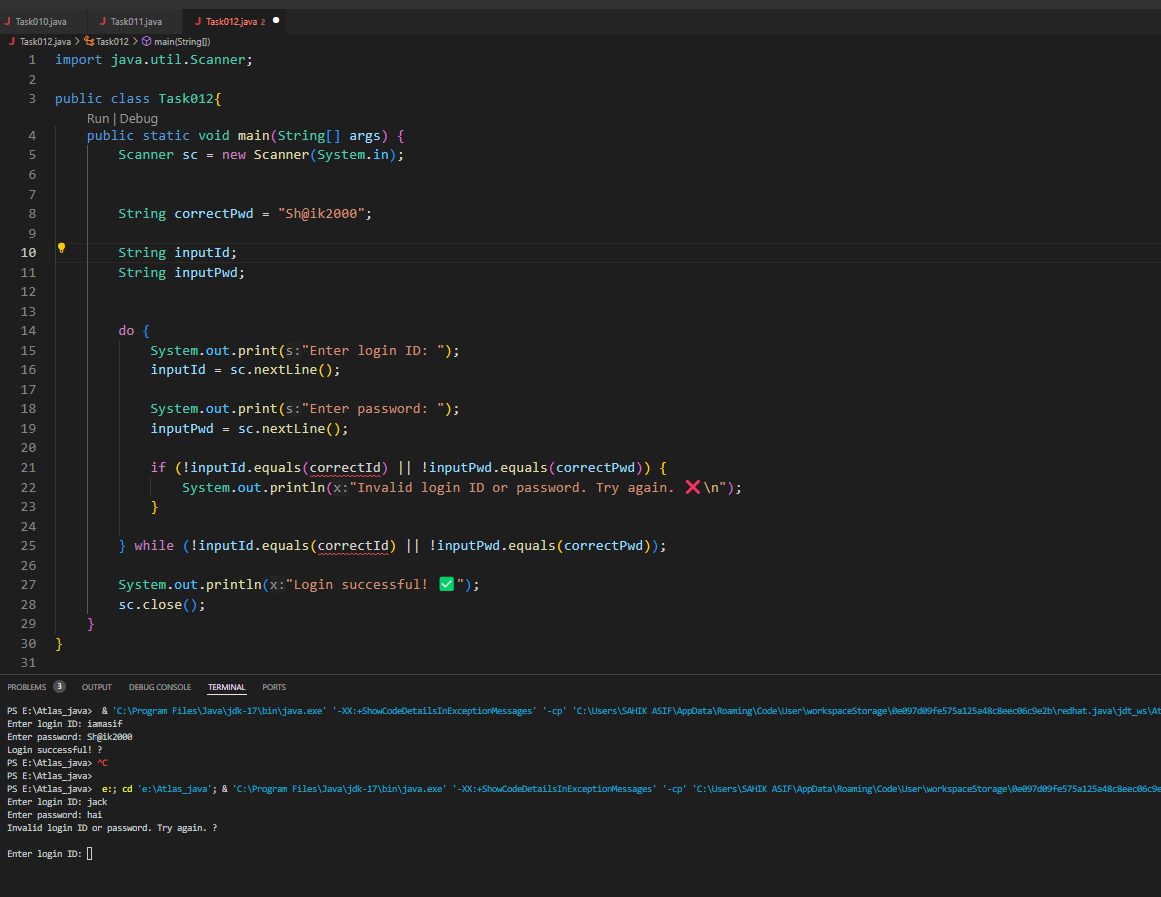
Wap to check loginid and password validation

