(23CSE111) Object Oriented Programming

LAB MANUAL

NAME: S. charan reddy

Roll no: AV.SC.U4CSE24321

SECTION: CSE-A

**WEEK1:**

**Program1:** How to download and install java

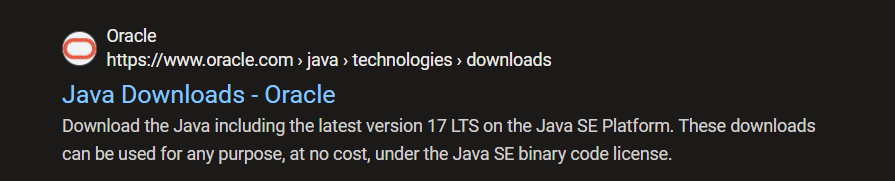
**Aim:** To download and install java

**Procedure:**

**Step1**: Go to browser and search “download java ”.

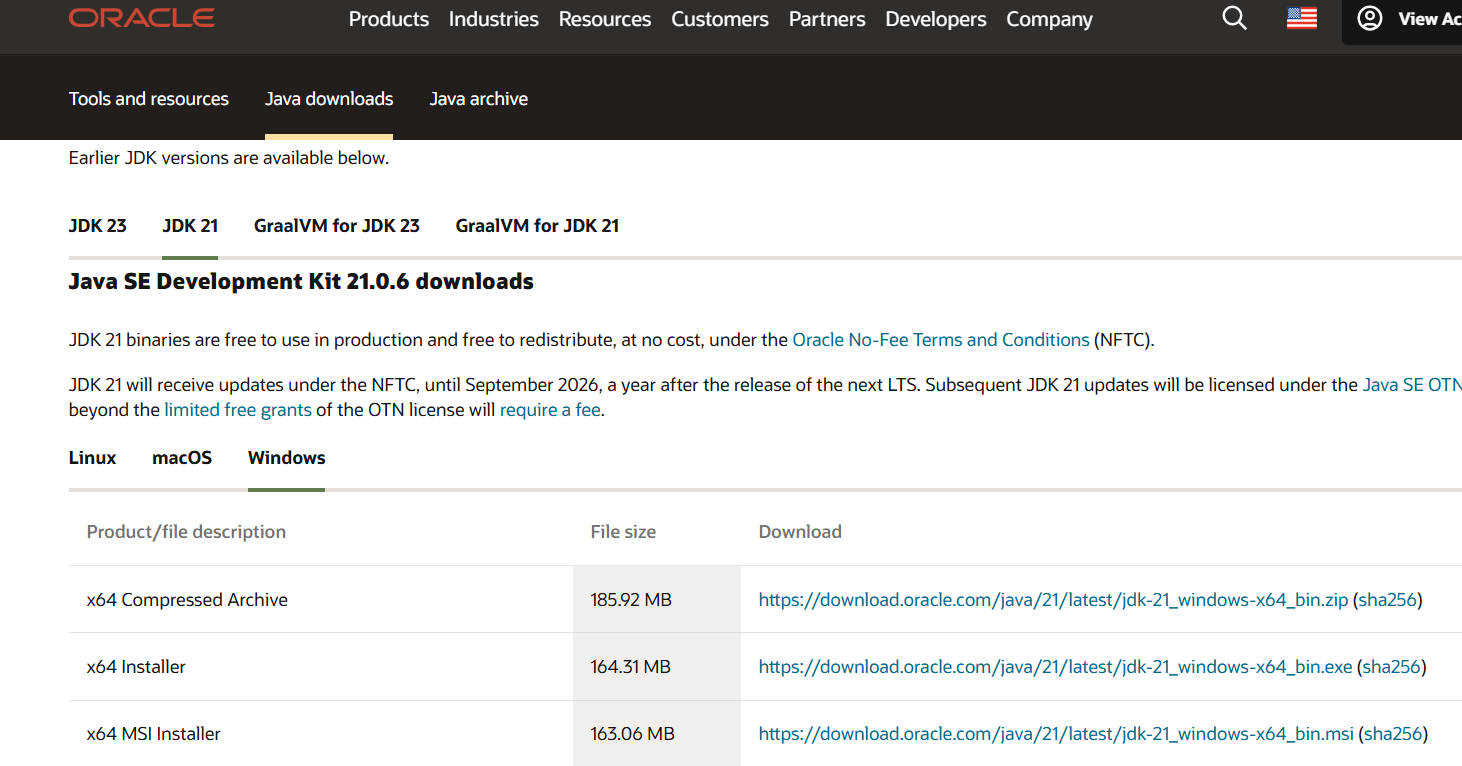
**Step2**: when you search you can see oracle website

to download java , click on it.



**Step3**: Download the java JDK21 version in

linux/macOS/windows in “64x installer”.



**Step4:** After downloading JDK 21 java version

download the installer of java



**Step5:** go to search the environmental variables

**Step6:** select system variables and add the selected Path.

A screenshot of a computer

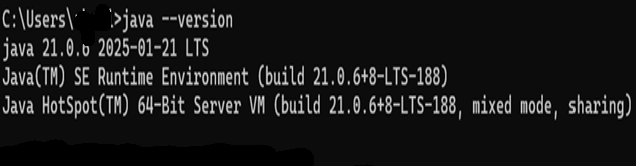
Description automatically generated

**Step7:** lastly do apply

**Step8:** for cross checking it was installed or not open

Command promt and type java --version

Then it will show like this

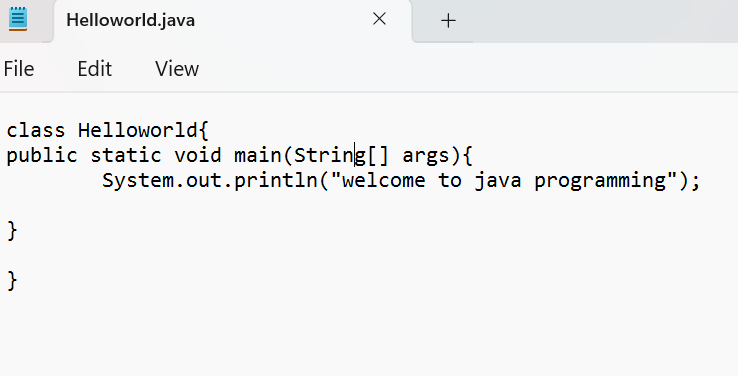


**Program2:** write a java program to print the message

‘welcome to java programming.’

