**LAB MANUAL**



**ROLLNO:AV.SC.U4CSE24124**

**NAME: G . Balaji Ajay**

**SECTION: CSE-B**

**INDEX**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.No.** | **Experiment Name** | **Page No.** | **Remarks** | **Signature** |
|  | **WEEK-1** |  |  |  |
| 1 | Installation Process of JDK |  |  |  |
| 2 | Simple Java Program for printing basic details of student |  |  |  |
|  | **WEEK-2** |  |  |  |
| 1 | Write a Java Program to find the factorial of a number |  |  |  |
| 2 | Write a Java Program to find the Fibonacci Series of given length |  |  |  |
| 3 | Write a Java Program to find the temperature from Celsius to Fahrenheit |  |  |  |
| 4 | Write a Java Program to find the Simple Interest |  |  |  |
| 5 | Write a Java Program to find the area of triangle using heron’s formula |  |  |  |
| 6 | Write a Java Program to find the area of rectangle |  |  |  |
|  | WEEK-3 |  |  |  |
| 1 | Write a java program with the following instructions:  a. Create class with name car.  b. Create 4 attributes named car color, car brand, fuel type, mileage.  c. Create 3 methods named start(), stop(), service().  d. Create 3 objects C1, C2, C3.  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. |  |  |  |
|  | **WEEK-5** |  |  |  |
| 1 | Create a calculator using the operations including addition, subtraction, multiplication and division using multilevel in heritance and display the desired output. |  |  |  |
| 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. |  |  |  |
|  | **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 colour using the constructor |  |  |  |
| 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% |  |  |  |
| 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 integer |  |  |  |
| 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. |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**WEEK-1:**

**Aim:** How to install jdk and first program on

printing student details*.*

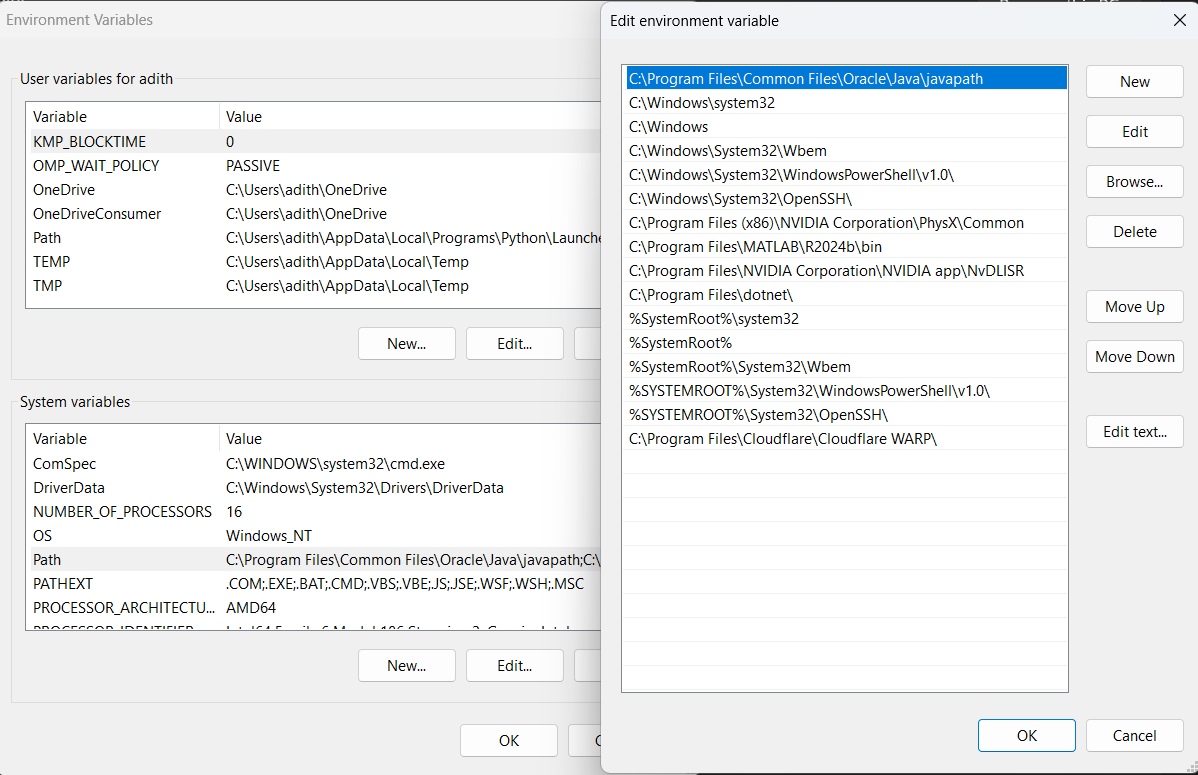
**Step-1:** Download JDK-21 from oracle website

**

**Step-2:**Install the JDK-21 with accepting terms and

conditions according to the respective windows.

**Step-3**:Setting up environmental variables.



\*Windows c -> C-drive -> program files ->Java -

>JDK-21->select bin

\*Select and open environmental variable in search

bar-> either select system variables or user

variables-> select path-> click edit->New-> paste

the bin-> finish the setup(apply the changes).

