23CSE111

OBJECT ORIENTED PROGRAMMING

LAB MANUAL



Department of Computer and Science Engineering

Amrita School of Engineering

Amrita Vishwa Vidyapeetham, Amaravati Campus

Name:-P LIKHITH SAI

Sec:-CSE-A

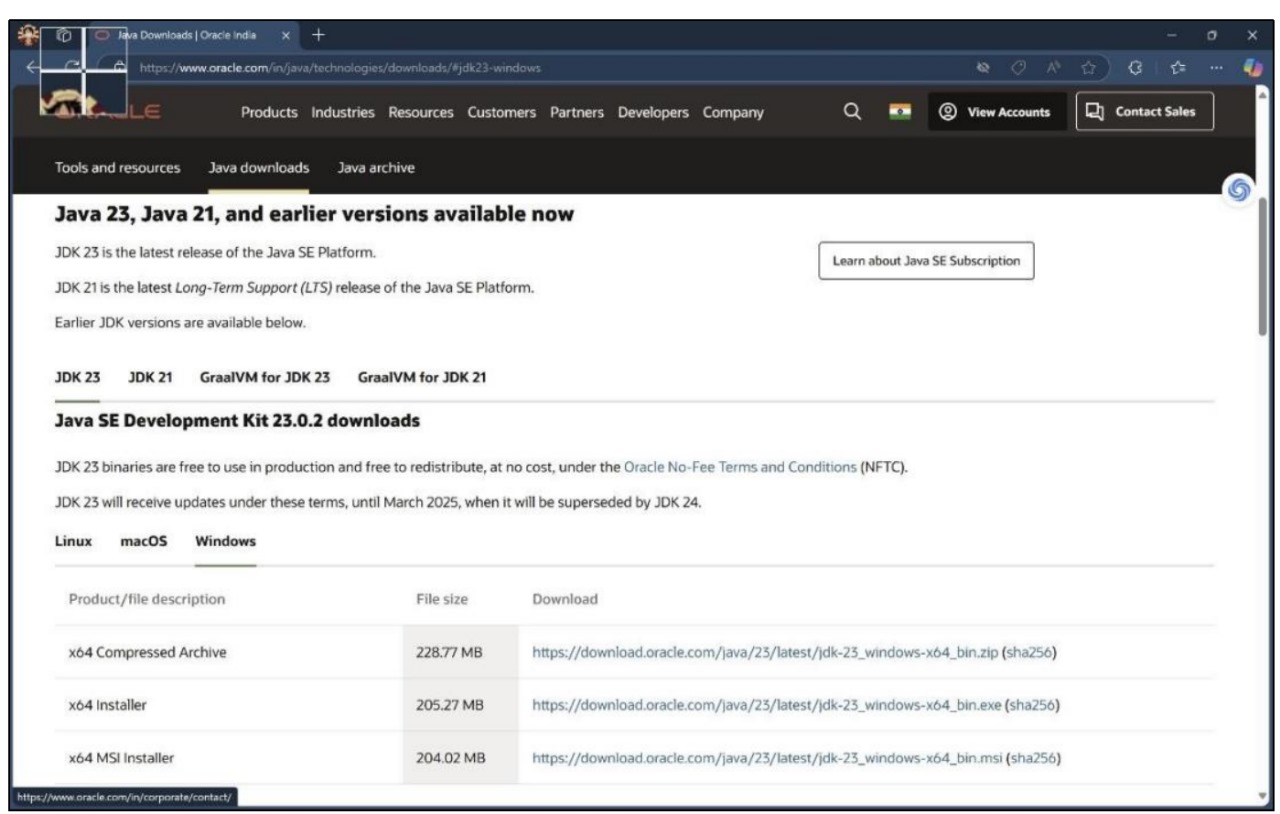
Roll.No:-AV.SC.U4CSE24306

Verified By:

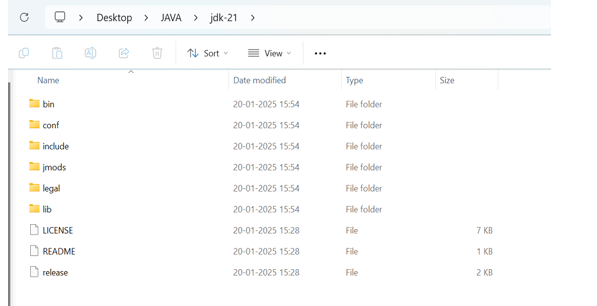
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.NO** | **ACTIVITY** | **PAGE** | **DATE** | **SIGN** |
|  | **WEEK-1** |  |  |  |
| 1 | Download and install JDK |  |  |  |
| 2 | Write a Java program to print the message Welcome to Java programming |  |  |  |
| 3 | Write a Java program to print the name, roll number, and section of the student |  |  |  |
|  | **WEEK-2** |  |  |  |
| 1 | Java program to calculate the area of a rectangle |  |  |  |
| 2 | Java program to convert temperature from Celsius to farenheit |  |  |  |
| 3 | Java program to calculate simple intrest |  |  |  |
| 4 | Java program to find largest of three numbers using ternary operator |  |  |  |
| 5 | Java program to find the factorial of an number |  |  |  |
|  | **WEEK-3** |  |  |  |
| 1 | Write a Java program with the following instructions.   1. Create a class with name car. 2. Create four attributes named car\_colour,car\_brand,fuel\_type,top speed. 3. Create three method named “Start\_Racing”,”End\_Race”.{ } 4. Create three objects named Car1,Car2,Car3. |  |  |  |
| 2 | Write a class by writing a Java program named Bank Account with two methods: “deposits and withdraw”. |  |  |  |
|  | **WEEK-4** |  |  |  |
| 1 | Write a java program with class named “Book”. The class should contain various attributes such as “Title of the book , author , year of publication “. |  |  |  |
| 2 | To create a java program with class named Myclass with a static variable “Count” of “int type”, Initialized to 0 and a constant variable “pi” of type double, initialized to 3.1415 as attributes of that class Now, define a constructor for “Myclass” that increments the “Count” variable each that an object of Myclass is created. Finally, print the final values of the “Count” and “pi” variables |  |  |  |
|  | **WEEK-5** |  |  |  |
| 1 | Create a calculator using the operations including addition using subtraction, multiplication and division using multilateral inheritance and display the desired output. |  |  |  |
| 2 | Create a calculator using the operations including addition using subtraction, multiplication and division using multilateral inheritance and display the desired output. |  |  |  |

***WEEK –1***

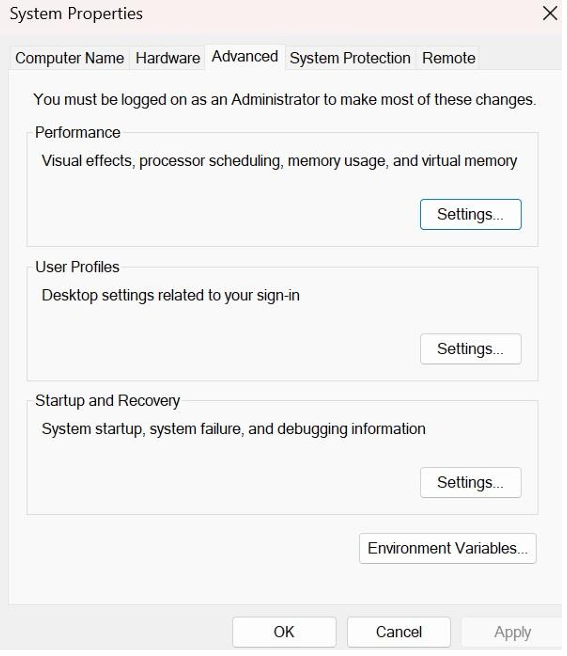
Task:-1) Write the steps to download and install Java. Aim: To download and install java. Procedure: i. Visit oracle.com website to download Java.