Hint use while loop



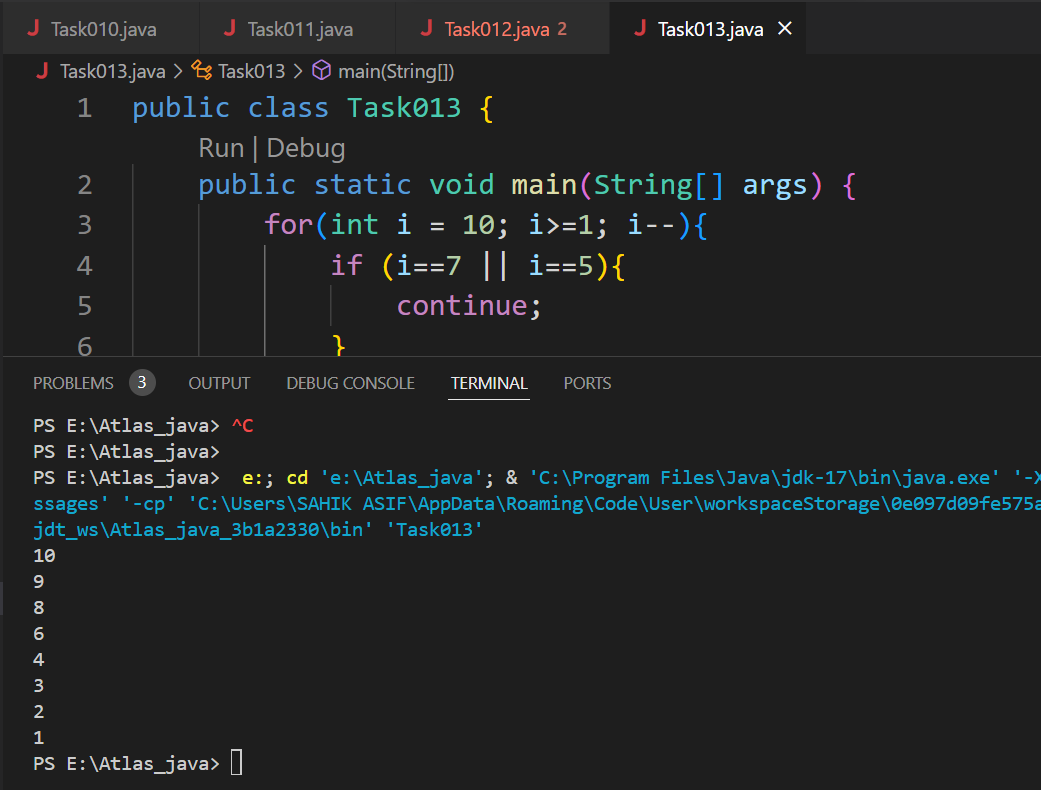
Task 012:

Same as above qn but use do while loop



Task 013:

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



Task 014:

Arrays:

Try the below code and display the output…

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

package Arrays;

public class Demo01 {

public static void main(String[] args) {

// TODO Auto-generated method stub

char[] arr = {'a','e','i','o','u'};

System.out.println(arr);

String[] names = {"Meena", "Tina", "Veena", "heena"};

System.out.println(names[0]);

names[1]= "Reena";

System.out.println(names[1]);

System.out.println(names.length);

