

**23CSE111**

**OBJECT ORIENTED PROGRAMMING**

**LAB REPORT**



**Department of Computer Science Engineering**

**Amrita School of Computing**

**Amrita Vishwa Vidyapeetham, Amaravati Campus**

**Verified By**

**Name: K.Koushik**

**Roll No: 24137**

## **INDEX**

<b>S.No.</b>	<b>Experiment Name</b>	<b>Page No.</b>	<b>Remarks</b>	<b>Signature</b>
	<b>WEEK-1</b>			
<b>1</b>	<b>Installation Process of JDK</b>			
<b>2</b>	<b>Simple Java Program for printing basic details of student</b>			
	<b>WEEK-2</b>			
<b>1</b>	<b>Write a Java Program to find the factorial of a number</b>			
<b>2</b>	<b>Write a Java Program to find the Fibonacci Series of given length</b>			
<b>3</b>	<b>Write a Java Program to find the temperature from Celsius to Fahrenheit</b>			
<b>4</b>	<b>Write a Java Program to find the Simple Interest</b>			
<b>5</b>	<b>Write a Java Program to find the area of triangle using heron's formula</b>			
<b>6</b>	<b>Write a Java Program to find the area of rectangle</b>			
	<b>WEEK-3</b>			
<b>1</b>	<b>Write a java program with the following instructions:</b>  <b>a. Create class with name car.</b>  <b>b. Create 4 attributes named car color, car brand, fuel type, mileage.</b>  <b>c. Create 3 methods named start(), stop(), service().</b>  <b>d. Create 3 objects C1, C2, C3.</b>			

	e. Create a constructor with parameters with car color, car brand, fuel type, mileage.			
2	Create a class named bank account with methods deposit and withdraw. Where the deposit method should accepts a parameter and when this method is called the deposited amount should be added to current balance. In addition to that when a withdraw method is called it has to verify whether the withdraw amount is less than the current balance. If not display message saying that there are insufficient funds. Use the constructor to display the details of the customer (Name, Account number, IFSC code, Branch). Also create two customer objects C1, C2.			
	WEEK-4			
1	Write a Java program with a class named book. The class should contain various attributes such as title, author, year of publication, and price. It should also contain a constructor with parameters that include title, author, year of publication, and price. Create a method that displays details of the book(Display the details of 3 books that is create 3 objects and display their details)			
2	Create a java program with a Class named "my class" with a static variable 'count' of int type static and initialized to zero and a constant variable 'pi' of type double initialized to 3.1415 as attributes of			

	that class now define a constructor of my class that increments the count variable each time an object of my class is created and finally prints the final values of count and pi variables.			
	<b>WEEK-5</b>			
<b>1</b>	Create a calculator using the operations including addition, subtraction, multiplication and division using multilevel inheritance and display the desired output.			
<b>2</b>	<p>Vehicle rental company wants to develop a system that maintains information about different types of vehicles available for rent. The company rents out cars and bikes and they need a program to store details about each vehicle such as brand and speed.</p> <p>i. Cars should have an additional property: number of doors, Seating capacity.</p> <p>ii. Bikes should have a property indicating whether they have gears or not.</p> <p>iii. The system should also include a function to display details about each vehicle and indicate when a vehicle is starting.</p> <p>iv. Each class should have a constructor.</p> <p><b>Questions:</b></p> <p>1. Which OOP concept is used in the above program? Explain why it is useful in this scenario.</p>			

	<p><b>2. If the company decides to add a new type of vehicle 'Truck', how would you modify the program?</b></p> <p><b>a. Truck should include and additional property capacity (in tons).</b></p> <p><b>b. Create a showTruck() method to display the truck's capacity.</b></p> <p><b>c. Write a constructor for truck that initializes all properties.</b></p> <p><b>3. Implement the truck class and update the main method to create a Truck object and also create an object for car and bike subclasses. Finally display the details.</b></p>			
	<b>WEEK-6</b>			
<b>1</b>	<p><b>Write a Java program to create a vehicle class with a method displayInfo(). Override this method in the car subclass to provide specific information about a car, model, fuel type, and colour using the constructor</b></p>			
<b>2</b>	<p><b>Create a Java program for the scenario.</b></p> <p><b>A college is developing an automated admission system that verifies student eligibility for undergraduate (UG) and postgraduate(PG) programs. Each program has different eligibility criteria based on the student's percentage in their previous qualification.</b></p> <p><b>i) UG admissions require a minimum of 60%</b></p> <p><b>ii) PG admissions require a minimum of 70%</b></p>			

3	<p>Write a Java Program to create a Calculator class with overloaded methods to perform addition: Take the integer values a and b from the user.</p> <p>i) Add two integers</p> <p>ii) Add two doubles</p> <p>iii) Add three integer</p>			
4	<p>Write a Java Program to create a shape class with a method calculateArea() that is overloaded for different shapes(e.g., Square, Rectangle ). Then create a subclass Circle that overrides the calculateArea() method for a circle.</p>			
	WEEK-7			
1	<p>Write a java program to create an abstract class Animal with an abstract method called sound() . Create subclasses Lion and Tiger that extends the animal class and implement the sound method to make a specific sound for each animal.</p>			
2	<p>Write a Java program to create an abstract class shape3D with abstract methods calculateVolume() and caculateSurfaccearea() .Create subclasses sphere and cube that extend the shape3D class and implement the respective methods to calculate the volume and surface area of each shape.</p>			
3	<p>Write a java program using an abstact class to define a method for pattern printing .Create an abstract method printPattern(int n) and a concrete method to display the pattern title</p> <p>→Implement two subclasses</p> <p>1)star pattern – prints a right angled triangle of stars.</p>			

	<b>2)Number pattern – prints a right angled triangle of numbers in increasing order.</b>			
	<b>Week-8</b>			
<b>1.</b>	<b>Write a java Program to create an interface shape with the getPerimeter() method. Create 3 classes Rectangle,circle,Triangle that implements the shape interface. Implement the getPerimeter() method for each of the three classes.</b>			
<b>2.</b>	<b>Write a Java Program to create an interface playable with method play() that takes no arguments and returns void. Create 3 classes Football,VolleyBall,BasketBall that implement the playable interface and override play() method to play the respective sports</b>			
<b>3.</b>	<b>Write a Java program to implement a login system using interfaces</b>			

