

23CSE111
OBJECT-ORIENTED
PROGRAMMING
LAB MANUAL

Department of computer science and
engineering
Amrita School of Engineering
Amrita Vishwa Vidyapeetham, Amaravati
Campus

Name: Jahnavi

Roll No: 24035

Class: CSE-A

Semester: 2

Verified By:

Date of Submission:

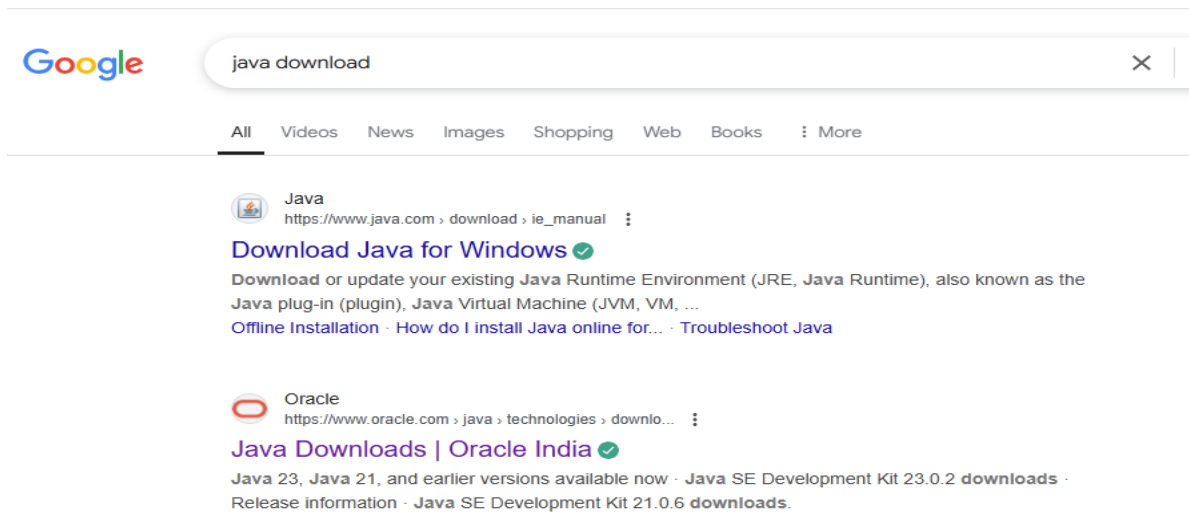
WEEK 1

Program 1

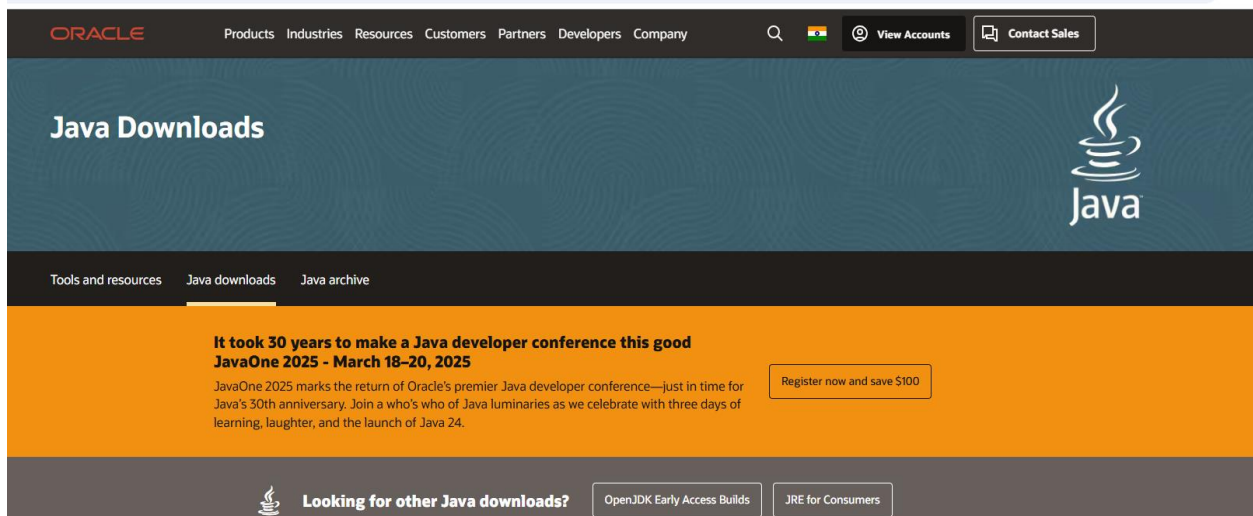
AIM: Downloading and Installing JAVA software