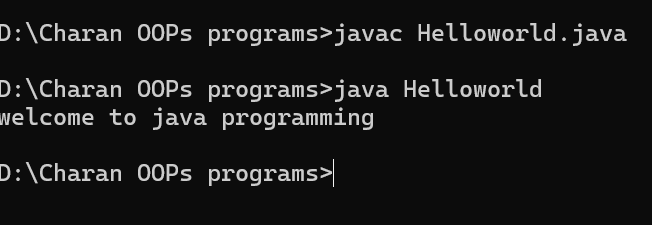
**Aim:** to print ‘welcome to java programming’

**Code:**

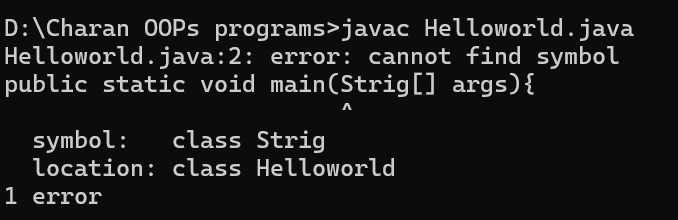


**Output:**

Positive case:



Negative case:



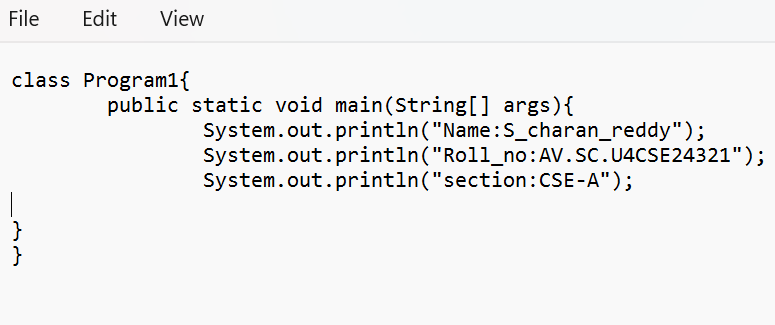
|  |  |  |
| --- | --- | --- |
| S.no | Error message | rectification |
| 1 | symbol: class Strig | symbol: class String |
| 2 | error: package system does not exist system.out.println("welcome to java programming"); | System.out.println(“welcome to java programming”); |

**Errors:**

**Program3:**

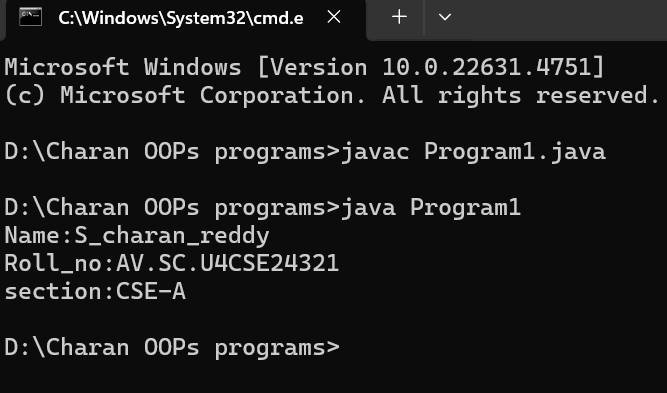
**Aim:** to print name, roll number and section

**Code:**

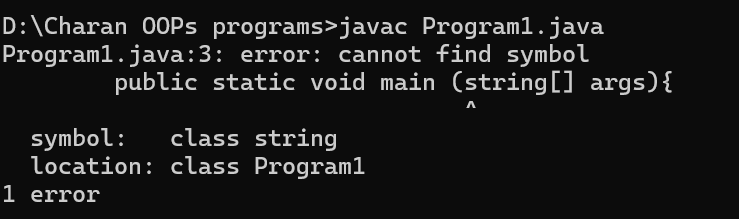


**Output:**

positive case:



Negative case:



**Error:**

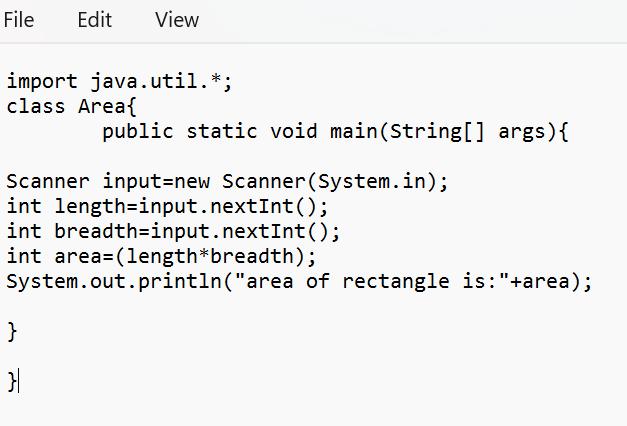
|  |  |  |
| --- | --- | --- |
| s.no | Error message | rectification |
| 1. | Symbol: class string | Symbol: class String |
| 2. | Error: Program1 (wrong name: program1) | class Program1 |