## WEEK-1

### 1) Explain the process of Installing JDK (Java Development Kit)

**AIM:** To install JDK(Java Development Kit).

#### Installing of JDK (Java Development Kit):

##### 1. Download JDK:

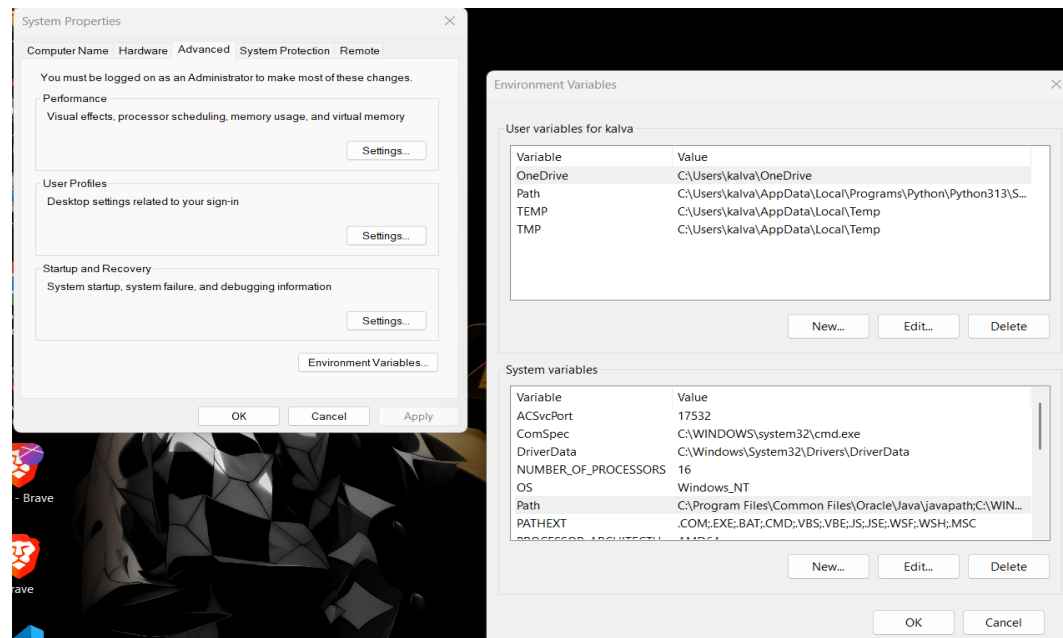
- Go to the Oracle JDK download page in your google and click on JDK-21 version which is Long term support (LTS) version.
- Click on the download link as per your operating system (Windows, macOS, or Linux).

##### 2. Install JDK:

- Once downloaded, run the installer.
- Follow the instructions and keep clicking "Next" until it's done.

##### 3. Set Environment Variables (Windows):

- Open file explorer, then right click on This PC next select on properties then it will take you to the settings app then click on advanced system settings and then click on **Environment Variables**.
- Click on path and new under **System Variables**:
  - **Variable value:** The folder address where JDK is installed (like C:\Program Files\Java\jdk-21\bin)
- Find Path under **System Variables**, click **New**, and add the path of the jdk-21(C:\Program Files\Java\jdk-21\bin)





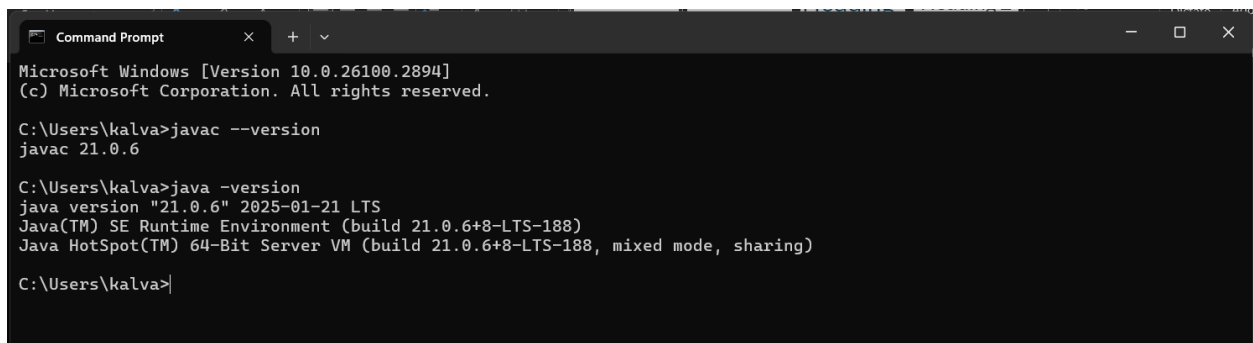
## **Checking of JDK Version:**

### **1. Open Command Prompt:**

- Press win+R, type cmd, and press Enter.

### **2. Check Version:**

- Type java -version and press Enter.
- Type javac --version and press Enter.



```
Microsoft Windows [Version 10.0.26100.2894]
(c) Microsoft Corporation. All rights reserved.

C:\Users\kalva>javac --version
javac 21.0.6

C:\Users\kalva>java -version
java version "21.0.6" 2025-01-21 LTS
Java(TM) SE Runtime Environment (build 21.0.6+8-LTS-188)
Java HotSpot(TM) 64-Bit Server VM (build 21.0.6+8-LTS-188, mixed mode, sharing)

C:\Users\kalva>
```

## **Important Points :**

- 1) Choose the right version
- 2) Download from the official source

## 2) Simple Java Program for printing Name, Class, Roll No, of a Student

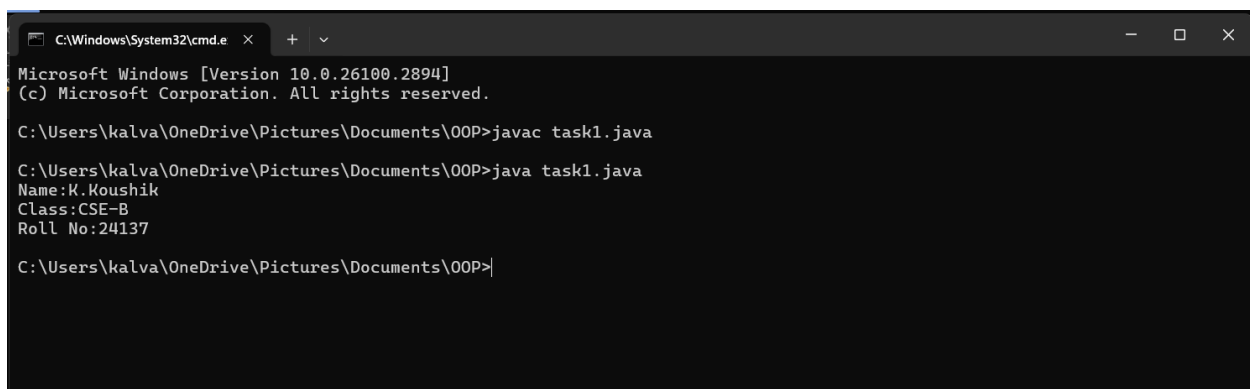
Write your code in Notepad and execute it in cmd prompt

