**LAB MANUAL**



**ROLLNO:AV.SC.U4CSE24244**

**NAME: P. Pranav**

**SECTION: CSE-C**

**WEEK-1:**

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

printing student details*.*

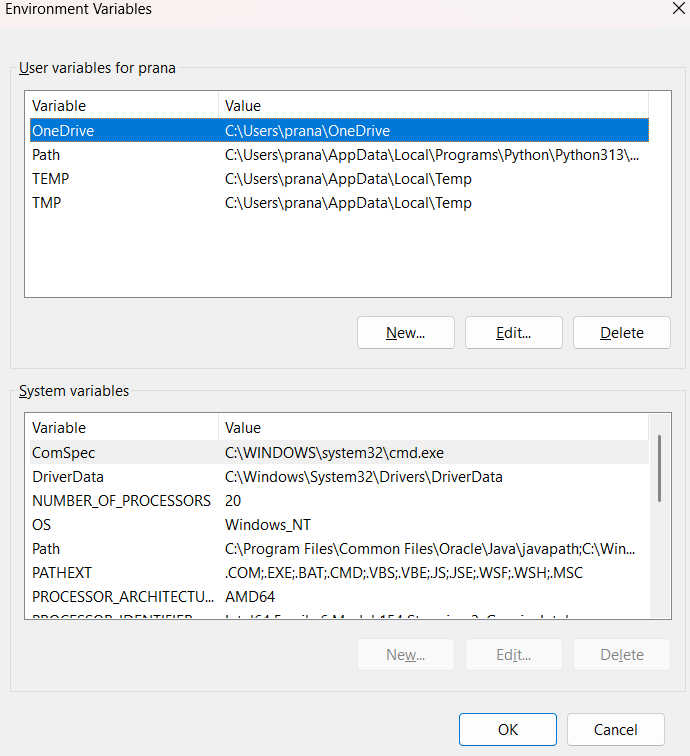
**Step-1:** Download JDK-21 from oracle website

**

**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

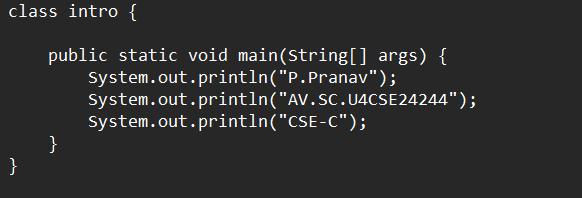
Javac filename.java ->compiling.

Java filename.java ->display

**PROGRAM-1:**

**Aim:** Write a program in java for displaying

student details.



**Output:**

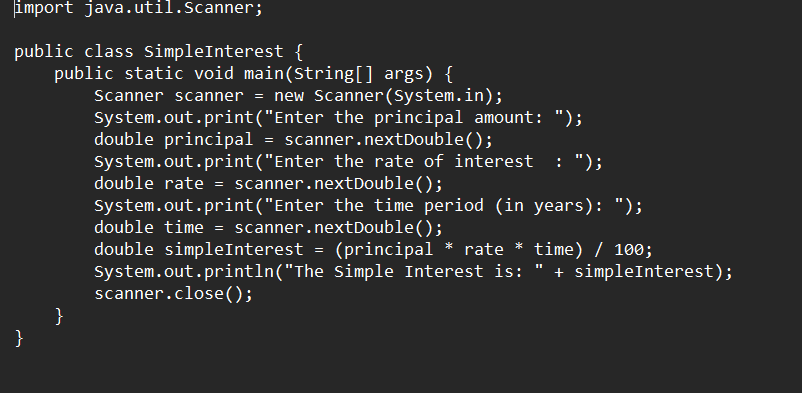
******

|  |  |  |
| --- | --- | --- |
| **Code \** | **Error** | **Rectification** |
| System.out.println  ("Class: CSE-C") | Semi colon(;) is  missing at the  end. | Add a semi colon(;) at the  end.  System.out.println("Class:  CSE-C"); |