~for verifying the installed version

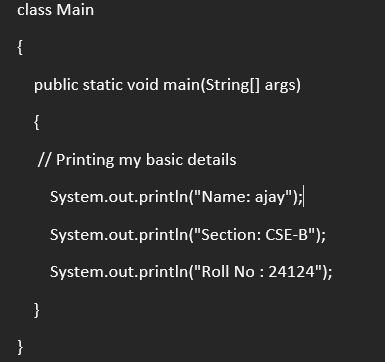
Open cmd-> type java --version

~command propt

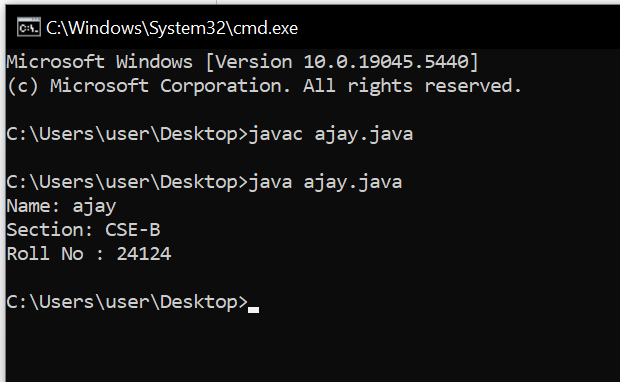
Javac filename.java ->compiling.

Java filename.java ->displaying

**PROGRAM-1(Rectified):**

******

**Output:**



**WEEK-2:**

**PROGRAM-1:**

**Aim:**Write a java program for SI

**

**Output:**

******

**ERROR TABLE*:***

|  |  |
| --- | --- |
| **Code Error** | **Code rectification** |
| 1.Giving space between next and Double.  2.Not giving parenthesis after closing the input. | 1.Should not give space between next and Double.  2.We must put parenthesis after closing the input. |

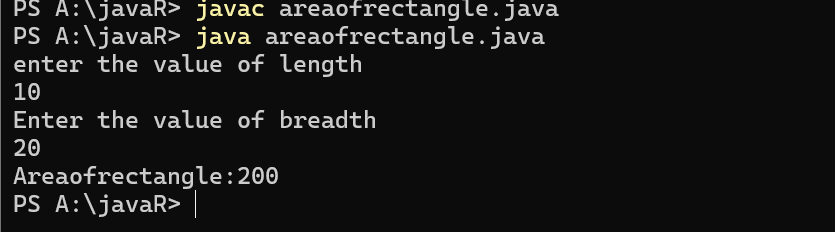
**PROGRAM-2:**

**Aim:**Write a program in java for area of

rectangle.

**

**Output:**

******

**ERROR TABLE:**

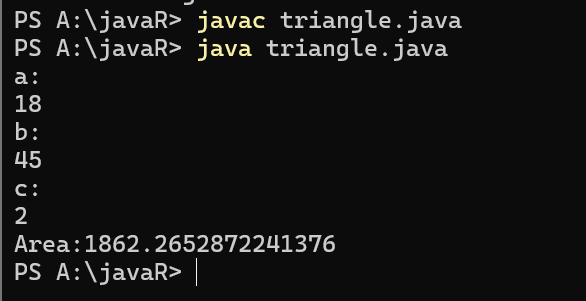
|  |  |
| --- | --- |
| **Code Error** | **Code rectification** |
| 1.While using for iteration, not giving the conditions correctly.  2.Declaring the data type as double instead of int. | 1.We should give iterative statements correctly.  2.We should give the data type as int for integers. |

**PROGRAM-3:**

**Aim:**Write a program in java for area oftriangle using heron’s formula.



**Output:**

******

ERROR TABLE:

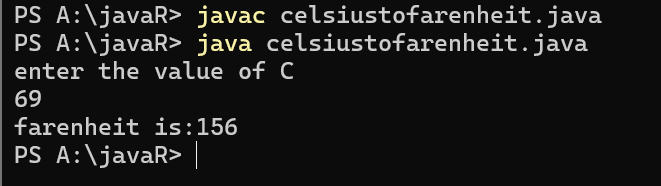
|  |  |
| --- | --- |
| **Code Error** | Code rectification |
| 1.While printing the variable not giving + sign.  2.Not closing the scanner. | 1.We should give correct indentation.  2.Closing the scanner is must. |

**PROGRAM-4(a):**

**Aim:**Write a program in java for converting temperature from celsius to fahrenite.



OUTPUT:

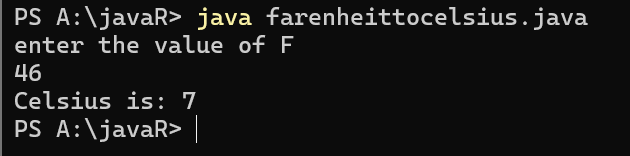


**PROGRAM-4(b):**

**Aim:**Write a program in java for converting temperature from fahrenite to celsius.

******

**Output:**

******

**ERROR TABLE:**

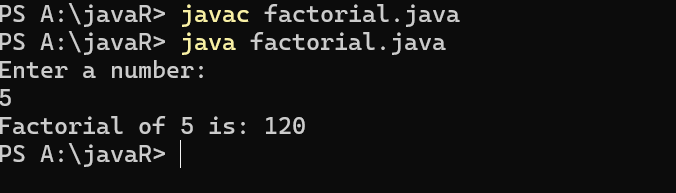
|  |  |
| --- | --- |
| **Code Error** | **Code rectification** |
| 1.While printing the variable not giving + sign.  2.Not closing the scanner. | 1.We should give correct indentation.  2.Closing the scanner is must. |

