**LAB 1**

**Q.**

Develop a Java program that prints all real solutions to the quadratic equation

ax2 +bx+c= 0. Read in a, b, c and use the quadratic formula. If the discriminate b2-4ac is negative, display a message stating that there are no real solutions.

**A.**

import java.util.Scanner; class QuadraticEquations

{

public static void main(String args[])

{

double a,b,c,D; double r1,r2;

double real\_part,imaginary\_part;

Scanner in = new Scanner(System.in);

System.out.println("Enter the constants a,b and c of the quadratic equation a(x)^2+b(x)+c=0 : ");

a = in.nextDouble(); b = in.nextDouble(); c = in.nextDouble();

System.out.println("\nInput Quadration Equation : "+a+"(x)^2 + "+b+"(x) +

"+c+" = 0");

D = (b\*b)-(4\*a\*c);

if(D>0)

{

System.out.println("\nRoots are real and unequal since Discriminant =

"+D);

r1 = (-b + Math.sqrt(D))/(2\*a); r2 = (-b - Math.sqrt(D))/(2\*a);

System.out.println("\nRoots of the Quadratic Equation are:\nRoot 1 = "+r1+"\nRoot 2 = "+r2);

}

else if(D==0)

{

System.out.println("\nRoots are real and equal since Discriminant =

"+D);

r1 = r2 = (-b)/(2\*a);

System.out.println("\nRoots of the Quadratic Equation are:\nRoot 1 =

Root 2 = "+r1);

}

else

{

System.out.println("\nRoots are unreal since Discriminant = "+D);

real\_part = (-b)/(2\*a);

imaginary\_part = (Math.sqrt(-D))/(2\*a);

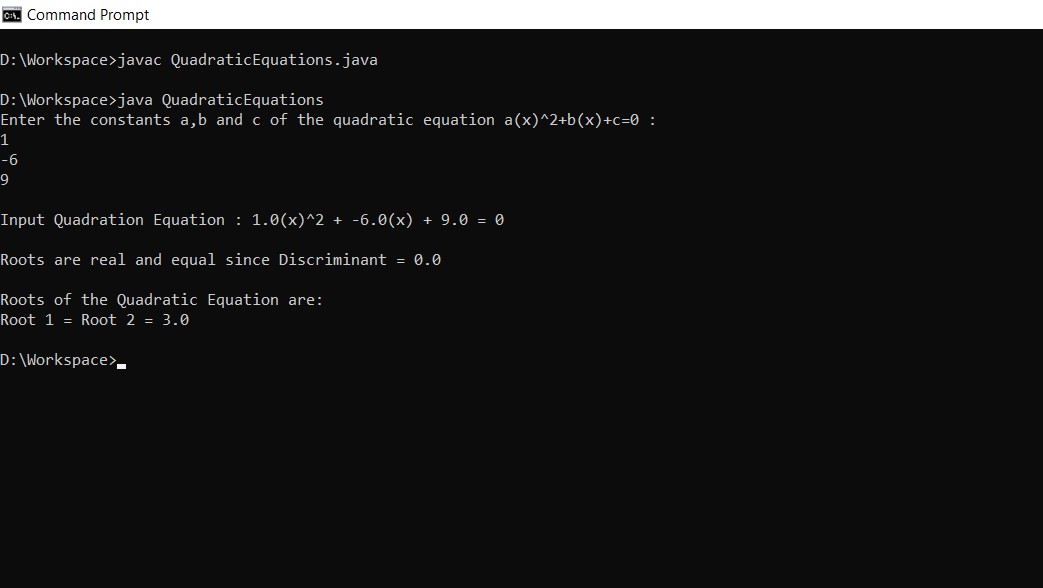
System.out.println("\nRoots of the Quadratic Equation are:\nRoot 1 = "+real\_part+"+"+imaginary\_part+"(i)\nRoot 2 = "+real\_part+"-"+imaginary\_part+"(i)");