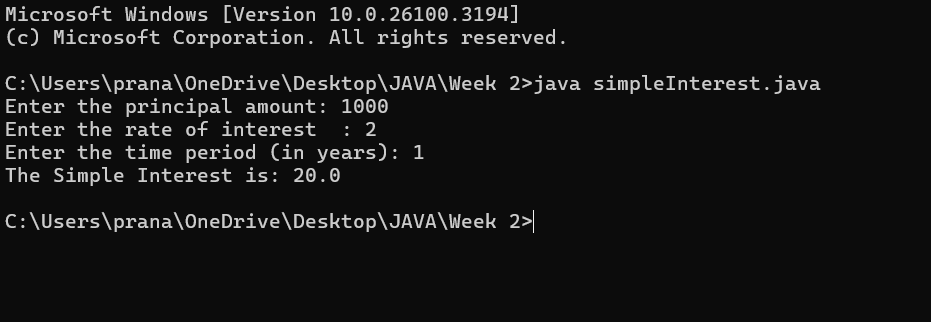
**WEEK-2:**

**PROGRAM-1:**

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

**

**Output:**

******

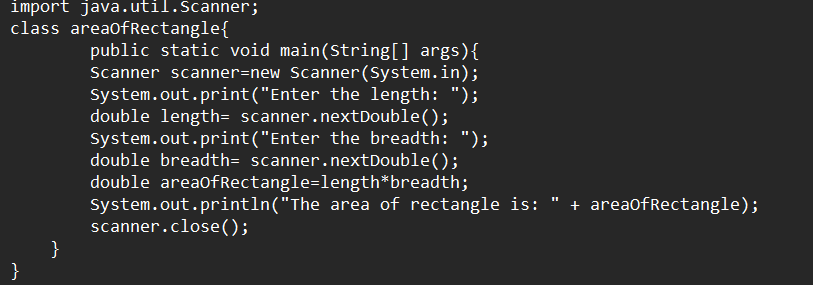
**ERROR TABLE*:***

|  |  |
| --- | --- |
| **Code Error** | **Code rectification** |
| 1.Giving space between next and Double.  2.Not using the closing semi colon. | 1.Should not give space between next and Double.  2.We must put semi colon after each line when required. |

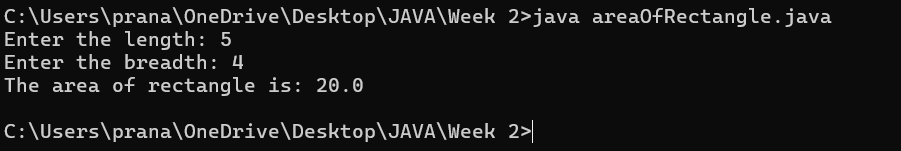
**PROGRAM-2:**

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

rectangle.

**

**Output:**

******

IMPORTANT POINTS:

1. Area of a rectangle is area = l\*b, where

L = length of a side of the rectangle,

B= breadth of a side of the rectangle.

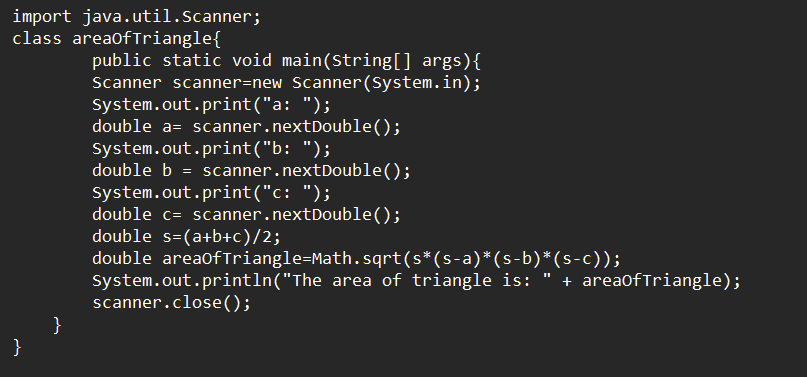
1. Here, we must be sure that all the expressions/conditions inside for the for loop must be given correctly.

**ERROR TABLE:**

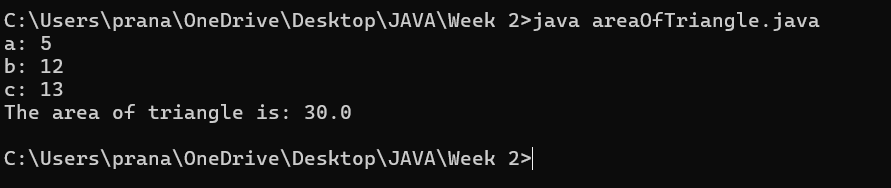
|  |  |
| --- | --- |
| **Code Error** | **Code rectification** |
| 1.While using for iteration, not giving the conditions correctly.  2.Declaring the data type. | 1.We should give iterative statements correctly.  2.We should give the data type. |

**PROGRAM-3:**

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



**Output:**

******

IMPORTANT POINTS:

1. Here, we’re finding the area of a triangle using heron’s formula.
2. Heron’s formula for finding a triangle is:

S = (a +b +c)/2

Where S is the semi-perimeter of the triangle.

Now the area formula is:

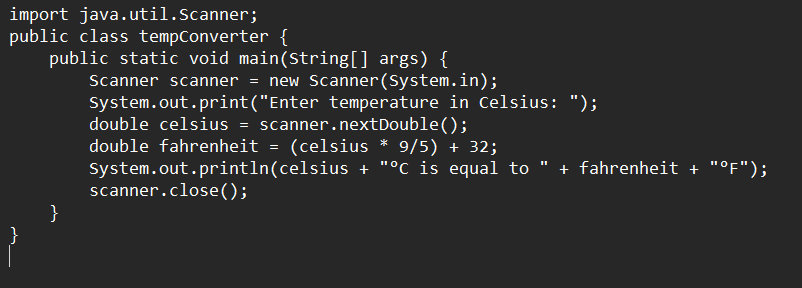
Area = sqrt(s\*(s-a)\*(s-b)\*(s-c)).