Procedure:

Step 1: Open Google and type java download



Step 2: open website linked with oracle



Step 3: click on JDK 21 and click on windows

Java 23, Java 21, and earlier versions available now

JDK 23 is the latest release of the Java SE Platform.

[Learn about Java SE Subscription](#)

JDK 21 is the latest *Long-Term Support (LTS)* release of the Java SE Platform.

Earlier JDK versions are available below.

JDK 23 **JDK 21** **GraalVM for JDK 23** **GraalVM for JDK 21**

JDK Development Kit 21.0.6 downloads

JDK 21 binaries are free to use in production and free to redistribute, at no cost, under the [Oracle No-Fee Terms and Conditions \(NFTC\)](#).

JDK 21 will receive updates under the NFTC, until September 2026, a year after the release of the next LTS. Subsequent JDK 21 updates will be licensed under the [Java](#) (OTN) and production use beyond the [limited free grants](#) of the OTN license will [require a fee](#).

Linux **macOS** **Windows**

Product/file description	File size	Download
x64 Compressed Archive	185.92 MB	https://download.oracle.com/java/21/latest/jdk-21_windows-x64_bin.zip (sha256)
x64 Installer	164.31 MB	https://download.oracle.com/java/21/latest/jdk-21_windows-x64_bin.exe (sha256)
x64 MSI Installer	163.06 MB	https://download.oracle.com/java/21/latest/jdk-21_windows-x64_bin.msi (sha256)

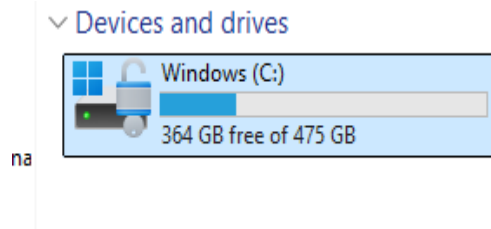
Step 4: After that click on x64 installer

Step 5: the downloading starts

Step 6: After downloading the next step is Installation starts

Step 7: After installation we need to set a path

Step 8: Go to “C” drive in C drive we must select the program files



Name	Date modified	Type	Size
PerfLogs	5/7/2022 10:54 AM	File folder	
Program Files	1/28/2025 6:12 PM	File folder	
Program Files (x86)	9/26/2024 1:54 PM	File folder	
SWSetup	8/31/2024 10:54 AM	File folder	
TC	7/21/2024 8:32 AM	File folder	
Temp	8/2/2024 2:50 PM	File folder	
Users	7/5/2024 7:20 PM	File folder	
Windows	1/27/2025 8:20 PM	File folder	
YOGISOFT	7/21/2024 8:34 AM	File folder	
dosbox-0.74-3-installer_AZR-1K1	7/22/2024 2:28 PM	Application	1,731 KB
tc	7/22/2024 1:46 PM	Application	1,730 KB
turboc	7/22/2024 1:47 PM	Application	1,459 KB

Step 9: Double click on the folder and double click on the java

Avast Software	1/24/2025 4:13 PM	File folder
Common Files	1/22/2025 8:07 PM	File folder
Google	7/5/2024 7:51 PM	File folder
HP	10/4/2024 2:08 PM	File folder
HPCommRecovery	12/15/2023 7:09 PM	File folder
Internet Explorer	10/27/2024 4:58 PM	File folder
Java	1/28/2025 6:31 PM	File folder
MATLAB	10/4/2024 11:04 AM	File folder
McAfee	7/5/2024 7:31 PM	File folder

