**23CSE111**

OOPS

**LAB MANUAL**

A logo with pink letters

Description automatically generated

**Department of CSE**

**Amrita School of Engineering**

**Amrita Vishwa Vidyapeetham, Amaravati Campus**

**Verified By Name: E. Hima Teja Goud**

**Roll No: 24048**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| S.NO | Programs | Date | Pg:No | Signature |
| 1 | 1. Download and Install Java Software. 2. Write a java program to print message “Welcome to java programming”. 3. Write a java program that prints name,roll number,section of a student. |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Week-1

* Program : 1
* Aim : Download and Install Java Software.
* Step 1 : Visit chrome and search “ java download”.And select Oracle website.

A screenshot of a computer

Description automatically generated

* Step 2 : Now open Oracle website scroll down and now select “JDK 21” for

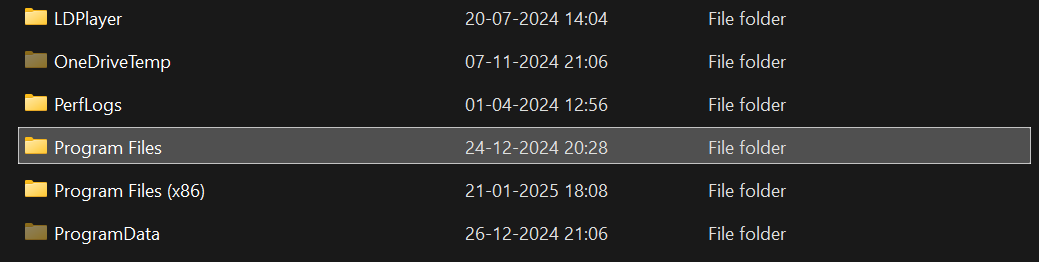
Windows and select “X64 installer” and download it.

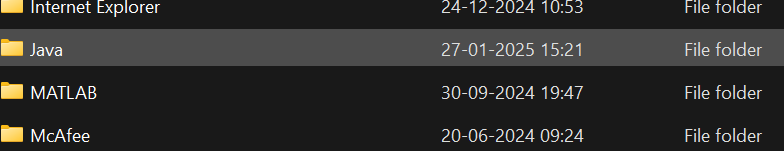
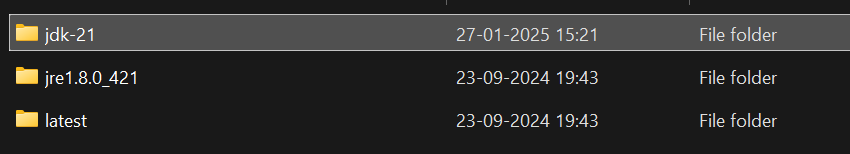
A screenshot of a computer