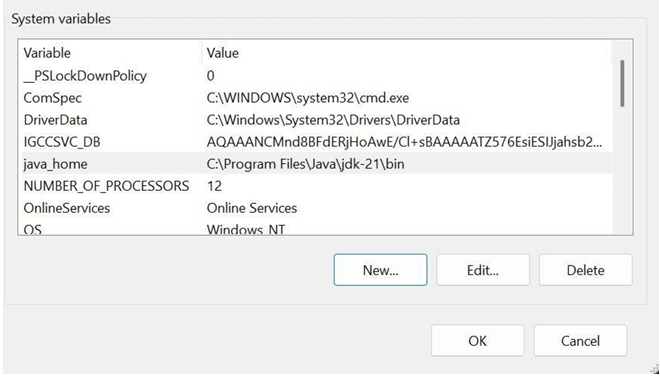


ii. Download the version which supports LTS (JDK 21) x64 installer for windows. iii. Install and copy the path.

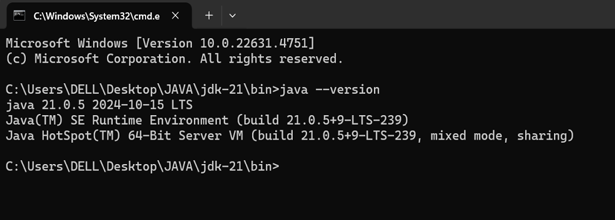


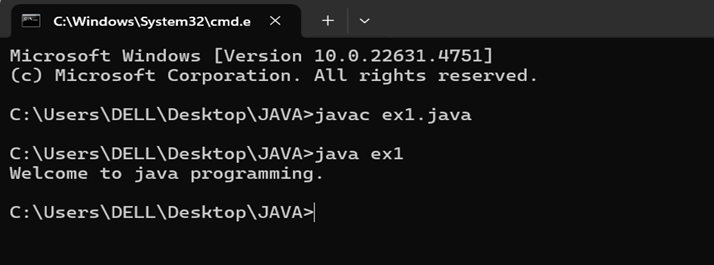
iv. Open environmental variables and add a new file with path.





v. Verify java version in command window.



Task:-2) Write a java program to print the message “Welcome to java programming”. *Code:-* public class Main { public static void main(String[] args){ System.out.println("Welcome to java programming."); } } Output:- 

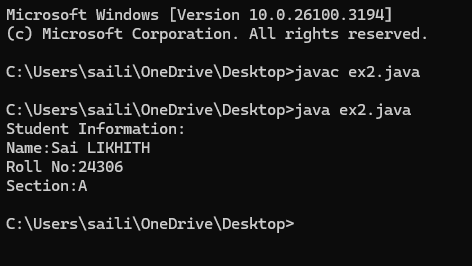
*Error:-*

|  |  |  |
| --- | --- | --- |
| S.No | Expected Error | Reason |
| 1 | ; | ; is expected at end |
| 2 | S | Capital S is expected for String and System. |

Task:-3)Write a java program that prints student details(name, roll number and section of a student).

Code: class ex2{ public static void main(String[] args){ String name = "Sai LIKHITH"; int rollNo = 24306; String section = "A"; System.out.println("Student Information:"); System.out.println("Name:" + name); System.out.println("Roll No:" + rollNo); System.out.println("Section:" + section);

} } **Output:-**



|  |  |  |
| --- | --- | --- |
| S.No | Expected Error | Reason |
| 1 | S | Capital S is expected for String and System. |

**WEEK-2**

**1)JAVA PROGRAM TO CALCULATE THE AREA OF THE RECTANGLE:**

**CODE:**

import java.util.Scanner;

public class area {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter the length of the rectangle: ");

double length = scanner.nextDouble();

System.out.print("Enter the width of the rectangle: ");

double width = scanner.nextDouble();

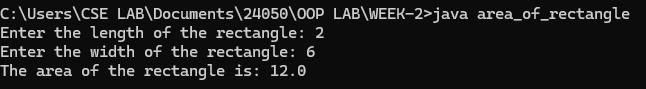
double area = length \* width;

System.out.println("The area of the rectangle is: " + area);

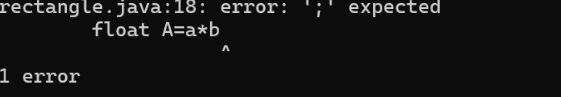
scanner.close();

