**23CSE111**

**OBJECT ORIENTED PROGRAMMING**

**LAB MANUAL**



**Department of Computer Science and Engineering**

**Amrita School of Engineering**

**Amrita Vishwa Vidyapeetham, Amaravati**

**Campus**

**Name: M. Krish Sashank**

**Verified by:**

**Roll.no: AV.SC.U4CSE24229**

**INDEX**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.NO** | **TITLE** | **DATE** | **SIGNATURE** |
| Week-1 | Download and  installation of java software | 26-01-25 |  |
| Java program to print the message” Welcome to java programming” |
| Java program to print details of a student. |

# 

# 

# WEEK-1

1. **Process of Installing JDK (Java Development Kit)**

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

* 1. **Download JDK:**
* Go to the Oracle JDK download page in google and click on JDK-21 version which is Long term support (LTS) version.
* Click the download link as your operating system (Windows, macOS, or Linux).
  1. **Install JDK:**
* Once downloaded, run the installer.
* Follow the given instructions and keep clicking "Next" until it is done.
  1. **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

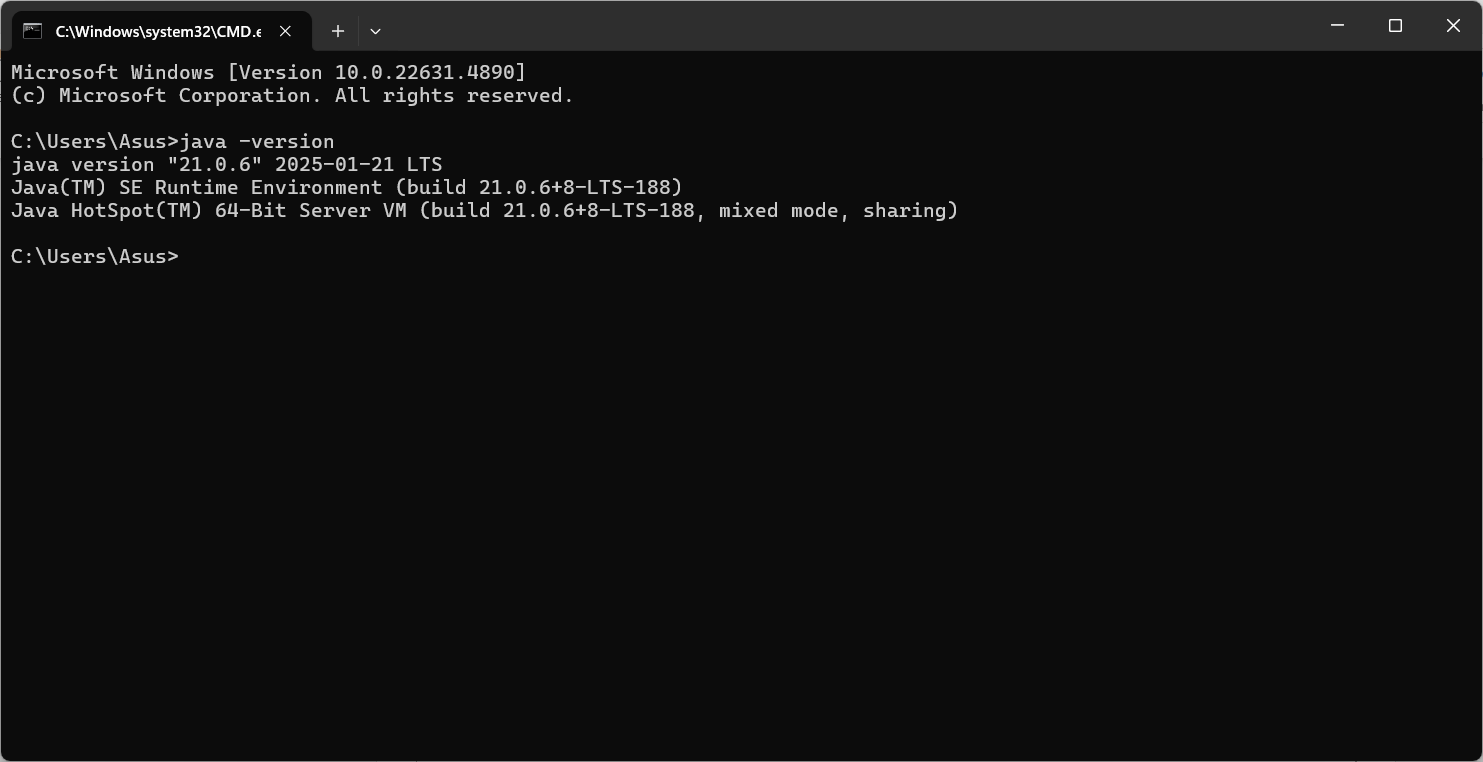
C:\Program Files\Java\jdk-21\bin)

* Find Path under **System Variables**, click **New**, and add the path of the jdk-21(C:\Program Files\Java\jdk-21\bin)



**Checking JDK Version: -**

* 1. **Open Command Prompt:**
* Presswin+R, typecmd, and press Enter.
  1. **Check Version:**
* Type java -version and press Enter.
* Type javac --version and press Enter.



**ERROR:- No Error found**

1. **Simple Java Program for printing Name, Class, Roll No, of a Student**

Write your code in Notepad and execute it in cmd prompt

**CODE: -**

class Main

{

public static void main(String[] args)

{

System.out.println("Name:M.Krish");

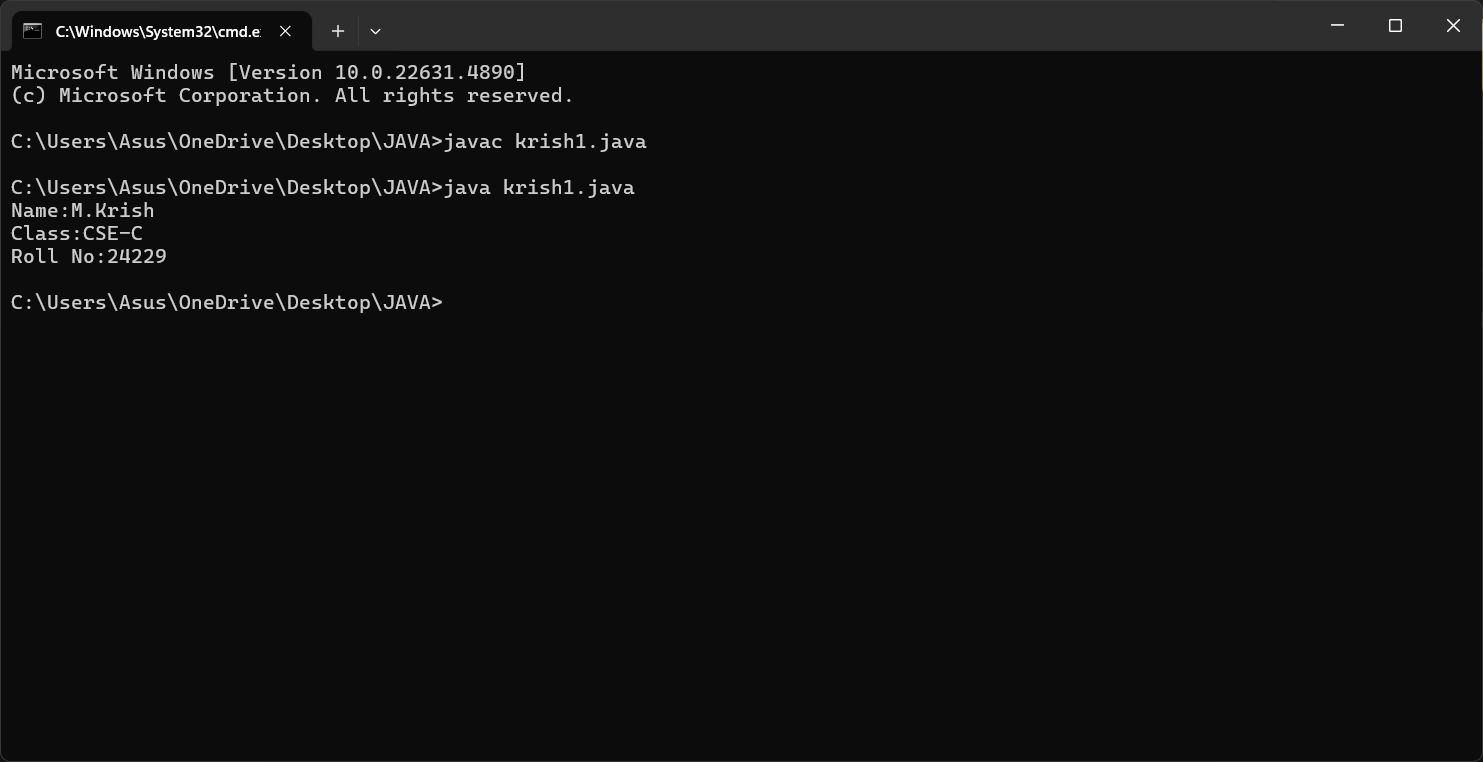
System.out.println("Class:CSE-C");