**PROGRAM-5:**

**Aim:**Write a program in java for factorial of a number.

******

OUTPUT:



ERROR TABLE:

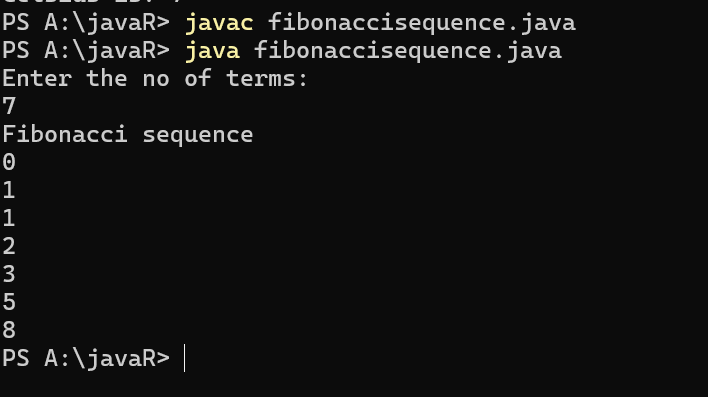
|  |  |
| --- | --- |
| **Code Error** | **Code rectification** |
| 1.While using for iteration, not giving the conditions correctly.  2.Declaring the data type as double instead of int. | 1.We should give iterative statements correctly.  2.We should give the data type as int for integers. |

**PROGRAM-6:**

**Aim:**Write a program in java for fibonacci series.



OUTPUT:



ERROR TABLE:

|  |  |
| --- | --- |
| **Code Error** | **Code rectification** |
| 1.Giving space between next and Double.  2.Not giving parenthesis after closing the input. | 1.Should not give space between next and Double.  2.We must put parenthesis after closing the input. |

**WEEK -3:**

**PROGRAM-1:**

**AIM: To create java program with following instructions :**

**1.Create a class with name Car**

**2.Create four attributes named car\_color,car\_brand, fuel\_type, mileage**

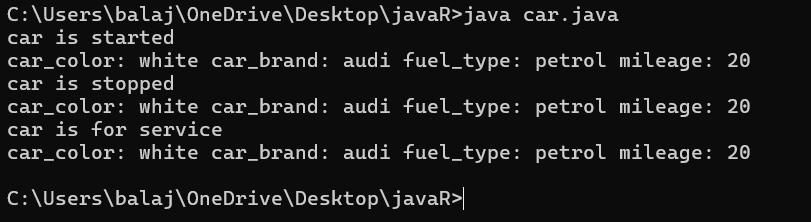
**3.Create these methods named start(),stop(),service()**

**4.Create the objects named car, car1,car2**

**CODE:**

****

**OUTPUT:**

****

**Error table:**

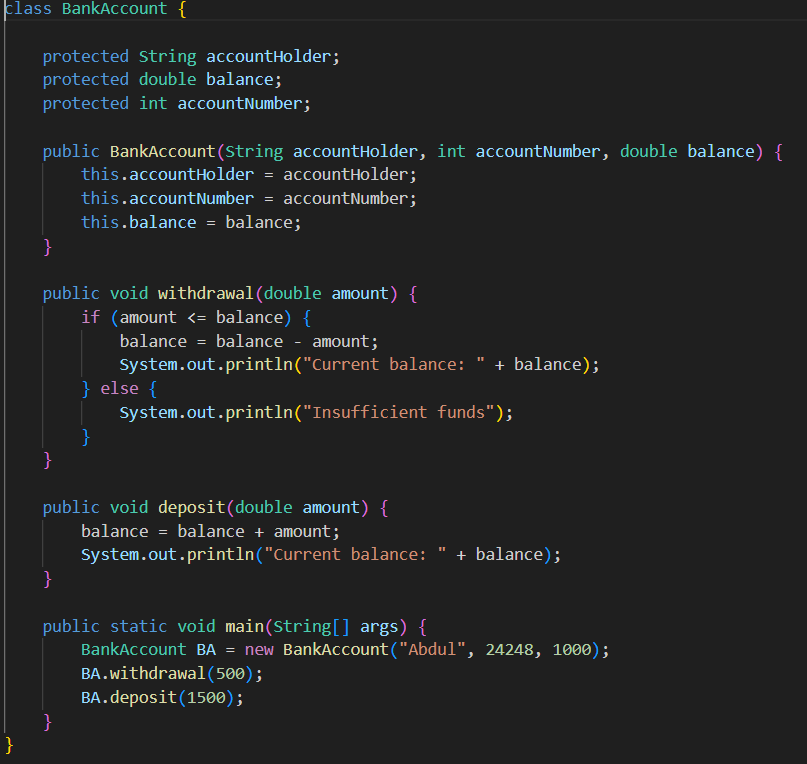
|  |  |  |  |
| --- | --- | --- | --- |
| **S.no** | **Error name** | **Cause of error** | **Rectification** |
| **1** | **Syntax Error** | **Missing ‘{‘** | **‘{‘ added** |
| **2** | **Compile time Error** | **Mispelled Variable call** | **Rectified with**  **Correct variable name** |
| **3** | **Case sensitive error** | **Uppercase and lowercase** | **rectified** |