**WEEK2:**

**Program1:**

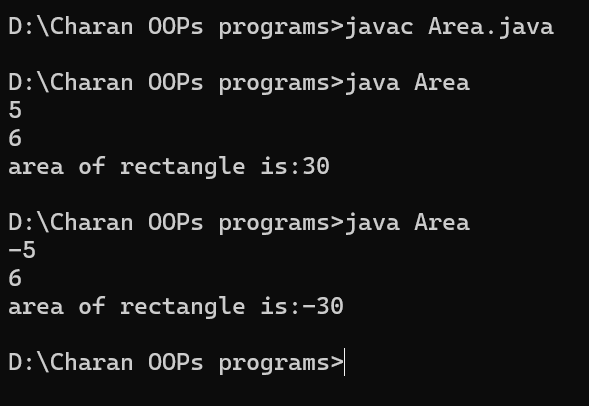
**Aim:** To find area of rectangle

**Code:**

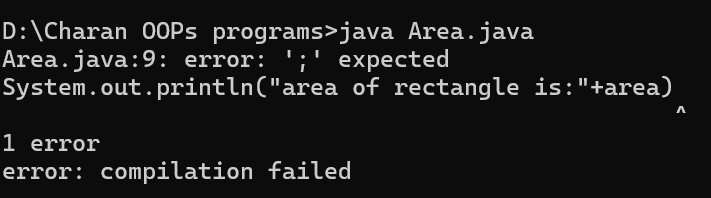


**Output:**

positive case:



Negative case:



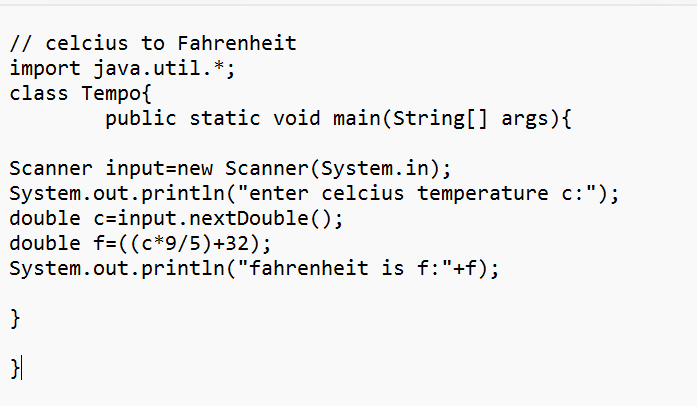
**Error:**

|  |  |  |
| --- | --- | --- |
| S.no | Error message | Rectification |
| 1 | Syntax error | ; |
| 2 | Symbol: class string | Symbol: class String |

**Program2:**

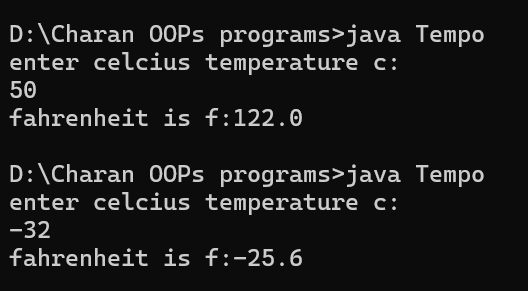
**Aim:** To convert Celsius to Fahrenheit

**Code:**

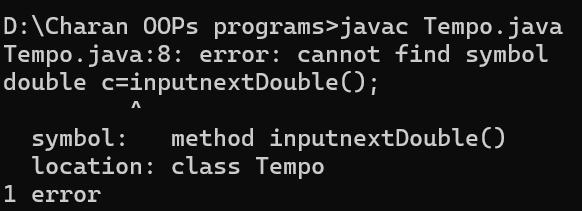


**Output:**

Positive case:



Negative case:



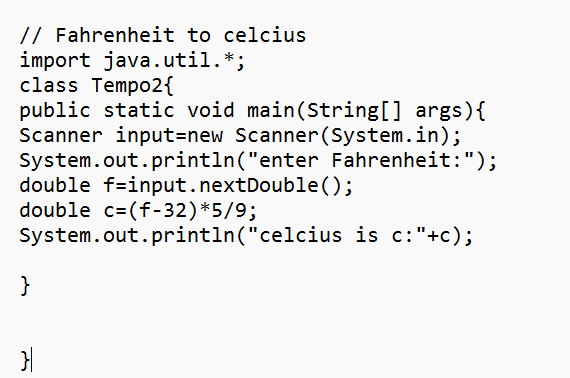
**Error**:

|  |  |  |
| --- | --- | --- |
| S.no | Error message | Rectification |
| 1 | Symbol: method inputnextDouble() | Input.nextDouble() |
| 2 | Syntax missing | ; |

**Program3:** write a java program to convert Fahrenheit to Celsius

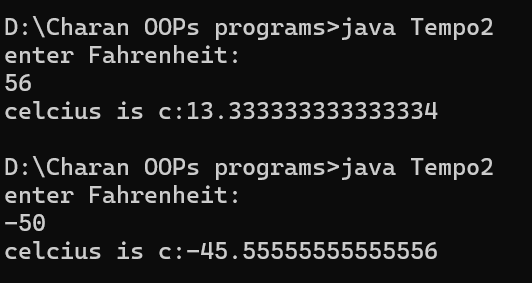
**Aim:** to convert Fahrenheit to Celsius

**Code:**

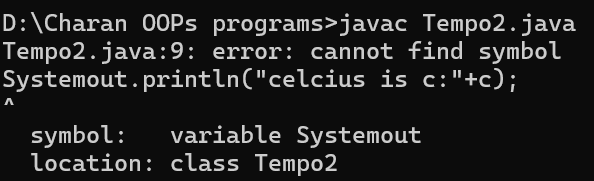


**Output:**

Positive case:



Negative case:

****

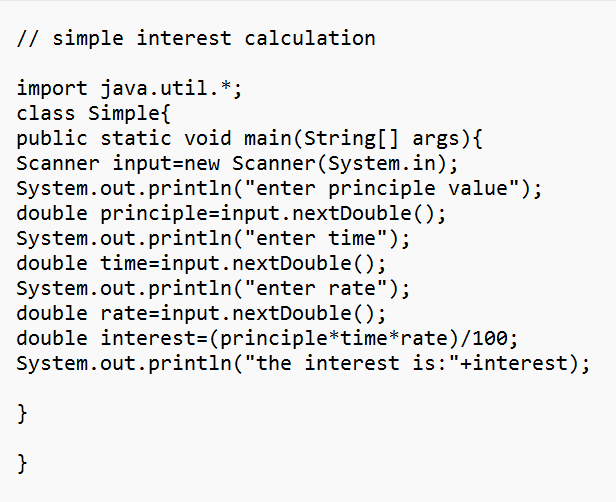
**Error:**

|  |  |  |
| --- | --- | --- |
| S.no | Error message | Rectification |
| 1 | Symbol: Systemout | System.out |
| 2 | package javautil does not exist  import javautil.\*; | Import java.util.\*; |

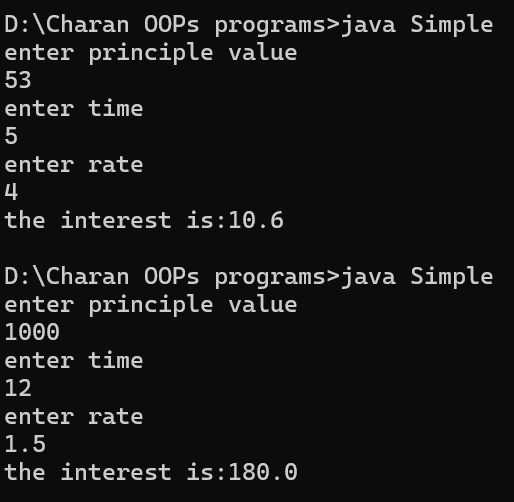
**Program4**: write a java program to calculate simple interest

**Aim:** to calculate simple interest

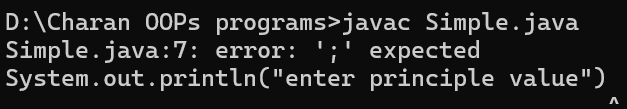
**Code:**



**Output: positive case:**



Negative case:



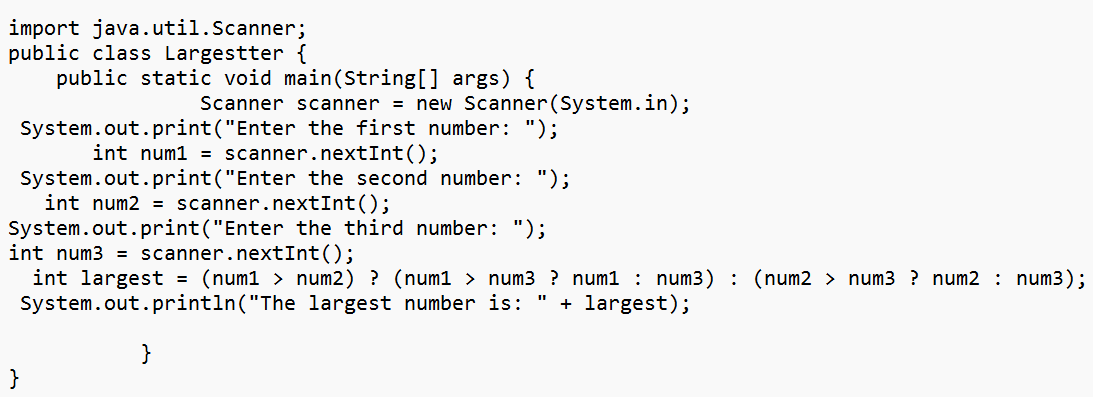
Error:

|  |  |  |
| --- | --- | --- |
| S.no | Error message | rectification |
| 1 | Syntax missing | ; |
| 2 | Symbol: principe | principle |

**Program5**:

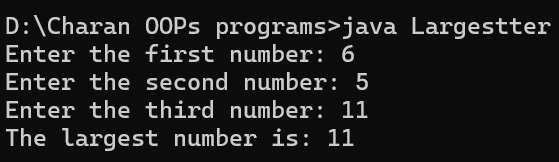
**Aim:** To find largest of three numbers using ternary operator

**Code:**

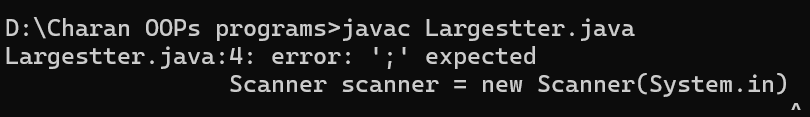
****

**Output:**

Positive case:



Negative case:



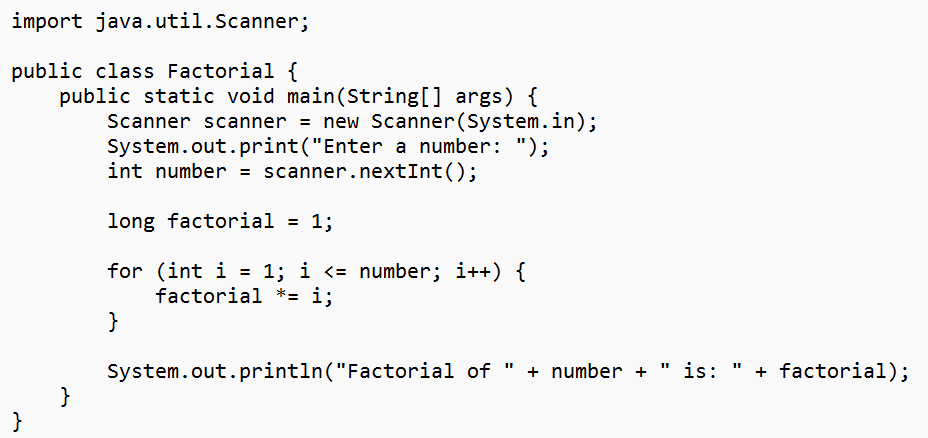
**Error:**

|  |  |  |
| --- | --- | --- |
| S.no | Error message | Rectification |
| 1 | Scanner scanner = new Scanner(System.in) | Scanner scanner = new Scanner(System.in); |
| 2 | Variable : num not found | num1 |