System.out.println("Roll No:24229");

}

}

**Output: -**



**ERROR:-**

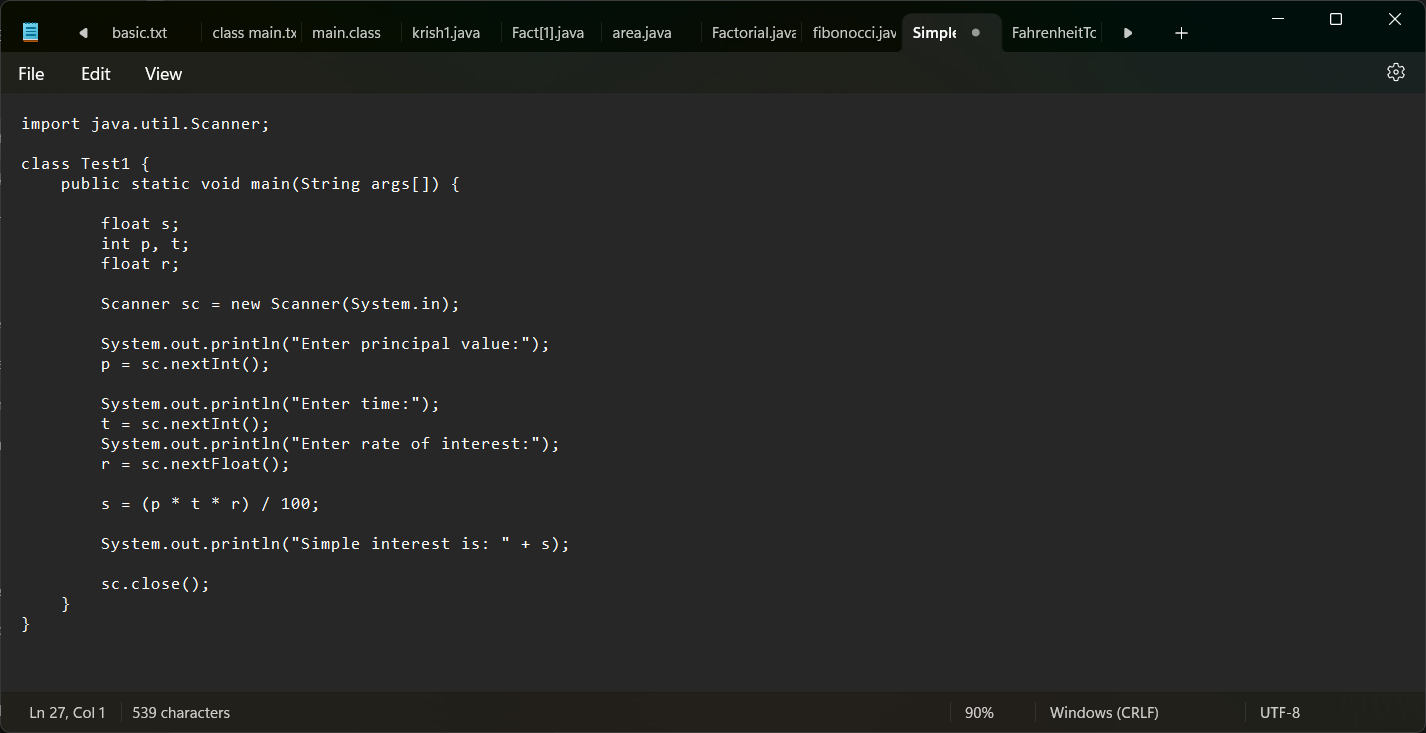
|  |  |
| --- | --- |
| **CODE ERROR** | **CODE RECTIFICATION** |
| **writing small “S” in place of ”S”**  **In system.out.println()** | **code is rectified by keeping capital “S”** |

**WEEK-2**

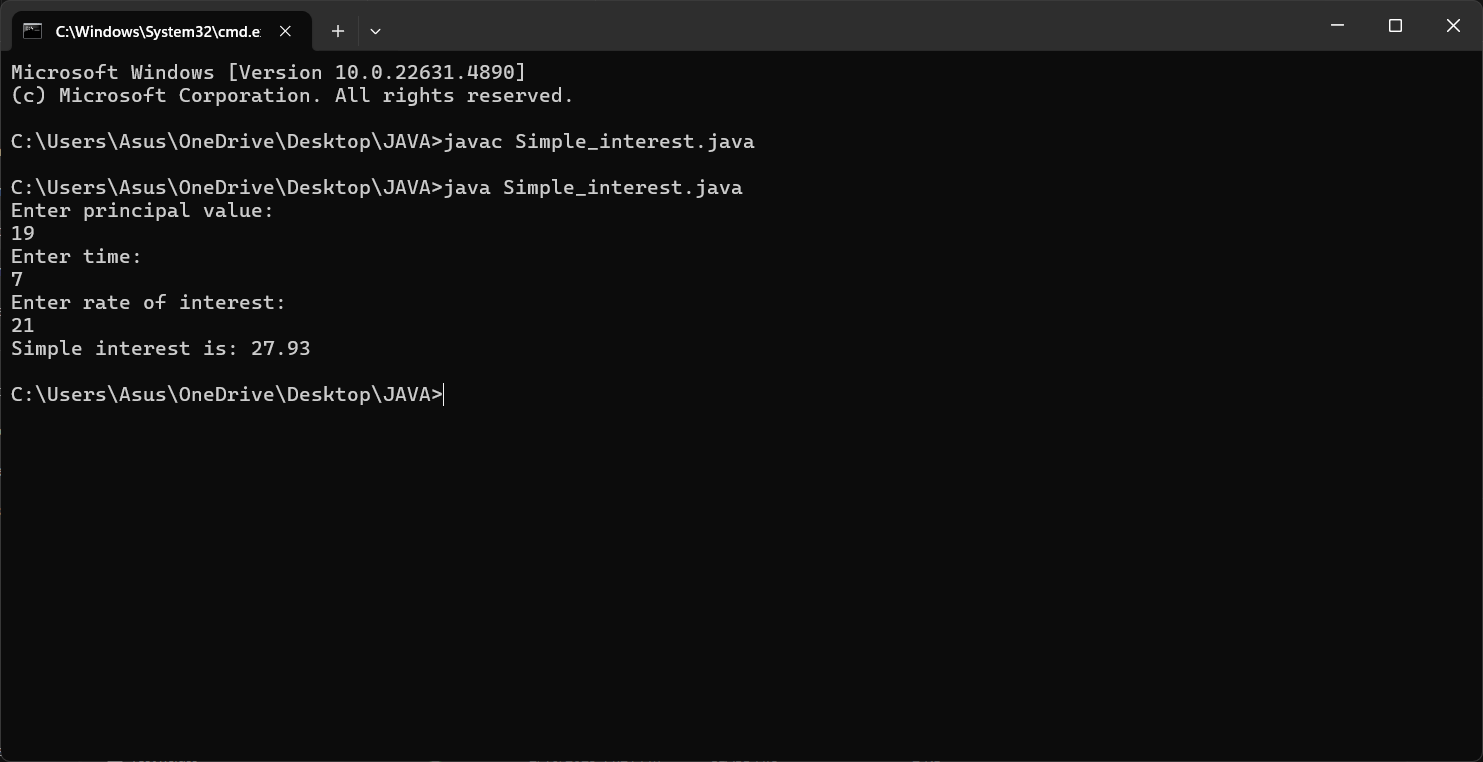
**PROGRAM-1 :-**

**AIM :- Simple Java Program for finding simple interest by taking input from User**

**CODE :-**

****

**OUTPUT :-**

****

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **ERROR** | **REASON** | **RECTIFICATION** |
| **1** | **Syntax error** | **; missing** | **Added ;** |
| **2** | **Formula mistake** | **Placed % instead of \*** | **Corrected the mistake** |