**Class diagram:**

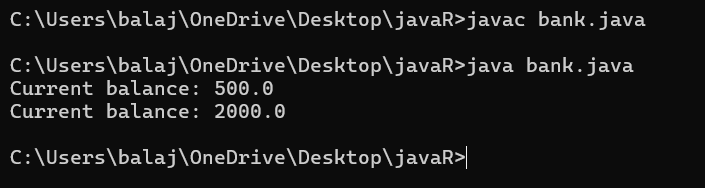
|  |
| --- |
| **car**  **---------------------- --- -**  **-car\_color:string**  **-car\_brand:string -**  **-fuel\_type:string**  **-milage:double**  **---------------------- ---**  **+start():void**  **+stop():void**  **+service():void** |

**PROGRAM-2:**

**Aim:To create a class BankAccount with methods deposit() and withdraw() create two subclasses savingsaccount and checkingaccount override the withdraw method in each subclass to impose different withdrawal limits and fees  
  
CODE:**

****

**OUTPUT:**

****

**Error table:**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.no** | **Error name** | **Error name** | **Rectification** |
| **1** | **Name Error** | **Undefined name** | **Correct variable**  **Name replaced** |
| **2** | **Syntax Error** | **Missing Parenthesis** | **Parenthesis Added** |
| **3** | **Logical Error** | **Incorrect Condition** | **Condition Rectified** |

**Class diagram:**

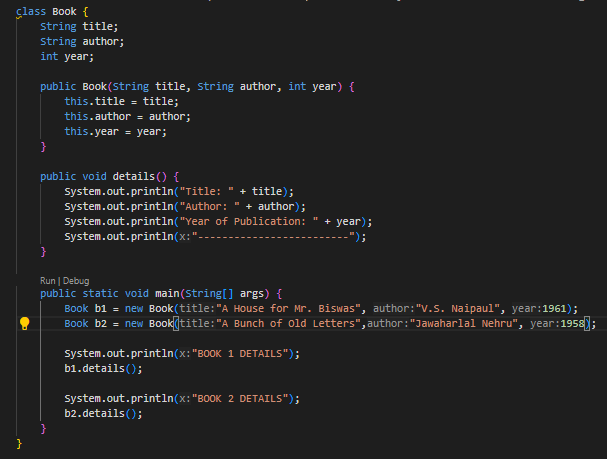
|  |
| --- |
| **BankAccount**  **----------------------------------------------------------**  **-balance: double**  **----------------------------------------------------------**  **+BankAccount(intialBalance: double)**  **+deposit(amount: double):void**  **+withdraw(amount: double):void** |

**Week-4**

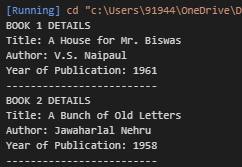
**Program 1:**

**AIM : Write a java program with class named book the class should contain various attributes such as title ,author, year of publication .It should also contain a constructor with parameters with initializes title ,author and year of publication . create a method which displays the details of the book (display the details of two book i.e, create two books and objects with details).**

**Code:**

****

**Output:**

****

**Error :**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Error type** | **Reason for error** | **Rectification** |
| **1** | **syntax error** | **Forgot to keep }**  **At last** | **} is added** |
| **2** | **Logical error** | **Incorrect logic** | **Correct logic** |
|  |  |  |  |

**Class diagram:**

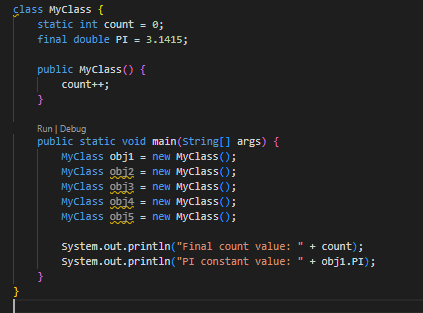
|  |
| --- |
| **Book** |
| **title : String**  **author : String**  **year : int** |
| **+ book(title: String, author: String, year: int)**  **+ details(): void** |

**Program 2:**

**AIM :**

**Write a java program to create a class named myclass with a static variable count of int type and initialize to zero and a constant variable pie of double data type ,initialize to 3.1415 as attributes of that class now define a constructor for my class that increments the count variables each time an object of my class is created variable each time an object of myclass is created. Finally print the final values of count and pie variables.**

**Code:**



**Output:**

**C:\Users\91944\OneDrive\Pictures\Screenshots\Screenshot 2025-03-12 013521.png**

Error:

|  |  |  |  |
| --- | --- | --- | --- |
| S.No | Error type | Reason for error | Rectification |
| 1 | syntax error | String forgot in main function | String is added |
| 2 | Logical error | Incorrect logic | Correct logic |