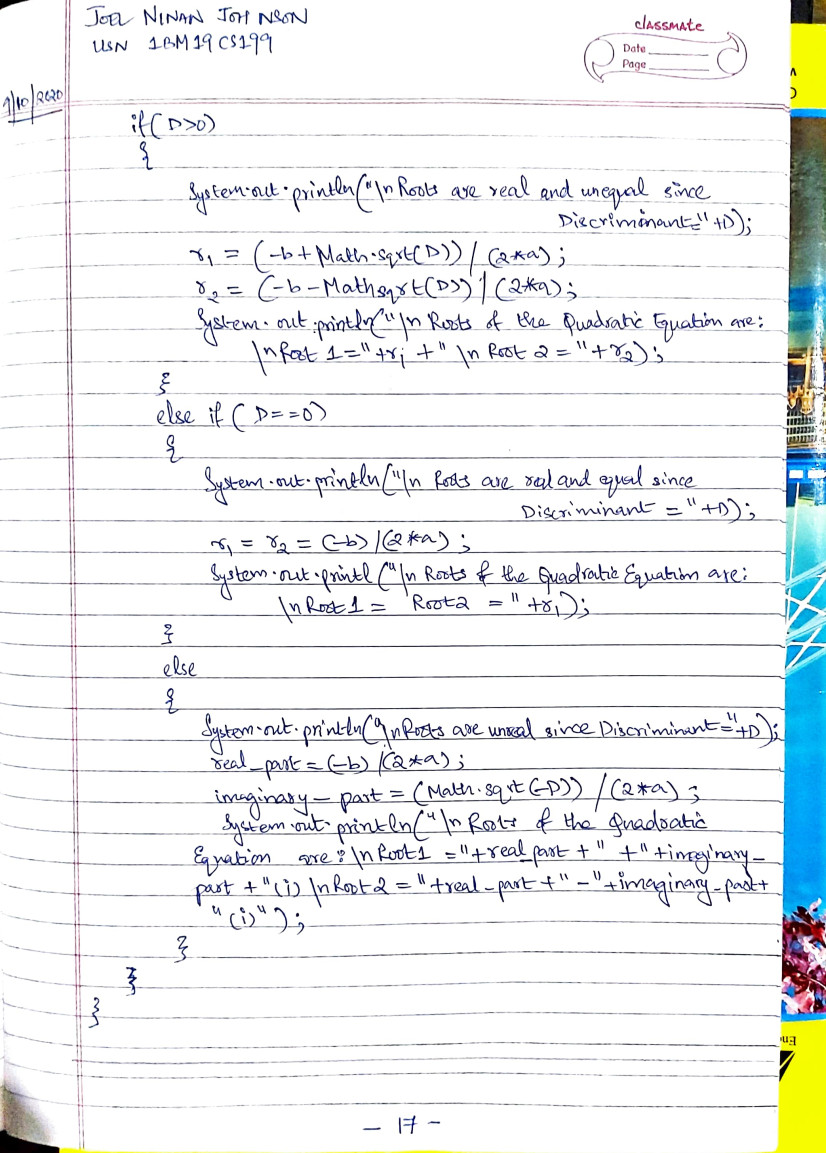
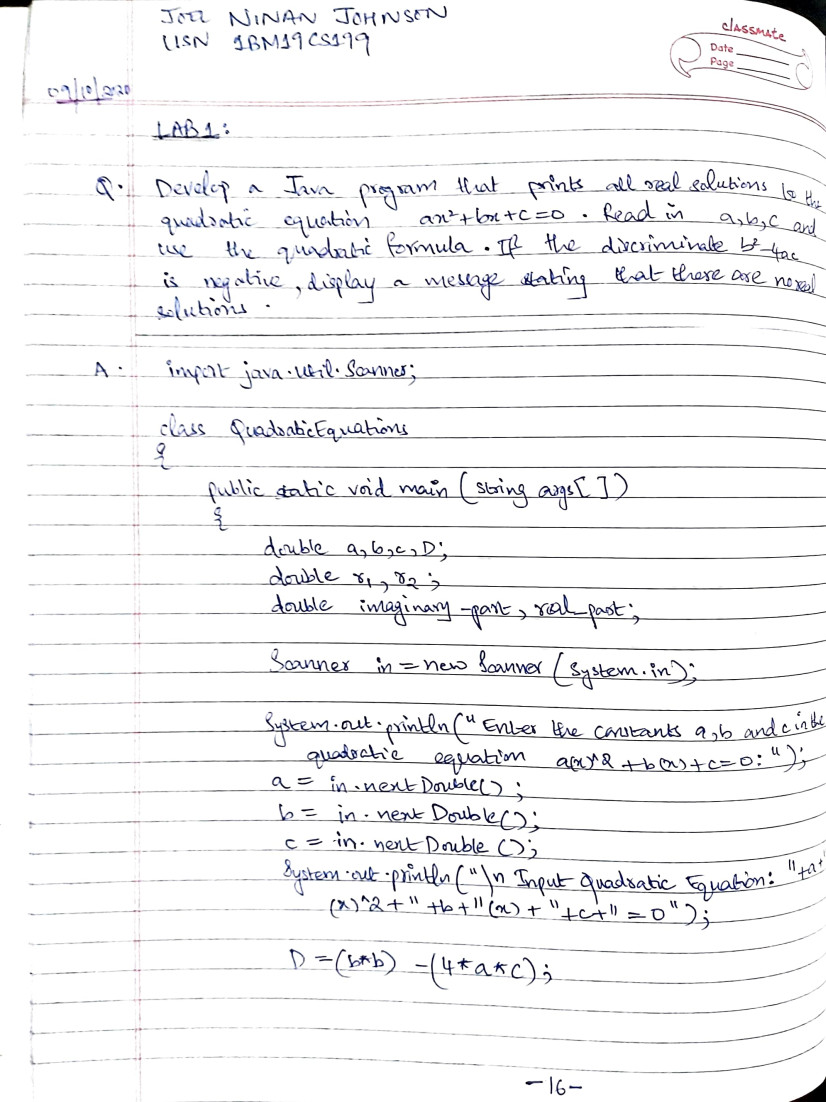
}

}

}

**OUTPUT**





**LAB PROGRAM 2**

**Q.** Develop a Java program to create a class Student with members usn, name, an array credits and an array marks. Include methods to accept and display details and a method to calculate SGPA of a student.