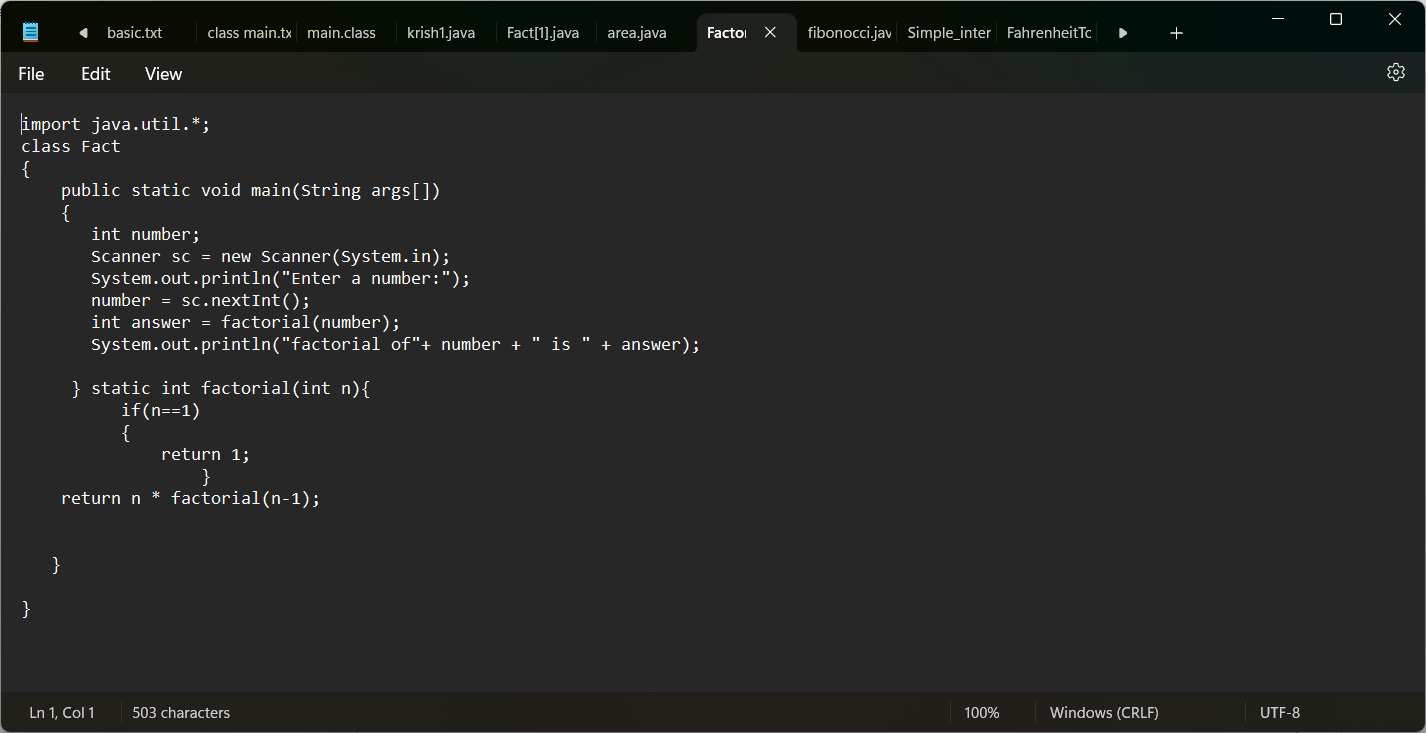
**Concepts to be known:**

1. import java.util.Scanner; - To accept input from user, Scanner class under util package has to be imported.
2. double p=input.nextDouble(); - Used to read double data type stored under the object created
3. System.out.println(“ “); - It is used to print string inside the quotes. After printing, the cursor moves to the beginning of the next line.

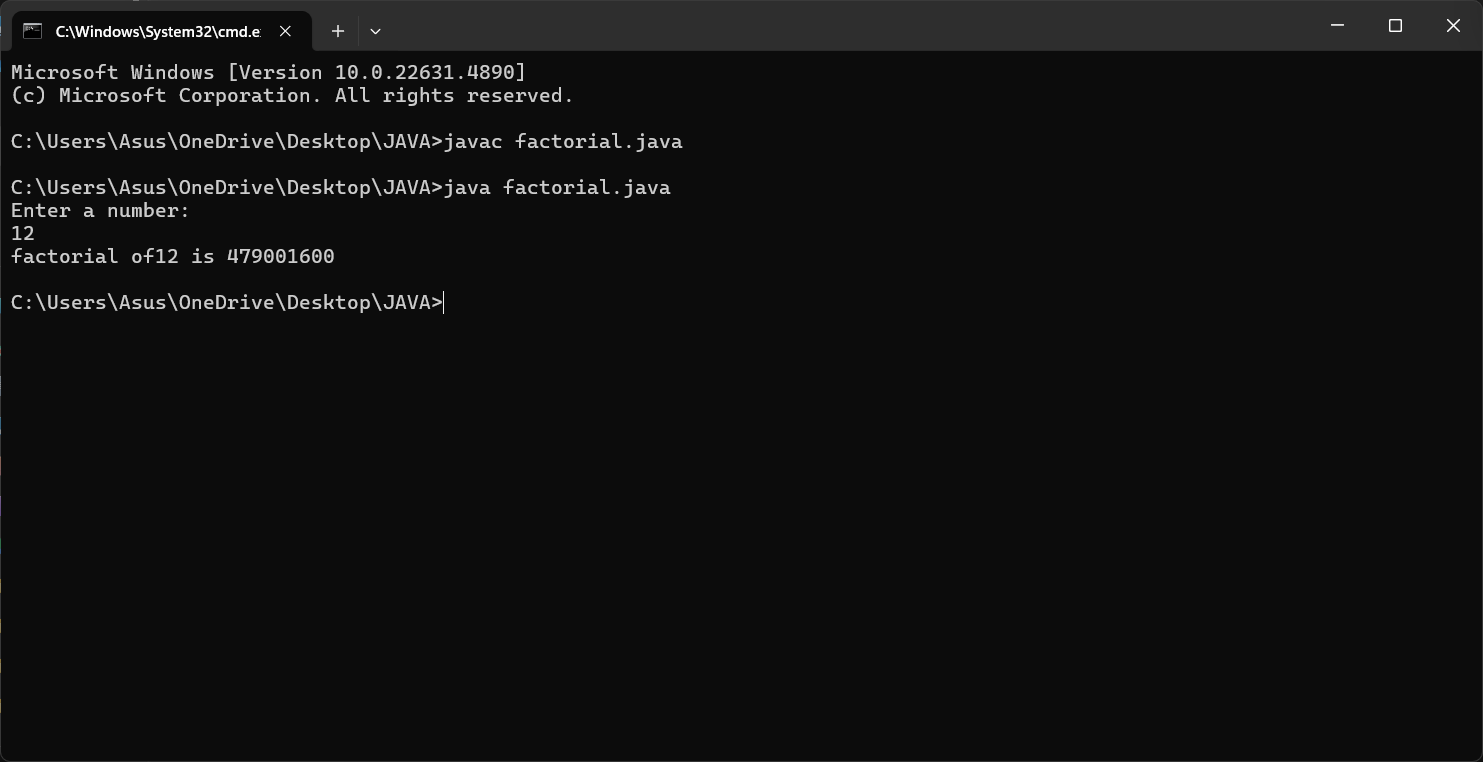
**PROGRAM-2 :-**

**AIM :- Write a simple program to calculate factorial of a number and read the input from user**

**INPUT :-**

****

**OUTPUT :-**

****

**ERROR:-**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.NO** | **ERROR** | **REASON** | **RECTIFICATION** |
| **1** | **Syntax error** | **Haven’t added ; in line 6** | **Added ;** |
| **2** | **Conditional error** | **Wrong formula** | **Formula rectified** |