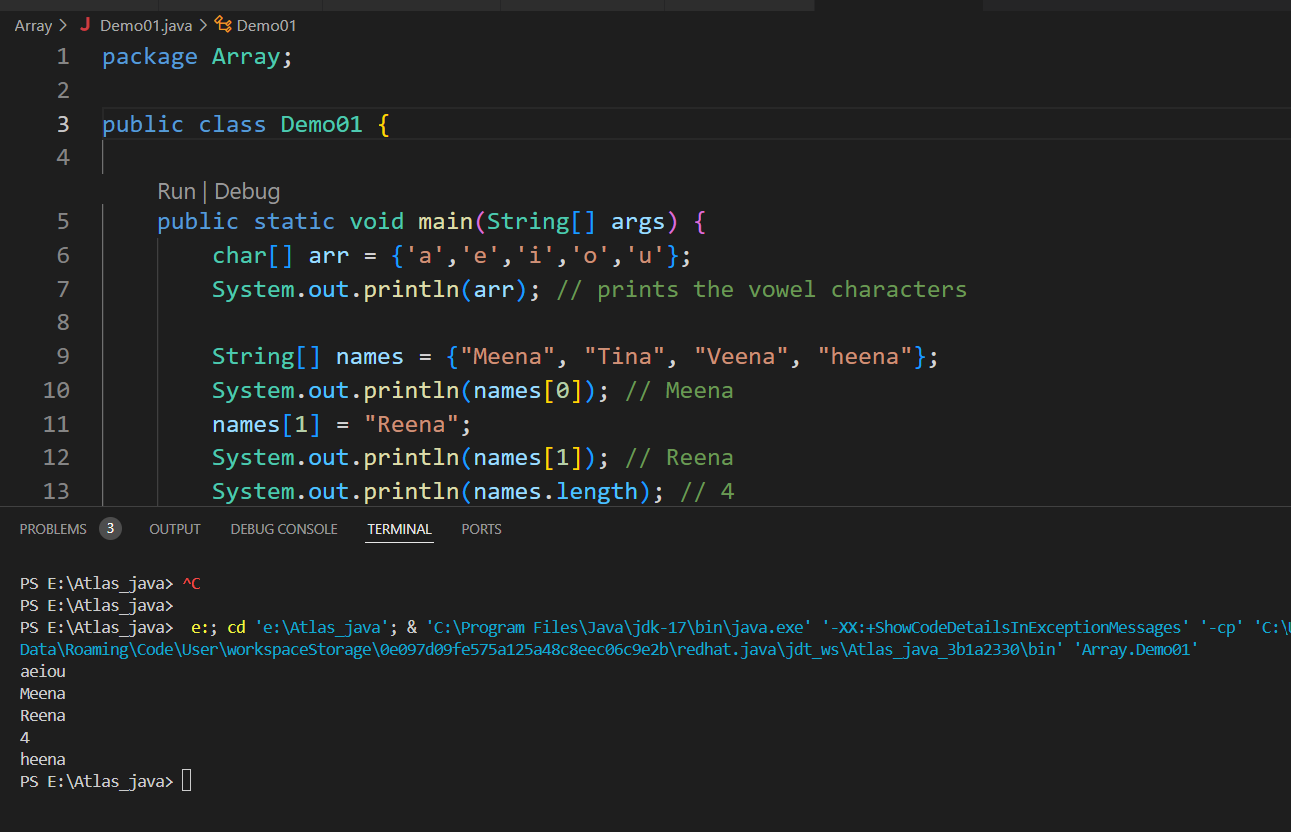
System.out.println(names[4]);

//Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException

}

}

Strings:



Task 015:

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

– immutable ⇒ cannot be changed

String Name = “Prasunamba is a trainer”;

Name = “Hello”;

Variables are mutable ⇒ which can be chaged

package StringHandling;

public class Demo01 {

public static void main(String[] args) {

// TODO Auto-generated method stub

String str1 = "Java Strings "; // string Literal

String str2 = new String(str1); // obj of the string - new keyword

String str3 = new String("are easy to learn ");

