23CSE111

**OBJECT ORIENTED PROGRAMMING**

**LAB REPORT**



**Department of Computer Science Engineering**

**Amrita School of Computing**

**Amrita Vishwa Vidyapeetham, Amaravati Campus**

**Name: S.Mohana Maheswari**

**Verified By Roll No: 24314**

**WEEK-1**

**1)How to install the java and download java software ?**

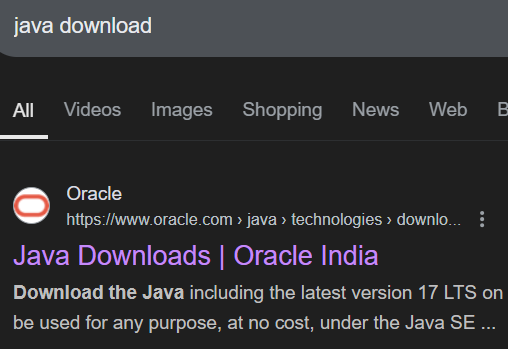
**Aim: To Explain the process of Installing JDK (Java Development Kit)**

**Procedure:**

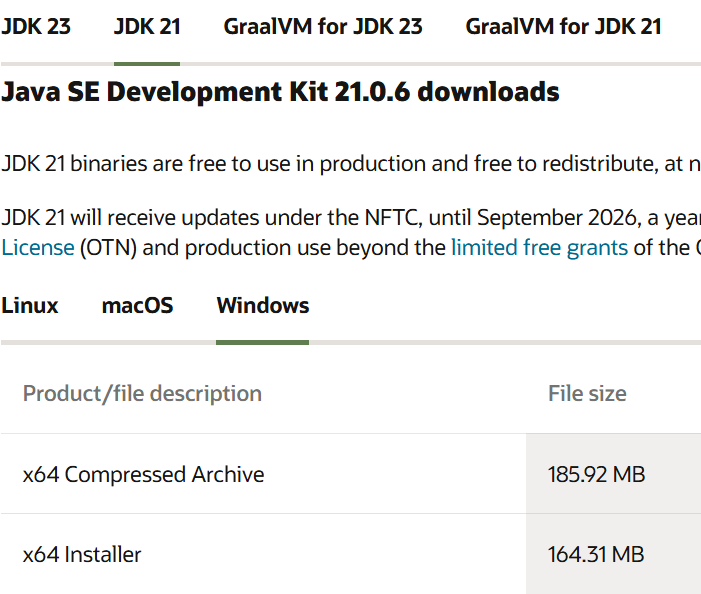
**Installing of JDK (Java Development Kit):**

**1. First go to google and search java Download**

**2 . Go to the Oracle JDK download page in your web browser**

****

**3.** **click on JDK-21 version which is Long term support (LTS) version.**



4. Click on the download link for your operating system (Windows, macOS, or Linux).

5.Next move on to installing JDK

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

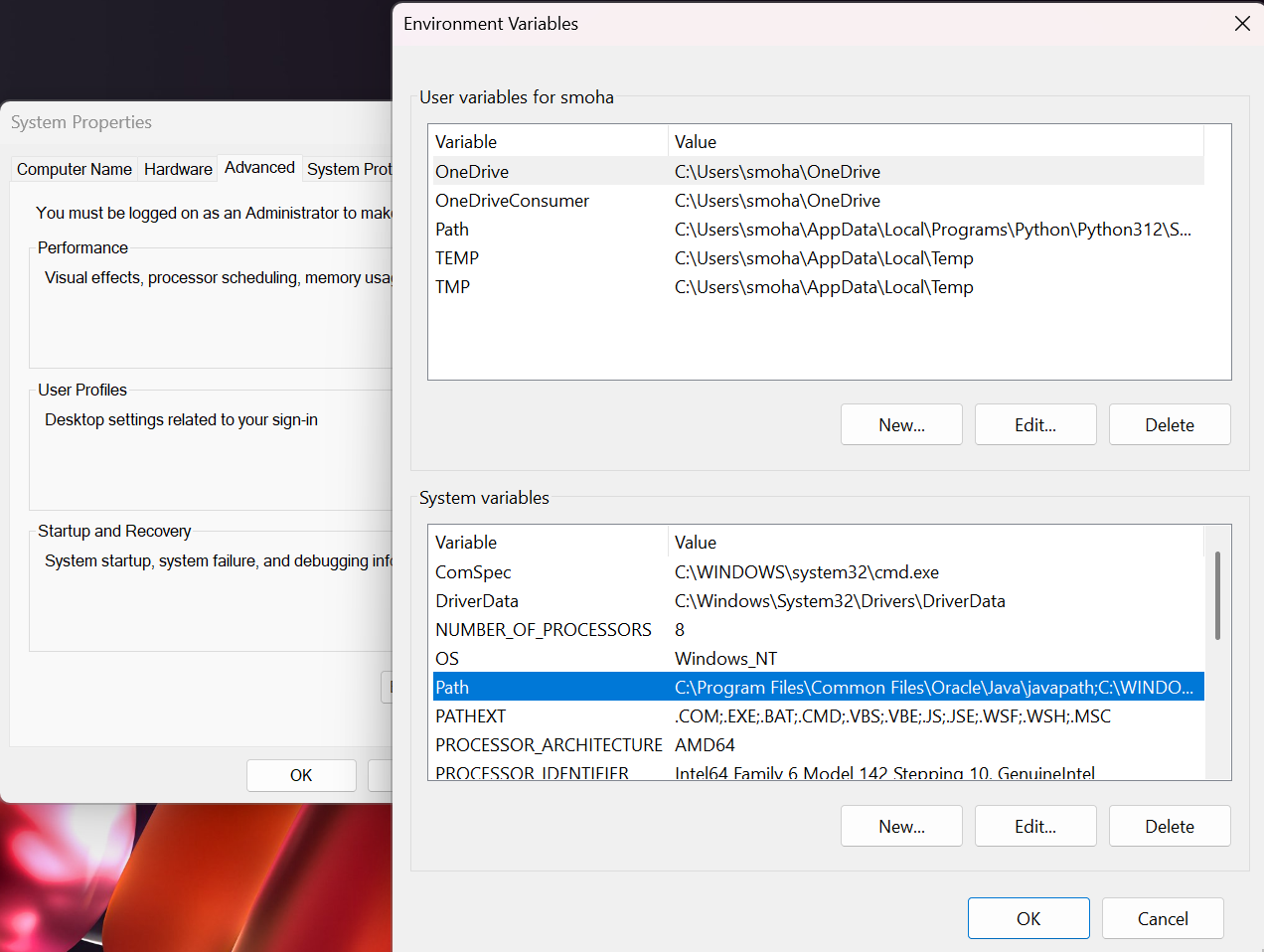
7.Follow this Navagation ,at last we get an link, then copy past that link in path