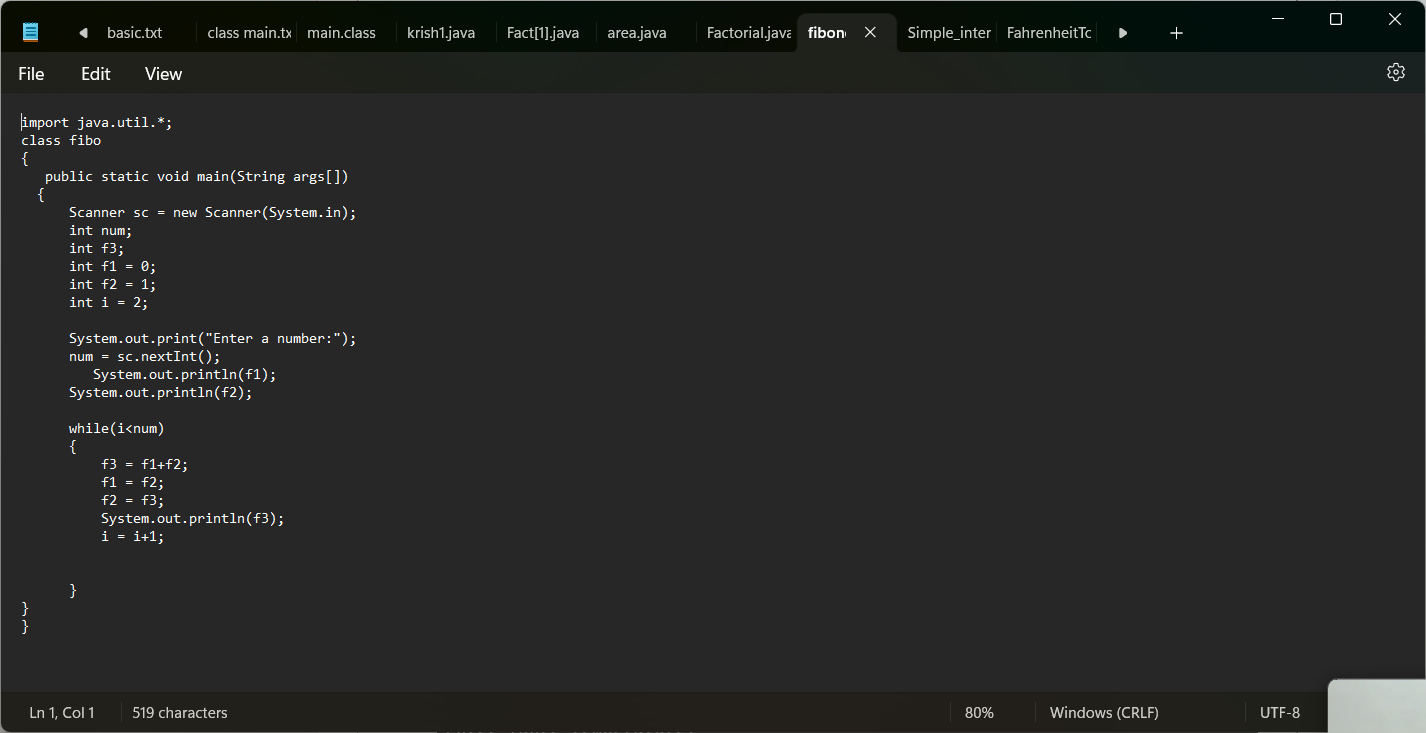
**Concepts to be known:**

1. for (int i=1; n>=i;--n){ } - For loop syntax: for(initial expression; test expression; update expression){} The loop is executed, until the test expression evaluates to be false.