**Program 6:**

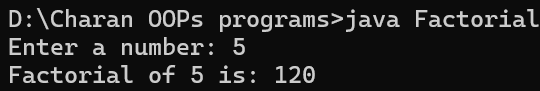
**Aim :** To find factorial of a number

**Code:**

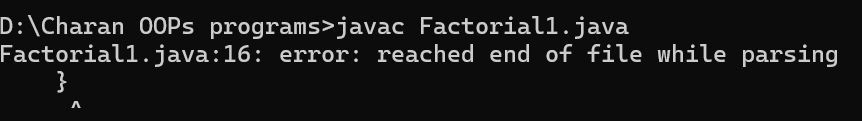
****

**Output :**

Positive case:



Negative case:



**Error :**

|  |  |  |
| --- | --- | --- |
| S.no | Error message | Rectification |
| 1 | error: reached end of file while parsing  } | } |
| 2 | Scanner not closed | Add read.close(); at the end of the program |

**Week 3:**

**Program 1:**

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

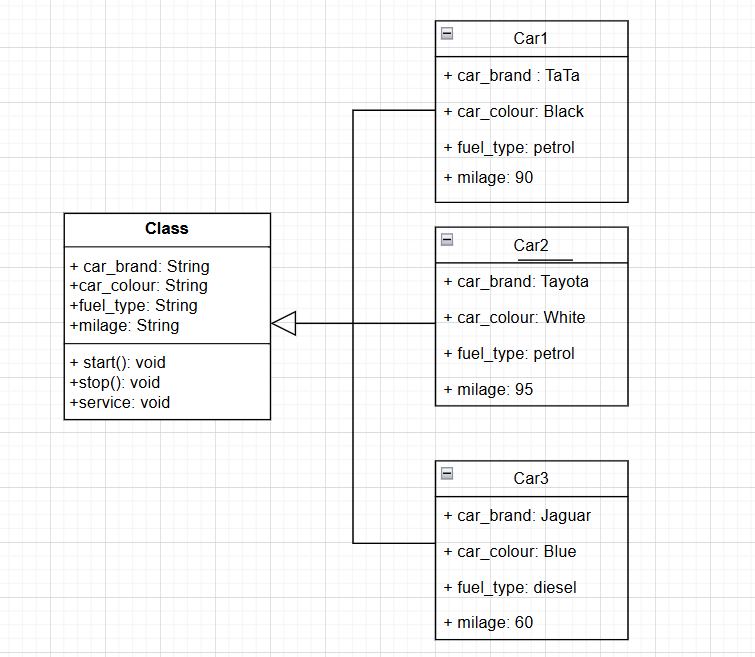
1.Create a class with name car

2. Create four attributes named car\_colour, Car\_brand ,fuel\_type, mileage

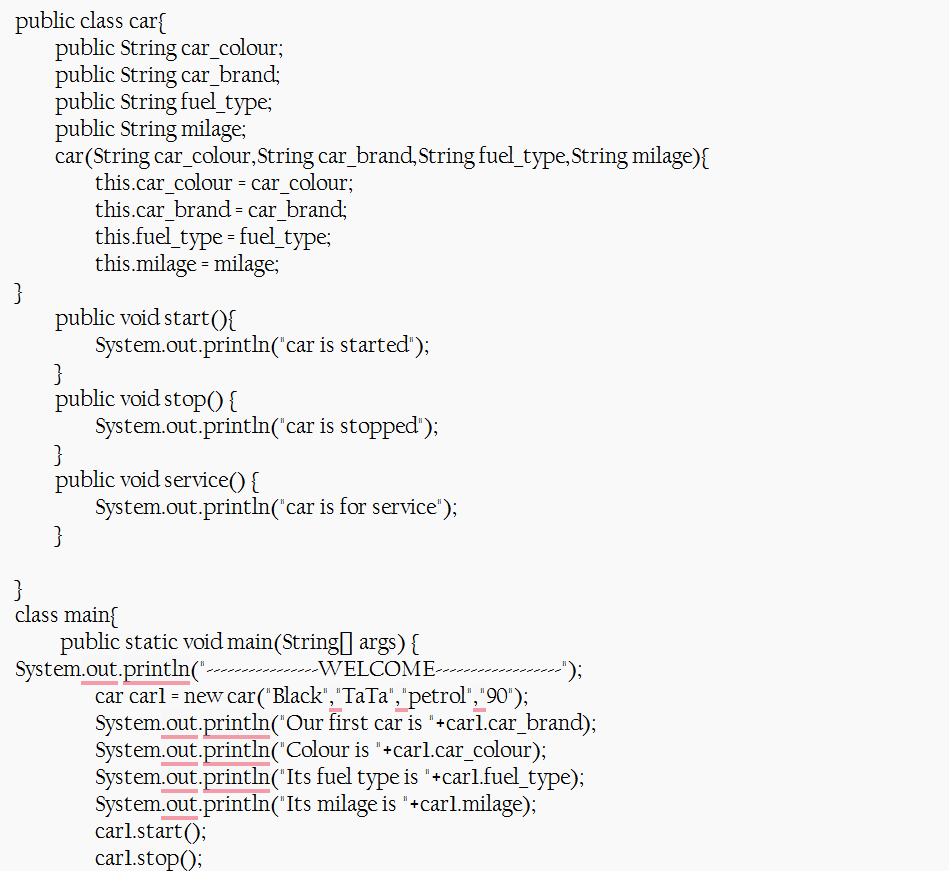
3. Create three methods named start(), stop(). Service()