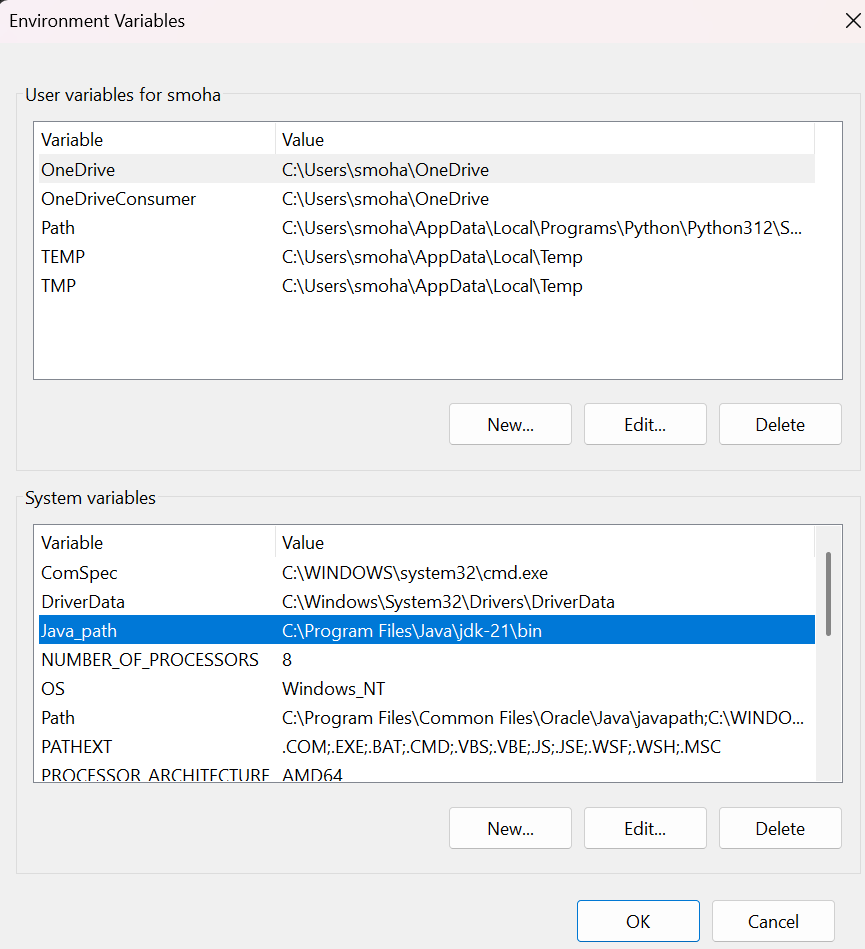
8.Search for Environment variables in search option which is below the pc. Then click on **Environment Variables**.