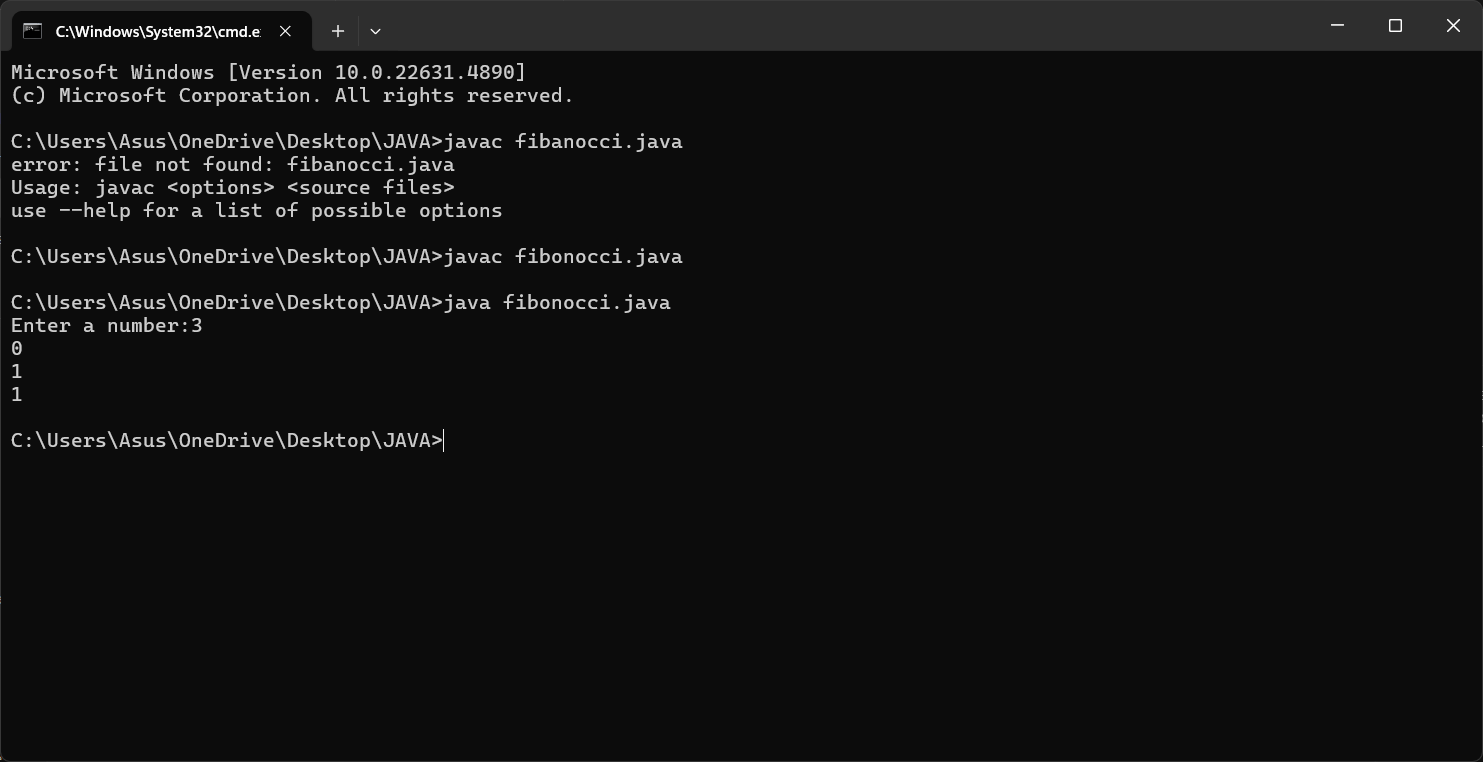
**PROGRAM-3 :-**

**AIM :- Write a program to to calculate the fibonacii sequence and take the input from user.**

**INPUT :-**

****

**OUTPUT :-**

****

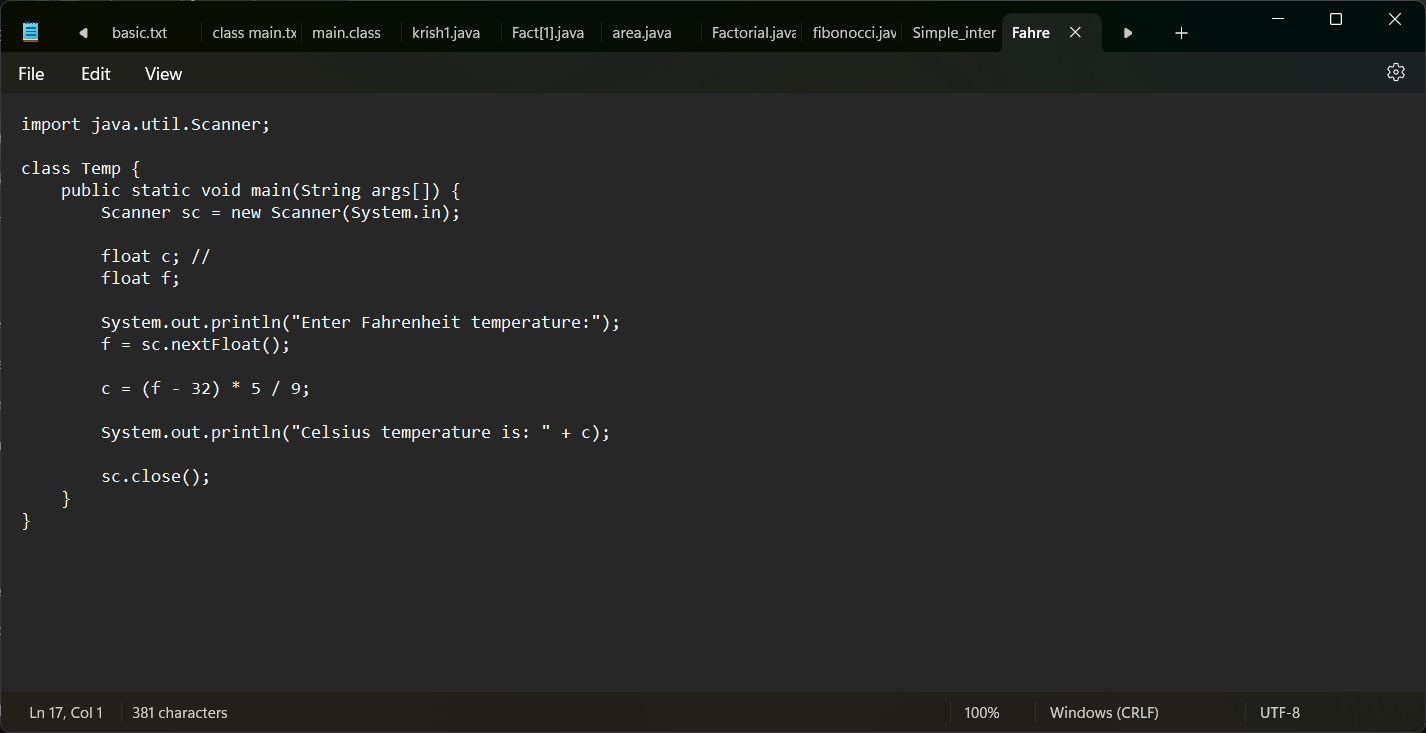
**ERROR:-**

|  |  |  |
| --- | --- | --- |
| **S.No** | **Error** | **Code Rectification** |
| **1** | **Inserted “p” in capital** | **Changed it into small letter** |
|  |  |  |

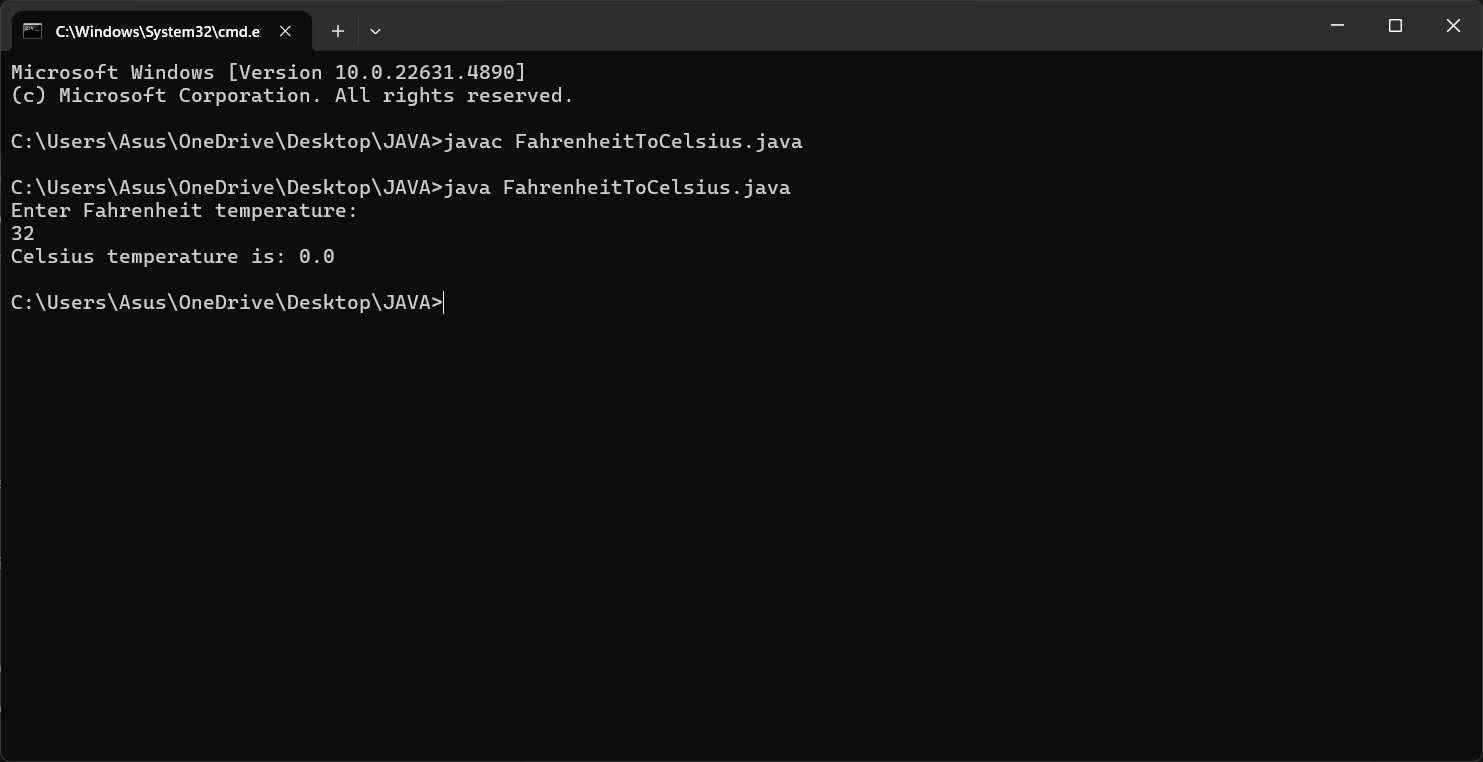
**PROGRAM - 4:-**

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

**INPUT :-**

****

**OUTPUT :-**

****

**Concepts to be known:**

1. import java.util.Scanner; - To accept input from user, Scanner class under util package has to be imported.

## 2. Scanner input=new Scanner(System.in); - Used to create a Scanner object

1. double fh=input.nextDouble(); - Used to read double data type stored under the object created
2. System.out.println(“ “); - It is used to print string inside the quotes. After printing, the cursor moves to the beginning of the next line.