Name	Date modified	Type	Size
jdk-21	1/28/2025 6:31 PM	File folder	
jdk-23	1/21/2025 9:37 AM	File folder	

Step 10: click on jdk-21 and select bin

bin	1/28/2025 6:31 PM	File folder
conf	1/28/2025 6:31 PM	File folder
include	1/28/2025 6:31 PM	File folder

Step 11: copy the link and paste it in environment

Step 12: select the system variables and select the path

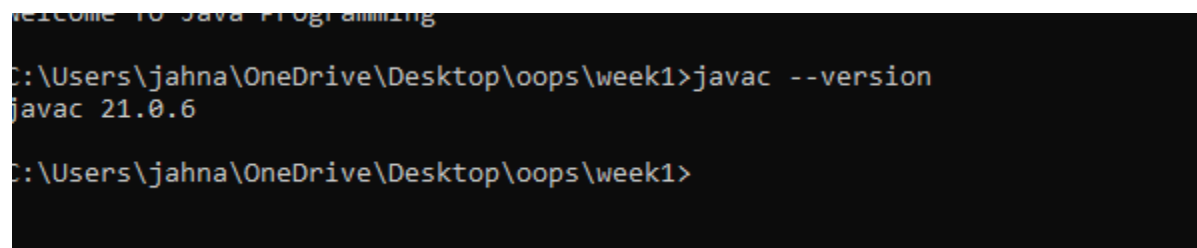
Step 13: after selecting the path we need select new one and need to paste the link

Step 14: after pasting the link we need to click the ok and the apply

Step 15: in this way the downloading and installation of java takes place

Step 16: After installation go to command prompt and type javac – version

Step 17: after we will get the version of the java

A screenshot of a Windows command prompt window. The title bar at the top reads "Welcome To Java Programming". The command prompt shows the directory path "C:\Users\jahna\OneDrive\Desktop\oops\week1" followed by the command "javac --version". The output of the command is "javac 21.0.6". The prompt then returns to "C:\Users\jahna\OneDrive\Desktop\oops\week1>".

```
Welcome To Java Programming
C:\Users\jahna\OneDrive\Desktop\oops\week1>javac --version
javac 21.0.6
C:\Users\jahna\OneDrive\Desktop\oops\week1>
```

Program 2:

AIM: Write a java program to print the message “Welcome To Java Programming”

CODE:

```
class Example1{
    Public static void main(String[] args){
        System.out.println(“Welcome To Java Programming”);
    }
}
```

Output:

```
Microsoft Windows [Version 10.0.22631.4751]
(c) Microsoft Corporation. All rights reserved.

C:\Users\jahna\OneDrive\Desktop\oops\week1>javac Example1.java

C:\Users\jahna\OneDrive\Desktop\oops\week1>java Example1
Welcome To Java Programming

C:\Users\jahna\OneDrive\Desktop\oops\week1>_
```

Program 3:

AIM: write a java code for printing student name, roll no, class, section

Code:

```
class Example1 {
    public static void main(String[] args) {
        System.out.println("name: Jahnavi");
        System.out.println("roll no: 24035");
        System.out.println("class: CSE");
        System.out.println("Section: CSEA");
    }
}
```

Output:

```
C:\Users\jahna\OneDrive\Desktop\oops\week1>javac Example1.java

C:\Users\jahna\OneDrive\Desktop\oops\week1>java Example1
name: Jahnavi
roll no: 24035
class: CSE
Section: CSEA

C:\Users\jahna\OneDrive\Desktop\oops\week1>_
```

WEEK-2

Program-2

Aim: Write a java program to calculate the area of rectangle

Input:

```
class Areaofrectangle{
    public static void main(String[] args){
        int l=5;
        float b=(float)6.5;
        System.out.println("Area of rectangle="+l*b);
    }
}
```

Output:

```
C:\Users\jahna\OneDrive\Desktop\oops\week1>javac Areaofrectangle.java
C:\Users\jahna\OneDrive\Desktop\oops\week1>java Areaofrectangle
Area of rectangle=32.5
```