9. this may dispay on our screen after we click on environment variables



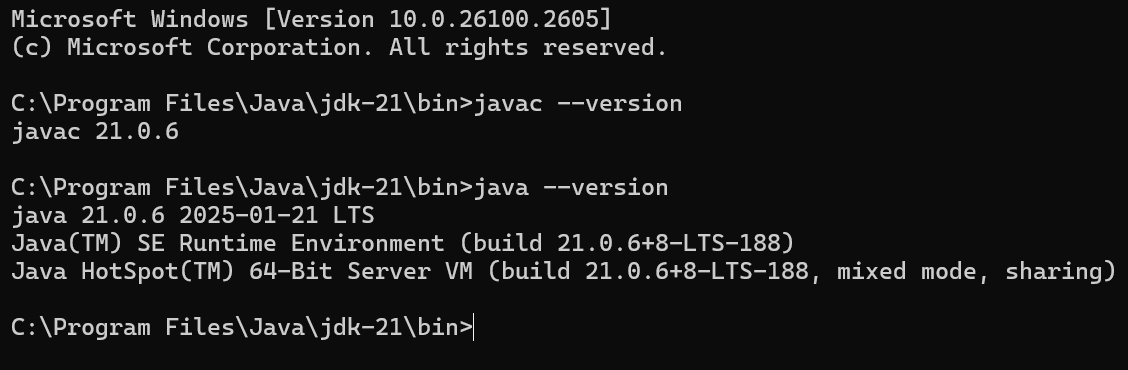
10. To set up the Click **New** under **System Variables**:

* **Set Variable name as:** java\_path
* **Variable value:** The folder address where JDK is installed (like C:\Program Files\Java\jdk-21\bin)
* After this click on ok



11. **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.



**2) Write a Java Program to print the message “Welcome to Java programming”.**

**AIM:**  To write a Java Program to print the message “Welcome to Java programming”.

**CODE:**

//java program that prints a welcome message

//Define a class named Example1

class Example1{

public static void main(String[]args){

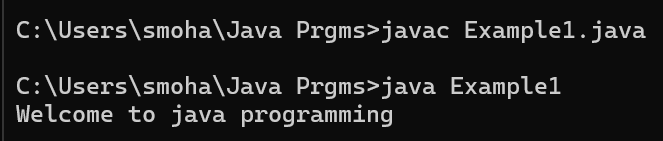
//print "welcome to java programming"

System.out.println("Welcome to java programming");

}

}

**OUTPUT:**

**Write a java program that prints Name , Roll no and Section of a student**

**AIM:** To Write a java program that prints Name , Roll no and Section of a student

**CODE:**

//java program that prints a Details of student

//Define a class named Example2

class Example2{

public static void main(String[]args){

//print Name, Roll no and Section of student

System.out.println("Name:Mohana Maheswari");

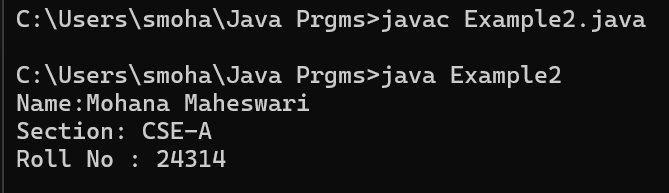
System.out.println("Section: CSE-A");

System.out.println("Roll No : 24314");

}

}

**OUTPUT:**

****

**WEEK-2**

**3.Write a java program to calculate area of rectangle:**

**Code:**

import java.util.Scanner;

class rectangle {

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

// Getting float input

System.out.print("Enter a : ");

float a = input.nextFloat();

System.out.println(" entered a value = " + a);

System.out.print("Enter b: ");

float b = input.nextFloat();

System.out.println(" entered b value = " + b);

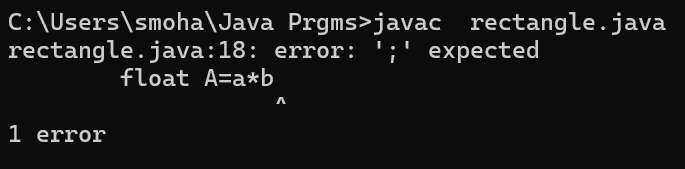
float A=a\*b

System.out.println("Area of Rectangle is "+A);

}

}

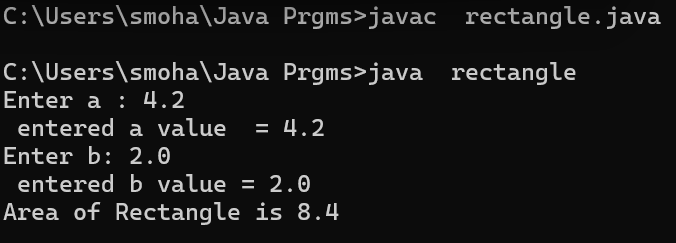
**Output:**

****

**Error: Here there is an one error that is semicolon( ; ) is not present at the end of the expression.**

|  |  |  |
| --- | --- | --- |
| **S.no** | **ERROR MESSAGE** | **ERROR RECTIFICTION** |
| **1** | **Error:expected** | **Keep the semicolon at the end.** |

**Output: After the error is rectified with semicolon(;)**



**Important points:**

**Class Definition**: The code defines a class named rectangle.

**Main Method**: The main method is the entry point for the program. It contains the logic for user input and calculations.