**PROGRAM:**

import java.util.Scanner; class Student

{

String name; String usn; int credits[] = new int[5]; int gradepoints[] = new int[5]; double marks[] = new double[5]; int i; int total\_credits=0; int sgpa=0;

Scanner in = new Scanner(System.in);

void getdetails()

{

System.out.println("Enter the Name of Student: ");

name = in.nextLine();

System.out.println("Enter the USN of Student: "); usn = in.nextLine();

System.out.println("Enter the Credits in the subjects: "); for(i=0;i<5;i++)

{

credits[i] = in.nextInt(); total\_credits = total\_credits + credits[i];

}

System.out.println("Enter the Marks in the subjects(0-

100): ");

for(i=0;i<5;i++)

{

marks[i] = in.nextDouble();

}

}

void displaydetails()

{

System.out.println("Student Name: "+name); System.out.println("Student USN: "+usn); for(i=0;i<5;i++)

{

System.out.println("Subject "+(i+1)+" --> Marks

scored: "+marks[i]+" Credit: "+credits[i]);

}

System.out.println("SGPA: "+(double)sgpa/total\_credits);

}

void sgpa()

{

for(i=0;i<5;i++)

{

if(marks[i]>=91 && marks[i]<=100)

{

gradepoints[i] = 10;

}

else if(marks[i]>=81)

{

gradepoints[i] = 9;

}

else if(marks[i]>=71)

{

gradepoints[i] = 8;

}

else if(marks[i]>=61)

{

gradepoints[i] = 7;

}

else if(marks[i]>=51)

{

gradepoints[i] = 6;

}

else if(marks[i]>=41)

{

gradepoints[i] = 5;

}

else if(marks[i]>0 && marks[i]<40)

{

gradepoints[i] = 0;

}

sgpa = sgpa + (gradepoints[i]\*credits[i]);

}

}

}

class Lab2

{

public static void main(String args[])

{

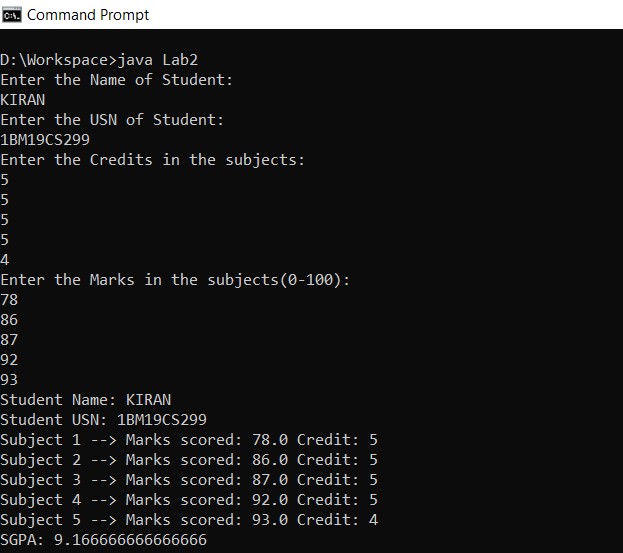
Student std1 = new Student(); std1.getdetails();