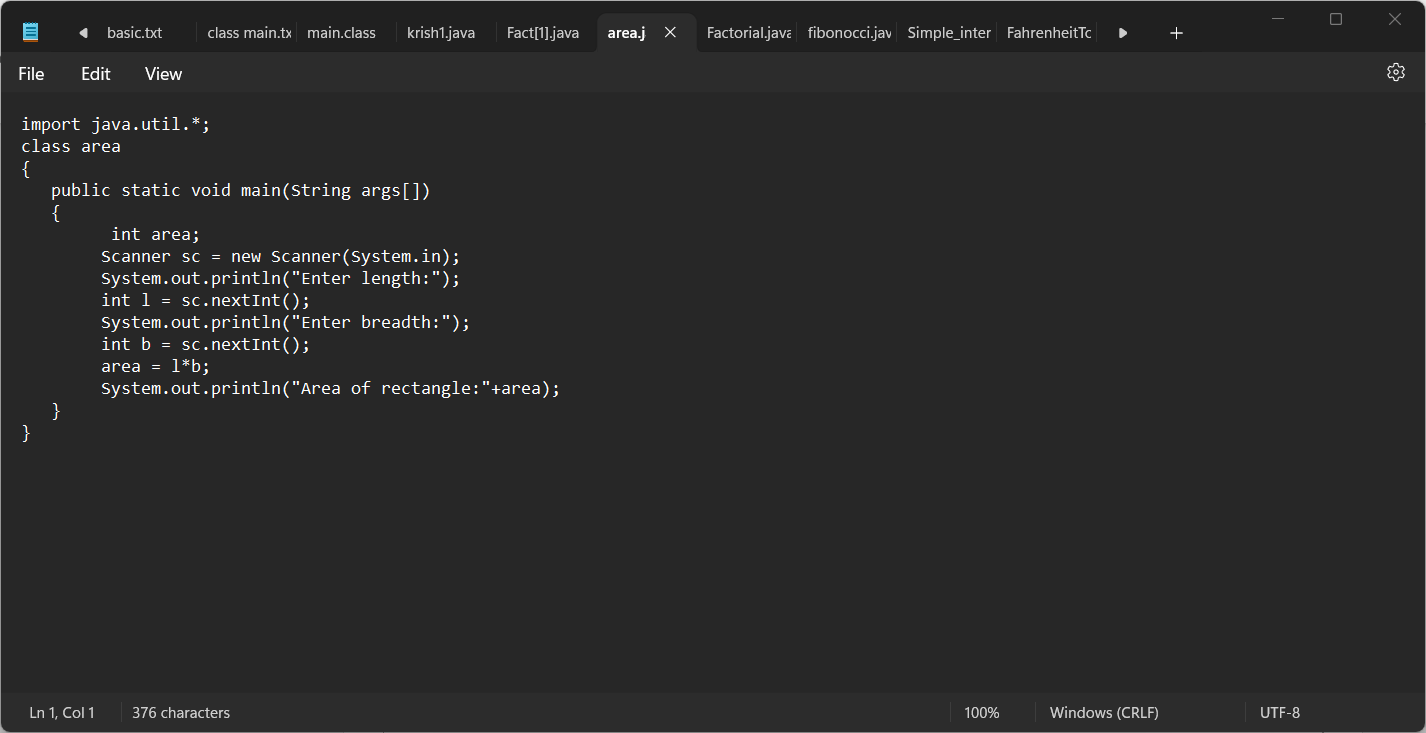
**ERROR:-**

|  |  |  |
| --- | --- | --- |
| **S.NO** | **ERROR** | **CODE RECTIFICATION** |
|  | **No error** |  |
|  |  |  |

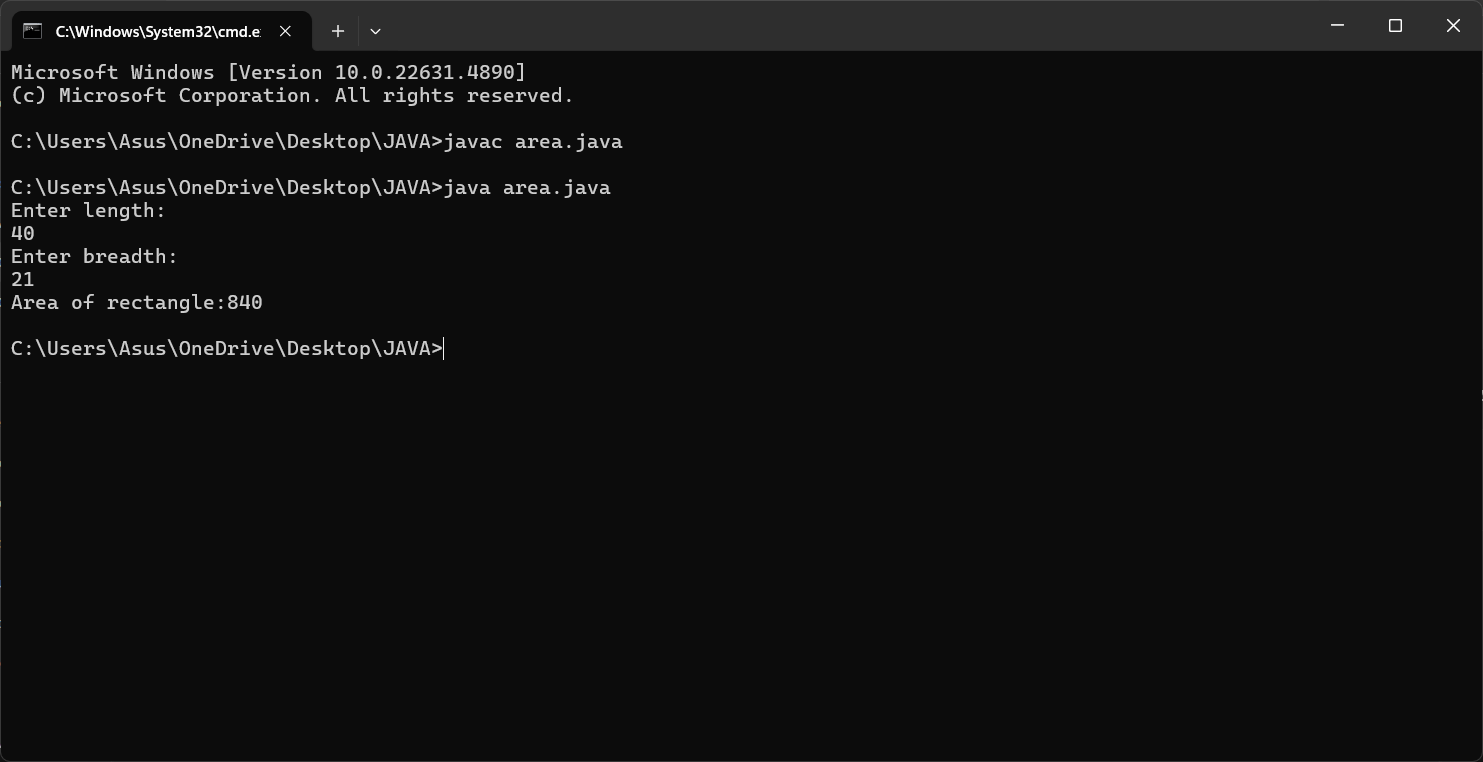
**PROGRAM-5:-**

**AIM :- Write a simple program to find the area of rectangle:**

**CODE :-**

****

**OUTPUT :-**

****

**ERROR:-**

|  |  |  |
| --- | --- | --- |
| **S.NO** | **ERROR** | **CODE RECTIFICATION** |
| **1** | **In “sc.next” n was written in capital** | **Changed it into small** |
|  | **Line no.13**  **Written “area of rectangle=”**  **Instead of “area of rectangle:”** | **Changed the syntax** |

**Concepts to be known:**

1. import java.util.Scanner; - To accept input from user, Scanner class under util package has to be imported.

## 2. Scanner input=new Scanner(System.in); - Used to create a Scanner object

1. int ln=input.nextInt(); - Used to read the integer data type stored under the object created
2. System.out.println(“ “); - It is used to print string inside the quotes. After printing, the cursor moves to the beginning of the next line.