**User Input**: The code uses a Scanner object to get user input. It prompts the user to enter two float values (a and b).

**Questions:**

**What is the purpose of the main method in Java?**

* The main method in Java serves a crucial purpose as the entry point of any standalone Java application. When the Java Virtual Machine (JVM) starts running a program, it looks for the main method to begin execution.

**2.Write a java program to convert temperature from fahenhit to celsius:**

**Formula:(**F-32)\*5/9

**Code:**

import java.util.Scanner;

public class FahrenheitToCelsius {

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

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

double fahrenheit = input.nextDouble();

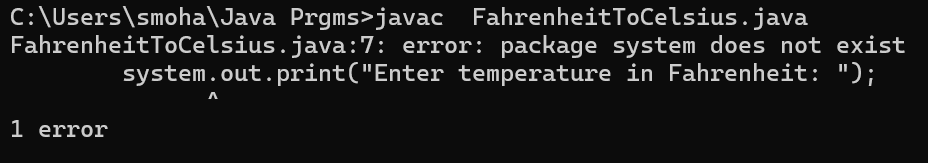
double celsius = (fahrenheit - 32 )\* 5 / 9;

System.out.println("Temperature in Celsius: " + celsius);

input.close();

}

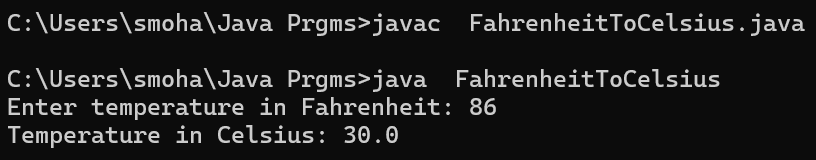
}



Error:Here there is one error that is “s” in System is small letter, but it should be always Capital letter “S”

|  |  |  |
| --- | --- | --- |
| **S.no** | **ERROR MESSAGE** | **ERROR RECTIFICTION** |
|  | **Package system does not exist.** | **Keep Catical S instead of small s.** |

Output: **After the error is rectified**



**Important points:**

**Import Statement**:

* The import java.util.Scanner; statement imports the Scanner class from the java.util package. The Scanner class is used to read input from various input sources, including user input from the console.

**Class Declaration**:

* The class is named FahrenheitToCelsius, and it contains a main method.

**Main Method**:

* The main method is the entry point of the program. It is where the program execution starts.

**Question:**

**How is the Scanner class used in this code?**

* **Answer**: The Scanner class is used to get user input from the console. An instance of Scanner is created to read the input temperature in Fahrenheit

**2.Write a java program to convert temperature from Celsius to Fahrenheit:**

**Formula:**

**Code:**

import java.util.Scanner;

public class CelsiusToFahrenheit {

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

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

double Celsius = input.nextDouble();

double Fahrenheit = (Celsius\*9/5)+32;

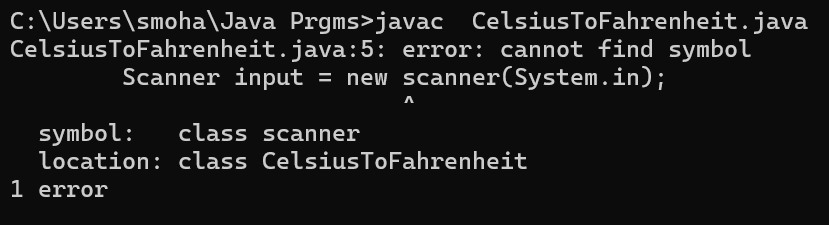
System.out.println("Temperature in Fahrenheit : " + Fahrenheit );

input.close();

}

}

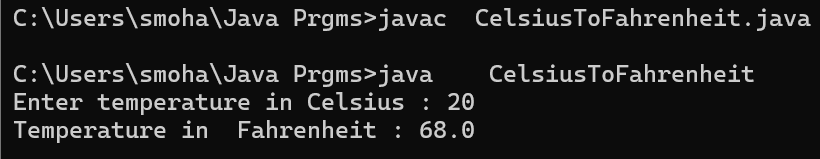
Output:



Error: here the error is ” S” in Scanner is small letter(s),but it should be in capital letter(S).

|  |  |  |
| --- | --- | --- |
| **S.no** | **ERROR MESSAGE** | **ERROR RECTIFICTION** |
| **1** | Error:cannot finf symbol. | S” in Scanner is small letter(s),but it should be in capital letter(S). |

Output: **After the error is rectified**



**Important points:**

**Import Statement**:

* The import java.util.Scanner; statement imports the Scanner class from the java.util package. The Scanner class is used to read input from various input sources, including user input from the console.

**Class Declaration**:

* The class is named CelsiusFahrenheitTo, and it contains a main method.

**Main Method**:

* The main method is the entry point of the program. It is where the program execution starts.

Question:

Why is it important to close the Scanner object in this code?

* **Answer**: Closing the Scanner object releases the resources associated with it, preventing resource leaks. This is done using the input.close() method.