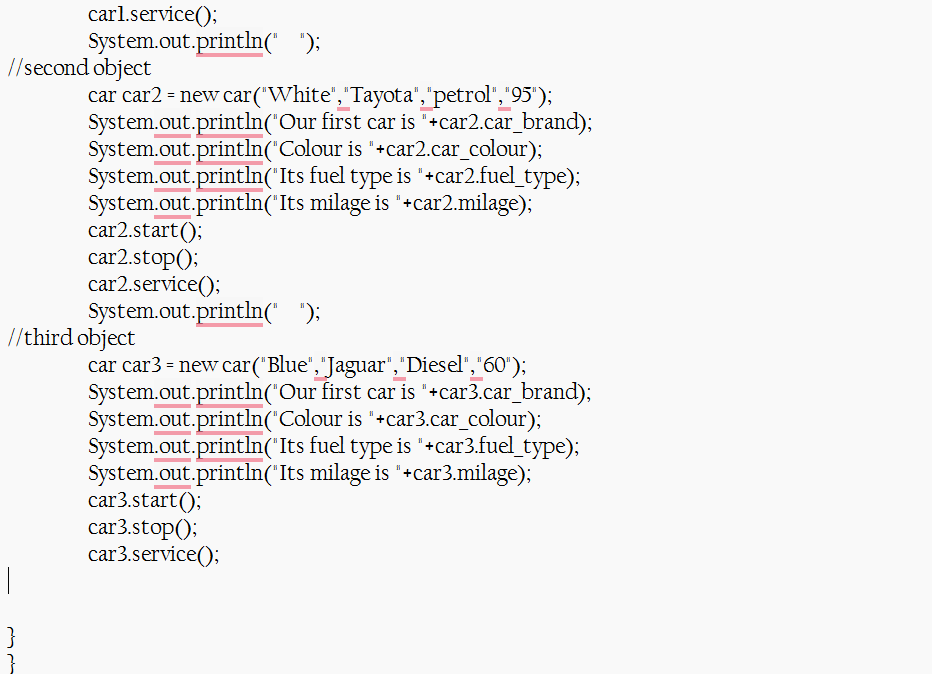
4. Create three objects named car1,car2 and car3

**Class diagram:**

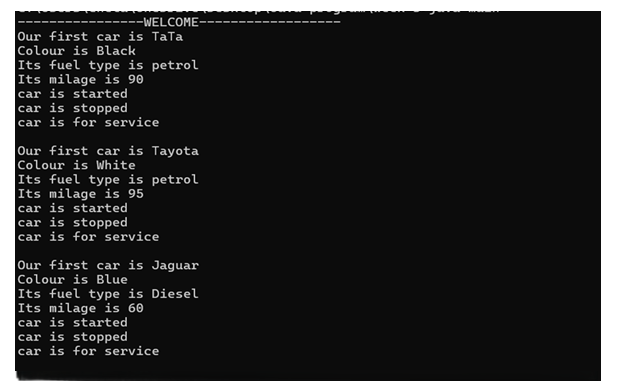


**Code:**





**Output:**

****

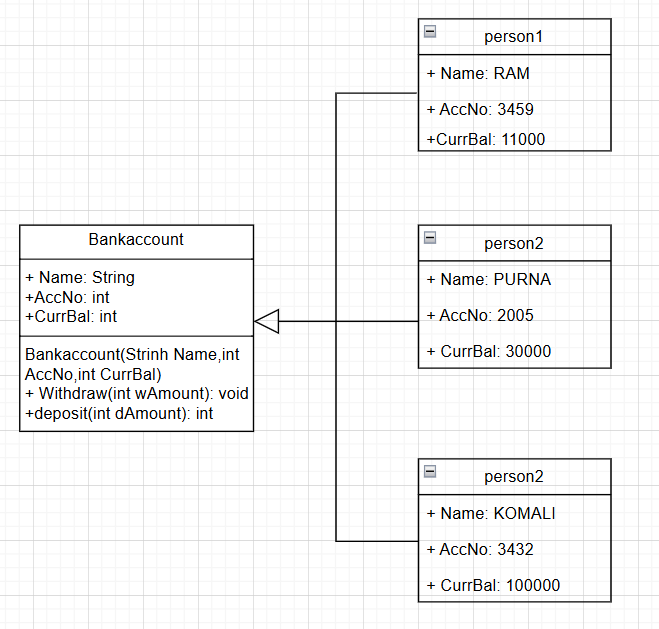
**Error:**

|  |  |  |
| --- | --- | --- |
| S.no | Error message | Rectification |
| 1 | class main{ | class Main{ |
| **2** | "Our first car is "+car2.car\_brand; | "Our first car is "+car2.car\_brand; |