ERROR TABLE:

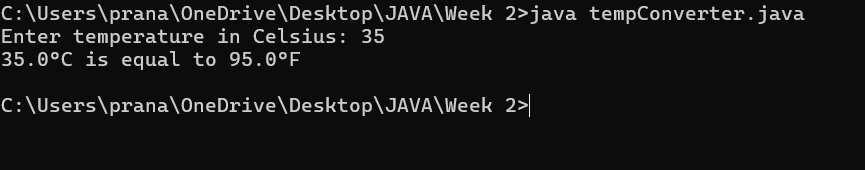
|  |  |
| --- | --- |
| **Code Error** | Code rectification |
| 1.While printing the variable not giving + sign.  2.Declaring the data type. | 1.We should give correct indentation.  2.We should give the data type. |

**PROGRAM-4(a):**

**Aim:** Write a program in java for converting temperature from Celsius to Fahrenheit.

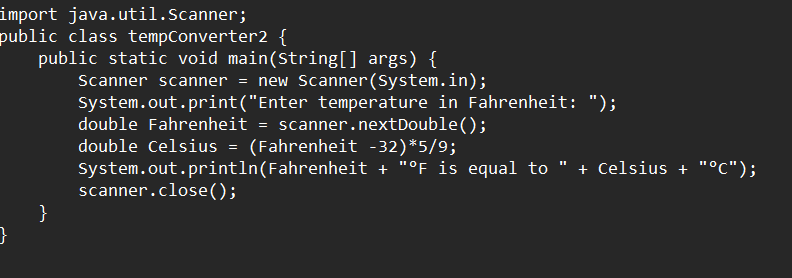


OUTPUT:

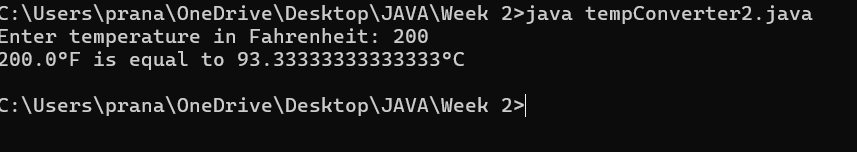


**PROGRAM-4(b):**

**Aim:** Write a program in java for converting temperature from Fahrenheit to Celsius.

******

**Output:**

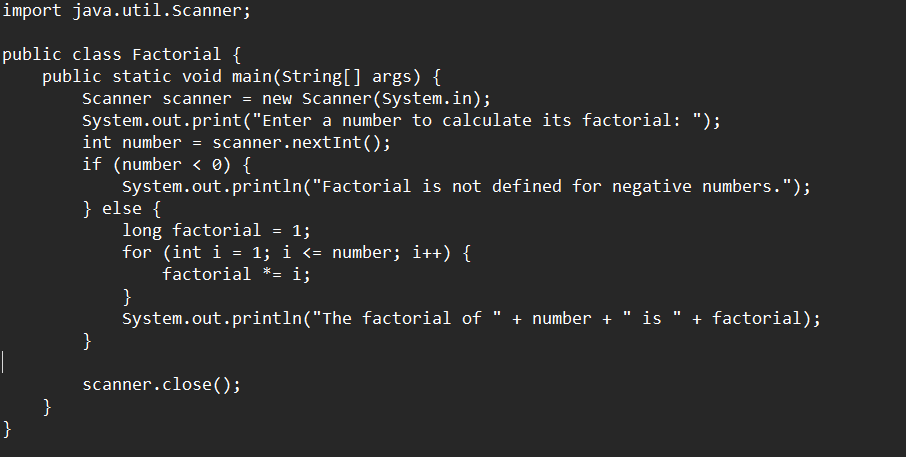
******

**ERROR TABLE:**

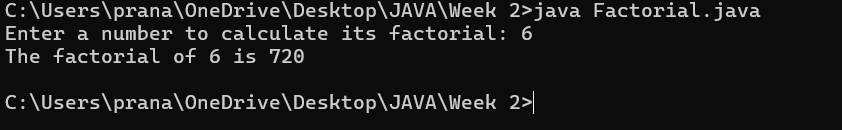
|  |  |
| --- | --- |
| **Code Error** | **Code rectification** |
| 1.While printing the variable not giving + sign.  2.Not closing the scanner. | 1.We should give correct indentation.  2.Closing the scanner is must. |

**PROGRAM-5:**

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

******

OUTPUT:

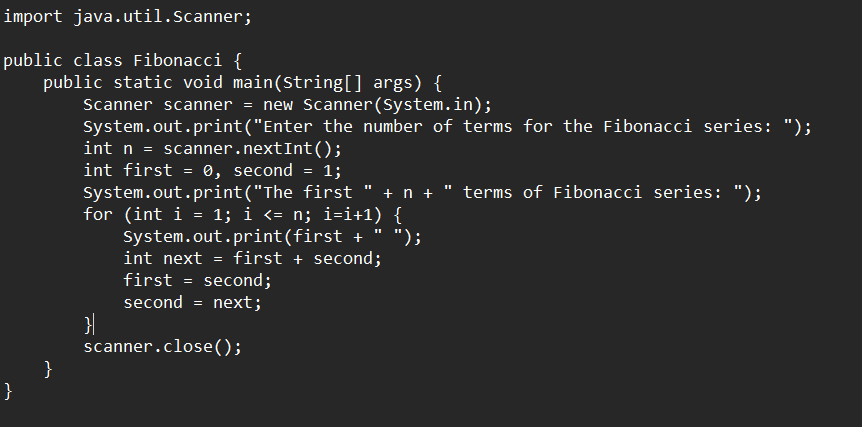


ERROR TABLE:

|  |  |
| --- | --- |
| **Code Error** | **Code rectification** |
| 1.While using for iteration, not giving the conditions correctly. | 1.We should give iterative statements correctly. |

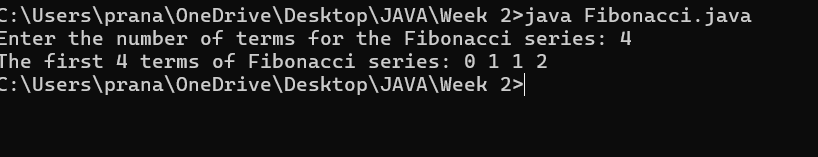
**PROGRAM-6:**

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



IMPORTANT POINTS:

1. In the Fibonacci sequence, the sum value is given to the second variable, and the value of the second variable is given to the first variable.
2. This process is repeated a certain number of times until the conditions are met.

OUTPUT: 

ERROR TABLE:

|  |  |
| --- | --- |
| **Code Error** | **Code rectification** |
| 1.Giving space between next and Double.  2.Not putting flower brackets in the end. | 1.Should not give space between next and Double.  2.We must put giving flower brackets in the end. |

**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:**

public class Car {

private String car\_color;

private String car\_brand;

private String fuel\_type;

private String mileage;

public void start() {

System.out.println("car is started");

}

public void stop() {

System.out.println("car is stopped");

}

public void service() {

System.out.println("car is for service");

}

public static void main(String args[]) {

Car car = new Car();

car.car\_color = "white";

car.car\_brand = "audi";

car.fuel\_type = "petrol";

car.mileage = "20";

car.start();

System.out.println("car\_color: " + car.car\_color + " car\_brand: " + car.car\_brand + " fuel\_type: " + car.fuel\_type + " mileage: " + car.mileage);

Car car1 = new Car();

. car1.car\_color = "white";

car1.car\_brand = "audi";

car1.fuel\_type = "petrol";

car1.mileage = "20";

car1.stop();

System.out.println("car\_color: " + car1.car\_color + " car\_brand: " + car1.car\_brand + " fuel\_type: " + car1.fuel\_type + " mileage: " + car1.mileage);

Car car2 = new Car();

car2.car\_color = "white";

car2.car\_brand = "audi";

car2.fuel\_type = "petrol";

car2.mileage = "20";

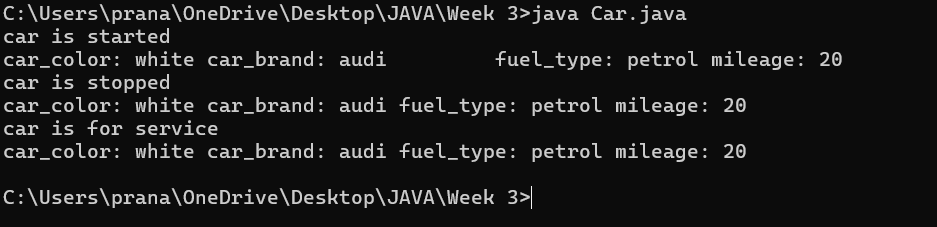
car2.service();

System.out.println("car\_color: " + car2.car\_color + " car\_brand: " + car2.car\_brand + " fuel\_type: " + car2.fuel\_type + " mileage: " + car2.mileage);

}

}

**OUTPUT:**

****

**Error table:**

|  |  |  |
| --- | --- | --- |
| S.no | error | Rectification |
| 1 | Missing ‘;‘ | ‘;‘ added |
| 2 | Mispelled Variable call | Rectified with  Correct variable name |
| 3 | Uppercase and lowercase | rectified |

IMPORTANT POINTS:

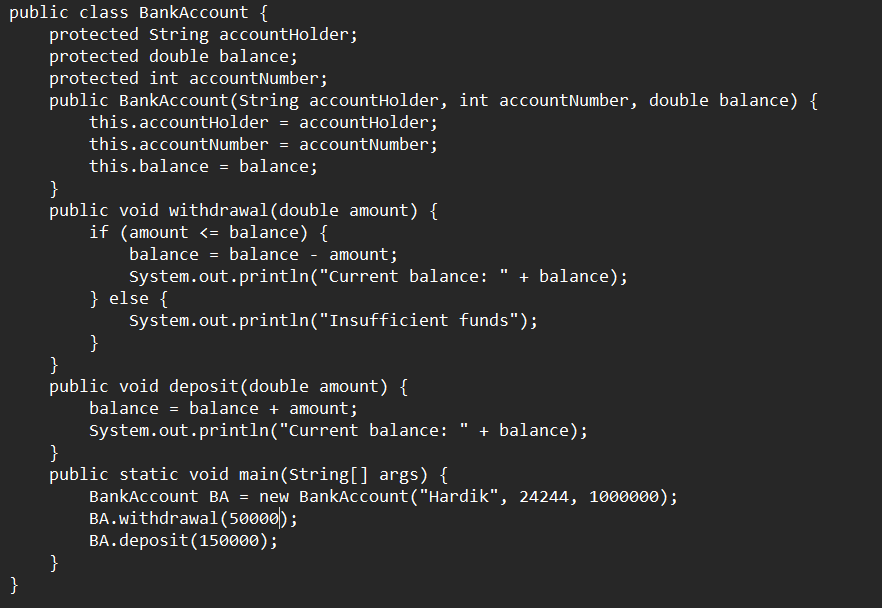
1. Before calling the function we should write the method properly.
2. Here, the “public void start( )” indicates that we are writing a method to call the function.
3. When we call a certain method, the process inside it will be printed as an output of the code.
4. Here the details inside the function are called objects, we can give any objects

**Class diagram:**

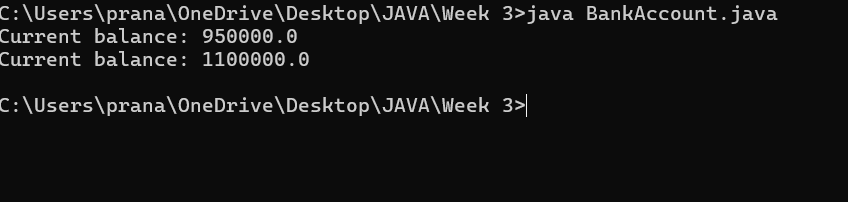
|  |
| --- |
| **car**  **----------------------**-  -car\_color:string  -car\_brand:string  -fuel\_type:string  -milage:double  ----------------------  +start():void  +stop():void  +service():void |

**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

****

**OUTPUT:**

****

IMPORTANT POINTS:

1. The condition inside the if statement must be correct.
2. It explains that if the withdrawal money is less than the money in the bank account, then we can withdraw the amount.

**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 and Year of Publication .It should also contain a constructor with parameter which initializes Title ,Author and Year of publication .Create a method which displays the details of the book .Display the details of two books.

**CODE:**

class book{

public String title;

public String author;

public String year\_of\_publication;

public void book(){

this.title=title;

this.author=author;

this.year\_of\_publication=year\_of\_publication;

}

public static void main(String[] args){

book book1=new book();

book book2=new book();

book1.book();

book1.title="Sherlock Holmes ";

book1.author="Arthur Conan Doyle";

book1.year\_of\_publication="1887";

book2.book();

book2.title="Harry Potter";

book2.author="J.K. Rowling";

book2.year\_of\_publication="1997";

System.out.println("Book-1");

System.out.println("Title :" +book1.title);

System.out.println("Author :" +book1.author);

System.out.println("Year of publication :" +book1.year\_of\_publication);

System.out.println("Book-2");

System.out.println("Title :" +book2.title);

System.out.println("Author :" +book2.author);

System.out.println("Year of publication :" +book2.year\_of\_publication);

}

}

**OUTPUT:**

****

**Class Diagram:**

|  |
| --- |
| **Book** |
| * Title: String * Author: String * Year of publication: int |
| + Book(title: String,  Author: String;  Year of publication: int  + displayDetails( ): void |

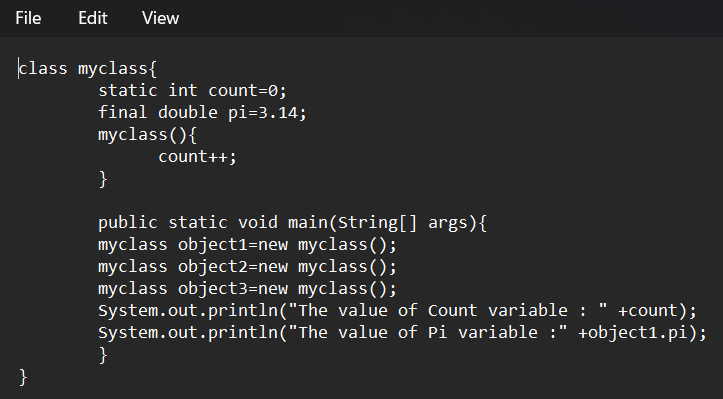
IMPORTANT POINTS:

1. While defining two classes for a code, we must be sure that we save both the classes in separate files.
2. While defining a method we should also define a function to call that method.