**AIM :** Printing simple java program

### CODE:

```
class Main
{
    public static void main(String[] args)
    {
        System.out.println("Name:K.Koushik");
        System.out.println("Class:CSE-B");
        System.out.println("Roll No:24137");
    }
}
```

### Output:

A screenshot of a Windows Command Prompt window. The title bar shows 'C:\Windows\System32\cmd.exe'. The window content displays the following text: 'Microsoft Windows [Version 10.0.26100.2894] (c) Microsoft Corporation. All rights reserved. C:\Users\kalva\OneDrive\Pictures\Documents\OOP>javac task1.java C:\Users\kalva\OneDrive\Pictures\Documents\OOP>java task1.java Name:K.Koushik Class:CSE-B Roll No:24137 C:\Users\kalva\OneDrive\Pictures\Documents\OOP>'. The output of the Java program is printed on three separate lines: 'Name:K.Koushik', 'Class:CSE-B', and 'Roll No:24137'.

### Important points :

- 1) Make sure that print all the symbols in the correct order.

**Errors:**

S.NO	Error Name	Error Rectification
1	Syntax/ Compilation Error	Absence of Semicolon
2	Closing Brackets	Need to Close the brackets

## **WEEK-2**

Task-1:Write a java program to calculate the simple interest for user wanted details?

Sol

Program-

```
import java.util.Scanner;

class SI{
public static void main(String[] args)
{
    Scanner input=new Scanner(System.in);

    System.out.print("Enter the amount - ");
    double num=input.nextDouble();
    System.out.println("Amount = " + num);

    System.out.print("Enter the rate of interest in % - ");
    double num1=input.nextDouble();
    System.out.println("Rate of interest = " + num1);

    System.out.print("Enter the Time in years - ");
    double num2=input.nextDouble();
    System.out.println("Time = " + num2);

    double output=(num*num1*num2)/100;
    System.out.println("The simple interest is "+output);

    input.close();
}
}
```

**Output:**

```
C:\Users\kalva\OneDrive\Desktop\OOP>javac SI.java
C:\Users\kalva\OneDrive\Desktop\OOP>java SI.java
Enter the amount - 100000
Amount = 100000.0
Enter the rate of interest in % - 15
Rate of interest = 15.0
Enter the Time in years - 3
Time = 3.0
The simple interest is 45000.0
```

**Error Table -**

s.no	Error	Rectification
1	Error : cannot find symbol	Here in nextdouble()  D should be capital  Or else it is not considered as data type

**Task-2(a):** Write a java program to calculate the Area of rectangle for user wanted details?

**Sol**

**Program -**

```
import java.util.Scanner;

class arear{
    public static void main(String[] args){
        Scanner input=new Scanner(System.in);
        System.out.println("enter the length :");
        int a=input.nextInt();
        System.out.println("enter the breadth:");
        int b=input.nextInt();

        double area=a*b;
        System.out.println("Area of rectangle :"+area);
        input.close();  }
}
```

**Output –**

```
C:\Users\kalva\OneDrive\Desktop\00P>javac arearect.java

C:\Users\kalva\OneDrive\Desktop\00P>java arearect.java
enter the length :
50
enter the breadth:
20
Area of rectangle :1000.0
```

**Error Table -**

s.no	Error	Rectification
1	We can get symbol not found error means the system can't find given symbols.	Here in nextInt() I should be capital Or else it is not considered as data type
2	Here we use a,b values as int so we should int values.	Here we should give input values as int domain values.

**Task-2(b):** Write a java program to calculate the Area of triangle for user wanted details?

**Sol**

**Program -**

```
import java.util.Scanner;

class areat{
    public static void main(String[] args){
        Scanner input=new Scanner(System.in);
        System.out.println("enter the side a :");
        int a=input.nextInt();
        System.out.println("enter the side b :");
        int b=input.nextInt();
        System.out.println("enter the side c :");
        int c=input.nextInt();
        double s=(a+b+c)/2;
        double area=(Math.sqrt(s*(s-a)*(s-b)*(s-c)));
        System.out.println("Area of triangle :"+area);
        input.close(); }
}
```

**output -**

```
C:\Users\kalva\OneDrive\Desktop\OOP>javac areatri.java

C:\Users\kalva\OneDrive\Desktop\OOP>java areatri.java
enter the side a :
3
enter the side b :
3
enter the side c :
3
Area of triangle :2.0
```



**Error Table -**

s.no	Error	Rectification
1	Here at output we may end up getting Error at end product.	We should use float or double datatypes.
2	Here we can get wrong value at the end	We should make sure that we should give correct indentation <code>Math.sqrt(s*(s-a)*(s-b)*(s-c))</code> .

**Task-3:**Write a java program to calculate the factorial for user wanted numbers?

**Sol**

**Program -**

```
import java.util.Scanner;

class Main{

    public static void main(String[] args){

        Scanner input=new Scanner(System.in);

        System.out.println("enter a number: ");

        int num=input.nextInt();

        int fact=1;

        for(int i=num;i>0;i--){

            fact=fact*i;

        }

        System.out.print("the factorial of "+num+" is "+fact);

        input.close();

    }

}
```

output –

```

C:\Users\kalva\OneDrive\Desktop\OOP>javac facto.java
C:\Users\kalva\OneDrive\Desktop\OOP>java facto.java
enter the number :
6
the factorial of 6 is 720

```

Error Table -

s.no	Error	Rectification
1	Here at output we may end up getting Error at end product.	We should prefer using the Long datatype for could flexibility
2	Error in recursion may possible	We should make sure that The recursion steps be prefect.

Task-4:Write a java program to calculate the Fibonacci series for user wanted numbers?

Sol

AIM : To find the fibonacci series of any number given by an user.

Program -

```
import java.util.Scanner;
```

```
class Main {  
    public static void main(String[] args){  
        Scanner input=new Scanner(System.in);  
  
        System.out.println("Name : K.Koushik ");  
        System.out.println("Sec : CSE-B" );  
        System.out.println("Roll No : 24137");  
        System.out.print("Enter the Length of fibonacci Series:");  
        int n=input.nextInt();  
        int a=0;  
        int b=1;  
        System.out.println("Series:");  
  
        for(int i=0;i<n;i++){  
            System.out.print(" "+a);  
            int c=a+b;  
            a=b;  
            b=c;  
            input.close();  
        }  
    }  
}  
output –
```

```

C:\Users\kalva\OneDrive\Desktop\SEM-2\OOP\Lab>java fib.java
Name : K.Koushik
Sec : CSE-B
Roll No : 24137
Enter the Length of fibonacci Series:6
Series:
 0  1  1  2  3  5

```

Important points :

1) Understand the logic i.e.,

-c=a+b

-a=b

-b=c

Error Table -

s.no	Error	Rectification
1	We might end up getting wrong output at the end.	Make sure variables are correctly assigned
2	Error in recursion steps	We should make sure that The recursion steps be prefect.