Errors:

s.no	Error	Correction
1)	Missing semicolon	Keeping semicolon
2)	End of the flower braces	Closing flower braces

Aim: Write a java program to calculate the simple interest

Input:

```

import java.util.Scanner;

public class SimpleInterest {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        |
        // Input Principal amount
        System.out.print("Enter Principal amount: ");
        double principal = scanner.nextDouble();

        // Input Rate of Interest
        System.out.print("Enter Rate of Interest (per annum): ");
        double rate = scanner.nextDouble();

        // Input Time period in years
        System.out.print("Enter Time period (in years): ");
        double time = scanner.nextDouble();

        // Calculate Simple Interest
        double simpleInterest = (principal * rate * time) / 100;

        // Display the result
        System.out.println("Simple Interest: " + simpleInterest);

        scanner.close();
    }
}

```

Output:

```

C:\Users\jahna\OneDrive\Desktop\oops\week1>javac SimpleInterest.java

C:\Users\jahna\OneDrive\Desktop\oops\week1>java SimpleInterest
Enter Principal amount: 5
Enter Rate of Interest (per annum): 600
Enter Time period (in years): 2
Simple Interest: 60.0

```

Errors:

S.no	Error	Correction
1)	';' excepted Scanner.close()	Add ";" Scanner.close();

Aim: Write a java program to convert temperature from degree farhenit To degree celusis

Input:


```

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

Scanner input=new Scanner(System.in);
System.out.println("enter the temperature in celsius");
double celsius=input.nextDouble();
double farhenit=(celsius*9/5)+32;
System.out.println("temperatute in farhenit:"+farhenit);

}
}

```

Output:

```

C:\Users\jahna\OneDrive\Desktop\oops\week1>javac farhenit.java

C:\Users\jahna\OneDrive\Desktop\oops\week1>java farhenit
enter the temperature in celsius
56
temperatute in farhenit:132.8

```

Errors:

S.no	Error	Correction
1)	error: cannot find symbol fh=(tp*9/5)+32;	Declare the variable: double fh=(tp*9/5)+32;

Aim: Write a program to convert temperature from degree celusis To degree farhenit

Input:

```

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

Scanner input=new Scanner(System.in);
System.out.println("enter the temperature in fahrenheit");
double fahrenheit=input.nextDouble();
double celsius=(fahrenheit-32)*5/9;
System.out.println("temperatute in celsius:"+celsius);

}
}

```

Output:

```
C:\Users\jahna\OneDrive\Desktop\oops\week1>javac celsius.java

C:\Users\jahna\OneDrive\Desktop\oops\week1>java celsius
enter the temperature in fahrenheit
78
temperatute in celsius:25.555555555555557
```

Aim: write a java program for factorial number

Input:

```
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: ");
        int number = scanner.nextInt();
        long factorial = 1;

        for (int i = 1; i <= number; i++) {
            factorial *= i;
        }

        System.out.println("Factorial of " + number + " is " + factorial);
    }
}
```

Output:

```
C:\Users\jahna\OneDrive\Desktop\oops\week1>javac Factorial.java

C:\Users\jahna\OneDrive\Desktop\oops\week1>java Factorial
Enter a number: 6
Factorial of 6 is 720
```

Errors:

S.no	Error	Correction
1)	',' excepted factorial*=i	Add "',' factorial*=i;

Aim: write a java program for largest number

Input:

```
import java.util.Scanner;
class LargestNumber{
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        System.out.println("enter a: ");
        int a = input.nextInt();
        System.out.println("enter b: ");
        int b = input.nextInt();
        System.out.println("enter c: ");
        int c = input.nextInt();
        int largest = (a >= b) ? ((a >= c) ? a : c) : ((b >= c) ? b : c);
        System.out.println("Largest Number: " + largest);
    }
}
```

Output:

```
C:\Users\jahna\OneDrive\Desktop\oops\week1>javac LargestNumber.java
C:\Users\jahna\OneDrive\Desktop\oops\week1>java LargestNumber
enter a:
56
enter b:
45
enter c:
90
Largest Number: 90
C:\Users\jahna\OneDrive\Desktop\oops\week1>|
```

Errors:

S.no	Error	Correction
1)	';' excepted int c=input.nextInt()	Add ";" int c=input.nextInt();

WEEK-3

Program-3

Aim: Create the java program with the following instructions.