**Class diagram:**

|  |
| --- |
| Myclass |
| Count : int  PIE : double |
| +Myclass() |

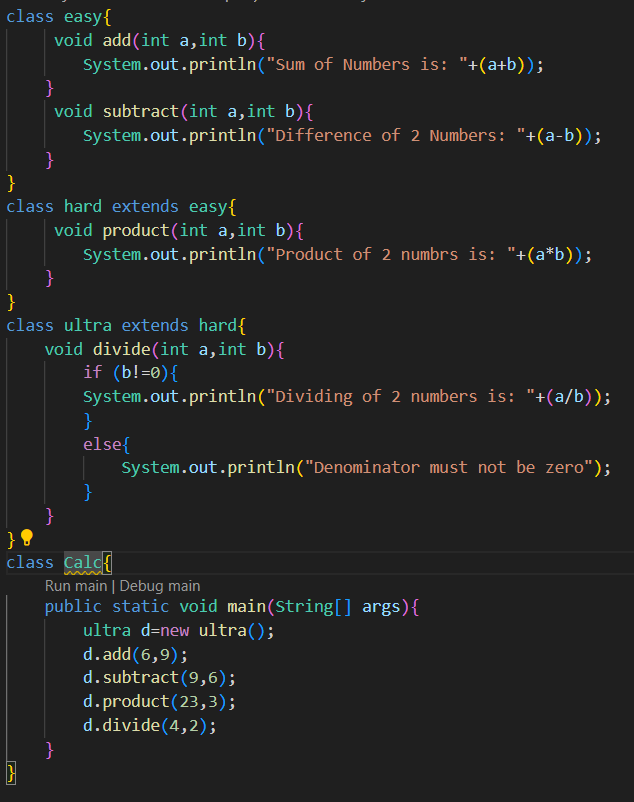
**WEEK-5  
  
1) Create a calculator using the operations including addition, subtraction, multiplication and division using multilevel in heritance and display the desired output.**

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

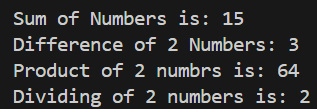
**- Important Points:**

* + - * **Understand the calling of a Constructor**
      * **Giving class name correctly**
      * **Give the parameters Correctly**

**CODE:**

****

**OUTPUT:**

 **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).**

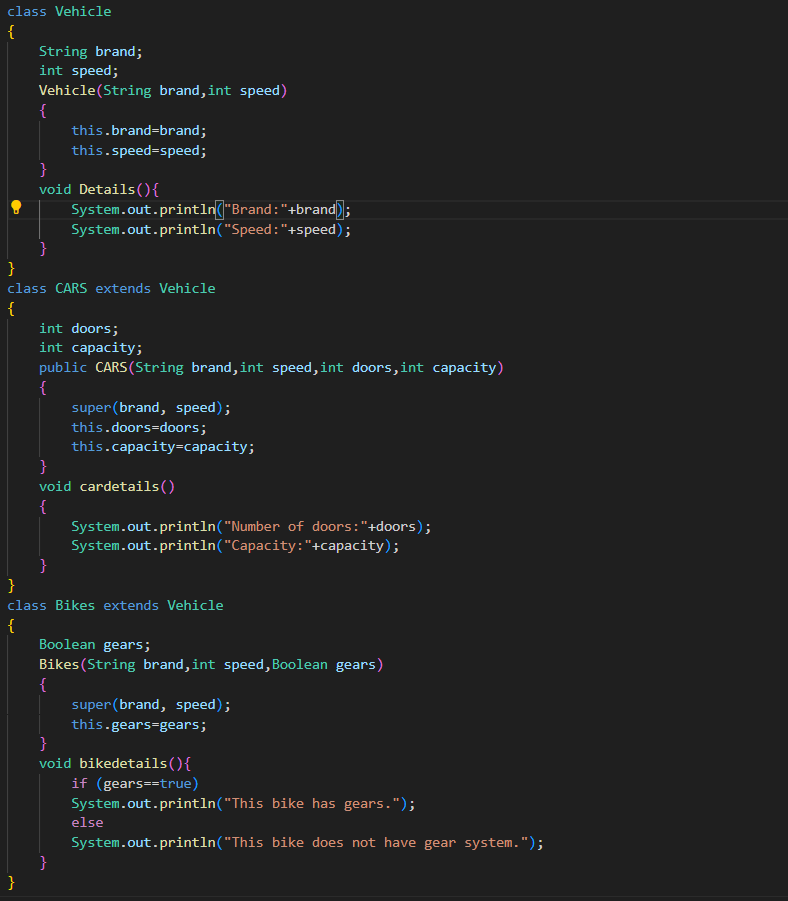
**b. Create a showTruck() method to display the truck’s capacity.**

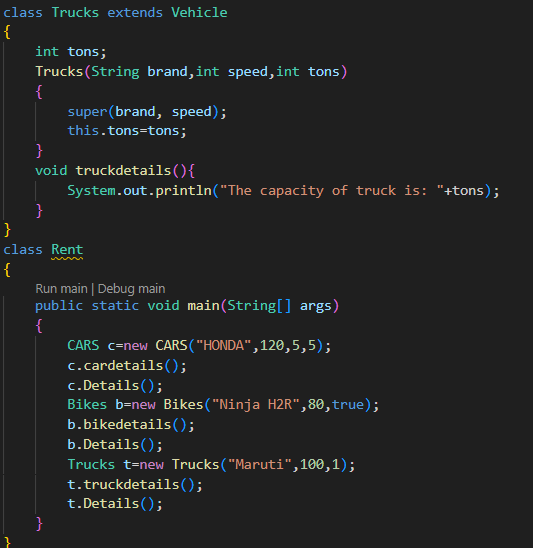
**c. Write a constructor for truck that initializes all properties.**

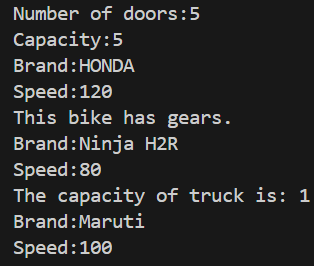
**3. Implement the truck class and update the main method to create a Truck object and also create an object for car and bike subclasses. Finally display the details.  
  