}

}

**OUTPUT:**

zError:-



|  |  |  |
| --- | --- | --- |
| S.No | Expected Error | Reason |
| 1 | ; | That is  semicolon( ; )  is not present at the  end of the  expression. |

**2)**

**A)JAVA PROGRAM TO CONVERT TEMPERATURE FROM CELCIUS TO FAHRENHEIT:**

**CODE:**

import java.util.Scanner;

public class celcius\_to\_fahrenheit {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter temperature in Celsius: ");

double celsius = scanner.nextDouble();

double fahrenheit = (celsius \* 9/5) + 32;

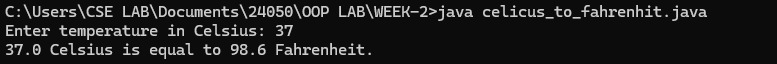
System.out.println(celsius + " Celsius is equal to " + fahrenheit + " Fahrenheit.");

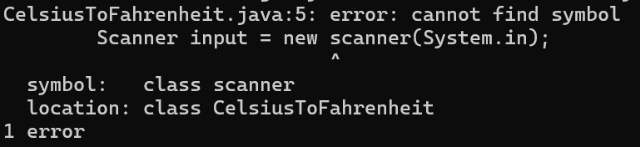
scanner.close();

}

}

**OUTPUT:**



Error:-

|  |  |  |
| --- | --- | --- |
| S.No | Expected Error | Reason |
| 1 | S | Capital S is expected for String and System. |

**B)JAVA PROGRAM TO CONVERT FAHRENHEIT TO CELCIUS :**

**CODE:**

import java.util.Scanner;

public class ftoc {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter temperature in Fahrenheit: ");

double fahrenheit = scanner.nextDouble();

double celsius = (fahrenheit - 32) \* 5 / 9;

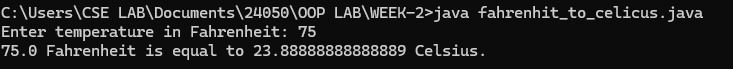
System.out.println(fahrenheit + " Fahrenheit is equal to " + celsius + " Celsius.");

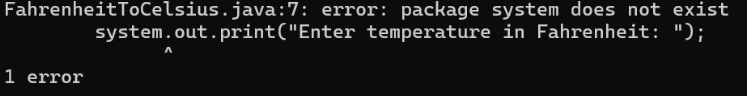
scanner.close();

}

}

**OUTPUT:**



Error:-

|  |  |  |
| --- | --- | --- |
| S.No | Expected Error | Reason |
| 1 | S | Capital S is expected for String and System. |

**3)JAVA PROGRAM TO CALCULATE THE SIMPLE INTEREST:**

**CODE:**

import java.util.Scanner;

public class SimpleInterest {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter the principal amount: ");

double principal = scanner.nextDouble();

System.out.print("Enter the rate of interest (in %): ");

double rate = scanner.nextDouble();

System.out.print("Enter the time period (in years): ");

double time = scanner.nextDouble();

double simpleInterest = (principal \* rate \* time) / 100;

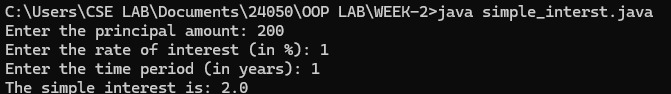
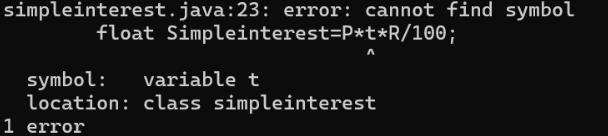
System.out.println("The simple interest is: " + simpleInterest);

scanner.close();

}

}

**OUTPUT:**

Error:-

|  |  |  |
| --- | --- | --- |
| S.No | Expected Error | Reason |
| 1 | T | Capital T is expected for String and System. |

**4)JAVA PROGRAM TO FIND THE LARGEST AMONG THREE NUMBER USING TERNARY OPERATOR:**

**CODE:**

import java.util.Scanner;

public class largest3 {

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();

System.out.print("Enter the third number: ");

int num3 = scanner.nextInt();