**4.Write the java program to calculate the simple interest:**

**Formula:P\*T\*R/100**

**Code:**

import java.util.Scanner;

class simpleinterest {

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

System.out.print("Enter P : ");

float P = input.nextFloat();

s System.out.println(" entered P value = " + P);

System.out.print("Enter T: ");

float T = input.nextFloat();

System.out.println(" entered T value = " + T);

System.out.print("Enter R: ");

float R = input.nextFloat();

System.out.println(" entered R value = " + R);

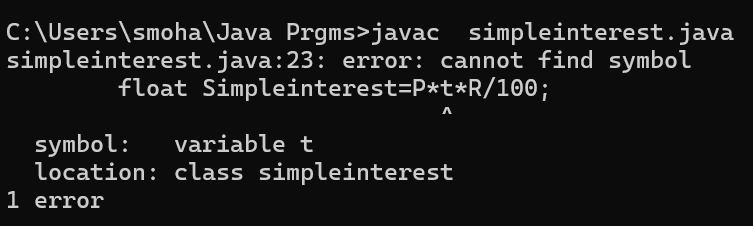
float Simpleinterest=P\*T\*R/100;

System.out.println("Simple interest is:"+Simpleinterest);

}

}

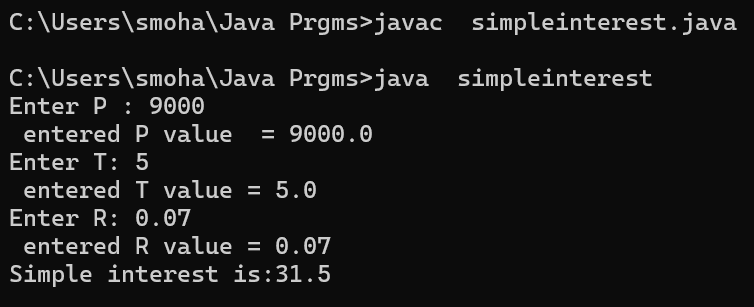
Output:



Error: Here the error is variable “t” is not found ,instead we have to keep “T”.

|  |  |  |
| --- | --- | --- |
| **S.no** | **ERROR MESSAGE** | **ERROR RECTIFICTION** |
| **1** | **Error:cannot find symbol** | Variable “t” is not found ,instead we have to keep “T”. |

Output: **After the error is rectified**



**Important points:**

**Class Definition**: The code defines a class named simpleinterest.

**Main Method**: The main method is the entry point for the program, containing the logic for user input and calculations.

**User Input**: The code uses a Scanner object to get user input. It prompts the user to enter three float values (P, T, and R).

**Question:**

**How can you improve this code to handle invalid input?**

* **Answer**: You can add a loop with a try-catch block to repeatedly prompt the user until valid input is entered:

**5.Write a java program for largest of two number using ternary operator:**

**Code:**

import java.util.Scanner;

class largest {

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

System.out.println("Enter your n1: ");

int n1 = input.nextInt();

System.out.println("Enter your n2: ");

int n2 = input.nextInt();

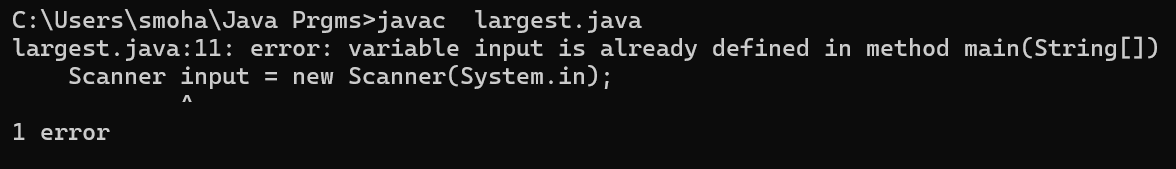
int largest = (n1 >= n2) ? n1 : n2 ;

System.out.println("Largest Number: " + largest);

}

}

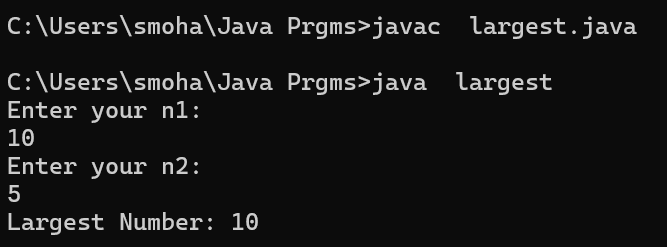
Output:



Error: Variable input is already defines in method main(String []).

|  |  |  |
| --- | --- | --- |
| **S.no** | **ERROR MESSAGE** | **ERROR RECTIFICTION** |
| **1** | Variable input is already defines in method main(String []). | Variable input is already defines in method main(String []). |

Output:



**Important points:**

**Class Definition:** The code defines a class named largest.

**User Input**: The code uses a Scanner object to get user input. It prompts the user to enter two integer values (n1 and n2).

**Conditional (Ternary) Operator**: The code uses the ternary operator to determine which of the two input numbers is larger. The expression (n1 >= n2) ? n1 : n2 assigns the larger value to the largest variable.