- 1. Create a class with name car**
- 2. Create four attributes named car-color, car-brand, fuel-type, milage.**
- 3. Create three methods named start, stop, service.**
- 4. Create three objects named car1, car2, car3**

Class diagram:

Car
-car_color: String -car_brand: String -car_fuel_type: String -car_milage: String
+Start(): void +Stop(): void

Input:

```

class Car{
private String car_color;
private String car_brand;
private String car_fuel_type;
private String car_mileage;
public void start()
{
System.out.println("Car started");
}
public void stop()
{
System.out.println("car stopped");
}
public Car()
{
System.out.println("welcome to car garage");
}
public static void main(String[]args){
Car car1=new Car();
car1.car_color ="red";
car1.car_fuel_type="petrol";
car1.car_brand="Honda";

Car car2=new Car();
car2.car_color ="Blue";
car2.car_fuel_type="Diesel";
car2.car_brand="Toyato";

Car car3=new Car();
car3.car_color ="white";
car3.car_fuel_type="Electric";
car3.car_brand="FORD";
car1.start();
car1.stop();

car2.start();
car2.stop();

car3.start();
car3.stop();

System.out.println("car1_color: "+car1.car_color+" brand:"+car1.car_brand+"fuel_type:"+car1.car_fuel_type);
System.out.println("car2_color: "+car2.car_color+" brand:"+car3.car_brand+"fuel_type:"+car2.car_fuel_type);
System.out.println("car3_color: "+car3.car_color+" brand:"+car3.car_brand+"fuel_type:"+car3.car_fuel_type);

}
}

```

Output:

```

C:\Users\jahna\OneDrive\Desktop\oops\week1>javac Car.java

C:\Users\jahna\OneDrive\Desktop\oops\week1>java Car
welcome to car garage
welcome to car garage
welcome to car garage
Car started
car stopped
Car started
car stopped
Car started
car stopped
car1_color: red brand:Hondafuel_type:petrol
car2_color: Blue brand:FORDfuel_type:Diesel
car3_color: white brand:FORDfuel_type:Electric

```

Errors:

S.no	Error	Correction
1)	illegal start of type Public void stop{	Add “)” Public void stop(){
2)	Cannot find symbol thiscar_brand=car_brand	Add “.” this.car_brand=car _brand

Aim: Write a java program for Bank account by using constructor and for withdrawal and deposit .

Class diagram:

BankAccount
-Name: String -AccNo: String -currbal : String
+withdraw(int WAmt): int +deposit(int DAmt): int

Input:

```

class BankAccount {
    String Name;
    int AccNO, currbal;

    BankAccount(String Name, int AccNO, int currbal) {
        this.Name = Name;
        this.AccNO = AccNO;
        this.currbal = currbal;
        System.out.println("The customer details are: " + this.Name + " " + this.AccNO + " " + this.currbal);
    }

    public void withdraw(int withdrawalAmount) {
        if (withdrawalAmount <= currbal) {
            currbal = currbal - withdrawalAmount;
            System.out.println("Remaining balance: " + currbal);
        } else {
            System.out.println("Insufficient funds");
        }
    }

    public int deposit(int depositAmount) {
        currbal = currbal + depositAmount;
        return currbal;
    }

    public static void main(String[] args) {
        BankAccount ram = new BankAccount("ram", 1234, 1000);
        ram.withdraw(500); // Example of valid withdrawal
        ram.withdraw(500); // Another example of valid withdrawal
        int finalAmount = ram.deposit(5001);
        System.out.println("Final balance: " + finalAmount);
    }
}

```

Output:

```

C:\Users\jahna\OneDrive\Desktop\oops\week1>javac BankAccount.java

C:\Users\jahna\OneDrive\Desktop\oops\week1>java BankAccount
The customer details are: ram 1234 1000
Remaining balance: 500
Remaining balance: 0
Final balance: 5001