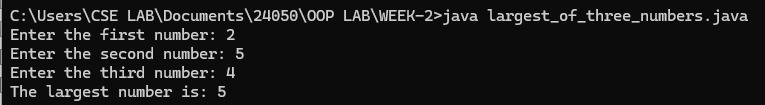
int largest = (num1 >= num2 && num1 >= num3) ? num1 : (num2 >= num1 && num2 >= num3) ? num2 : num3;

System.out.println("The largest number is: " + largest);

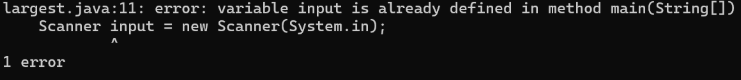
scanner.close();

}

}

**OUTPUT:**

**Error:-**



|  |  |  |
| --- | --- | --- |
| S.No | Expected Error | Reason |
| 1 | main(String []). | Variable input is already defines in method main(String []). |

**5)JAVA PROGRAM TO FIND FACTORIAL OF A NUMBER:**

**CODE:**

import java.util.Scanner;

public class factorial {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter a number to find 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);

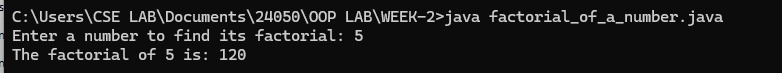
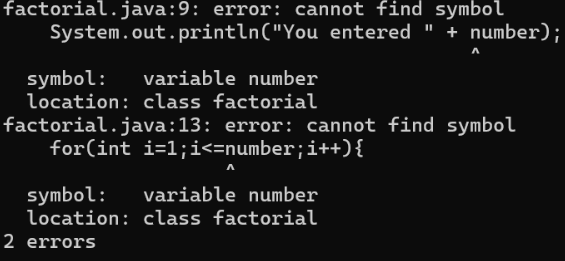
}

scanner.close();

}

}

**OUTPUT:**

Error:- 

WEEK – 3

1) Create the java program with the following instructions i) Create a class with name Car ii) Create 4 attributes named Car\_Color , Car\_brand, fuel\_type, mileage iii) Create 3 method named Start( ) , Stop( ), Service( ) iv) Create 3 objects Car1 , Car2 , Car3 v) Create a constructor which should print “Welcome to Car Garage” public class Car{

    public String carColor;

    private String carBrand;

    private String fuelType;

    public int mileage;

    Car(String carColor , String carBrand , String fuelType , int mileage){

    this.carColor =  carColor;

    this.carBrand = carBrand;

    this.fuelType = fuelType;

    this.mileage = mileage;

    System.out.println(carColor + " " + carBrand + " " + fuelType + " " +

mileage);

    }

    public void Start(){

    System.out.println("The car has just started");

    }

    public void Stop(){

    System.out.println("The car has just stopped");

    }

    public void Service(){

    System.out.println("The car is in good condition");

     }

    public static void main(String[] args){

   Car Car1 = new Car("Black","BMW","Petrol",20);

   Car Car2 = new Car("White","AUDI","Diesel",17);

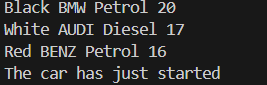
   Car Car3 = new Car("Red","BENZ","Petrol",16);

   Car1.Start();

   }

   }

OUTPUT:-



*Error:-*

|  |  |  |
| --- | --- | --- |
| S.No | Expected Error | Reason |
| 1 | } | } is expected at end of the class |
| 2 | Setting the parameters inside the constructor | We cannot pass the values inside constructor without setting them first |

Class Diagram:

|  |
| --- |
| Car |
| + carColor : String - carBrand : String - fuelType : String + mileage : int |
| + Car() : void + Start( ) : void + Stop( ) : void + Service( ) : void |

2)Write a java program to create a class BackAccount with two methods deposit( ) and withdraw( ) i) In deposit( ) whenever an amount is deposited it has to be updated with current amount ii) In withdraw( ) whenever an amount is withdrawn it has to be less than current amount else print “Insufficient funds”. Code: - public class BankAccount {

    private String Name;

    private int AccNo, CurrBal;

    public BankAccount(String Name, int AccNo, int CurrBal) {

        this.Name = Name;

        this.AccNo = AccNo;

        this.CurrBal = CurrBal;