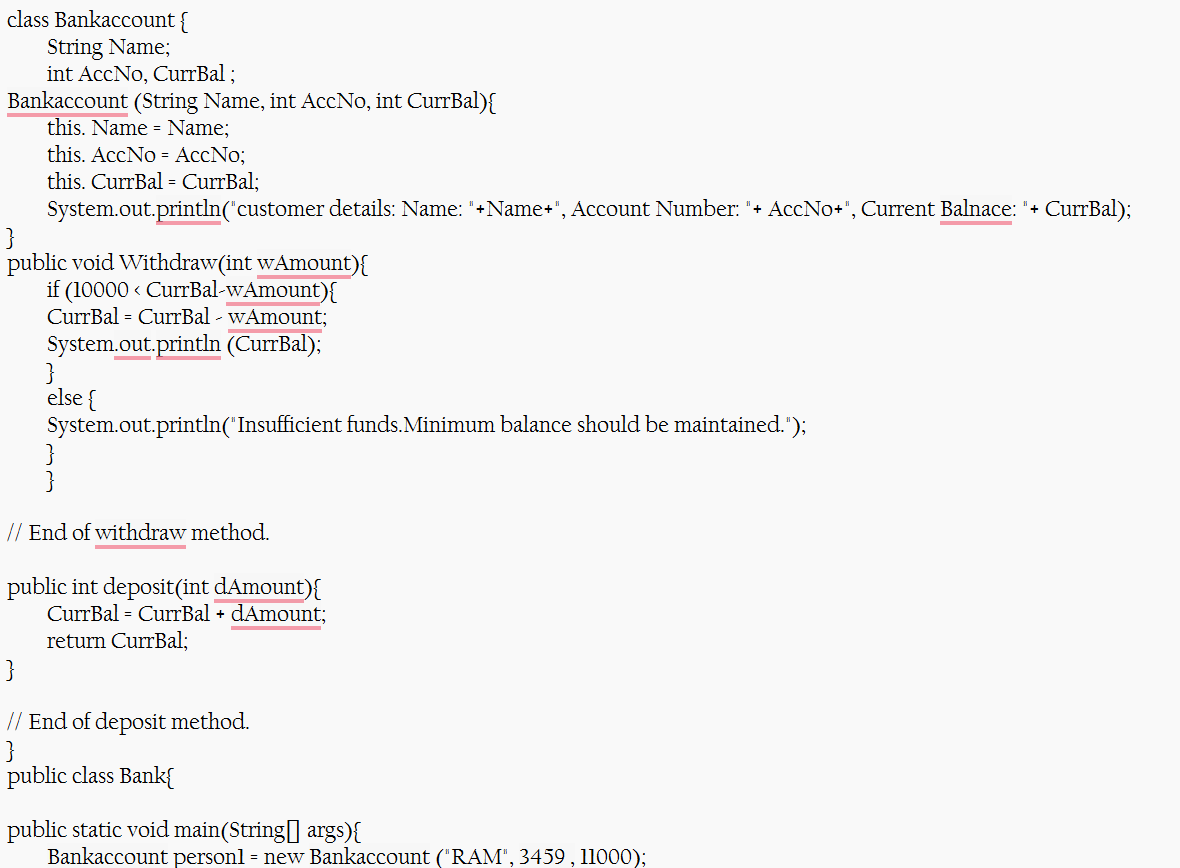
**Program 2:**

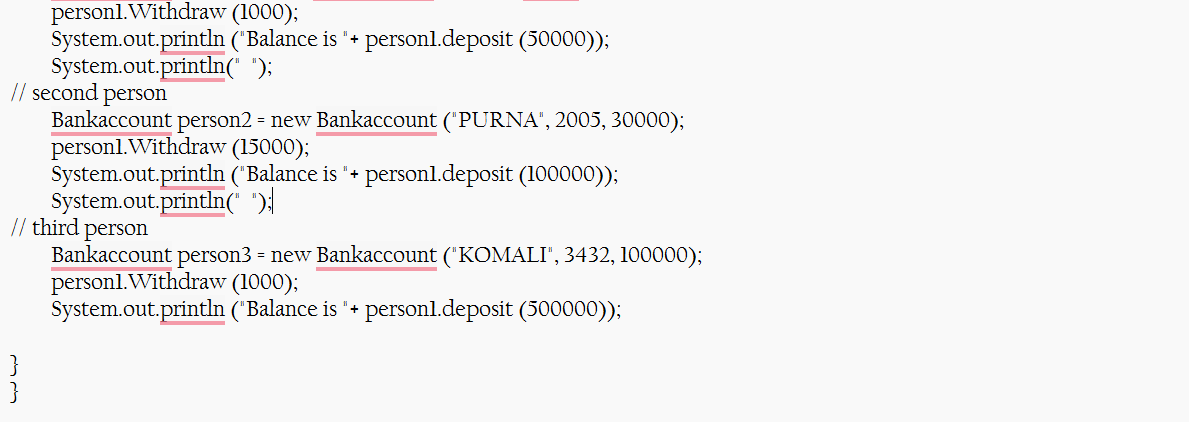
**Aim:** Create the java program to withdraw and deposit money in the bank account

**Class diagram:**

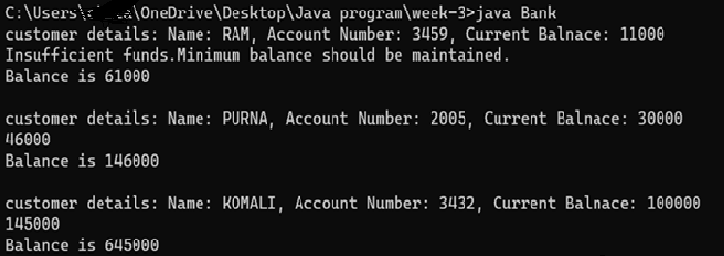


**Code:**





**Output:**

****

**Error:**

|  |  |  |
| --- | --- | --- |
| S.no | Error message | Rectification |
| 1 | class Bankaccount | class BankAccount (Java follows PascalCase for class names) |
| 2 | BankAccount person-1 (hyphen is not allowed) | BankAccount person1 |