  
  
- Write your code in VS CODE and execute**

**- Important Points:**

* + - * **Understand the calling of a Constructor**
      * **Giving class name correctly**
      * **Give the parameters Correctly**

**CODE:  
**



**OUTPUT:**

**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 |

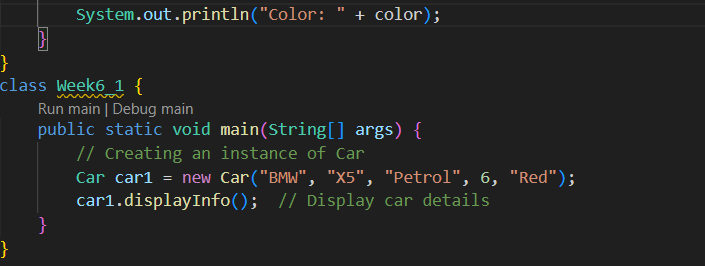
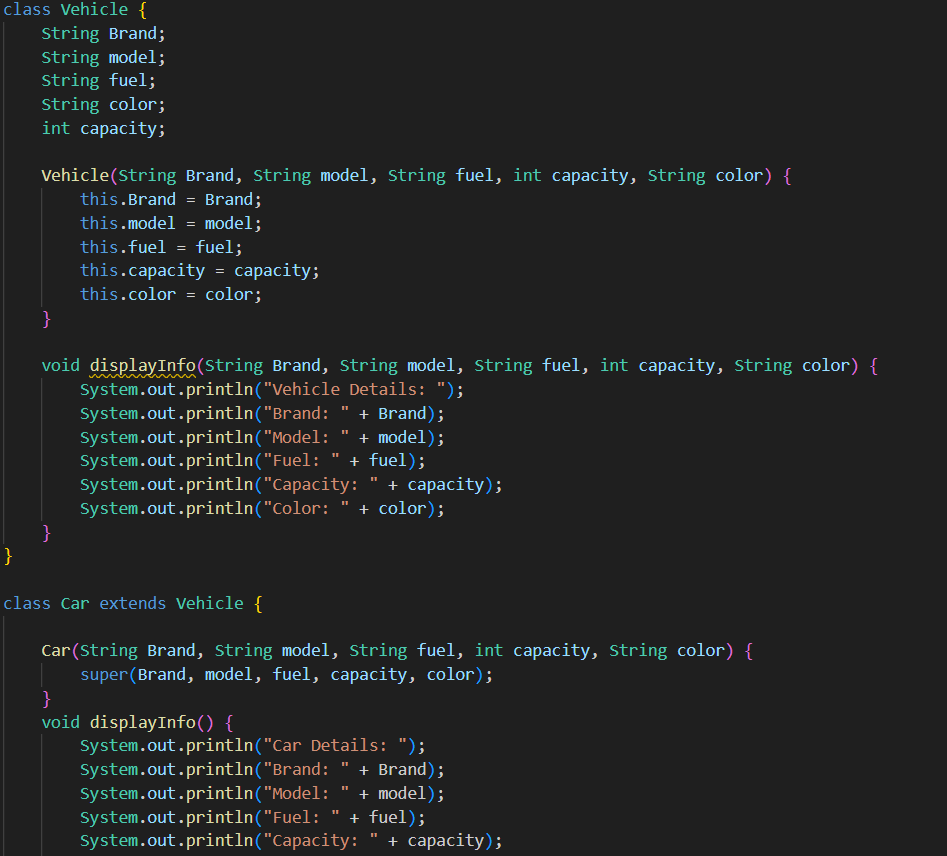
**WEEK-6**

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**

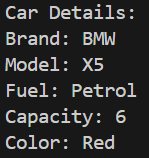
- Write your code in VS CODE and execute

**- Important Points:**

* + - * Understand the calling of a Constructor
      * Giving class name correctly
      * Give the parameters Correctly

**CODE:**  


**OUTPUT:**



**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%**

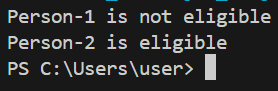
**- Write your code in VS CODE and execute**

**- Important Points:**

* + - * **Understand the calling of a Constructor**
      * **Giving class name correctly**
      * **Give the parameters Correctly**

**CODE:**

**  
  
  
  
  
  