**Resource Management**: The Scanner object is not closed in the code, which is good practice to prevent resource leaks.

**Question:**

**How does the Scanner class work for getting user input in this code?**

* **Answer**: The Scanner class is used to get user input from the console. An instance of Scanner is created to read the input values n1 and n2.

**6.Write a java program to find factorial of a number:**

**Code:**

import java.util.Scanner;

class factorial {

public static void main(String[] args) {

Scanner input = new Scanner (System.in);

System.out.println("Enter n value: ");

int number = input.nextInt();

System.out.println("You entered " + number);

int fact=1;

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

fact=fact\*i;

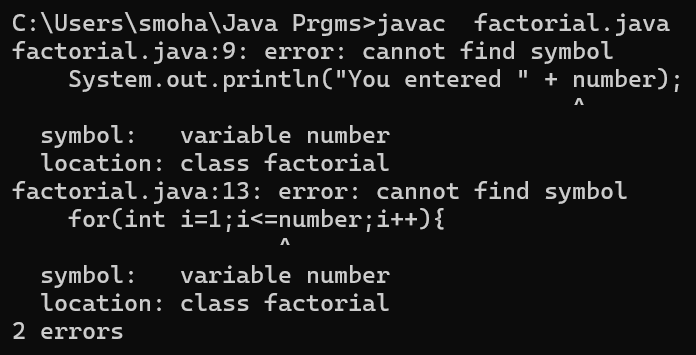
System.out.println(fact);

}

}

}

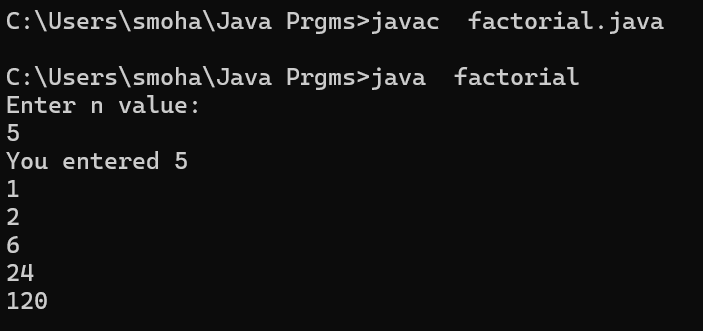
**Output:**



Error:there are two errors number variable is not indicated properly.

|  |  |  |
| --- | --- | --- |
| **S.no** | **ERROR MESSAGE** | **ERROR RECTIFICTION** |
| **1** | **Error:cannot find symbol** | number variable should indicate properly. |

**Output:**



**Important points:**

**Class Definition**: The code defines a class named factorial.

**Main Method**: The main method is the entry point for the program, containing the logic for user input and factorial calculation.

**User Input**: The code uses a Scanner object to get user input. It prompts the user to enter an integer value (number).

**Factorial Calculation**: The code calculates the factorial of the given number using a for loop. The factorial is computed as the product of all integers from 1 to the input number.

**Question**:

How is the factorial of a number calculated in this code?

* **Answer**: The factorial is calculated using a for loop that multiplies the value of fact by the loop variable i for each iteration from 1 to number.

**WEEK-3**

**1)** PROGRAME-1

AIM: To write a program for car color and all respective complextions using constructor and method.

Step 1:open notepad<<save the note pad in the path[desktop<<oops<<week 1<<car.java

**Code:**

class car{

//creating the attributes required fo the class

String car\_color,car\_brand,fuel\_type;

int maleage;

//constructor

car(String car\_color, String car\_brand, String fuel\_type,int maleage){

this. car\_color=car\_color;

this.car\_brand=car\_brand;

this. fuel\_type=fuel\_type;

this.maleage=maleage;

}

//creating methods for the class

public void start(){

System.out.println("The "+car\_brand+ " "+"car gets started" +" "+ "which is in"+" "+ car\_color +"color");

}

public void stop(){

System.out.println("The car is stopped due to less"+" " + fuel\_type);

}

public void service(){

System.out.println("The car is in servicing " +"has "+ maleage+" "+ "maleage");

}

public static void main(String[] args) {

//creating the objects for the class

car car1=new car("navy blue","maruthi""petrol",300);

car1.start();

car car2=new car("navy blue","Honda","petrol",400);

car2.stop();

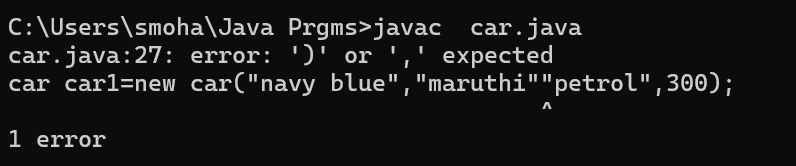
car car3=new car("black","maruthi","petrol",500);

car3.service();