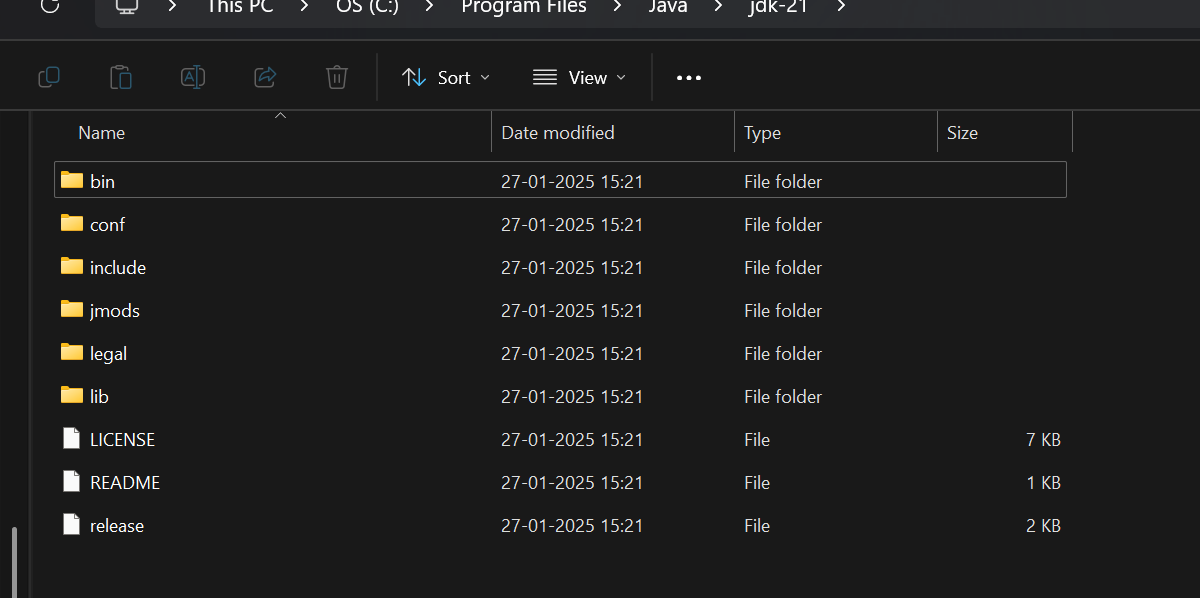
Description automatically generated

* Step 3 : After downloading open “this pc” in our laptop and open “program

files”,open “java”,open “JDK 21”