```

Errors:

S.no	Errors	Correction
1)	; excepted Ram.withdraw(500)	Add ";" Ram.withdraw(500);
2)	Cannot find symbol thisCurrBal=CurrBal	Add "." this.CurrBal=CurrBal

WEEK-4

Program-1

Aim: Write java program with class name Book. This class should contain various attributes such as title of the book, author, year of publication. It should also contain constructor with parameter which initializes the title of the book, author, year of publication. create a method which displays the details of the book that is title of the book, author, year of publication display details of two books by creating two objects.

Class Diagram:

Book
-title: String -author: String -yearOfPublication: int
+Book(): void +display() : void

Input:

```

public class Book {
    private String title;
    private String author;
    private int yearOfPublication;

    public Book(String title, String author, int yearOfPublication) {
        this.title = title;
        this.author = author;
        this.yearOfPublication = yearOfPublication;
    }

    public void displayDetails() {
        System.out.println("Title: " + title);
        System.out.println("Author: " + author);
        System.out.println("Year of Publication: " + yearOfPublication);
    }

    public static void main(String[] args) {

        Book book1 = new Book("To Kill a Mockingbird", "Harper Lee", 1960);
        Book book2 = new Book("1984", "George Orwell", 1949);

        System.out.println("Book 1 Details:");
        book1.displayDetails();

        System.out.println("\nBook 2 Details:");
        book2.displayDetails();
    }
}

```

Output:

```

C:\Users\jahna\OneDrive\Desktop\oops\week1>javac Book.java

C:\Users\jahna\OneDrive\Desktop\oops\week1>java Book
Book 1 Details:
Title: To Kill a Mockingbird
Author: Harper Lee
Year of Publication: 1960

Book 2 Details:
Title: 1984
Author: George Orwell
Year of Publication: 1949

```

Errors:

s.no	Error	Correction
1)	Illegal start of type public void display({	Add “)” public void dispaly(){
2)	Cannot find symbol thisauthor=author;	Add “.” this.author=author;

Program-2

Aim: To create a java class named MyClass with astatic variable "count" of int type initialized to zero and a constant variable pi of type double initialized to 3.1415 as attributes of that class. define a constructor for myclass that increments the count variable each time an object of my class is created. finally print the final values of "count" and "pi"

Class diagram:

MyClass
+count: int +pi : double

Input:

```
public class MyClass {
    public static int count = 0;
    public static final double pi = 3.1415;

    public MyClass() {
        count++;
    }

    public static void main(String[] args) {

        MyClass obj1 = new MyClass();
        MyClass obj2 = new MyClass();
        MyClass obj3 = new MyClass();

        |
        System.out.println("Final Count: " + count);
        System.out.println("Value of Pi: " + pi);
    }
}
```

Output:

```
C:\Users\jahna\OneDrive\Desktop\oops\week1>javac MyClass.java

C:\Users\jahna\OneDrive\Desktop\oops\week1>java MyClass
Final Count: 3
Value of Pi: 3.1415
```

Errors:

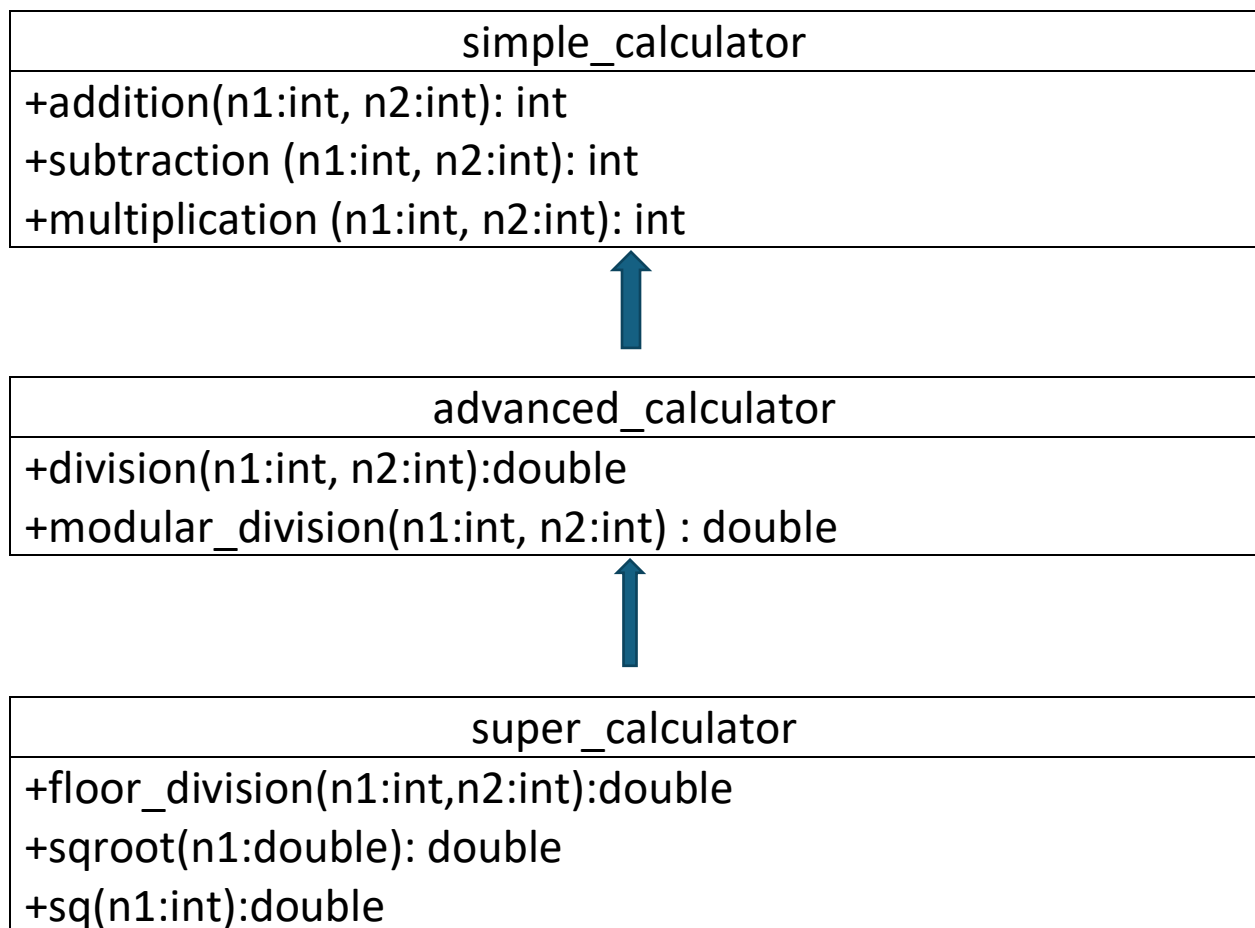
S.no	Error	Correction
1)	;expected MyClass()	Add ; MyClass():
2)	Closing Syntax }	Keeping }

Week-5

Program-1

Aim: Create a calculator using the operations including addition, subtraction, multiplication, division, square, square root, cube, floor using multiple inheritance and display desired output

Class Diagram:



Input:

```

import java.util.Scanner;

class simple_calculator {
    // Methods
    public int addition(int n1, int n2) {
        return n1 + n2;
    }

    public int subtraction(int n1, int n2) {
        return n1 - n2;
    }

    public int multiplication(int n1, int n2) {
        return n1 * n2;
    }
}

class advanced_calculator extends simple_calculator {
    // Methods
    public double division(int n1, int n2) {
        if (n2 != 0) {
            return (double) n1 / n2; // Cast to double to ensure correct division
        } else {
            System.out.println("Division with 0 is not possible.");
            return Double.NaN; // Return NaN for invalid division
        }
    }

    public double modular_division(int n1, int n2) {
        if (n2 != 0) {
            return n1 % n2;
        } else {
            System.out.println("Modular division with 0 is not possible.");
            return Double.NaN; // Return NaN for invalid modulo operation
        }
    }
}

// Derived class from advanced_calculator
class super_calculator extends advanced_calculator {
    // Methods
    public int floor_division(int n1, int n2) {
        if (n2 != 0) {
            return Math.floorDiv(n1, n2); // Proper floor division
        } else {
            System.out.println("Floor division with 0 is not possible.");
            return Integer.MIN_VALUE; // Return a placeholder for invalid operation
        }
    }

    public double sqroot(double n1) {
        if (n1 >= 0) {
            return Math.sqrt(n1);
        } else {
            System.out.println("Square root of negative number is not possible.");
            return Double.NaN; // Return NaN for invalid square root
        }
    }

    public double sq(int n1) {
        return n1 * n1;
    }
}