}

}

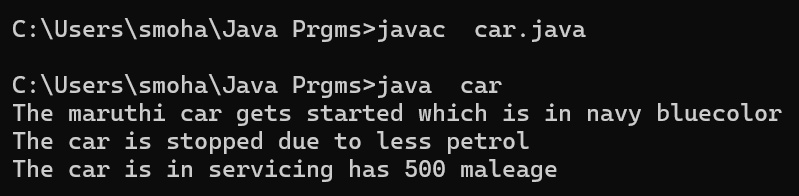
**Output:**



**Error:**Here there should be coma in between that two quotation marks.

|  |  |  |
| --- | --- | --- |
| **S.no** | **ERROR MESSAGE** | **ERROR RECTIFICTION** |
| **1** | Error: ’)’ or ‘ , ‘ expected | Keep coma in between that two quotation marks. |

**Output:**



**Important points:**

 **Constructor**: The constructor initializes the attributes with values provided as arguments.

**Methods**: The class has three methods: start, stop, and service. Each method performs a specific action and prints a message related to the car.

**Main Method**: The main method creates objects of the car class and calls the methods on these objects to demonstrate their functionality.

**Syntax Error**: There is a syntax error in the code. The car1 object creation line has an extra double-quote character. It should be car car1 = new car("navy blue", "maruthi", "petrol", 300);.

**Question:**

**How are objects of the car class created in the main method?**

**Answer**: Objects of the car class are created using the new keyword, followed by a call to the constructor with the appropriate arguments. For example: car car1 = new car("navy blue", "maruthi", "petrol", 300);.

2) AIM: To write a program for car color and all respective complextions using constructor and method.

Step 1:open notepad<<save the note pad in the path[desktop<<oops<<week 1<<BANK.java

import java.util.Scanner;

class BankAccount {

// Class-level variable to store balance

private float existing;

private Scanner input; // Single Scanner instance for input

public String name;

// Constructor

public BankAccount() {

input = new Scanner(System.in);

System.out.println("Enter the account holder name :");

this.name=input.next();

System.out.print("Enter existing amount in bank account: ");

this.existing = input.nextFloat();

}

// Deposit method

public void deposit() {

System.out.print("Enter amount to be deposited: ");

float deposit = input.nextFloat();

existing += deposit;

System.out.println("Existing amount now is: " + existing);

}

// Withdrawal method

public void withdrawal() {

System.out.print("Enter amount to be withdrawn: ");

float withdrawal = input.nextFloat();

if (existing < withdrawal) {

System.out.println("Not sufficient balance.");

} else {

existing -= withdrawal;

System.out.println("Remaining balance: " + existing);

}

}

// Main method

public static void main(String[] args) {

BankAccount customer1 = new BankAccount();

customer1.deposit();

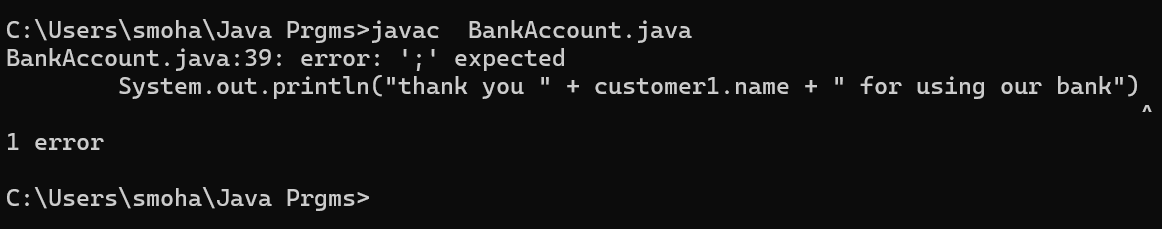
customer1.withdrawal();

System.out.println("thank you " + customer1.name + " for using our bank");

}

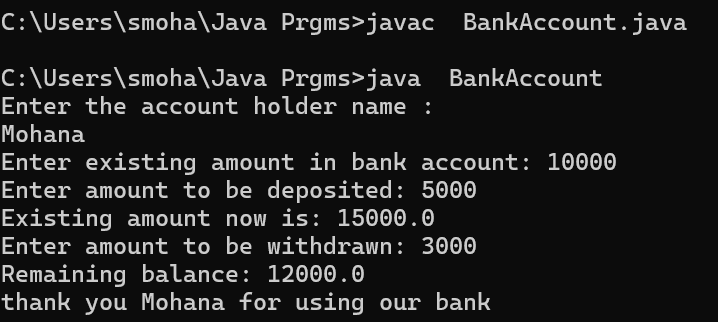
}

**Output:**



**Error:**

|  |  |  |
| --- | --- | --- |
| **S.no** | **ERROR MESSAGE** | **ERROR RECTIFICTION** |
| **1** | Error: ‘;’ expected | Spectify the semicolon at the end. |



**IMPORTANT POINTS:**

* Java constructor is used to save the variables present in different or same class or methods.
* In Java, the this keyword refers to the current instance of a class. It is commonly used to distinguish between instance variables and parameters with the same name, or to refer to the current object from within a method or constructor.
* In Java, a method is a block of code that performs a specific task and can be invoked to execute that task. It typically consists of a method signature (name, return type, and parameters) and the body of the method, which contains the logic.