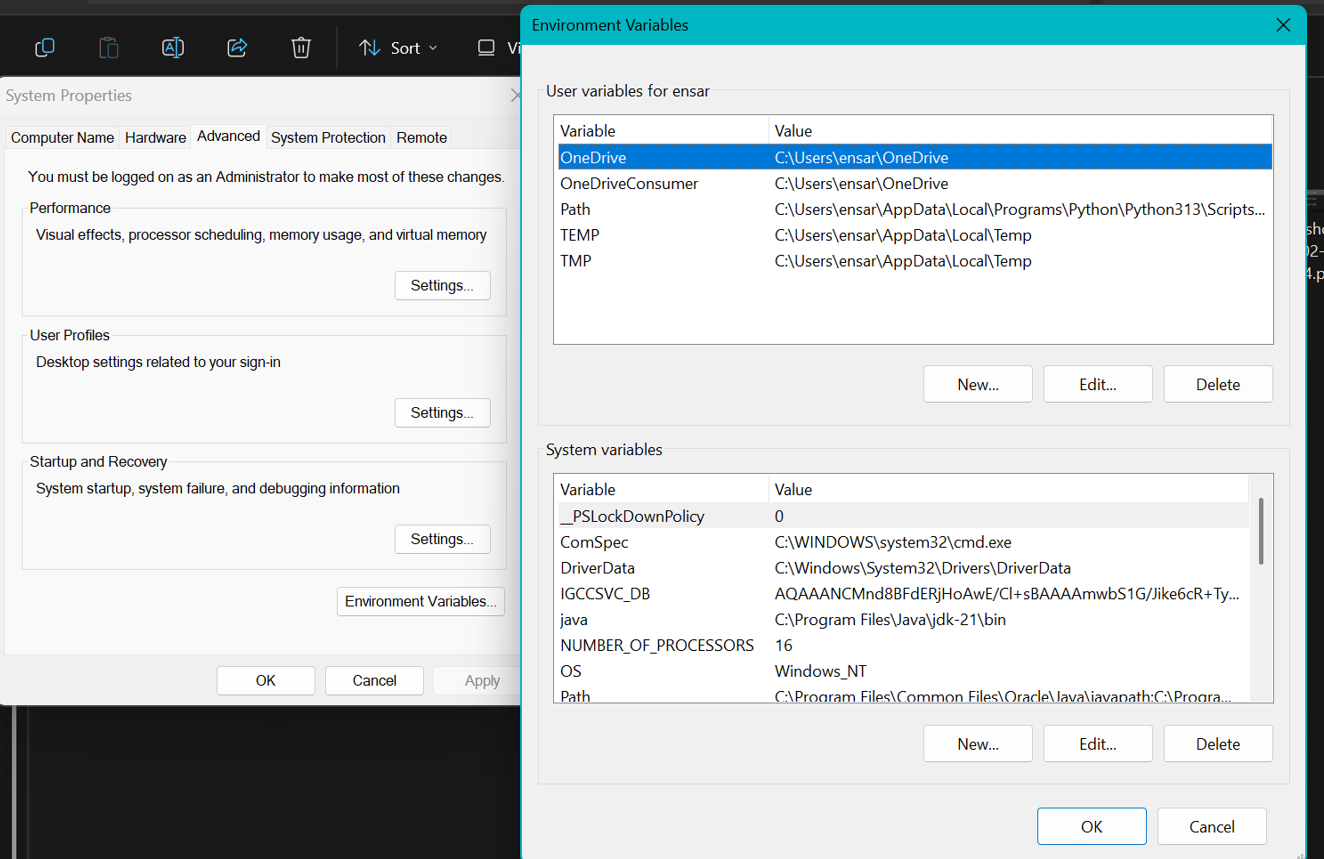
 



* Step 4 : In the task bar search and open “environment variables of system”,after opening environment variables, go to the system variables and see for java if there leave it. Or click path and add “JAVA” in ‘variable name’ and copy link in ‘variable value’

A screenshot of a computer

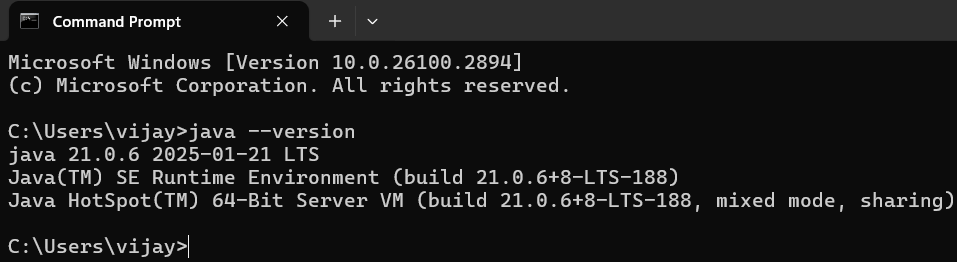
Description automatically generated



* Step 5 : Verifying Installation of Java. Again open task bar and search “cmd”,

open it ant type “java –version” and press enter. It will show the

version of installation of java.



Successfully Java is installed and it will show the version otherwise it will show error and command is not recognized.

* Program : 2

Q) Write a java program to print the message “welcome to java program”.

class Main{

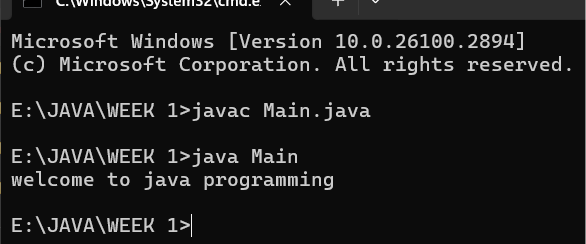
public static void main(String[] args){

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

}

}

OUTPUT



* Program : 3

Q) Write a java program that prints name,roll number,section of a student.

public class my\_profile{

public static void main(String[] arg){

System.out.println("name:E.Hima Teja Goud");