OUTPUT:**

****

**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**

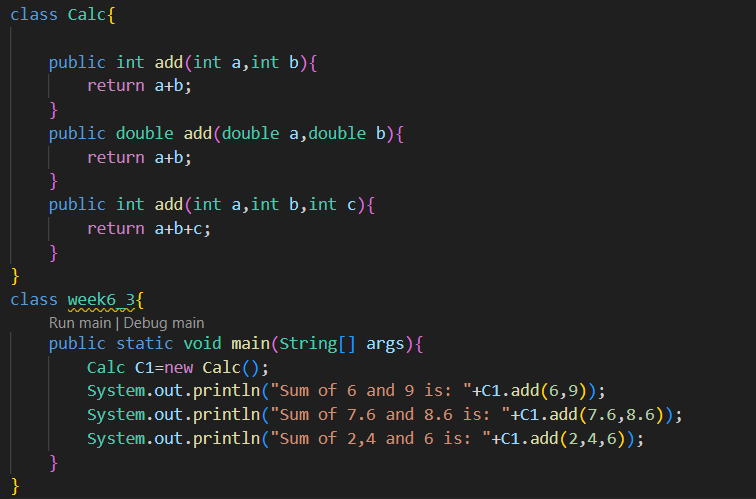
**iii) Add three integers**

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

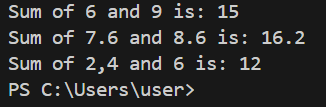
**- Important Points:**

* + - * **Understand the calling of a Constructor**
      * **Giving class name correctly**
      * **Give the parameters Correctly**

**CODE:**

****

**OUTPUT:**

****

**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 |

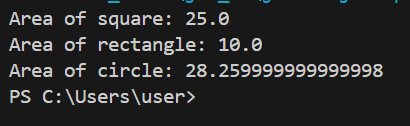
**4) Write a Java Program to create a shape class with a method calculateArea() that is overloaded for different shapes(e.g., Square, Rectangle ). Then create a subclass Circle that overrides the calculateArea() method for a circle.**

- Write your code in VS CODE and execute

**- Important Points:**

* + - * Understand the calling of a Constructor
      * Giving class name correctly
      * Give the parameters Correctly

**CODE:  
  
  
  
  
  
  
  
  
OUTPUT:**

****

**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) **Aim: 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.**

**CODE:**

 abstract class Animal {

public abstract void sound();

}

class Lion extends Animal {

@Override

public void sound() {

System.out.println("The lion roars.");

}

}

 class Tiger extends Animal {

@Override  
  
System.out.println("The tiger growls.");

}

}

public class q17 {

public static void main(String[] args) {

    System.out.println("Name Ajay;Roll No 24124;Sec CSe-B");

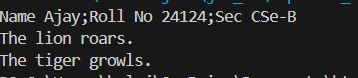
Lion lion = new Lion();

Tiger tiger = new Tiger();

lion.sound();

tiger.sound();

}

}  
  
**output:**  


**ERRORS:**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Error type** | **Reason for error** | **Rectification** |
| 1 | Syntax error | Abstract key word is missed before method | Abstract keyword is added |
| 2 | Logical error | Incorrect logic in subclass method | Corrected logic in subclass method |

**2)Aim :** 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  
  
**CODE:**abstract class area{

    abstract double   calculatevolume();

    abstract double  calculatesurfacearea();

}

class sphere extends area{

    double radius;

    double calculatevolume(){

        return 4/3\*3.14\*radius\*radius\*radius;

        }

        double calculatesurfacearea(){

            return 4\*3.14\*radius\*radius;

            }

    }

    class cube extends area{

        double side;

        double calculatevolume(){

            return side\*side\*side;

            }

            double calculatesurfacearea(){

                return 6\*side\*side;

                }

                }

class q18{

    public static void main(String[] args) {

        System.out.println("Name Ajay;Roll No 24124;Sec CSe-B");

        sphere s=new sphere()

        s.radius=5;

        System.out.println("volume of sphere is "+s.calculatevolume());

        System.out.println("surface area of sphere is "+s.calculatesurfacearea());

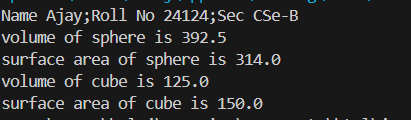
        cube c=new cube();

        c.side=5;

        System.out.println("volume of cube is "+c.calculatevolume());

        System.out.println("surface area of cube is "+c.calculatesurfacearea());

        }

    }  
  
**output:** **ERROR:**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Error type** | **Reason for error** | **Rectification** |
| 1 | Syntax error | package is missed before abstract class | Package is imported |
| 2 | Logical error | Incorrect logic in subclass method | Corrected logic in subclass method |

**3)Aim :** Write a java program using an abstract class to define a method for pattern printing Create an abstract class named pattern printer with 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 triangles of increasing numbers.

In the main() method, create Objects

Star Pattern Number pattern

\* 1

\*\* 1 2

\*\*\* 1 2 3

\*\*\*\* 1 2 3 4

\*\*\*\*\* 1 2 3 4 5

**CODE:**  
  
abstract class PatternPrinter {

    abstract void printPattern(int n);

    void displayTitle(String title) {

        System.out.println(title);

    }

}

class StarPattern extends PatternPrinter {

    void printPattern(int n) {

        for (int i = 1; i <= n; i++) {

            for (int j = 1; j <= i; j++) {

                System.out.print("\* ");

            }

            System.out.println();

        }

    }

}

class NumberPattern extends PatternPrinter {

    void printPattern(int n) {

        for (int i = 1; i <= n; i++) {

            for (int j = 1; j <= i; j++) {

                System.out.print(j + " ");

            }

            System.out.println();

        }

    }

}

 class pat {

    public static void main(String[] args) {

        StarPattern sp = new StarPattern();

        NumberPattern np = new NumberPattern();

        System.out.println("Name Ajay;Roll No 24124;Sec CSe-B");

        System.out.println("    ");

        sp.displayTitle("Star Pattern");

        sp.printPattern(5);

        System.out.println("    ");