Task-5(a):Write a java program to calculate the temperature of Celsius to Fahrenheit for user wanted numbers?

Sol

Program – Celsius to Fahrenheit-

```
import java.util.Scanner;

class tempf{

    public static void main(String[] args){

        Scanner input=new Scanner(System.in);

        System.out.println("enter the temperature in celsius :");

        double c=input.nextInt();

        double f=((9*c/5)+32);

        System.out.println("the temperature in fahrenheit:"+f);

        input.close(); }

}
```

output –

```
C:\Users\kalva\OneDrive\Desktop\00P>javac tempf.java

C:\Users\kalva\OneDrive\Desktop\00P>java tempf.java
enter the temperature in celsius :
37
the temperature in fahrenheit:98.6
```

Important points :

1)Always remember the formula of Fahrenheit to Celsius

Error Table -

s.no	Error	Rectification
1	Error in datatypes .	Make sure variables are double or float datatypes.
2	We might end up with wrong values at the output	We should make sure that Brackets and multiplication Symbols at correct order.

Task-5(b):Write a java program to calculate the temperature of Fahrenheit to Celsius for user wanted numbers?

Sol

Program – (1) Fahrenheit to Celsius -

```
import java.util.Scanner;

class tempc{
    public static void main(String[] args){
        Scanner input=new Scanner(System.in);
        System.out.println("enter the temperature in fahrenheit:");
        double f=input.nextInt();
        double c=((f-32)*5)/9;
        System.out.println("the temperature in celsius:"+c);
        input.close();  }
}
```

output –

```
C:\Users\kalva\OneDrive\Desktop\00P>javac tempc.java

C:\Users\kalva\OneDrive\Desktop\00P>java tempc.java
enter the temperature in fahrenheit:
98
the temperature in celsius:36.666666666666664
```



## Error Table -

s.no	Error	Rectification
1	Error in datatypes .	Make sure variables are double or float datatypes.
2	We might end up with wrong values at the output	We should make sure that Brackets and multiplication Symbols at correct order.

## Week-3

### Task-1:

Write a java program with the following instructions:

- Create class with name car.
- Create 4 attributes named car color, car brand, fuel type, mileage.
- Create 3 methods named start(), stop(), service().
- Create 3 objects C1, C2, C3.
- Create a constructor with parameters with car color, car brand, fuel type, mileage.

### Program:

```
class Car{
String color;
String brand;
String fuel_type;
double milage;

//constructor
public Car(String color,String brand,String fuel_type,double milage){
this.brand=brand;
this.color=color;
this.fuel_type=fuel_type;
this.milage=milage;
System.out.println("car brand:"+brand+" car color:"+color+" car fuel-type:"+fuel_type+" car milage:"+milage+"km");
}
//end

//methods
public void start(){
System.out.println(brand+" it is well known car "+color+" is iconic color "+fuel_type+" is source and of milage "+milage+"km");
}

public void stop(){
System.out.println(brand+" it is well known car "+color+" is good color "+fuel_type+" is source and of milage "+milage+"km");
}

public void service(){
System.out.println(brand+" it is well known car for its speed "+color+" is beautiful color"+fuel_type+" is source and of milage "+milage+"km");
}
//end

public static void main(String[] args){
Car c1=new Car("BMW","Black","petrol",150);
c1.start();
Car c2=new Car("petrol","oddi","Blue",130);
c2.stop();
Car c3=new Car("red","Tesla","e-power",200);
c3.service();
}
}
```

### Output:

```
C:\Users\kalva\OneDrive\Desktop\OOP>javac car.java

C:\Users\kalva\OneDrive\Desktop\OOP>java car.java
car brand:Black car color:BMW car fuel-type:petrol car milage:150.0km
Black it is well known car BMW is iconic color petrol is source and of milage 150.0km
car brand:oddi car color:petrol car fuel-type:Blue car milage:130.0km
oddi it is well known car petrol is good color Blue is source and of milage 130.0km
car brand:Tesla car color:red car fuel-type:e-power car milage:200.0km
Tesla it is well known car for its speed red is beautiful colore-power is source and of milage 200.0km
```

Error Table -

S.NO	Error	Rectification
1	Syntax/ Compilation Error	Absence of Semicolon
2	Closing Brackets	Need to Close the brackets
3	Class Name Error	Give the class name correctly

Task-2: Create a class named bank account with methods deposit and withdraw. Where the deposit method should accepts a parameter and when this method is called the deposited amount should be added to current balance. In addition to that when a withdraw method is called it has to verify whether the withdraw amount is less than the current balance. If not display message saying that there are insufficient funds. Use the constructor to display the details of the customer (Name, Account number, IFSC code, Branch). Also create two customer objects C1, C2.

### Program:

```
import java.util.Scanner;
class Bank_account{
    long current_balance;
    String name;
    String account_number;
    String IFSE;
    String branch;

    Scanner input=new Scanner(System.in);

    public Bank_account(long current_balance,String name,String account_number,String IFSE,String branch){
        this.current_balance=current_balance;
        this.name=name;
        this.account_number=account_number;
        this.IFSE=IFSE;
        this.branch=branch;
        System.out.println("User name:"+name+" account_number:"+account_number+" IFSE details:"+IFSE+" branch number:"+branch);
    }

    public void deposit(){
        System.out.println("enter the depositing amount: ");
        long deposit_amount=input.nextLong();
        long sum=current_balance+deposit_amount;
        System.out.println("the current blance after depoisting is "+ sum);
    }

    public void withdraw(){
        System.out.println("enter the withdrawing amount: ");
        long withdraw_amount=input.nextLong();
        long dum=current_balance-withdraw_amount;
        if(dum>0){
            System.out.println("the current blance after withdrawal is "+ dum);}
        else{
            System.out.println("the current blance is insufficient ");
        }
    }

    public static void main(String[] args){
        Scanner input=new Scanner(System.in);
        System.out.println("enter the Balance amount: ");
        long amount=input.nextLong();
        Bank_account a1=new Bank_account(amount,"balu","147852369","abcd","502278");
        a1.deposit();
        System.out.println("enter the Balance amount: ");
        long amount1=input.nextLong();
        Bank_account a2=new Bank_account(amount1,"sri","123654789","abcd","502278");
        a2.withdraw();
    }
}
```

**Output:**

```

C:\Users\kalva\OneDrive\Desktop\00P>javac bank.java

C:\Users\kalva\OneDrive\Desktop\00P>java bank.java
enter the Balance amount:
1500
User name:balu account_number:147852369 IFSE details:abcd branch number:502278
enter the depositing amount:
1200
the current blance after depoisting is 2700
enter the Balance amount:
2700
User name:sri account_number:123654789 IFSE details:abcd branch number:502278
enter the withdrawing amount:
2000
the current blance after withdrawal is 700

```

**Error Table -**

S.NO	Error	Rectification
1	Syntax/ Compilation Error	Absence of Semicolon
2	Closing Brackets	Need to Close the brackets
3	Class Name Error	Give the class name correctly

## Week-4

### Task-1:

Write a java program with named book the class should contain various attributes such as title,author,year of publication . It should also contain a constructor with parameters which initializes title,author,year of publication. Create a method which displays the details of the book (Display the details of two books that is create two objects and display their details).