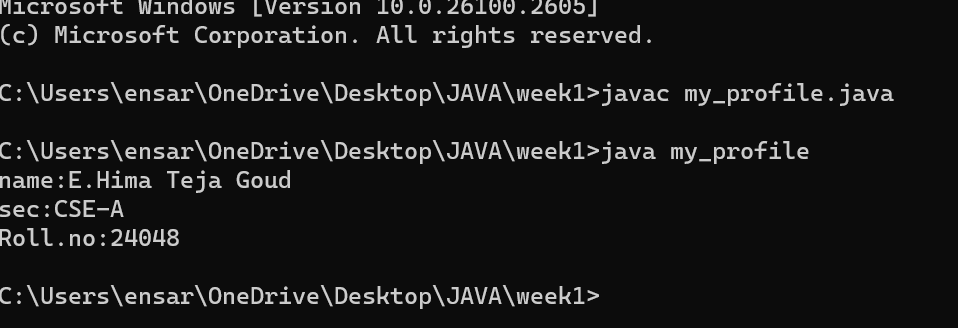
System.out.println("sec:CSE-A");

System.out.println("Roll.no:24048");

}

}

OUTPUT



Week-2

* Program : 1

import java.util.Scanner;

class rectangle{

public static void main(String[]args){

Scanner input=new Scanner(System.in);

System.out.println("enter the length");

int len=input.nextInt();

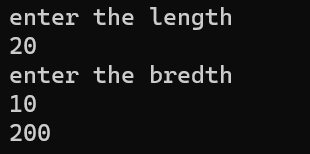
System.out.println("enter the bredth");

int bred=input.nextInt();

int area=len\*bred;

System.out.println(area);

}}

OUTPUT: 

|  |  |  |
| --- | --- | --- |
| s.no | EXPECTED ERROR | REASON |
| 1. | ; | **; is expected at end** |
| 2. | AREA | Declaration of int type variable |

* Program : 2

import java.util.Scanner;

class tem{

public static void main(String[]args){

Scanner input =new Scanner(System.in);

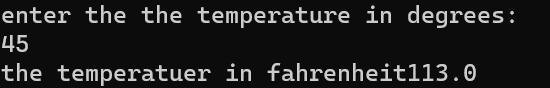
System.out.println("enter the the temperature in degrees:");

double deg=input.nextDouble();

System.out.println("the temperatuer in fahrenheit"+((deg\*9/5)+32));

}

}

OUTPUT: 

ERRORS:

|  |  |  |
| --- | --- | --- |
| S.No | EXPECTED ERROR | REASON |
| 1. | ; | ;expected at end |
| 2. | Input().close | The input is expected to closed |

* Program : 3

import java.util.Scanner;

class simpleintrest{

public static void main(String[]args){

Scanner input=new Scanner(System.in);

System.out.println("enter the p value");

int p=input.nextInt();

System.out.println("enter the t value");

int t=input.nextInt();

System.out.println("enter the r value");

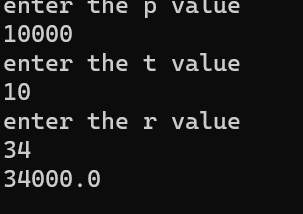
int r=input.nextInt();

float si=(p\*t\*r)/100;

System.out.println(si);

}

}

OUTPUT: 

|  |  |  |
| --- | --- | --- |
| S.No | EXPECTED ERROR | REASON |
| 1. | ; | ; is expected at end |
| 2. | Int t | **Without declaring t the compiler cannot execute the program.** |

* Program : 4

import java.util.Scanner;

class largest{

public static void main(String[]args){

Scanner input=new Scanner(System.in);

System.out.println("enter value of A");

int a=input.nextInt();

System.out.println("enter value of B");

int b=input.nextInt();

System.out.println("enter value of C");

int c=input.nextInt();

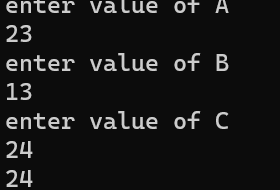
int large=(a>b)?((a>c)?a:c):((b>c)?b:c);

System.out.println(large);

}

}

OUTPUT:



ERRORS:

|  |  |  |
| --- | --- | --- |
| S.No | EXPECTED ERROR | REASON |
| 1. | ? | Checks the condition |
| 2. | : | Comparing between two variables |