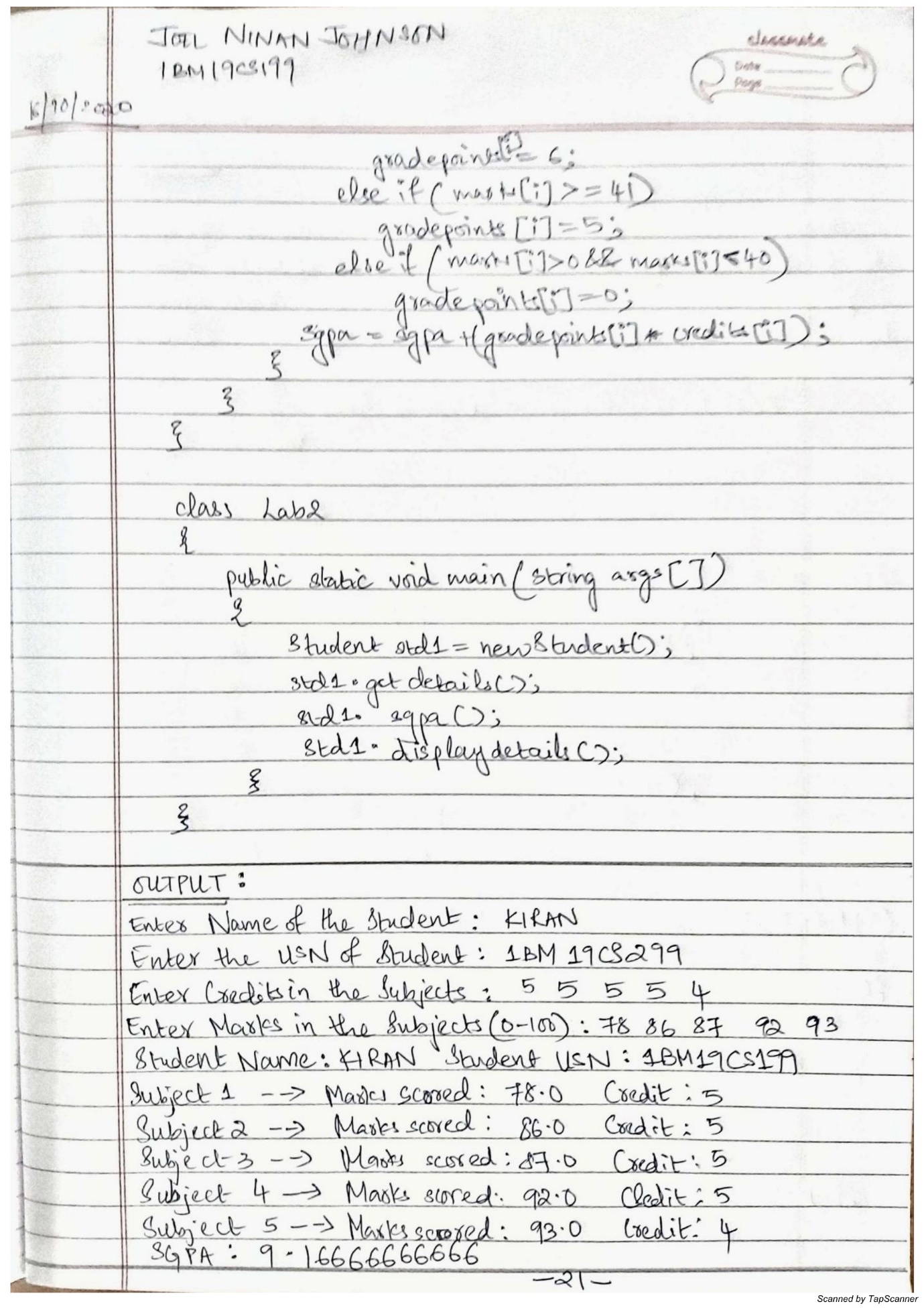
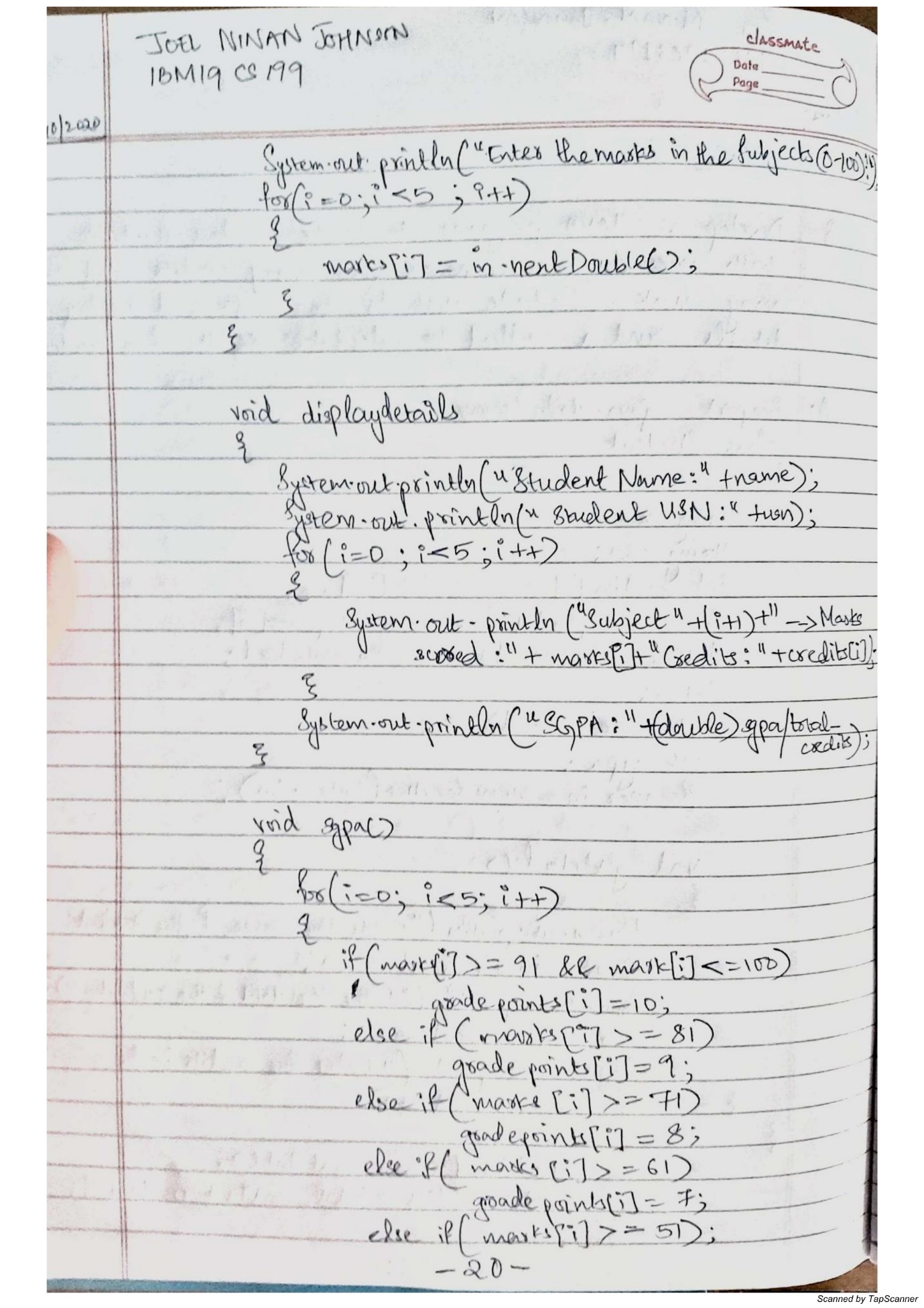