**Week 4:**

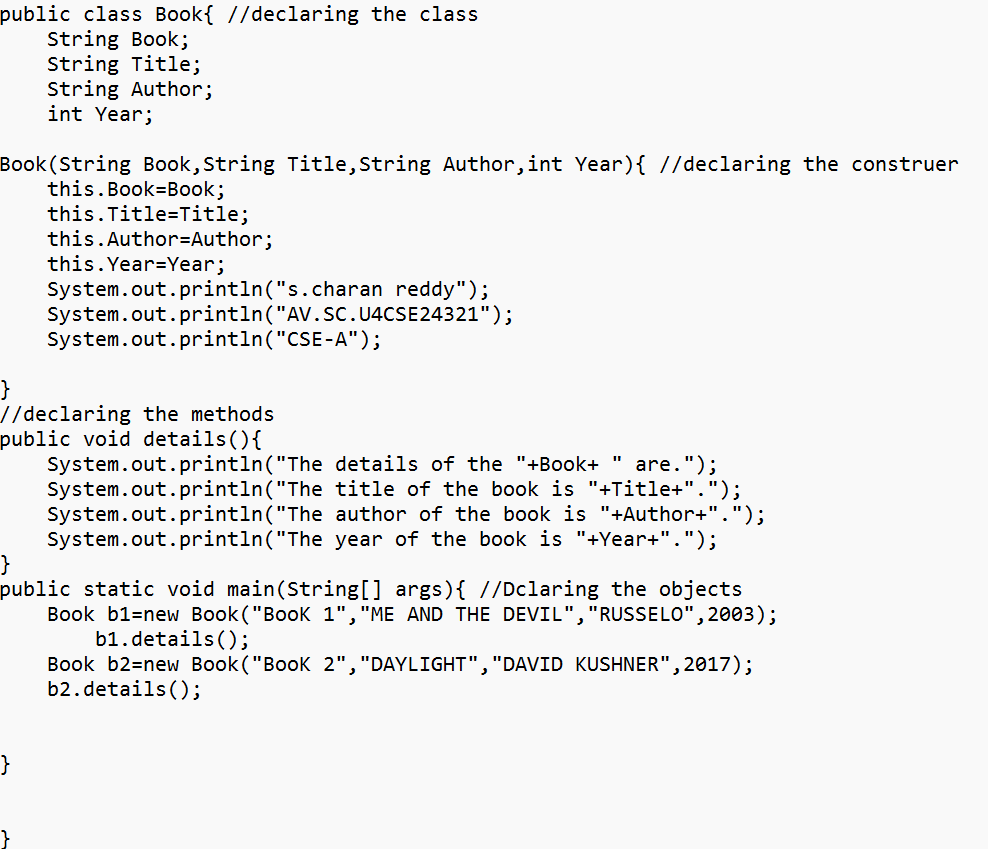
**Aim:** WRITE A JAVA PROGRAM WITH CLASS NAMED “Book”. THE CLASS SHOUKD CONTAIN VARIOUS ATTRIBUTES SUCH AS TITLE, AUTHOR, YEAR OF PUBLICATION. IT SHOULD ALSO CONTAIN A CONSTRUCTOR WITH PARAMETERS WHICH INITIALIZES TITLE, AUTHOR, YEAR OF PUBLICATION AND CREATE A METHOD WHICH DISPLAYS THE DETAILS OF 2 BOOKS.

PROGRAM:

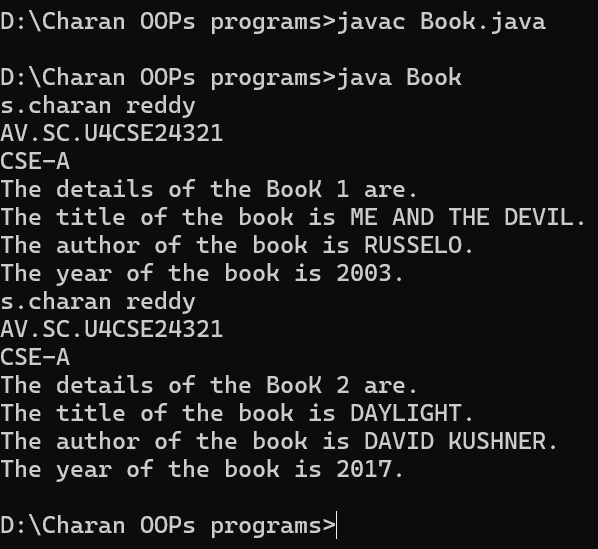
**CLASS DIAGRAM:**

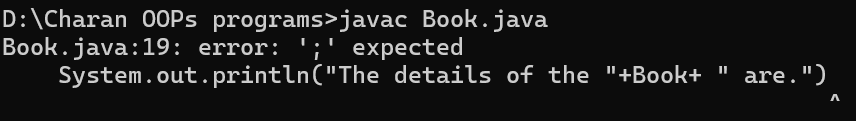
|  |
| --- |
| **Book** |
| **-title: String**  **-author: String**  **-year: int** |
| **+ Book(title: String, author:String, year: int) + displayDetails(): void** |

**Code :**

****

**Output:**

**Positive case:**

**Negative case:  
**

**Error :**

|  |  |  |
| --- | --- | --- |
| S.no | Error message | Rectification |
| 1 | System.out.println("The details of the "+Book+ " are.")  ^ | System.out.println("The details of the "+Book+ " are."); |
| 2 | Variable mistake: yea | year |

**Program 2:**

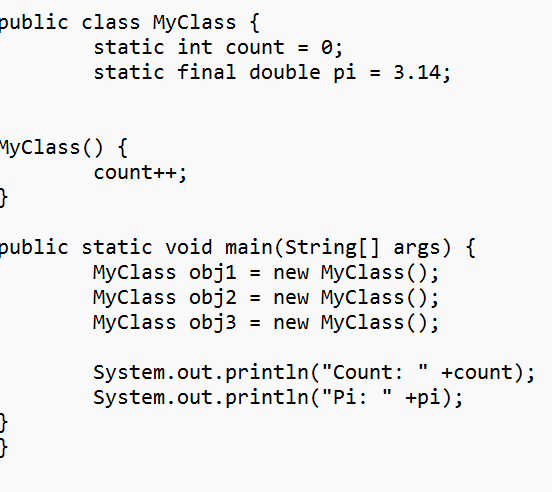
**Aim:**

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.

**Class diagram:**

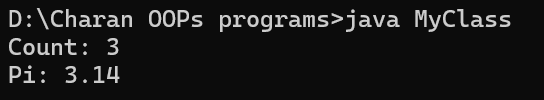
|  |
| --- |
| **MyClass** |
| **-count: int (static)**  **-pi: double (static, final)** |
| **+MyClass()**  **+main(args: String[]):void** |

**Code:**

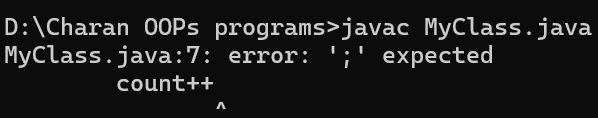
****

**Output:**

Positive case:

****

Negative case:



**Error:**

|  |  |  |
| --- | --- | --- |
| S.no | Error message | rectification |
| 1 | error: ';' expected  count++ | Count++; |
| **2** | : error: cannot find symbol  System.out.println("Pi: " +pi);  ^  symbol: variable pi | pi |