* Program : 5

import java.util.Scanner;

class factorial{

public static void main(String[]args){

Scanner input=new Scanner(System.in);

System.out.println("enter the number to find its factorial");

int n=input.nextInt();

int sum=1;

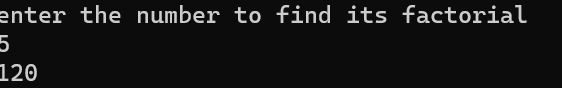
for(int i=1;i<=n;i++){

sum=sum\*i;}

System.out.println(sum);

}

}

OUTPUT: 

ERRORS:

|  |  |  |
| --- | --- | --- |
| S.No | EXPECTEED ERRORS | REASON |
| 1. | } | To close for loop |
| 2. | ; | ; expected |

WEEK-3

* Program : 1

Q) **Write a java program with the following instructions**.

1. Create a class with name car.
2. Create four attributes named car\_colour,car\_brand,fuel\_type,top\_speed.
3. Create three method named “Start\_Racing”,”End\_Race”.{ }
4. Create three objects named Car1,Car2,Car3.
5. Create a constructor which should print “Welcome to Garage”.

Class Diagram:

|  |
| --- |
| **Car** |
| * carColor: String |
| * carBrand: String |
| * fuelType: String |
| * topSpeed: int |
| + Car(String,String,String,int) |
| + startRacing() |
| + endRace() |

// Car.java

public class Car {

// Attributes

private String carColour;

private String carBrand;

private String fuelType;

private int topSpeed;

// Constructor

public Car(String carColour, String carBrand, String fuelType, int topSpeed) {

this.carColour = carColour;

this.carBrand = carBrand;

this.fuelType = fuelType;

this.topSpeed = topSpeed;

System.out.println("Welcome to car garage");

}

// Method to start racing

public void startRacing() {

System.out.println(carBrand + " (" + carColour + ") is starting the race with a top speed of " + topSpeed + " km/h and runs on " + fuelType + "!");

}

// Method to end race

public void endRace() {

System.out.println(carBrand + " (" + carColour + ") has finished the race!");

}

// Main method to create objects and demonstrate functionality

public static void main(String[] args) {

// Creating three objects

Car car1 = new Car("Red", "Ferrari", "Petrol", 200);

Car car2 = new Car("Blue", "Tesla", "Electric", 250);

Car car3 = new Car("Black", "BMW", "Diesel", 220);

// Starting and ending races

car1.startRacing();

car1.endRace();

car2.startRacing();

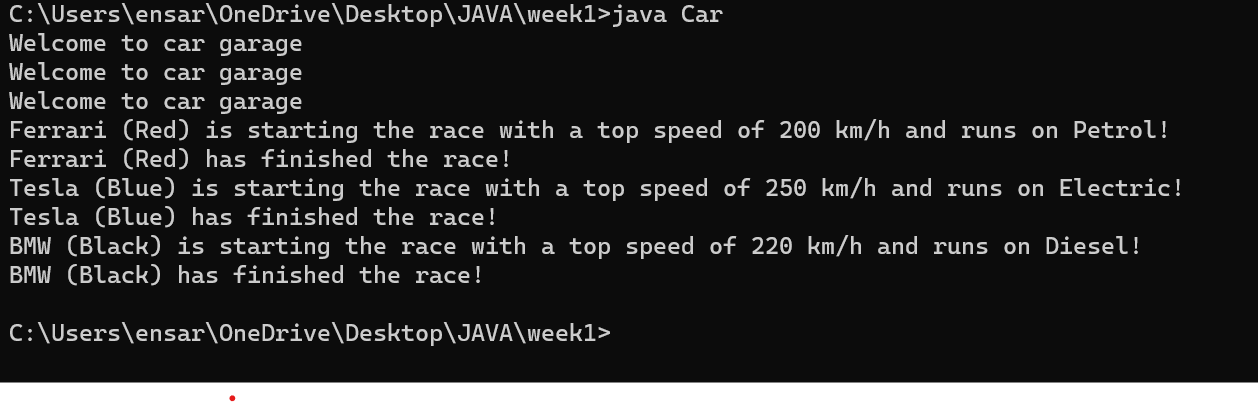
car2.endRace();

car3.startRacing();

car3.endRace();