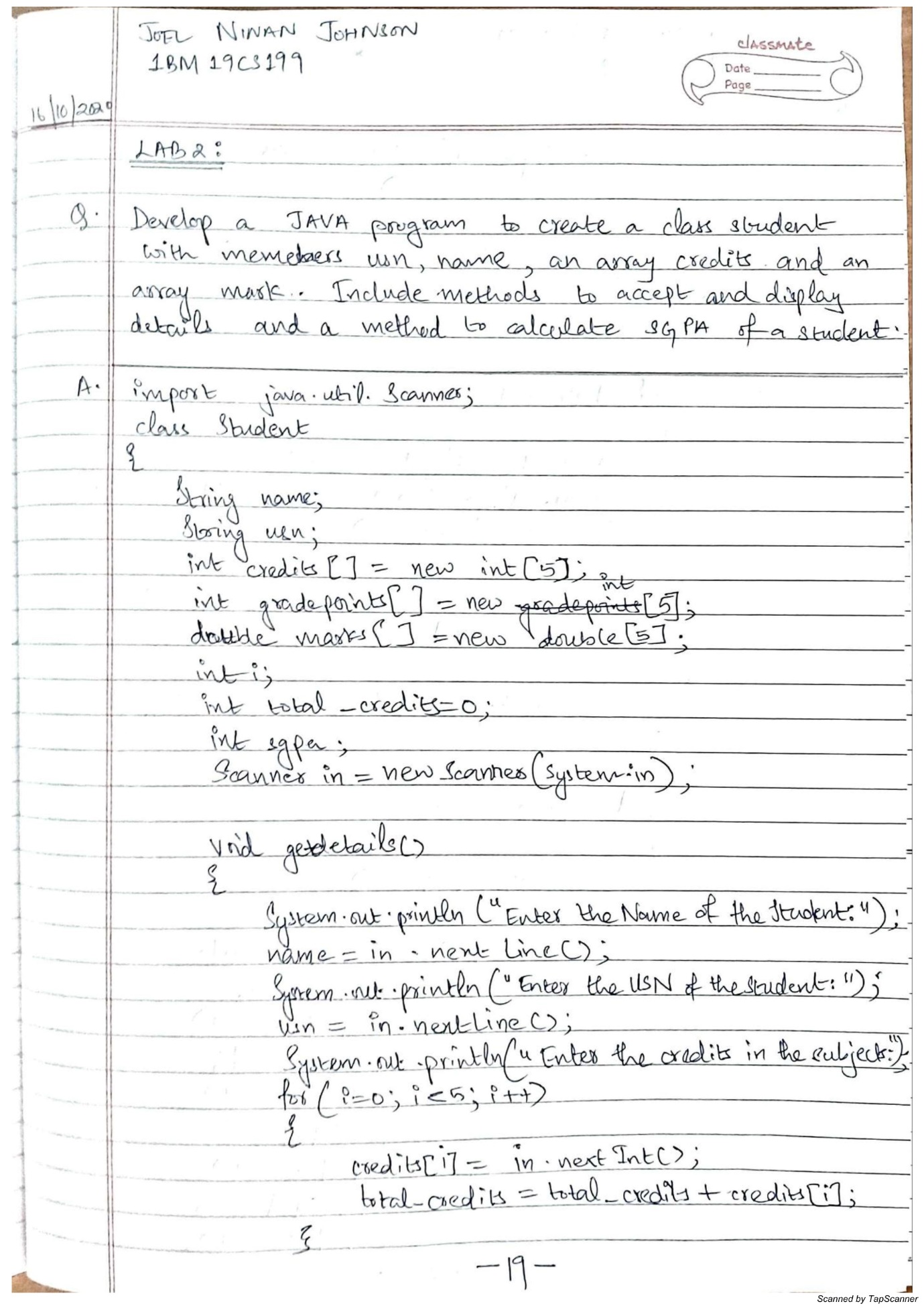
std1.sgpa(); std1.displaydetails();