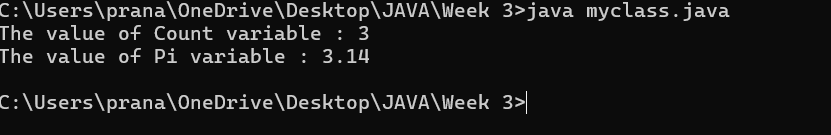
**Error Table:**

|  |  |  |
| --- | --- | --- |
| **S.No** | **Error** | **Rectification** |
| **1.** | **Missing “;” after calling method.** | **Added “;”** |

**PROGRAM-2:**

**Aim:** Create a java program with class named myclass with a static variable count of int type initialized to 0 and a constant variable “Pi” of type double initialized to 3.14 has attributes of that class .Now define a constructor for “myclass” that incerements the count variable each time an object of myclass is created finally print final values count and Pi variables .Create three objects

**OUTPUT:**

****

**Class Diagram:**

|  |
| --- |
| Myclass |
| - Count: int  - Pi: double |
| + myclass( )  + main(args: String[]): void |

IMPORTANT POINTS:

1. We must declare the initial value of the variable before declaring the final one.
2. Here the main objective is to increase the count according to the number of objects we make, i.e the count increases when the no.of objects are increasing.

Error Table:

|  |  |  |
| --- | --- | --- |
| S.No | Error | Rectification |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| 1. | Not typing “}“ at the end of the code. | Added “}”. |

**WEEK-5:**

**PROGRAM-1**

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

**Class Diagram:**

**| Calculator |**

**+----------------+**

**▲**

**│**

**+----------------+**

**| add |**

**+----------------+**

**| +addition(a,b) |**

**+----------------+**

**▲**

**│**

**+----------------+**

**| sub |**

**+----------------+**

**| +subtraction(a,b) |**

**+----------------+**

**▲**

**│**

**+----------------+**

**| mul |**

**+----------------+**

**| +multiplication(a,b) |**

**+----------------+**

**▲**

**│**

**+----------------+**

**| div |**

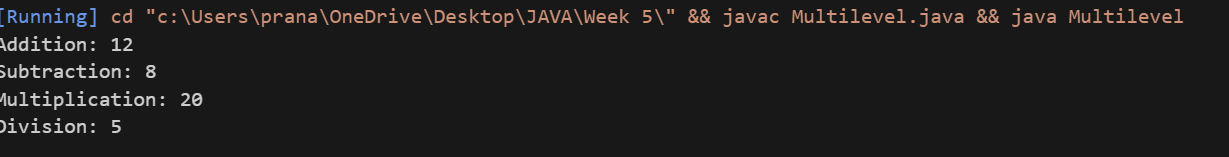
**+----------------+**

**| +division(a,b) |**

**+----------------+**

**Code:** ****

Output:

****

**ERROR TABLE :**

|  |  |  |
| --- | --- | --- |
| **s.no** | **Error** | **Error rectification** |
| **1** | **Syntax error on token ")”** | **‘{’ should be mentioned** |
| **2** | **calculator cannot be resolved to a type** | **‘c’ should be capital.** |