}

}



|  |  |  |
| --- | --- | --- |
| **S.NO** | **Errors** | **Rectification** |
| 1 | } | To close for loop |
| 2 | System.out.print(); | If we place the print statement inside the for loop it will print the each i value everytime but to print only the final value we must place it outside the for loop. |

* Program : 2

Q ) Write a class by writing java program named Bank Account

with two methods “deposits and withdraw”.

a) In deposit method whenever an amount is deposited it

has to be updated with current amount (logic C.A+D.A).

b) With draw amount whenever an amount is being

withdraw it has to be less than the current amount less

than the amount else print “Insufficient funds”.

Class Diagram:

|  |
| --- |
| Bank Account |
| * currentAmount: double |
| + BankAccount(initialAmount:double) |
| + deposit(amount: double):void |
| + withdraw(amount: double):void |
| + getCurrentAmount():double |

import java.util.Scanner;

class BankAccount {

String name;

int accountNumber;

int currentBalance;

// Constructor to initialize the bank account

BankAccount(String name, int accountNumber, int currentBalance) {

this.name = name;

this.accountNumber = accountNumber;

this.currentBalance = currentBalance;

System.out.println("Customer Details: " + name + ", Account Number: " + accountNumber + ", Current Balance: " + currentBalance);

}

// Method to withdraw an amount

public void withdraw(int withdrawAmount) {

if (withdrawAmount <= currentBalance) {

currentBalance -= withdrawAmount;

System.out.println("Withdrawn: " + withdrawAmount);

System.out.println("Current Balance: " + currentBalance);

} else {

System.out.println("Insufficient Funds");

}

}

// Method to deposit an amount

public int deposit(int depositAmount) {

currentBalance += depositAmount;

System.out.println("Deposited: " + depositAmount);

return currentBalance;

}

// Main method to run the program

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

// Input for account details

System.out.print("Enter your name: ");

String name = scanner.nextLine();

System.out.print("Enter your account number: ");

int accountNumber = scanner.nextInt();

System.out.print("Enter your initial balance: ");

int initialBalance = scanner.nextInt();

// Create a new bank account

BankAccount account = new BankAccount(name, accountNumber, initialBalance);

// Input for withdrawal and deposit

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

int withdrawAmount = scanner.nextInt();

account.withdraw(withdrawAmount);

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

int depositAmount = scanner.nextInt();

account.deposit(depositAmount);

// Final balance

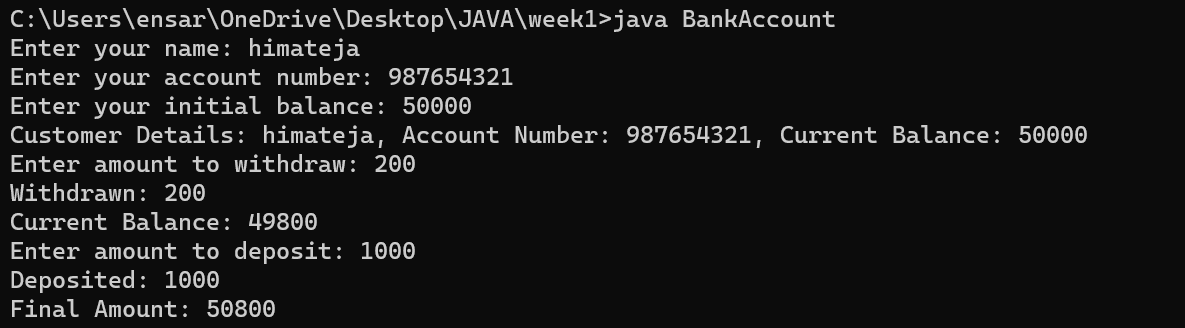
System.out.println("Final Amount: " + account.currentBalance);