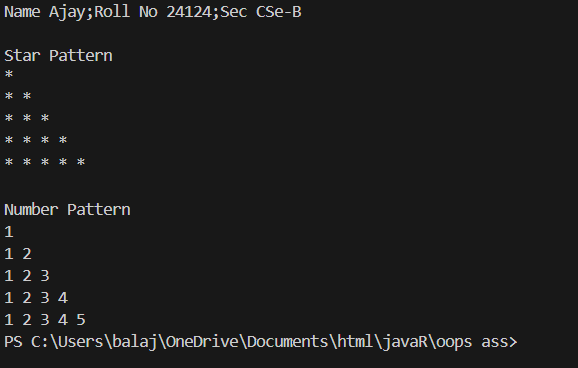
        np.displayTitle("Number Pattern");

        np.printPattern(5);

    }

}

**output:**

  
  
  
  
  
**ERROR:**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Error type** | **Reason for error** | **Rectification** |
| 1 | Syntax error | For loop increment condition is missed in subclass method | Increment condition is added in subclass method |
| 2 | Logical error | Incorrect logic in subclass method | Corrected logic in subclass method |

**WEEK-8**

1. **Aim :** 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.

**Code:**interface Shape {

double getPerimeter();

}

class Rectangle implements Shape {

private double length;

private double width;public Rectangle(double length, double width) {

this.length = length;

this.width = width;

}

@Override

public double getPerimeter() {

return 2 \* (length + width);

}

}

class Circle implements Shape {

private double radius;

public Circle(double radius) {

this.radius = radius;

}

@Override

public double getPerimeter() {

return 2 \* Math.PI \* radius;

}

}

class Triangle implements Shape {

private double side1;

private double side2;

private double side3;

public Triangle(double side1, double side2, double side3) {

this.side1 = side1;

this.side2 = side2;

this.side3 = side3;

}

@Override

public double getPerimeter() {

return side1 + side2 + side3;

}

}

public class q20 {

public static void main(String[] args) {

System.out.println("Name Ajay;Roll No 24124;Sec CSe-B");

Rectangle rectangle = new Rectangle(10, 5);Circle circle = new Circle(7);

Triangle triangle = new Triangle(6, 8, 10);

System.out.println("Rectangle Perimeter: " +

rectangle.getPerimeter());

System.out.println("Circle Perimeter: " +

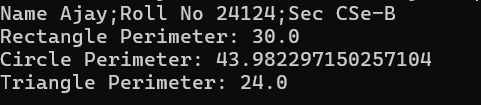
circle.getPerimeter());

System.out.println("Triangle Perimeter: " +

triangle.getPerimeter());

}

}

**Output:** **Error:**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Error type** | **Reason for error** | **Rectification** |
| 1 | Syntax error | Used extends keyword instead of implements keyword in inheritance | Implements key word is added in inheritance |

1. **Aim:** 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.

**Code:**interface Playable{

    public void play();

}

class Football implements Playable{

    @Override

    public void play(){

        System.out.println("Football is played between two teams of 11 players each, with the objective of scoring goals by kicking or heading the ball into the opposing team's net.");

    }

}

class Volleyball implements Playable{

    @Override

    public void play(){

        System.out.println("Volleyball is played between two teams of six players,where the teams try to score points by grounding a ball on the opponent's court, using a net and a maximum of three touches per side before sending the ball over. ");

    }

}

class Basketball implements Playable{

    @Override

    public void play(){

        System.out.println("Basketball is played between two teams of five players,where the teams try to score by shooting a ball through a hoop mounted high on a backboard, while preventing the opposing team from doing the same. ");

    }

}

public class play {

    public static void main(String[] args) {

           System.out.println("Name Ajay;Roll No 24124;Sec CSe-B");

        System.out.println();

        Football fb=new Football();

        fb.play();

        System.out.println();

        Volleyball vb=new Volleyball();

        vb.play();

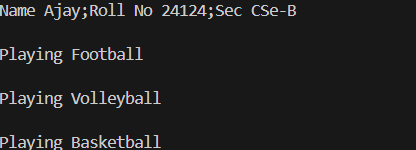
        System.out.println();

        Basketball bb=new Basketball();

        bb.play();

    }

}

**Output:** **ERROR:**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Error type** | **Reason for error** | **Rectification** |
| 1 | Syntax error | Error in calling play method in football class because object is not created for football class | Object created for football class |

**3) Aim:** write a java program to implement a login system using interfaces.

**Code:**interface LoginSystem {

    boolean Login(String ID, int pass);

}

class CollegePortal implements LoginSystem {

    public boolean Login(String ID, int pass) {

        if ((ID=="TEJA") && (pass==24138)){

            System.out.println("Login Successful..!");

            return true;

        }else {

            System.out.println("Invalid ID or Password");

            return false;

        }

    }

}

class LoginPortal {

    public static void main(String[] args) {

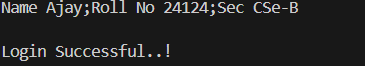
        CollegePortal CP = new CollegePortal();

        System.out.println("Name Ajay;Roll No 24124;Sec CSe-B");

        System.out.println("    ");

    }

}  
  
  
**output:**



**ERROR:**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Error type** | **Reason for error** | **Rectification** |
| 1 | Syntax error | Error in If statement condition | If statement condition is corrected |