### Program:

```
public class book {
    String title;
    String author;
    int yearOfPublication;
    public book(String title, String author, int yearOfPublication){
        this.title=title;
        this.author=author;
        this.yearOfPublication=yearOfPublication;
    }
    void display(){
        System.out.println("Title: "+title);
        System.out.println("Author: "+author);
        System.out.println("Year of Publication: "+yearOfPublication);
    }
    Run main | Debug main | Run | Debug
    public static void main(String[] args){
        book b1=new book(title:"The Alchemist",author:"Paulo Coelho",yearOfPublication:1988);
        book b2=new book(title:"The Da Vinci Code",author:"Dan Brown",yearOfPublication:2003);
        b1.display();
        b2.display();
    }
}
```

### Output:

```
PS C:\Users\kalva\OneDrive\Desktop\OOP> & 'C:\Program Files\Java\jdk-21\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\kalva\AppData\Roaming\Code\User\workspaceStorage\ddb582c04a81d1ea0cda6cd4e3502a60\redhat.java\jdt_ws\OOP_f2e7b4bf\bin' 'book'
Title: The Alchemist
Author: Paulo Coelho
Year of Publication: 1988
Title: The Da Vinci Code
Author: Dan Brown
Year of Publication: 2003
PS C:\Users\kalva\OneDrive\Desktop\OOP>
```

**Class diagram:**

<b>Book</b>
<b>title: String</b> <b>author: String</b> <b>yearOfPublication: int</b>
<b>Book(title: String, author: String, yearOfPublication: int, price: double)</b> <b>details(): void</b> <b>main(args: String[]): void</b>

**Error Table -**

S.NO	Name	Rectification
1	Syntax/ Compilation Error	Absence of Semicolon
2	Closing Brackets	Need to Close the brackets
3	Class Name Error	Give the class name correctly
4	Constructor Calling	Call the constructor correctly

**Task-2:** Create a java program with a Class named "my class" with a static variable 'count' of int type static and initialized to zero and a constant variable 'pi' of type double initialized to 3.1415 as attributes of that class now define a constructor of my class that increments the count variable each time an object of my class is created and finally prints the final values of count and pi variables.

**Program:**

```
class myclass
{
    static int count=0;
    final double Pi=3.1415;

    public myclass()
    {
        count++;
        System.out.println("The count Value is: " +count);
    }
    Run main | Debug main | Run | Debug
    public static void main(String[] args)
    {
        myclass C1=new myclass();
        myclass C2=new myclass();
        myclass C3=new myclass();
        myclass C4=new myclass();
        myclass C5=new myclass();

        myclass x= new myclass();
        System.out.println("The final pi value is: = "+x.Pi);
    }
}
```



**Output:**

```

PS C:\Users\kalva\OneDrive\Desktop\OOP> & 'C:\Program Files\Java\jdk-21\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\kalva\AppData\Roaming\Code\User\workspaceStorage\ddb582c04a81d1ea0cda6cd4e3502a60\redhat.java\jdt_ws\OOP_f2e7b4bf\bin' 'myclass'
The count Value is: 1
The count Value is: 2
The count Value is: 3
The count Value is: 4
The count Value is: 5
The count Value is: 6
The final pi value is: = 3.1415
PS C:\Users\kalva\OneDrive\Desktop\OOP>

```

**Class diagram:**

<b>myclass</b>
static count: int final Pi: double
myclass() main(args: String[]): void

**Error Table -**

S.NO	Error Name	Error Rectification
1	Syntax/ Compilation Error	Absence of Semicolon
2	Closing Brackets	Need to Close the brackets
3	Class Name Error	Give the class name correctly
4	Constructor Calling	Call the constructor correctly

## **WEEK-5**

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

- Write your code in VS CODE and execute

**- Important Points:**

0. Understand the calling of a Constructor
1. Giving class name correctly
2. Give the parameters Correctly

**CODE:**

```

class calsi{
    void add(int a,int b){
        System.out.println("Sum of Numbers is: "+(a+b));
    }
    void subtract(int a,int b){
        System.out.println("Difference of 2 Numbers: "+(a-b));
    }
}
class extra extends calsi{
    void product(int a,int b){
        System.out.println("Product of 2 numbrs is: "+(a*b));
    }
}
class pro extends extra{
    void divide(int a,int b){
        if (b!=0){
            System.out.println("Dividing of 2 numbers is: "+(a/b));
        }
        else{
            System.out.println(x:"Denominator must not be zero");
        }
    }
}
class Calc{
    Run | Debug | Run main | Debug main
    public static void main(String[] args){
        pro d=new pro();
        d.add(a:6,b:9);
        d.subtract(a:9,b:6);
        d.product(a:23,b:3);
        d.divide(a:4,b:2);
    }
}

```

**OUTPUT :**

```
Sum of Numbers is: 15
Difference of 2 Numbers: 3
Product of 2 numbrs is: 69
Dividing of 2 numbers is: 2
```

**Errors:**

S.NO	Error Name	Error Rectification
1	Syntax/ Compilation Error	Absence of Semicolon
2	Closing Brackets	Need to Close the brackets
3	Class Name Error	Give the class name correctly
4	Constructor Calling	Call the constructor correctly

**2) Vehicle rental company wants to develop a system that maintains information about different types of vehicles available for rent. The company rents out cars and bikes and they need a program to store details about each vehicle such as brand and speed.**

- i. Cars should have an additional property: number of doors, Seating capacity.**
- ii. Bikes should have a property indicating whether they have gears or not.**
- iii. The system should also include a function to display details about each vehicle and indicate when a vehicle is starting.**
- iv. Each class should have a constructor.**