// Close the scanner

scanner.close();

}

}



|  |  |  |
| --- | --- | --- |
| **S.NO** | **Errors** | **Rectification** |
| 1 | ; | ; is expected at end |
| 2 | Int t | Without declaring the compiler cannot execute the  program. |

**WEEK – 4**

1. **Write a java program with class named “Book”. The class should contain various attributes such as**

**“Title of the book , author , year of publication “. It should also contain a constructor with parameters**

**which initializes “ Title of the book, author, year of publication”. Create a method which displays the**

**details of the book. i.e. “ Title of the book, author and year of publication”. Display the details of two**

**books by creating two objects.**

**Code :**

**class Book {**

**// beginning of the class Book**

**public String title; // Changed Title to title for consistency**

**private String author;**

**public int yearOfPublication;**

**// beginning of constructor**

**Book(String title, String author, int yearOfPublication) {**

**this.title = title; // Changed Title to title for consistency**

**this.author = author;**

**this.yearOfPublication = yearOfPublication;**

**}**

**// constructor ends here**

**// method display starts here**

**public void display() {**

**System.out.println("Title of the book is: " + title +**

**", The name of the author is: " + author +**

**", The year of publication is: " + yearOfPublication);**

**}**

**// method display ends here**

**// creating objects**

**public static void main(String[] args) {**

**Book book1 = new Book("Harry Potter", "J.K. Rowling", 1993);**

**Book book2 = new Book("Someone Like You", "Nikitha Singh", 2010);**

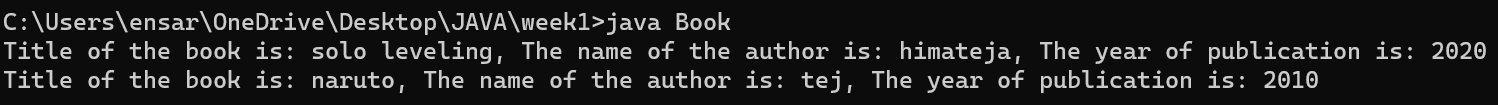
**book1.display();**

**book2.display();**

**}**

**}**

**// class ends here**

Output: 

**Errors :**

|  |  |  |
| --- | --- | --- |
| **s.no** | **Expected error** | **reason** |
| **1.** | **Setting the parameters inside the constructer** | **We cannot pass the values inside constructor without setting them first** |
| **2.** | **}** | **Ending the class and main method is required** |

1. b). **To 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 3.1415 as attributes of that class**

**Now, define a constructor for “Myclass” that increments the “Count” variable each that an object of**

**Myclass is created. Finally , print the final values of “Count” and “pi” variables .**

**Code:**

class Myclass{

// class starts here

static int Count = 0;

final double pi = 3.1415;

// the constructor starts here

Myclass(){

Count++;

}

// the constructor ends here

public static void main(String[] args){

Myclass c1 = new Myclass();

Myclass c2 = new Myclass();

System.out.println("Count : " + c1.Count);

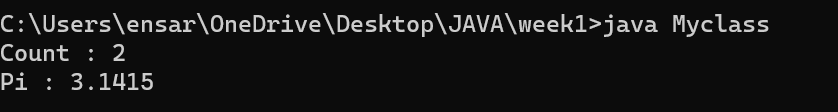
System.out.println("Pi : " + c1.pi);

}

}

// class ends here

Output:



**Errors :**

|  |  |  |
| --- | --- | --- |
| **S.No.** | **Expected Error** | **Reason** |
| **1** | **.variable** | **We must mention variable name to call the variable** |
| **2** | **static** | **Static variables contain only one value** |