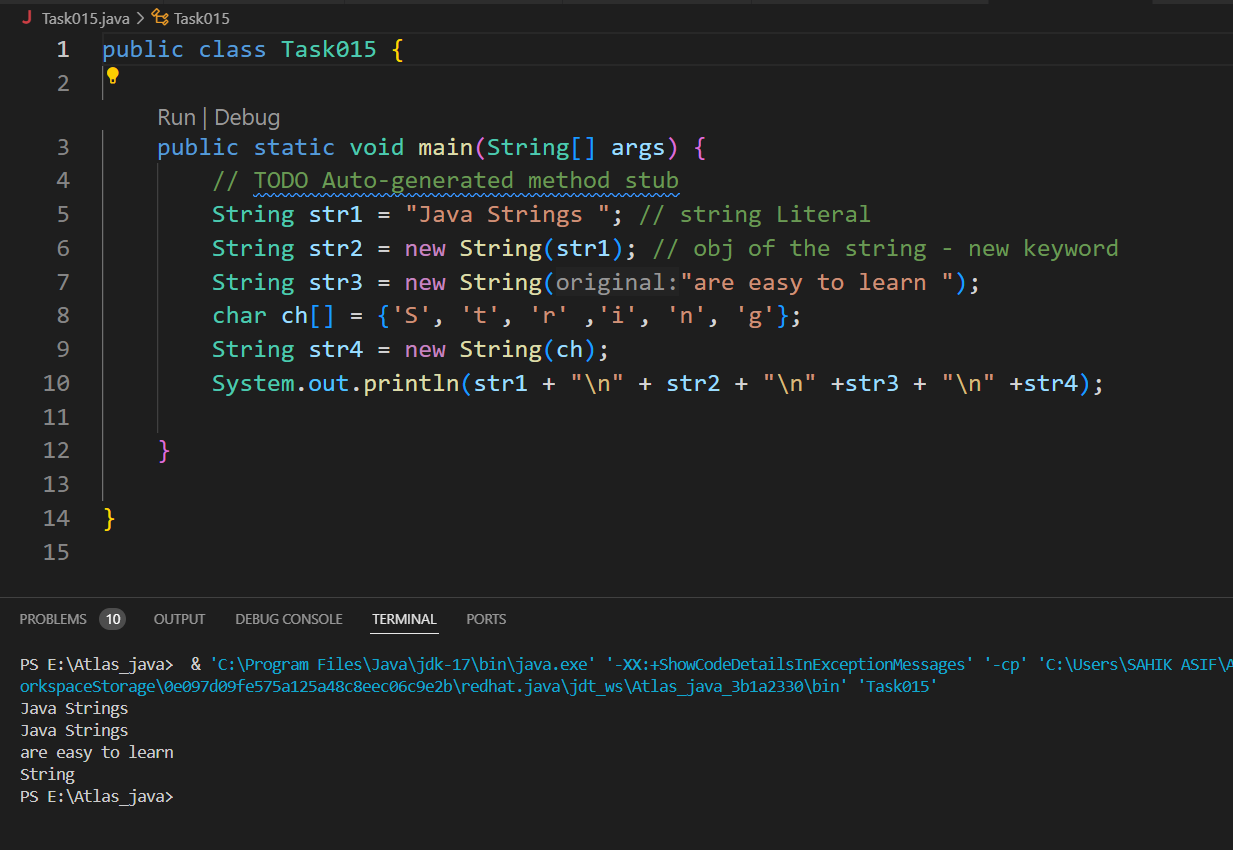
char ch[] = {'S', 't', 'r' ,'i', 'n', 'g'};

String str4 = new String(ch);

System.out.println(str1 + "\n" + str2 + "\n" +str3 + "\n" +str4);

}

}



Task 016

Enums or Enumerations

What is the output of the below code snippet

package Enumerations;