**Questions:**

- 1. Which OOP concept is used in the above program? Explain why it is useful in this scenario.**
- 2. If the company decides to add a new type of vehicle 'Truck', how would you modify the program?**
  - a. Truck should include and additional property capacity (in tons).**
  - b. Create a showTruck() method to display the truck's capacity.**
  - c. Write a constructor for truck that initializes all properties.**
- 3. Implement the truck class and update the main method to create a Truck object and also create an object for car and bike subclasses. Finally display the details.**

**- Write your code in VS CODE and execute**

**- Important Points:**

- 1. Understand the calling of a Constructor**
- 2. Giving class name correctly**
- 3. Give the parameters Correctly**

**CODE:**

```
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(x:"This bike has gears.");
        else
            System.out.println(x:"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
{
    Run | Debug | Run main | Debug main
    public static void main(String[] args)
    {
        CARS c=new CARS(brand:"Ferrari",speed:120,doors:2,capacity:2);
        c.caredetails();
        c.Details();
        Bikes b=new Bikes(brand:"Ninja H2R",speed:80,gears:true);
        b.bikedetails();
        b.Details();
        Trucks t=new Trucks(brand:"Maruti",speed:100,tons:1);
        t.truckdetails();
        t.Details();
    }
}

```

OUTPUT :

```

Number of doors:2
Capacity:2
Brand:Ferrari
Speed:120
This bike has gears.
Brand:Ninja H2R
Speed:80
The capacity of truck is: 1
Brand:Maruti
Speed:100

```

### Errors:

S.NO	Error Name	Error Rectification
1	Syntax/ Compilation Error	Absence of Semicolon
2	Closing Brackets	Need to Close the brackets
3	Class Name Error	Give the class name correctly
4	Constructor Calling	Call the constructor correctly

## **WEEK-6**

**1) Write a Java program to create a vehicle class with a method displayInfo(). Override this method in the car subclass to provide specific information about a car, model, fuel type, and color using the constructor**

**Aim:** To create a code on the vehicles

**Important Points:**

1. Understand the calling of a Constructor
2. Giving class name correctly
3. Give the parameters Correctly

**CODE:**

```
class Vehicle {
```

```
    String Brand;
```

```
    String model;
```

```
    String fuel;
```

```
    String color;
```

```
    int capacity;
```

```
    Vehicle(String Brand, String model, String fuel, int capacity, String color) {
```

```
        this.Brand = Brand;
```

```
        this.model = model;
```

```
        this.fuel = fuel;
```

```
        this.capacity = capacity;
```

```
        this.color = color;
```

```
    }
```

```
    void displayInfo(String Brand, String model, String fuel, int capacity, String color) {
```



```
        System.out.println("Vehicle Details: ");
        System.out.println("Brand: " + Brand);
        System.out.println("Model: " + model);
        System.out.println("Fuel: " + fuel);
        System.out.println("Capacity: " + capacity);
        System.out.println("Color: " + color);
    }
}

class Car extends Vehicle {

    Car(String Brand, String model, String fuel, int capacity, String color) {
        super(Brand, model, fuel, capacity, color);
    }

    void displayInfo() {
        System.out.println("Car Details: ");
        System.out.println("Brand: " + Brand);
        System.out.println("Model: " + model);
        System.out.println("Fuel: " + fuel);
        System.out.println("Capacity: " + capacity);
        System.out.println("Color: " + color);
    }
}

class Week6_1 {

    public static void main(String[] args) {
        // Creating an instance of Car
        Car car1 = new Car("BMW", "X5", "Petrol", 6, "Red");
        car1.displayInfo(); // Display car details
    }
}
```

```
}
```

**OUTPUT:**

```
Car Details:  
Brand: BMW  
Model: X5  
Fuel: Petrol  
Capacity: 6  
Color: Red
```

**Errors:**

S.NO	Error Name	Error Rectification
1	Syntax/ Compilation Error	Absence of Semicolon
2	Closing Brackets	Need to Close the brackets
3	Class Name Error	Give the class name correctly
4	Constructor Calling	Call the constructor correctly

## 2) Create a Java program for the scenario.

A college is developing an automated admission system that verifies student eligibility for undergraduate (UG) and postgraduate(PG) programs. Each program has different eligibility criteria based on the student's percentage in their previous qualification.

i) UG admissions require a minimum of 60%

ii) PG admissions require a minimum of 70%

**Aim:** To create a admission qualification for pg and ug students.

### Important Points:

- 1.Understand the calling of a Constructor
- 2.Giving class name correctly
- 3.Give the parameters Correctly

### CODE:

```
class College{
    String name;
    int percentage;
    void geteligibility(String name,int percentage){
        this.name=name;
        this.percentage=percentage;
    }
}

class UG extends College{
    void geteligibility(String name,int percentage){
        if (percentage>=60){
            System.out.println(name+" is eligible");
        }
        else{
```

```

        System.out.println(name+" is not eligible");
    }
}
}

class PG extends College{
    void geteligibility(String name,int percentage){
        if (percentage>=70){
            System.out.println(name+" is eligible");
        }
        else{
            System.out.println(name+" is not eligible");
        }
    }
}

class week6_2{
    public static void main(String[] args){
        UG ug=new UG();
        ug.geteligibility("Person-1",40);
        PG pg=new PG();
        pg.geteligibility("Person-2",80);
    }
}

```

### OUTPUT:

```

Person-1 is not eligible
Person-2 is eligible
PS C:\Users\kalva\OneDrive\Des

```

### Errors:

S.NO	Error Name	Error Rectification
1	Syntax/ Compilation Error	Absence of Semicolon
2	Closing Brackets	Need to Close the brackets
3	Class Name Error	Give the class name correctly
4	Constructor Calling	Call the constructor correctly

**3) Write a Java Program to create a Calculator class with overloaded methods to perform addition: Take the integer values a and b from the user.**

**i) Add two integers**

**ii) Add two doubles**

**iii) Add three integers**

**Aim:** Make a code to create a calculator for addition.

**Important Points:**

- 1.Understand the calling of a Constructor
- 2.Giving class name correctly
- 3.Give the parameters Correctly

### **CODE:**

```
class Calc{

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

class week6_3{

    public static void main(String[] args){
        Calc C1=new Calc();
```

```

System.out.println("Sum of 6 and 9 is: "+C1.add(6,9));

System.out.println("Sum of 7.6 and 8.6 is: "+C1.add(7,6.8));

System.out.println("Sum of 2,4 and 6 is: "+C1.add(2,4,8));

}

}

```

### OUTPUT:

```

Sum of 6 and 9 is: 15
Sum of 7.6 and 8.6 is: 13.8
Sum of 2,4 and 6 is: 14
PS C:\Users\kalva\OneDrive\De

```

### Errors:

S.NO	Error Name	Error Rectification
1	Syntax/ Compilation Error	Absence of Semicolon
2	Closing Brackets	Need to Close the brackets
3	Class Name Error	Give the class name correctly
4	Constructor Calling	Call the constructor correctly