**WEEK-4**

**1)** PROGRAME-1

AIM: To write a program for printing the title of the book and the author and year of publication using the constructors

Step 1:open notepad<<save the note pad in the path[desktop<<oops<<week 1<<person.java

class book{

String titleofthebook;

String Author;

int yearofpublication;

//creating constructor//

book(String titleofthebook,String Author,int yearofpublication){

this.titleofthebook=titleofthebook;

this.Author=Author;

this.yearofpublication=yearofpublication;

}

//creating a methods//

public void titleofbook(){

System.out.println("The Title of book is :"+titleofthebook);

}

public void Author(){

System.out.println("The Author of book is :"+Author);

}

public void yearofpublication(){

System.out.println("The book is published in the year :"+yearofpulication);

}

//creating objects//

public static void main(String[] args){

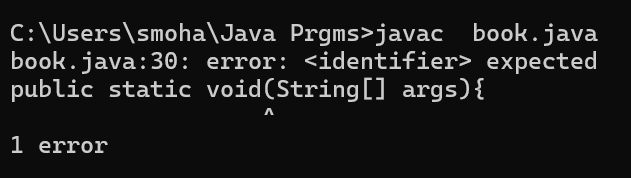
book b1=new book("the story of honey","Priya",2020);

b1.titleofbook();

}

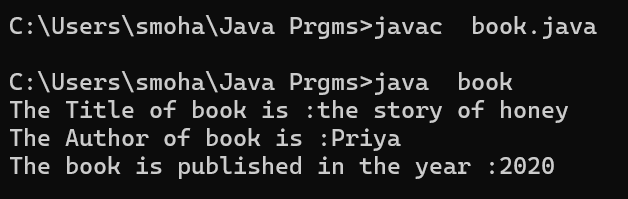
}

Output:



Error:

|  |  |  |
| --- | --- | --- |
| **S.no** | **ERROR MESSAGE** | **ERROR RECTIFICTION** |
| **1** | Error: <indentifier> expected | Spectify the main function. |



**Important points:**

Class Declaration: The class is named book and contains three instance variables: titleofthebook, Author, and yearofpublication.

Constructor: The constructor initializes the instance variables with the values passed as parameters.

Main Method: Creates an instance of the book class and calls the titleofbook() method to display the title.

Question:

Identify the typo in the code and explain its impact.

* **Answer**: There is a typo in the yearofpublication() method. The instance variable yearofpulication should be yearofpublication. Due to this typo, the method will not print the year of publication correctly, and the code will not compile if the method is called.

**2)** AIM: To write a program for printing the title of the book and the author and year of publication using the constructors

Step 1:open notepad<<save the note pad in the path[desktop<<oops<<week 1<<exam.java

class myclass{

//creating the variables

static int count=0;

final double pi=3.1415;

//creating a constructor

myclass(){

count++;

}

//method to print the values

public void values(){

System.out.println(+count);

System.out.println(+pi);

}

//object and the main function

public static void main(String[] args){

myclass one=new myclass();

one.values();

myclass two=new myclass();

two.values();

myclass three=new myclass();

three.values();

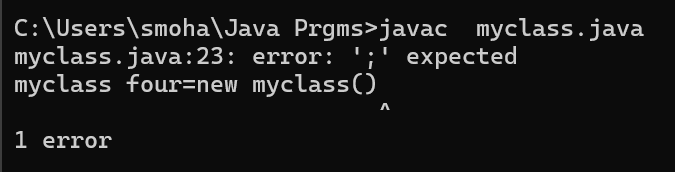
myclass four=new myclass();

four.values();

}

}

**Output:**

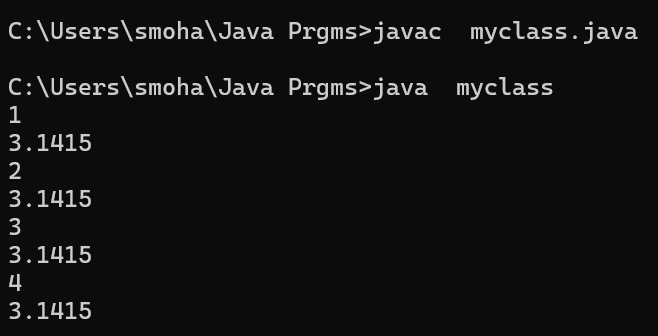


Error:

Error:

|  |  |  |
| --- | --- | --- |
| **S.no** | **ERROR MESSAGE** | **ERROR RECTIFICTION** |
| **1** | Error: ‘;’ expected | Spectify the semicolon at the end. |

Output:



IMPORTANT POINTS:

1. Java constructor is used to save the variables present in different or same class or methods.
2. In Java, the ++ operator increments a variable by 1, either as pre-increment (++x) or post-increment (x++).
3. In Java:
4. static: A static variable belongs to the class, not instances, meaning all objects share the same value.
5. final: A final variable cannot be modified once assigned, making it constant.