enum color{

red, blue, green, yellow

}

public class Demo01 {

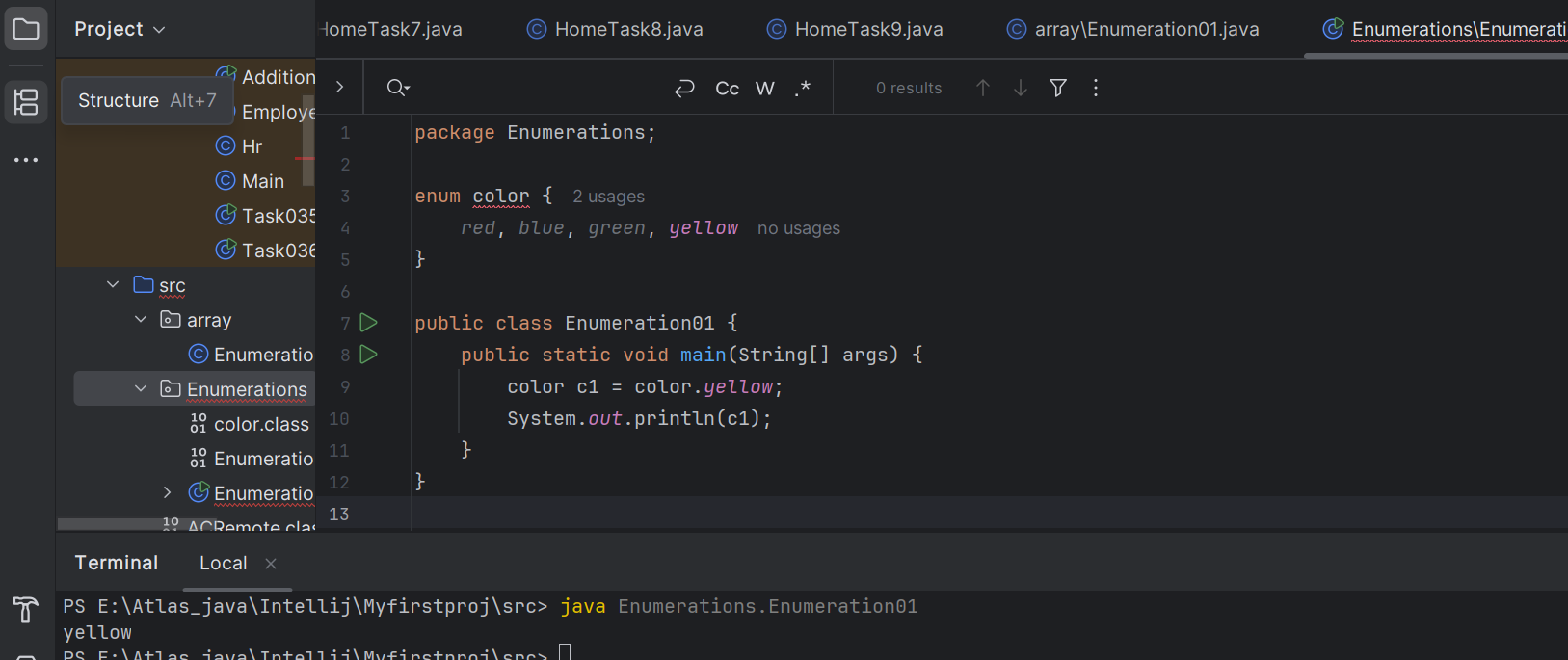
public static void main(String[] args) {

color c1 = color.yellow;

System.out.println(c1);

}

}



package Enumerations;