}

}

**OUTPUT**





**LAB 3**

**Q.** Create a class Book which contains four members: name, author, price, num\_pages. Include a constructor to set the values for the members. Include methods to set and get the details of the objects. Include a toString( ) method that could display the complete details of the book. Develop a Java program to create n book objects.

**PROGRAM:**

import java.util.Scanner; class Book{

String name; String author; double price; int num\_pages;

Scanner in = new Scanner(System.in);

Book(){

System.out.println("Enter the name of the Book: "); name = in.nextLine();

System.out.println("Enter the author of the Book: "); author = in.nextLine();

System.out.println("Enter the price of the Book: "); price = in.nextDouble();

System.out.println("Enter the number of pages of the Book: "); num\_pages = in.nextInt();

System.out.println();

}

void display(){

System.out.println("----BOOK-----");

System.out.println("Name--> "+name); System.out.println("Author--> "+author);

System.out.println("Price--> "+price);

System.out.println("Number of pages--> "+num\_pages);

}

public String toString(){

return (" Name: "+name+" | Author: "+author+" | Price: "+price+" | Number of pages: "+num\_pages);

}

}

class Lab3{

public static void main(String args[]){

Scanner in = new Scanner(System.in);

System.out.println("Enter the number of Books: "); int n = in.nextInt();

System.out.println(); Book ob[] = new Book[n]; for(int i=0;i<n;i++)

{

System.out.println("Book "+(i+1)); ob[i] = new Book();

}

for(int i=0;i<n;i++)

{

System.out.print("Book "+(i+1)+"--> ");

System.out.println(ob[i]);

}

System.out.println();

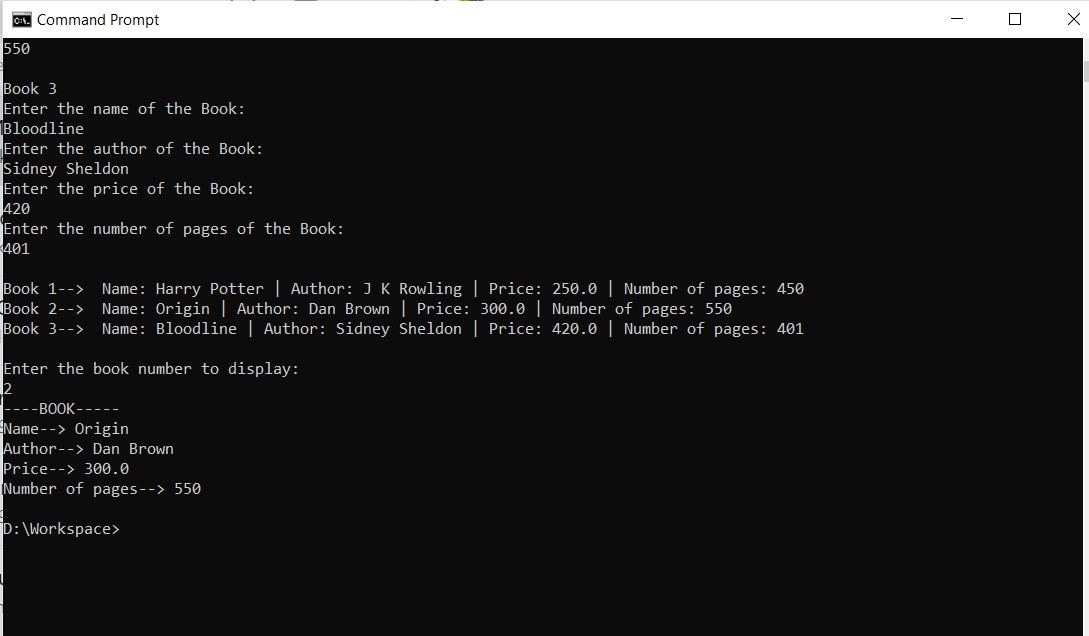
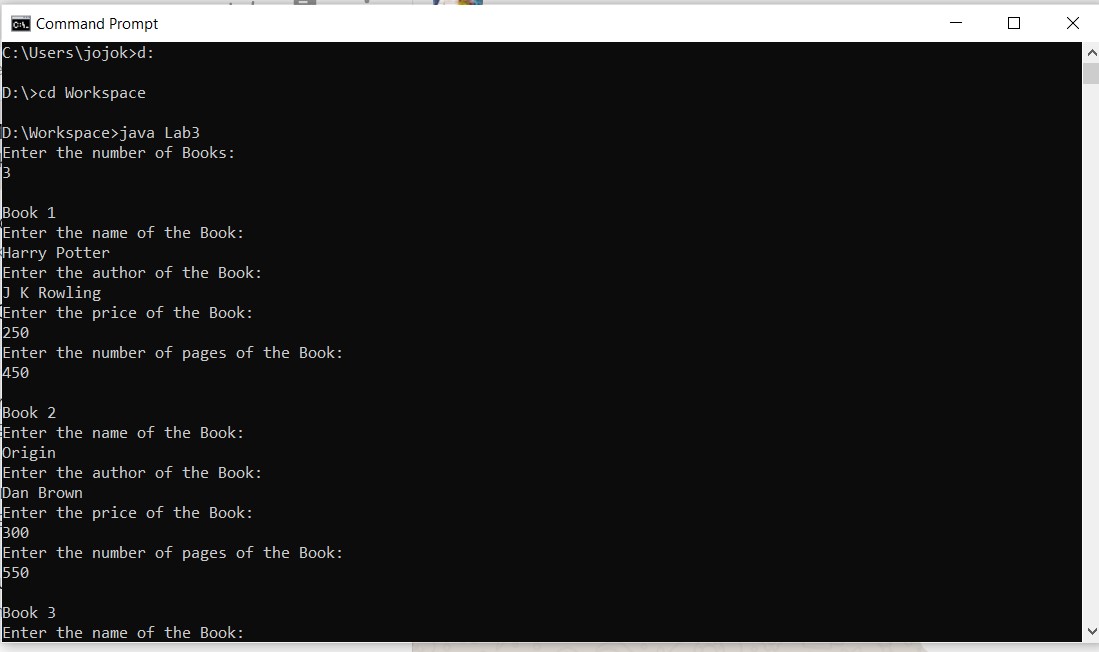
System.out.println("Enter the book number to display: "); int x = in.nextInt();