```

```
// Main class
public class Main {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        System.out.print("Enter the first number: ");
        int a = input.nextInt();
        System.out.print("Enter the second number: ");
        int b = input.nextInt();

        super_calculator sc = new super_calculator();
        System.out.println("Addition: " + sc.addition(a, b));
        System.out.println("Subtraction: " + sc.subtraction(a, b));
        System.out.println("Multiplication: " + sc.multiplication(a, b));
        System.out.println("Division: " + sc.division(a, b));
        System.out.println("Modular division: " + sc.modular_division(a, b));
        System.out.println("Floor Division: " + sc.floor_division(a, b));
        System.out.println("Square Root: " + sc.sqroot(a));
        System.out.println("Square: " + sc.sq(a));

        input.close(); // Fixed missing semicolon
    }
}
```

Output:

Error:

S.no	Error	Correction
1)	‘;’ excepted tv.v_final()	Add “;” tv.v_final();
2)	illegal start of type public void v_final({	Add “)” public void v_final(){

Program:2

Aim: A vehicle enter company wants to develop his system that maintains, information about different types of vehicles available for rent, the company rents out cars and boke and they need a program to store details about each vehicle such as brand and speed

Class Diagram:

Vehicle
brand: string Speed: int
Vehicle(String brand, int speed) DisplayDetails(): void



Input:


```

class Vehicle{
    String brand;
    int speed;
    Vehicle(String brand,int speed){
        this.brand=brand;
        this.speed=speed;
    }
    void Details(){
        System.out.println("Brand:"+brand);
        System.out.println("Speed:"+speed);
    }
}
class CARS extends Vehicle{
    int doors;
    int capacity;
    public CARS(String brand,int speed,int doors,int capacity){
        super(brand, speed);
        this.doors=doors;
        this.capacity=capacity;
    }
    void cardetails(){
        System.out.println("Number of doors:"+doors);
        System.out.println("Capacity:"+capacity);
    }
}
class Bikes extends Vehicle{
    Boolean gears;
    Bikes(String brand,int speed,Boolean gears){
        super(brand, speed);
        this.gears=gears;
    }
    void bikedetails(){
        if (gears==true)
            System.out.println("This bike has gears.");
        else
            System.out.println("This bike does not have gear system.");
    }
}
class Trucks extends Vehicle{
    int tons;
    Trucks(String brand,int speed,int tons){
        super(brand, speed);
        this.tons=tons;
    }
    void truckdetails(){
        System.out.println("The capacity of truck is: "+tons);
    }
}
class Rent{
    public static void main(String[] args){
        CARS c=new CARS("Tayota",120,5,5);
        c.cardetails();
        c.Details();
        Bikes b=new Bikes("KTM",80,true);
        b.bikedetails();
        b.Details();
        Trucks t=new Trucks("TATA",100,1);
        t.truckdetails();
        t.Details();
    }
}

```

Output:

```
C:\Users\jahna\OneDrive\Desktop\oops\week1>javac Vehicle.java

C:\Users\jahna\OneDrive\Desktop\oops\week1>java Rent
Number of doors:5
Capacity:5
Brand:Tayota
Speed:120
This bike has gears.
Brand:KTM
Speed:80
The capacity of truck is: 1
Brand:TATA
Speed:100
```

Errors:

S.no	Error	Correction
1)		
2)		