enum Weekdays{

Sunday , Monday , Tuesday

}

public class Demo01 {

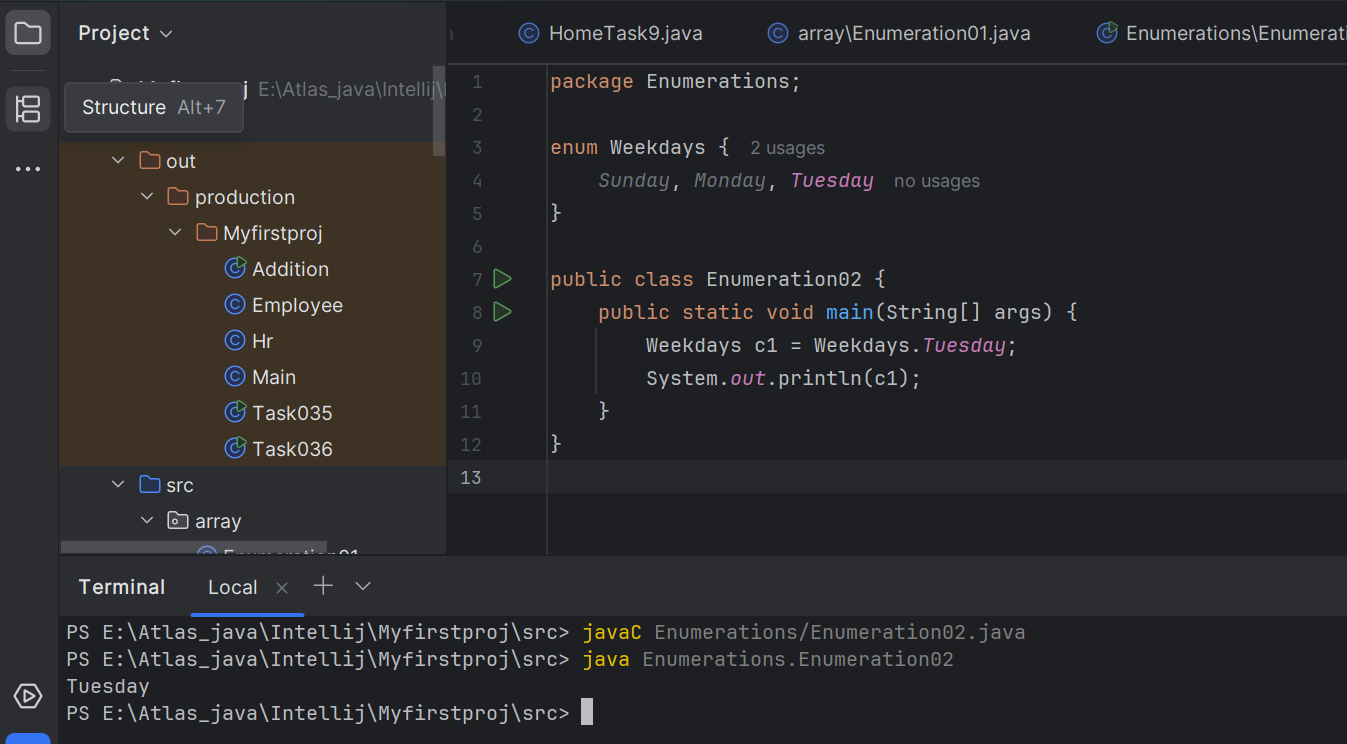
public static void main(String[] args) {

Weekdays c1 = Weekdays.Tuesday;

System.out.println(c1);

}

}



Task 017:

Getter and setter

Create a program name Person.java

public class Person {

private String name;

// Getter

public String getName() {

return name;

}

// Setter

public void setName(String newName) {

this.name = newName;