**4) Write a Java Program to create a shape class with a method calculateArea() that is overloaded for different shapes(e.g., Square, Rectangle ). Then create a subclass Circle that overrides the calculateArea() method for a circle.**

**Aim:** To find area of few shapes using overloading and overriding.

**Important Points:**

- 1.Understand the calling of a Constructor
- 2.Giving class name correctly
- 3.Give the parameters Correctly

**CODE:**

```
class Shape {

    double calculateArea(double side) {
        return side * side;
    }

    double calculateArea(double width, double height) {
        return width * height;
    }
}

class Circle extends Shape {

    double calculateArea(double radius) {
        return 3.14 * radius * radius;
    }
}

class Week6_4 {
```



```

public static void main(String[] args) {
    Shape S1 = new Shape();

    System.out.println("Area of square: " + S1.calculateArea(6));

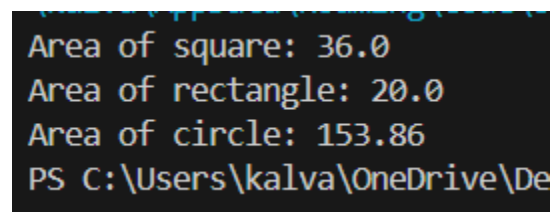
    System.out.println("Area of rectangle: " + S1.calculateArea(2, 10));

    Circle C1 = new Circle();

    System.out.println("Area of circle: " + C1.calculateArea(7));
}
}

```

### OUTPUT:



```

Area of square: 36.0
Area of rectangle: 20.0
Area of circle: 153.86
PS C:\Users\kalva\OneDrive\De

```

### Errors:

S.NO	Error Name	Error Rectification
1	Syntax/ Compilation Error	Absence of Semicolon
2	Closing Brackets	Need to Close the brackets
3	Class Name Error	Give the class name correctly
4	Constructor Calling	Call the constructor correctly

## WEEK – 7

1) Write a Java program to create an abstract class `Animal` with an abstract method called `sound()`. Create subclasses `Lion` and `Tiger` that extend the `Animal` class and implement the `sound` method to make a specific sound for each animal.

**Aim :** To create separate classes using an abstract class of the animals.

**Important points:**

- 1) Make sure that you are using an abstract class and the method.
- 2) Do not forget the order of the symbols.

**Code :**

```
abstract class Animal {  
    abstract void sound();  
}  
  
class Tiger extends Animal {  
    void sound() {  
        System.out.println("Tiger roars....");  
    }  
}  
  
class Lion extends Animal {  
    void sound() {  
        System.out.println("Lion roars.....");  
    }  
}  
  
class Main {  
    public static void main(String[] args) {  
        System.out.println("Name:Koushik Sec:CSE-B RollNo:241137");  
        Tiger t = new Tiger();  
        t.sound();  
        Lion l = new Lion();  
        l.sound();  
    }  
}
```

**Output:**

Positive -

```

Name:Koushik  Sec:CSE-B  RollNo:241137
Tiger roars....
Lion roars.....
PS C:\Users\kalva\OneDrive\Desktop\SEM-2\OOP\Lab\week-7>

```

Negative –

```

Exception in thread "main" java.lang.Error: Unresolved compilation problem:
    Syntax error, insert ";" to complete Statement

    at Main.main(animal.java:20)
PS C:\Users\kalva\OneDrive\Desktop\SEM-2\OOP\Lab\week-7>

```

**Error Table:**

S.NO	Error Name	Error Rectification
1	Syntax/ Compilation Error	Absence of Semicolon
2	Closing Brackets	Need to Close the brackets
3	Class Name Error	Give the class name correctly
4	Calling Method	Call the method correctly

2) Write a Java program to create an abstract class shape3D with abstract methods calculateVolume() and calculateSurfaceArea(). Create subclasses sphere and cube that extend the shape3D class and implement the respective methods to calculate the volume and surface area of each shape.

**Aim :** To find the volume and the surface area of the different 3D shapes

**Important points :**

- 1) Make sure that the entering formula must be right .
- 2) Also place the formula at the right place.

**Code:**

```
import java.lang.*;

abstract class Shape3D {

    abstract void CalculateVolume();

    abstract void CalculateSurfaceArea();

}

class Cube extends Shape3D {

    double edge;

    public Cube(double edge) {

        this.edge = edge;

    }

    public void CalculateVolume(){

        double e1=Math.pow(edge,3);

        System.out.println("Volume of Cube is "+e1);

    }

    public void CalculateSurfaceArea(){

        double e2= 6 * Math.pow(edge,2);

        System.out.println("Surface area of Cube is "+e2);

    }

}
```

```
class Sphere extends Shape3D {  
    double radius;  
    public Sphere(double radius){  
        this.radius = radius;  
    }  
    public void CalculateVolume(){  
        double r1= (4/3)*Math.PI*Math.pow(radius,3);  
        System.out.println("Volume of Sphere is "+r1);  
    }  
    public void CalculateSurfaceArea(){  
        double r2=4* Math.PI*Math.pow(radius,2);  
        System.out.println("Surface area of Sphere is "+r2);  
    }  
}
```

```
class sol{  
    public static void main(String[] args) {  
        System.out.println("Name:Koushik Sec:CSE-B RollNo:24137");  
        Cube c=new Cube(5);  
        c.CalculateVolume();  
        c.CalculateSurfaceArea();  
        Sphere s=new Sphere(7);  
        s.CalculateVolume();  
        s.CalculateSurfaceArea();  
    }  
}
```

**Output:**

```

Name:Koushik Sec:CSE-B RollNo:24137
Volume of Cube is 125.0
Surface area of Cube is 150.0
Volume of Sphere is 1077.5662801812991
Surface area of Sphere is 615.7521601035994
PS C:\Users\kalva\OneDrive\Desktop\SEM-2\OOP\Lab\week-7>

```

**Error Table:**

S.NO	Error Name	Error Rectification
1	Formula placement	Arrange the formula at the right place
2	Syntax/ Compilation Error	Absence of Semicolon
3	Closing Brackets	Need to Close the brackets
4	Class Name Error	Give the class name correctly
5	Calling Method	Call the method correctly

**3)** Write a java program using an abstract class to define a method for pattern printing .Create an abstract method printPattern(int n) and a concrete method to display the pattern title

→ Implement two subclasses

1) star pattern – prints a right angled triangle of stars.

2) Number pattern – prints a right angled triangle of numbers in increasing order.

**Aim :** To build a a star pattern and a number pattern.

**Important points :**

- 1) Use the nested loops in correct order.
- 2) Enter the logic in the correct way.