        System.out.println("The customer is: " + this.Name);

    }

    public int deposit(int dAmt) {

        CurrBal += dAmt;

        return CurrBal;

    }

    public void withdraw(int wAmount) {

        if (wAmount <= CurrBal) {  // Allowing withdrawal if balance is equal

            CurrBal -= wAmount;

            System.out.println("Remaining Balance: " + CurrBal);

        } else {

            System.out.println("Insufficient funds");

        }

    }

    public static void main(String[] args) {

        BankAccount SaiKrishna = new BankAccount("SaiKrishna", 2000, 15000);

        SaiKrishna.withdraw(13000);  // Should print "Insufficient funds"

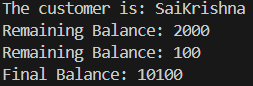
        SaiKrishna.withdraw(1900);   // Should print remaining balance

        int FinalAmount = SaiKrishna.deposit(10000);

        System.out.println("Final Balance: " + FinalAmount);

    }

}

OUTPUT:-

*Error:-*

|  |  |  |
| --- | --- | --- |
| S.No | Expected Error | Reason |
| 1 | } | } is expected at end of the class |
| 2 | Setting the parameters inside the constructor | We cannot pass the values inside constructor without setting them first |

Class Diagram:-

|  |
| --- |
| BankAccount |
| - Name : String - AccNo : String - CurrBal: String |
| + BankAccount( ) : void + deposit( ) : int + withdraw( ) : void |

WEEK4

**1)Aim:** Write a java program with class named “Book”. The class should contain various attributes such as “Title of the book , author , year of publication “.

**CODE:**

public class Book{

// atributes

private String title;

private String author;

private int yearofpublication;

// constructor

public Book( String title, String author, int yearofpublication ){

this.title= title;

this.author= author;

this. yearofpublication= yearofpublication;

}

// method to display the details of the book

public void displaydetails(){

System.out.println("the title of the book is:"+ title);

System.out.println("the autor of book is:"+ author);

System.out.println("year of publication:"+ yearofpublication );

}

// Main method to create and display details of two books

public static void main(String[] args){

// Creating two Book objects

Book book1= new Book("Solo levelling", "p.likhith", 2025);

Book book2= new Book("attack on titan", "p.likhith", 2026);

System.out.println("book 1 Details:");

book1.displaydetails();

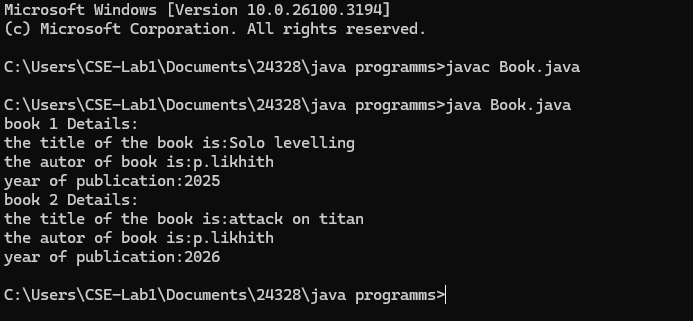
System.out.println("book 2 Details:");

book2.displaydetails();

}

}

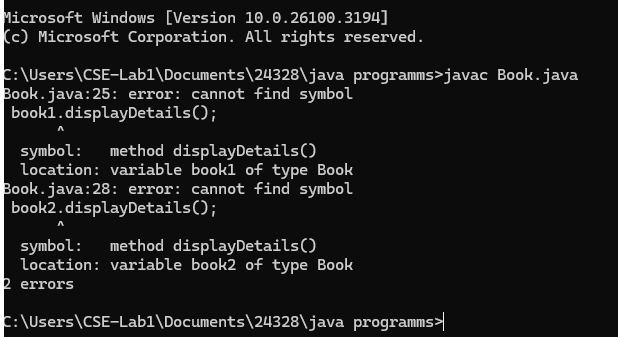
OUTPUT:



ERRORS:

**CLASS DIAGRAM**:

|  |
| --- |
| **Book** |
| - title: String  - author: String  - yearOfPublication: int |
| + Book(title: String, author: String,  yearOfPublication: int)  + displayDetails(): void |

2) **AIM:** To create a java program with class named Myclass with a static variable “Count” of “int type”, Initialized to 0 and a constant variable “pi” of type double, initialized to 3.1415 as attributes of that class Now, define a constructor for “Myclass” that increments the “Count” variable each that an object of Myclass is created. Finally, print the final values of the “Count” and “pi” variables.