# WEEK 03

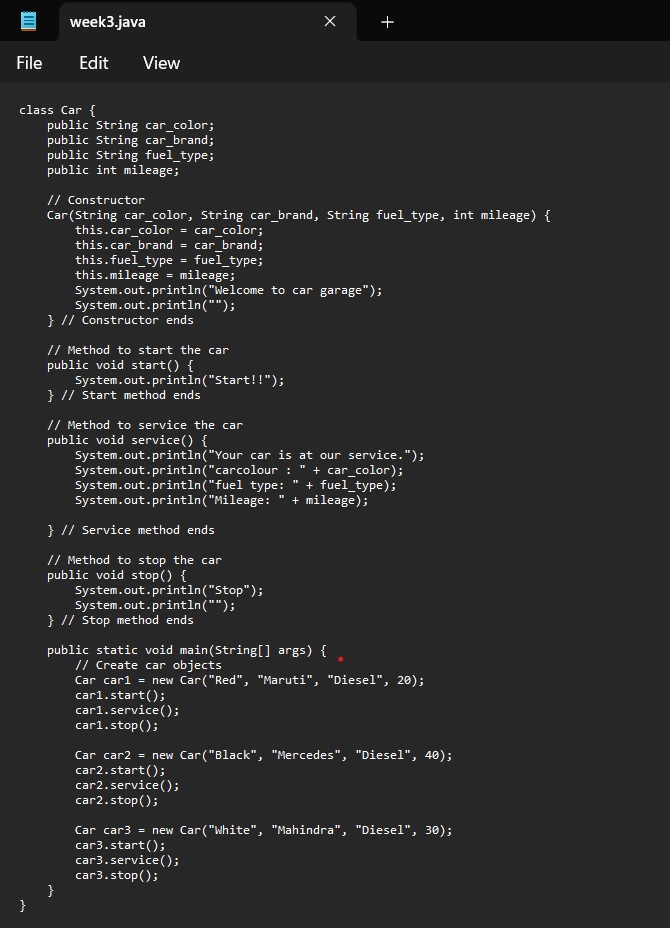
**PROGRAM-1:**

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

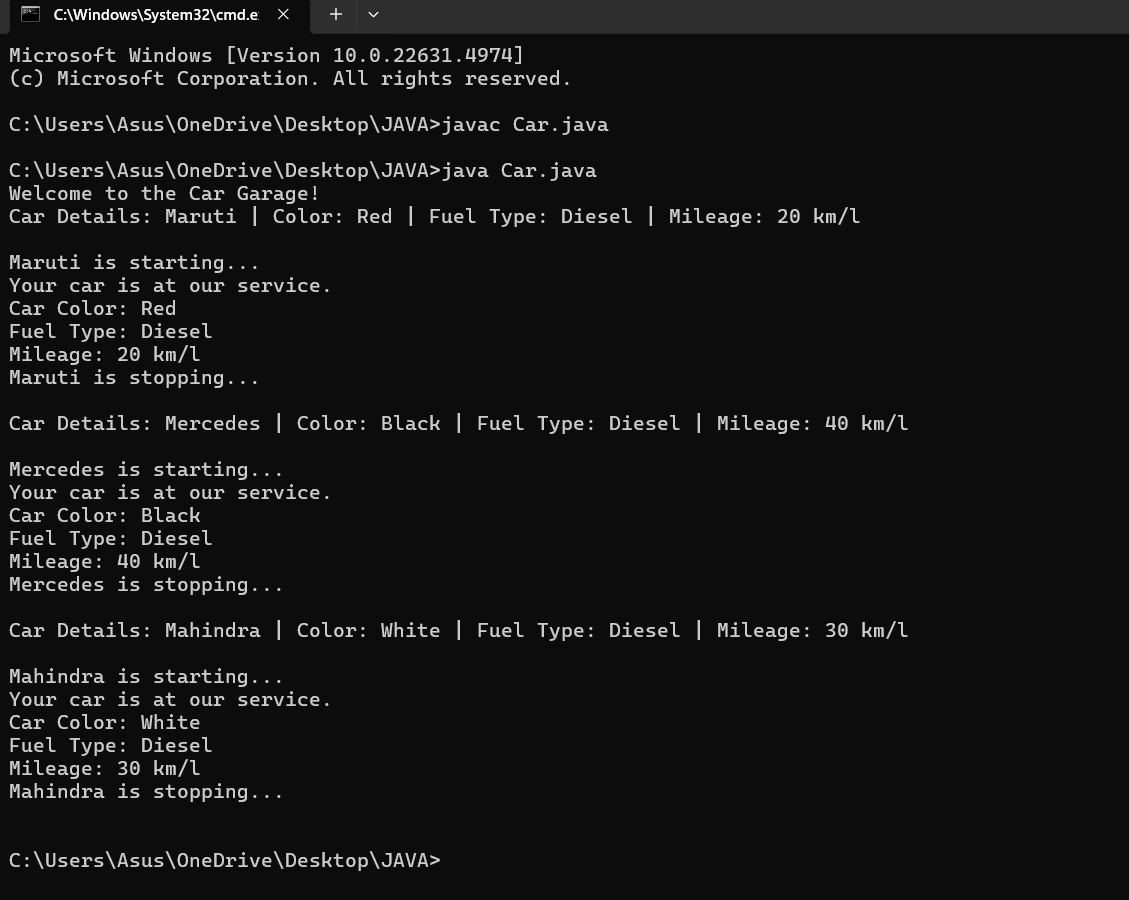
1. Create a class with name “Car”
2. Create 4 attributes, named: car\_color, car\_brand, fuel\_type, mileage
3. Create 3 methods, named: start(), service(), stop()
4. Create 3 objects, named: car1, car2, car3

Create a constructor, which should print, “Welcome to car garage”

**CODE:**



**OUTPUT:**



**ERRORS:**

|  |  |  |
| --- | --- | --- |
| Sno. | Error message | Error rectification |
| 1. | error: ';' expected car1.start() | Add a “;”  car1.start(); |
| 2. | error: illegal start of type public void stop({ | Add a “)”  public void stop(){ |
| 3. | error: cannot find symbol  thiscar\_brand=car\_brand; | Add a “.” this.car\_brand=car\_brand; |

**Class Diagram:**

|  |
| --- |
| Car |
| + car\_color: String  + car\_brand: String  + fuel\_type: String  + mileage: int |
| + Car(): void  + start(): void  + service(): void  + stop(): void |

**Concepts to be known:**

1. public String car\_color; - Used to declare a variable named car\_color, with data type as String with public accessibility.
2. Car(String car\_color,String car\_brand,String fuel\_type,int mileage){ } – It is a constructor (method with name same as class), which requires parameters such as car\_color (String data-type) and so on.
3. this.car\_color=car\_color; - “this” is a default method, which is used to point to the instance variables.
4. public void start(){} – used to declare a method, which will return nothing(void) in public accessibility.
5. Car car1=new Car("Red","Maruti","Diesel",20); - used to create a object in class Car, with object name as car1.

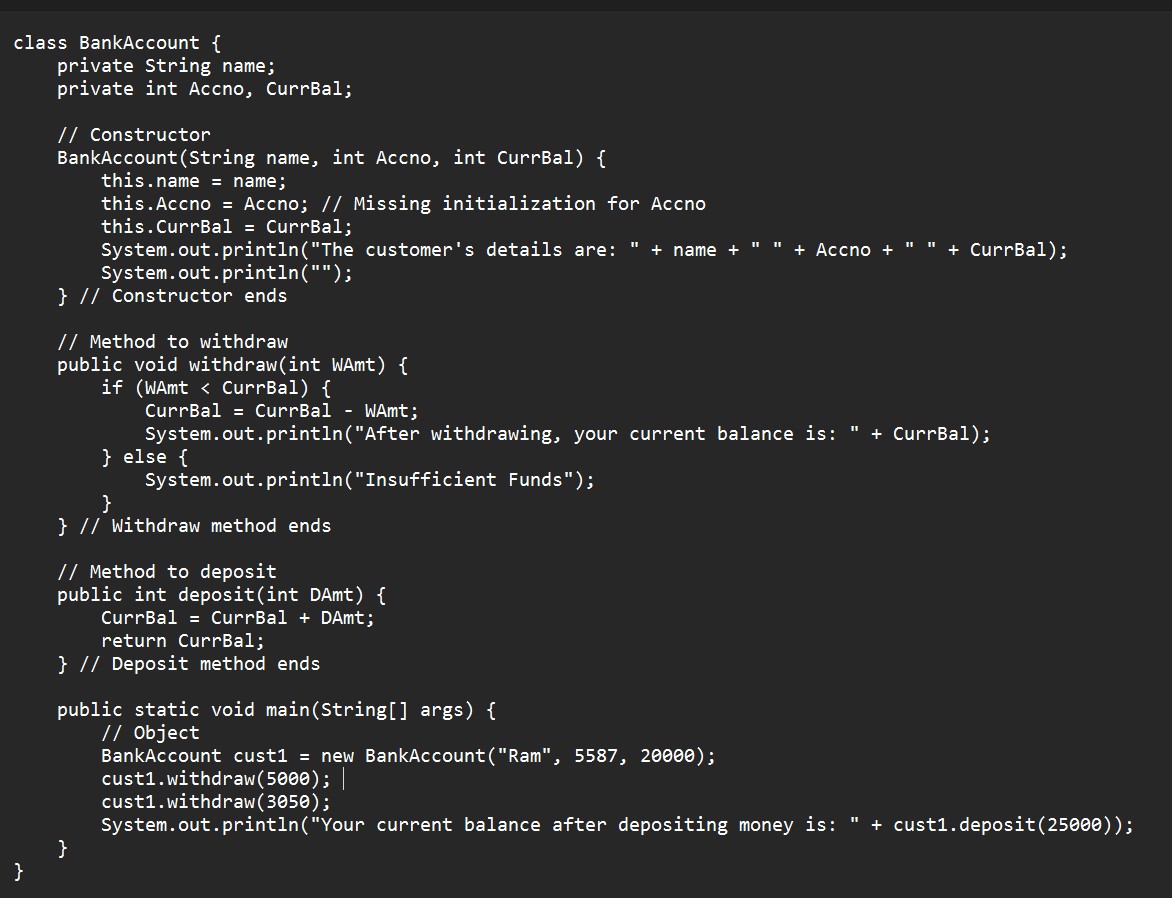
car1.start(); - Calling a method, under object car1.

**Program 2**

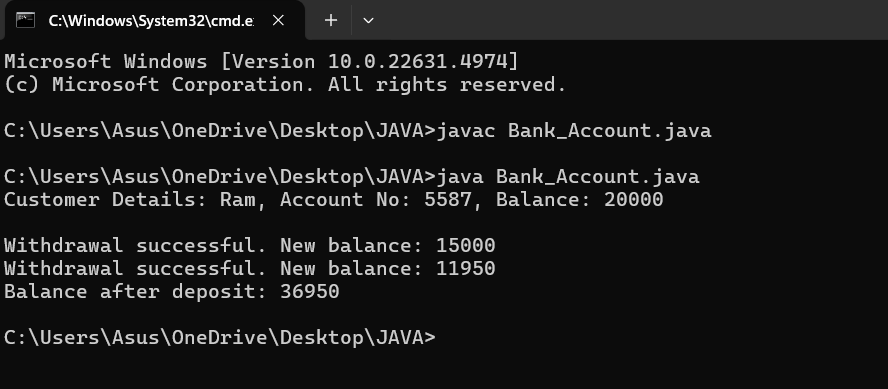
**Aim:** To write a java program to create a class named BankAccount, with 2 methods deposit() and withdraw().