}

}

Create another program named Task017.java

public class Task017{

public static void main(String[] args) {

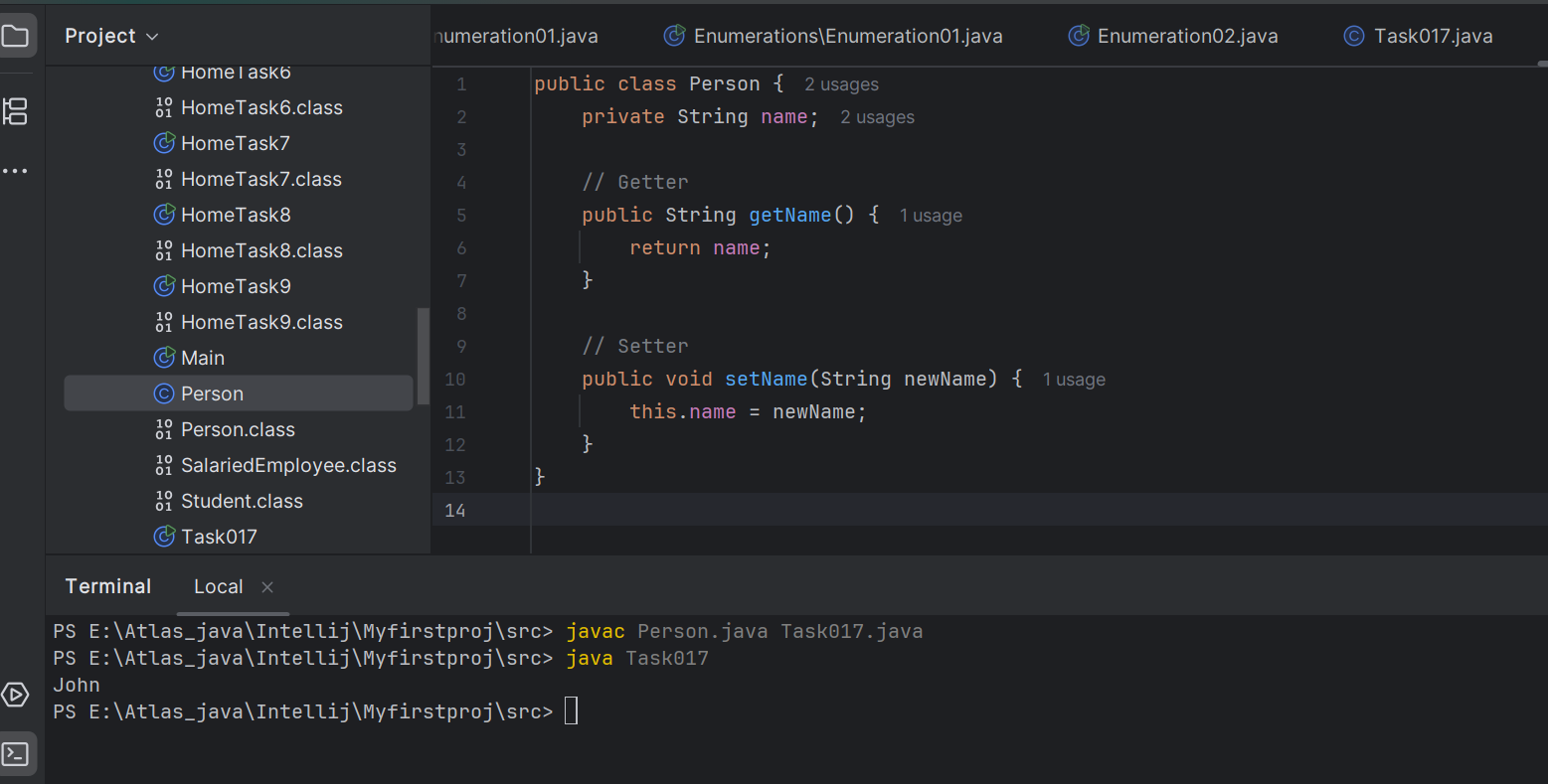
Person myObj = new Person();

myObj.name = "John";

System.out.println(myObj.name);

}

}



—----------------------------------what is the reason for the error —---------------explain

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());

}

}

Now —--------------think what is the output of the above code—--------------

=========================================================================================================================================================================

public class Task015{

public static void main(String[] args){

String Str1 = "Myname ";

Str1 = "java";

String str2 = "hello";

System.out.println(Str1);

System.out.println(str2);

}

}

//string - immutable

//strings with the same content share storage ina single pool to minimize creating the copy of the same value again and again..

//a string is created/ generated , its content cannot be changed

// string are considered as classes -===> we create objects to it..

//cannot be changed one initialized in string

// variable =---> mutable

// age = 10;

// age = 11;

class Customer01{

void accept(){

System.out.println("Accept method");

}

void display(){

System.out.println("display method");

}

}

public class Task111{

void method1(){

System.out.println("method1 called");

}

static void static\_method1(){

// method1(); ==> cannot access as it is a non static member

System.out.println(" static\_method1 called");

}

public static void main(String[] args){

Customer01 cobj = new Customer01();

cobj.accept();

cobj.display();

static\_method1();

Task111 tobj = new Task111();

tobj.method1();

// method1();

}

}