**IMPORTATNT POINTS:**

**We use multilevel inheritance in this**

**Each subclass inherits methods from its parent class, gaining access to addition, subtraction,**

**Multiplication and division.**

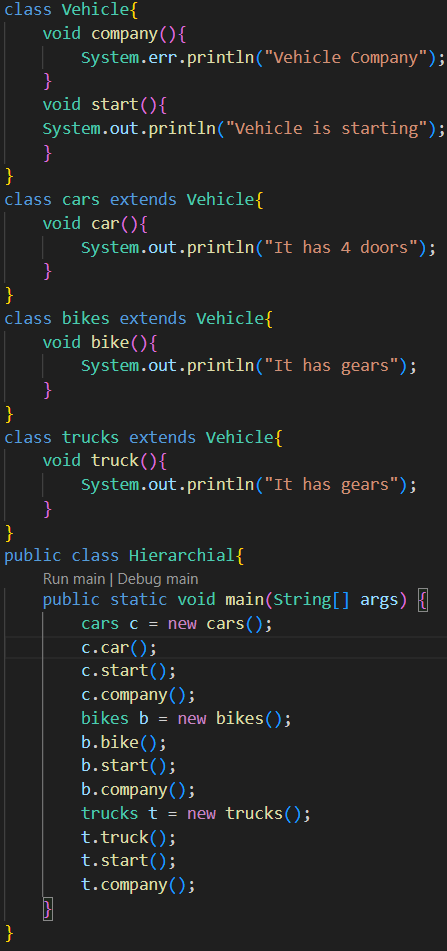
**PROGRAM-2**

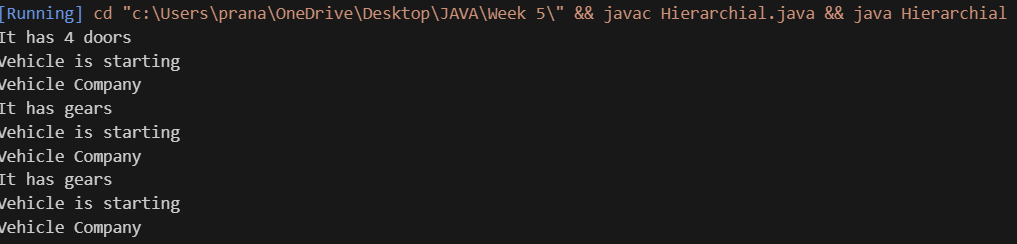
**AIM:**

**A vehicle rental company wants to develop a system that maintains information About different types of vehicles available for rent the company rents out cars bikes And they need a program to store details about each vehicle such as brand & speed.**

**Class Diagram:**

Code:



Output: 

IMPORTATNT POINTS:

The code demonstrates inheritance using a parent class Vehicle and three child classes: Car, Bike, and

Truck.

Each subclass has its own constructor that extends the parent class constructor

ERROR TABLE :

|  |  |  |
| --- | --- | --- |
| s.no | Error | Error rectification |
| 1 | Syntax error on token ")” | ‘{’ should be mentioned |
| 2 | Heirarchial cannot be resolved to a type | ‘h’ should be capital |

WEEK-6

Program 1

Aim: 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.

Class Diagram:

+------------------+

| Vehicle |

+------------------+

| |

+------------------+

▲

│

+------------------+

| Car |

+------------------+

| +displayInfo() |

+------------------+

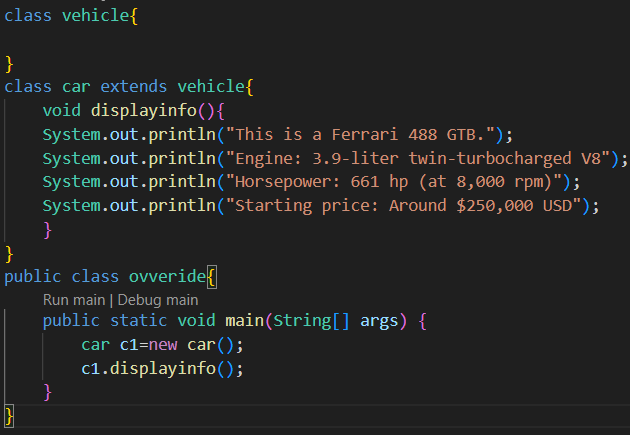
+------------------+

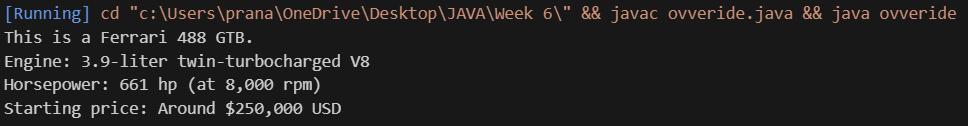
| Override |

+------------------+

| +main(String[]) |

+------------------+

Code: 

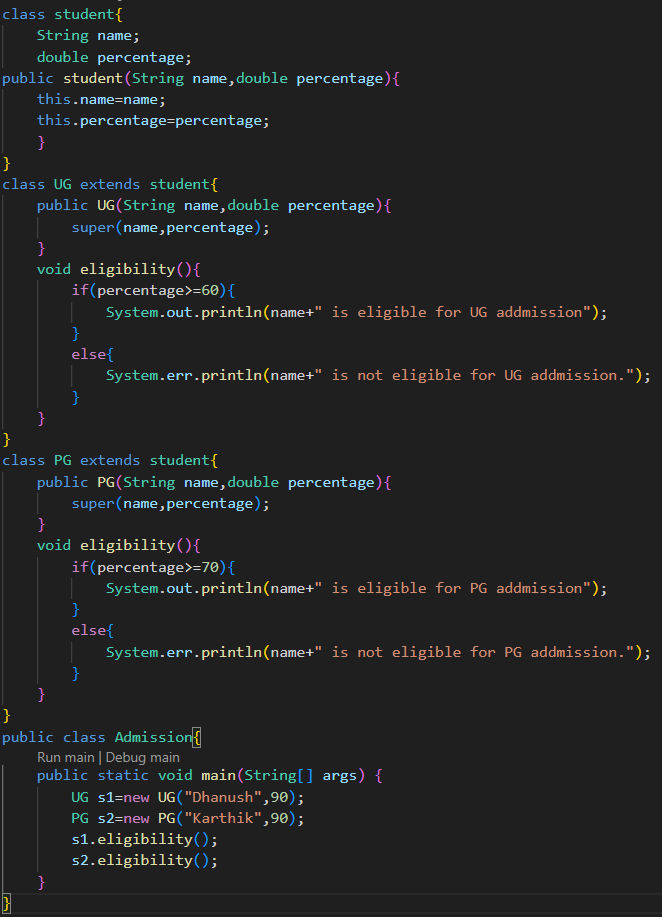
Output: 