1. deposit(): Whenever an amount is deposited, it has to be update the current amount.
2. withdraw(): Whenever an amount is withdrawn, it has to be less than the current amount , else print (“Insufficient funds”)

**Code:**



**Output:**



**ERRORS:**

|  |  |  |
| --- | --- | --- |
| **Sno.** | **Error message** | **Error rectification** |
| 1. | error: ';' expected  cust1.withdraw(3050) | Add a “;”    cust1.withdraw(3050); |
| 2. | error: cannot find  symbol  thisCurrBal=CurrBal; | Add a “.”    this.CurrBal=CurrBal; |

**Class Diagram:**

|  |
| --- |
| BankAccount |
| * name: String * Accno: int * CurrBal: int |
| BankAccount: void  + withdraw(int WAmt): void  + deposit(int DAmt): int |

**Concepts to be known:**

1. Private String name; - Used to declare a variable named name, with data type as String with private accessibility.
2. BankAccount(String name,int Accno,int CurrBal){ } – It is a constructor (method with name same as class), which requires parameters such as name (String data-type) and so on.
3. this.CurrBal=CurrBal; - “this” is a default method, which is used to point to the instance variables.
4. public void withdraw(int WAmt){ } – used to declare a method, which will return nothing(void) in public accessibility, which requires a parameter WAmt(integer data type).
5. public int deposit(int DAmt){} - used to declare a method, which will return integer data type in public accessibility, which requires a parameter DAmt(integer data type).
6. BankAccount cust1=new BankAccount("Ram",5587,20000); - used to create a object in class BankAccount, with object name as cust1.
7. cust1.withdraw(50000); - Calling a method, under object cust1, by passing a parameter.

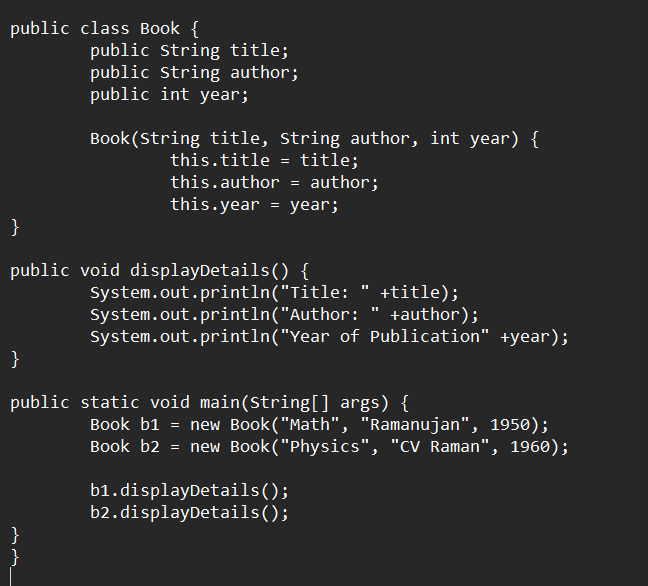
System.out.println("Your current balance after depositing money is:"+cust1.deposit(25000)); - Deposit method will return the value, which will be directly printed.

**WEEK-4**

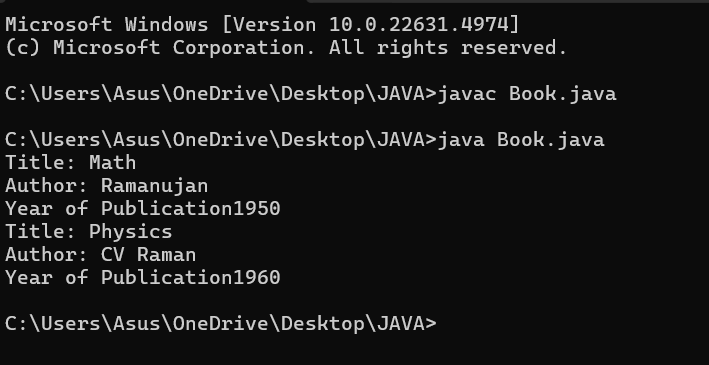
**PROGRAM-1 :-**

**AIM:** WRITE A JAVA PROGRAM WITH CLASS NAMED “Book”. THE CLASS SHOUKD 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 AND CREATE A METHOD WHICH DISPLAYS THE DETAILS OF 2 BOOKS.

**INPUT:-**

****

**OUTPUT :-**

****

**ERROR:**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **ERROR TYPE** | **Reason for error** | **Rectification** |
| **1.** | Syntax error | No semicolon | Semicolon added |
| **2.** | Runtime error | Incorrect path | Copied correct path |

**CLASS DIAGRAM:**

|  |
| --- |
| Book |
| -title: String  -author: String  -year: int |
| + Book(title: String, author:String, year: int) + displayDetails(): void |

**IMPORTANT POINTS:**

1. **Constructor**:

* The constructor Book(String, String, int) is used to initialize the object when it is created.
* The keyword **this** is used to differentiate between class attributes and constructor parameters.

2.**Method**:

* The method displayDetails() is used to display the book details.
* The **System.out.println()** method prints the details to the console.

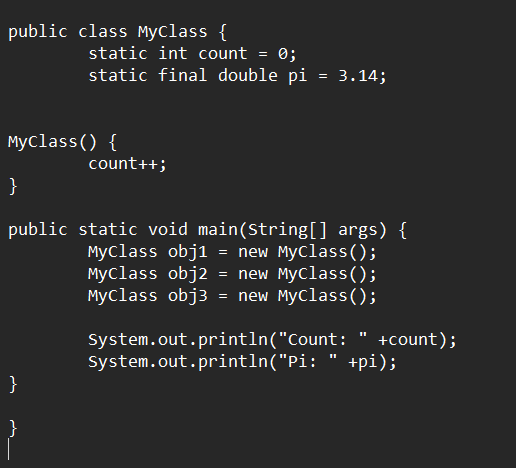
3. **Object Creation**:

* Two objects b1 and b2 are created using the constructor.

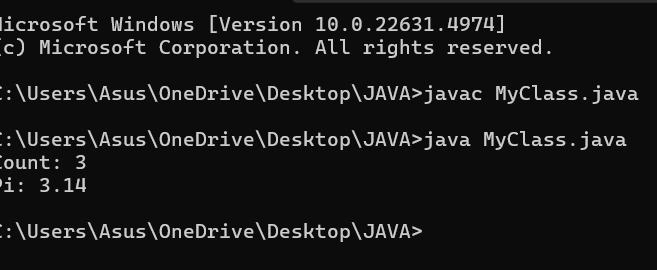
**PROGRAM-2 :-**

**AIM:** WRITE A JAVA PROGRAM WITH CLASS NAMED “MyClass” WITH A STATIC VARIABLE COUNT OF INT TYPE. INTIALIZE IT TO ZERO AND A CONSTANT VARIABLE “Pi” OF TYPE DOUBLE INITIALIZED TO “3.14” AS ATTRIBUTES OF THAT CLASS. NOW DEFINE A CONSTRUCTOR FOR “MyClass”, THAT INCREMENTS THE COUNT VARIABLE EACH TIME AN OBJECT OF “MyClass” IS CREATED. FINALLY, PRINT THE FINAL VALUES OF ‘COUNT’ AND ‘PI’ VARIABLES AND CREATE 3 OBJECTS.

**INPUT :-**

****

**OUTPUT:**

****

**ERROR:**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Error Type** | **Reason for error** | **Rectification** |
| **1.** | No class | No class name declared | Created class named ‘MyClass’ |
| **2.** | Syntax error | Not added keyword | Added keyword named ‘new’ |

**CLASS DIAGRAM:**

|  |
| --- |
| MyClass |
| -count: int (static)  -pi: double (static, final) |
| +MyClass()  +main(args: String[]):void |

**IMPORTANT POINTS:**

**1.Static Keyword**

* Static members belong to the **class, not to individual objects**.
* Only one copy of the static variable is maintained for all objects.

**2.Static Variable**

* **static int count**:
  + Shared among all objects of the class.
  + It is initialized only once and not for every object.
  + It increments every time the constructor is called.

**3.Final Variable**

* **static final double pi**:
  + The **final** keyword makes the variable constant.
  + Its value **cannot be changed** once assigned.
  + It must be initialized at the time of declaration.