**Code:**

```
abstract class patternPrinter {
    abstract void printpattern(int n);
    public void pattern_title(String title){
        System.out.println("Pattern of "+title);
    }
}
```

```
class star extends patternPrinter{
    public void printpattern(int n){
        for(int i=0;i<=n;i++){
            for(int j=0;j<=i;j++){
                System.out.print("* ");
            }
            System.out.print("\n");
        }
    }
}
```

```
class Number extends patternPrinter{
```

```
public void printpattern(int n){
    for(int i=0;i<=n;i++){
        for(int j=0;j<=i;j++){
            System.out.print(j+1+" ");
        }
        System.out.print("\n");
    }
}

public class Main{
    public static void main(String[] args) {
        System.out.println("Name:Koushik Sec:CSE-B RollNo:24137");
        star s = new star();
        s.pattern_title("stars");
        s.printpattern(5);
        System.out.println("-----");
        Number n = new Number();
        n.pattern_title("numbers");
        n.printpattern(5);
    }
}
```



**Output:**

```

Name:Koushik Sec:CSE-B RollNo:24137
Pattern of stars
*
* *
* * *
* * * *
* * * * *
* * * * *
-----
Pattern of numbers
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
1 2 3 4 5 6
PS C:\Users\kalva\OneDrive\Desktop\SEM-2\OOP\Lab\week-7>

```

**Error Table:**

S.NO	Error Name	Error Rectification
1	Logic error	By learning the correct logic
2	Syntax/ Compilation Error	Absence of Semicolon
3	Closing Brackets	Need to Close the brackets
4	Class Name Error	Give the class name correctly
5	Calling Method	Call the method correctly

## WEEK-8

**1) Write a Java program to create an interface Shape with the getPerimeter() method. Create three classes Rectangle, Circle, and Triangle that implement the Shape interface. Implement the getPerimeter() method for each of the three classes.**

**AIM:**

**To create a java program to find the perimeter of the different shapes.**

**IMPORTANT POINTS:**

- 1)** Demonstrates interface implementation across different classes.
- 2)** Encourages polymorphism with interface references.
- 3)** Enforces consistent method contract across unrelated classes.

**Code:**

```
interface Shape{
    void getPerimeter();
}

class Rectangle implements Shape{
    double l,b;

    public Rectangle(double l, double b){
        this.l=l;
        this.b=b;
    }

    public void getPerimeter(){
        System.out.println("Perimeter of Rectangle: " + (2*(l+b)));
    }
}
```

```
class Circle implements Shape{  
    double r;  
  
    public Circle(double r){  
        this.r=r;  
    }  
  
    public void getPerimeter(){  
        System.out.println("Perimeter of Circle: " + (2*Math.PI*r));  
    }  
}
```

```
class Triangle implements Shape{  
    double a,b,c;  
  
    public Triangle(double a,double b,double c){  
        this.a=a;  
        this.b=b;  
        this.c=c;  
    }  
  
    public void getPerimeter(){  
        System.out.println("Perimeter of Triangle: " + (a+b+c));  
    }  
}
```

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

```
System.out.println("Name:Koushik Sec:CSE-B RollNo:24137");
```

```
Rectangle rect = new Rectangle(12,5);
```

```
rect.getPerimeter();
```

```
Circle circle = new Circle(4);
```

```
circle.getPerimeter();
```

```
Triangle triangle = new Triangle(5,7,9);
```

```
triangle.getPerimeter();
```

```
}
```

```
}
```

**Output:**

```
Name:Koushik Sec:CSE-B RollNo:24137
Perimeter of Rectangle: 34.0
Perimeter of Circle: 25.132741228718345
Perimeter of Triangle: 21.0
```

**Error:**

SI.NO	ERROR MESSAGE	ERROR RECTIFICATION
1.	PatternPrinter.java:22: error: class, interface, enum, or record expected	Condition needs to be verified
2.	Class Name Error	Give the class name correctly

**2)Write a Java program to create an interface Playable with a method play() that takes no arguments and returns void. Create three classes Football, Volleyball, and Basketball that implement the Playable interface and override the play() method to play the respective sports.**

**Aim :** Create a java program to sports like football,volleyball and basketball.

**Important point:**

- 1) Implement play() differently for each sport
- 2)Interface implementation
- 3)Specific sport behaviors

**Code:**

```
interface playable{  
    public void play();  
}
```

```
class football implements playable{  
    @Override  
    public void play() {  
        System.out.println("Playing Football");  
    }  
}
```

```
class basketball implements playable{  
    @Override  
    public void play() {  
        System.out.println("Playing Basketball ");  
    }  
}
```

```

class volleyball implements playable{
    @Override
    public void play() {
        System.out.println("Playing Volleyball ");
    }
}

```

```

public class question_26 {
    public static void main(String[] args) {
        System.out.println("Name:Koushik Sec:CSE-B RollNo:24137");
        football f = new football();
        basketball b = new basketball();
        volleyball v = new volleyball();
        f.play();
        b.play();
        v.play();
    }
}

```

### **Output:**

```

Name:Koushik Sec:CSE-B RollNo:24137
Playing Football
Playing Basketball
Playing Volleyball
PS C:\Users\kalva\OneDrive\Desktop\SEM-2\OOP\Assignment>

```

### **Errors:**

S.No	error	Error rectification
1	play() not overridden in sports classes.	Implement play()
2	Error: ";" expected	Giving the ';' in wanted places

**3)Write a java program to implement login system using interfaces.**

**Aim: To make a login system using interfaces.**

**Important points:**

- 1) Add if statement in the code for error of the password.
- 2) Add Boolean method so that it tells whether the given input is right or wrong.

**Code:**

```
interface loginportal {
    boolean login(String id,String pass);
}
class Universityportal implements loginportal{
    public boolean login(String id,String pass){
        if ((id == "Balu") && (pass == "2707")){
            System.out.println("Login success.....");
            return true;
        }
        else{
            System.out.println("Invalid Credentials..??");
            return false;
        }
    }
}
class login{
    public static void main(String [] args){
        System.out.println("Name:Koushik Sec:CSE-B RollNo:24137");
        Universityportal l = new Universityportal();
        l.login("Balu", "2707");
    }
}
```

**Output:**

```
PS C:\Users\kalva\OneDrive\Desktop\SEM-2\OOP\Lab\week-8>
age\767912bc6ea646cd011bb7182195206f\redhat.java\jdt_ws\
Name:Koushik Sec:CSE-B RollNo:24137
Login success.....
```

**Errors:**

S.No	error	Error rectification
1	Error in if statement without any “{“	Placing { for it
2	Error: ”;” expected	Giving the ‘;’ in wanted places