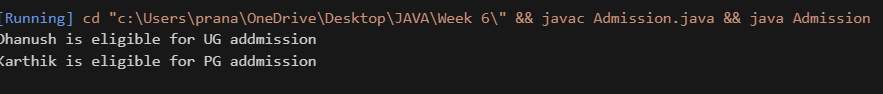
Program 2

Aim: A college is developing an automated admission system that verifies students eligibility for undergraduate(UG) and post graduate(PG) programs Each program has different eligibility criteria based on theh student’s percentage in the previous qualification.

UG admission require a minimum of 60%.

PG admission require a minimum of 70%.

Code: 

Output: 

Program 3

Aim: Create a calculator class with overloaded methods to perform addition.

1. Add 2 integers.
2. Add 2 doubles.
3. Add 3 integers.

Class Diagram:

+-------------------+

| Student |

+-------------------+

| - name: String |

| - percentage: double |

+-------------------+

| + Student(name, percentage) |

+-------------------+

▲

│

├───────────────┐

│ │

------------------- -------------------

| UG | | PG |

|------------------- -------------------

| + UG(name, percentage) | | + PG(name, percentage) |

| + eligibility() | | + eligibility() |

+------------------- -------------------

-------------------

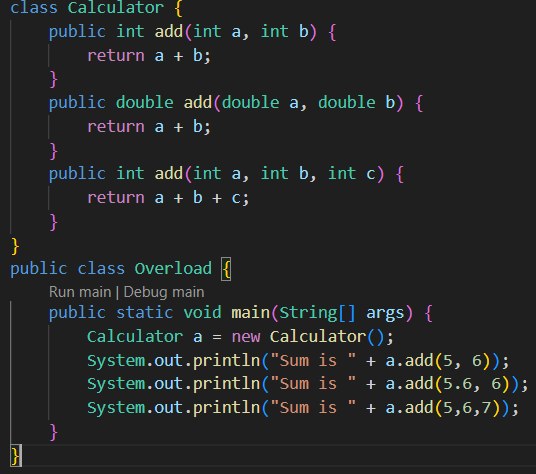
| Admission |

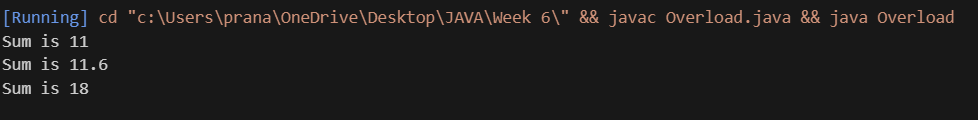
-------------------

| + main(String[]) |

-------------------

Code:

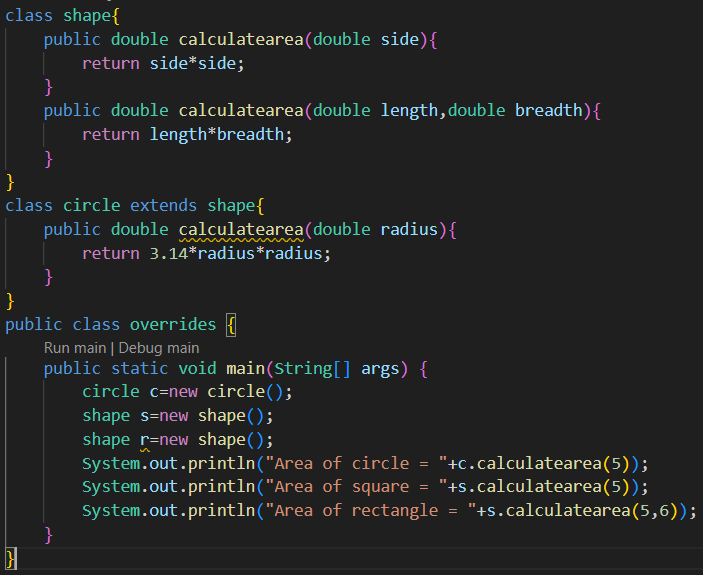


Output: 

Program 4

AIM: Create a shape class with a method calculateArea() that is overloaded for different shapes. Then create a subclass circle that overrides the calculator method for a circle.

Code:



Output: 