**CODE:**

public class MyClass {

// Static variable count of type int, initialized to zero

static int count = 0;

// Constant variable pi of type double, initialized to 314.15

final double pi = 314.15;

// Constructor for MyClass

public MyClass() {

// Increment the count variable each time an object is created

count++;

}

public static void main(String[] args) {

// Creating objects of MyClass

MyClass obj1 = new MyClass();

MyClass obj2 = new MyClass();

MyClass obj3 = new MyClass();

// Print the final value of count and pi

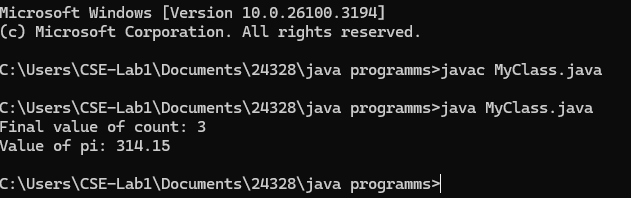
System.out.println("Final value of count: " + count);

System.out.println("Value of pi: " + obj1.pi); // Pi is the same for all objects

}

}

OUTPUT:



**CLASS DIAGRAM:**

|  |
| --- |
| **MyClass** |
| - Count: int  + pi: double |
| + MyClass()  + getCount(): int |

**WEEK 5**

**1)Aim:** Create a calculator using the operations including addition using subtraction, multiplication and division using multilateral inheritance and display the desired output.

**CODE:**

class addition{

int x,y;

int sum;

public void sum(){

sum = x + y;

System.out.println("The sum of the numbers is: "+sum);

}}

class subtraction extends addition{

int diff;

public void difference(){

diff = x - y;

System.out.println("The sum of the numbers is: "+diff);

}}

class multiplication extends subtraction{

int multi;

public void multi(){

multi = x\*y;

System.out.println("The product of the numbers is: "+multi);

}}

class division extends multiplication{

double div;

public void div(){

div = x/y;

System.out.println("The division of the numbers is: "+div);

}}

class calculator extends division{

public calculator(int x,int y){

this.x=x;this.y=y;

}}

class Main{

public static void main(String args[]){

calculator c = new calculator(10,5);

c.sum();

c.difference();

c.multi();

c.div();

}}

**OUTPUT:**

**2)AIM:** Create a calculator using the operations including addition using subtraction, multiplication and division using multilateral inheritance and display the desired output.

**CODE:** class vehicle{

String brand;

int speed;

public vehicle (String brand,int speed){

this.brand=brand;

this.speed=speed; }

public void display(){

System.out.println("The brand of the vehicle is: "+brand+" "+"The speed of the vehicle is: "+speed+" "); }

public void start(){

System.out.println("The vehicle is starting");

}}

class car extends vehicle{

int doors;

public car(String brand,int speed,int doors){

super(brand, speed);

this.doors=doors;

}

public void details(){

super.display();

System.out.println("The number of dooes are: "+doors);

}}

class bike extends vehicle{

boolean gears;

public bike(String brand,int speed,boolean gears){

super(brand, speed);

this.gears=gears; }

public void details(){

super.display();

System.out.println("Does the bike have gears: "+gears);

}}

class truck extends vehicle{

double capacity;

public truck(String brand,int speed,double capacity){

super(brand, speed);

this.capacity=capacity; }

public void details(){

super.display();

System.out.println("The capacity of the truck is: "+capacity);}}

class system{

public static void main(String[] args){

car c = new car("benz",100,4);

bike b = new bike("T.V.S",80,true);

truck t = new truck("Mahindra",120,60);

c.start();

c.display();

b.start();

b.display();

t.start();

t.display();

}}

**OUTPUT:**