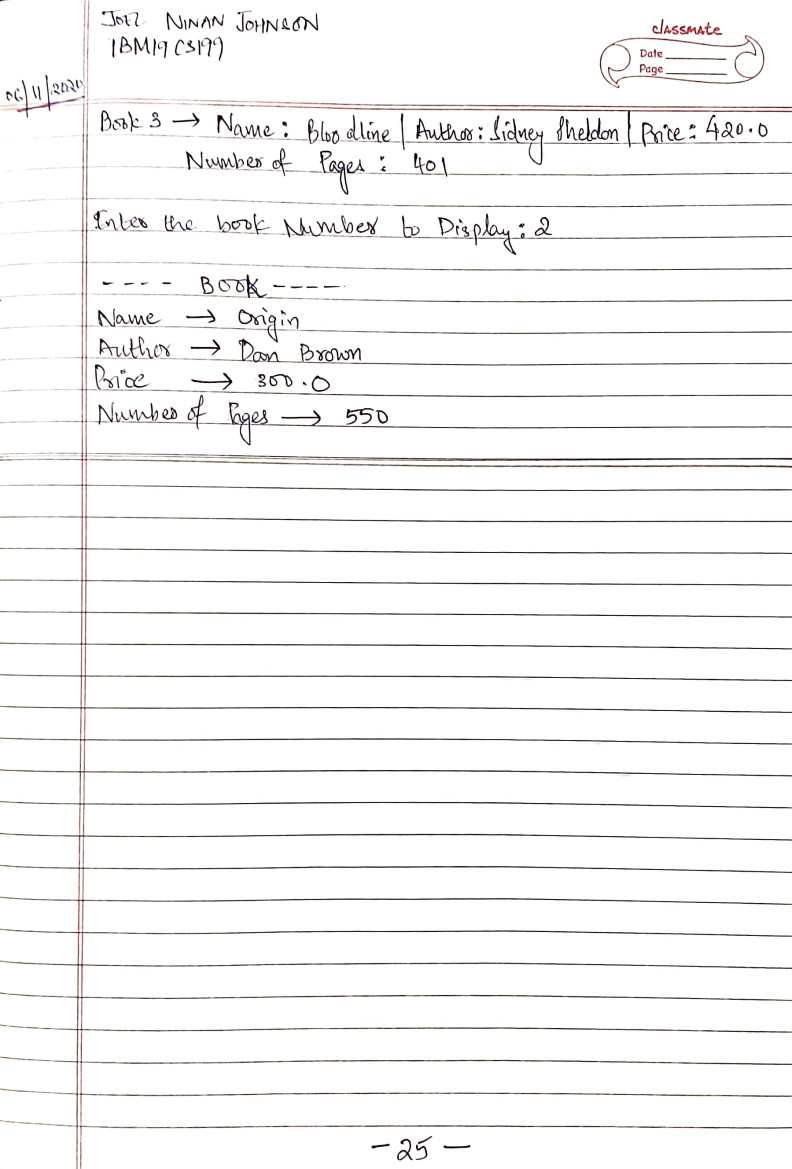
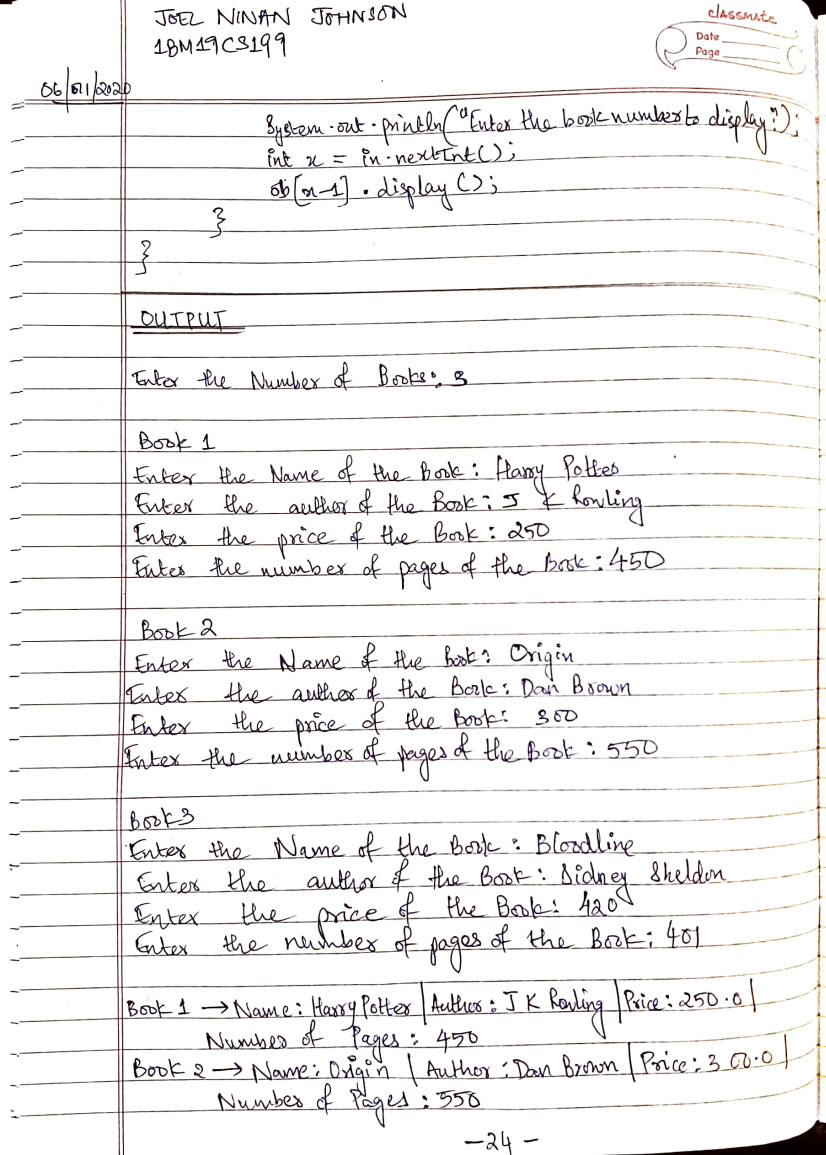
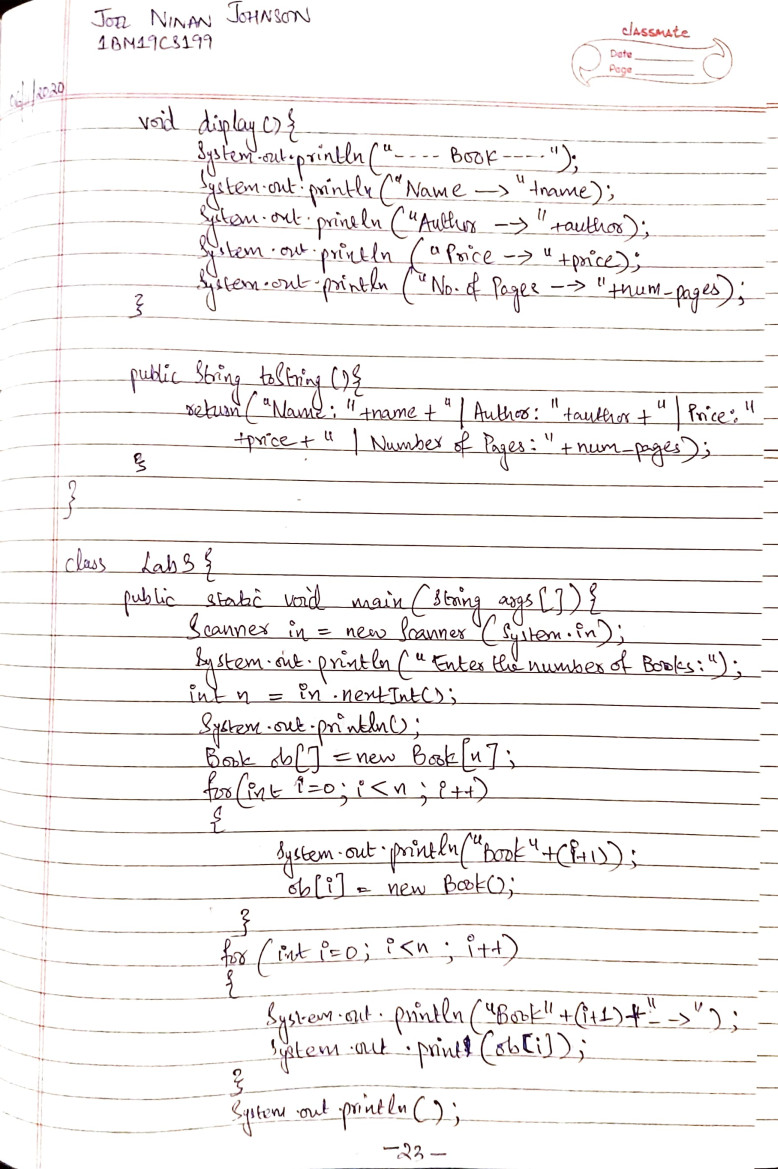
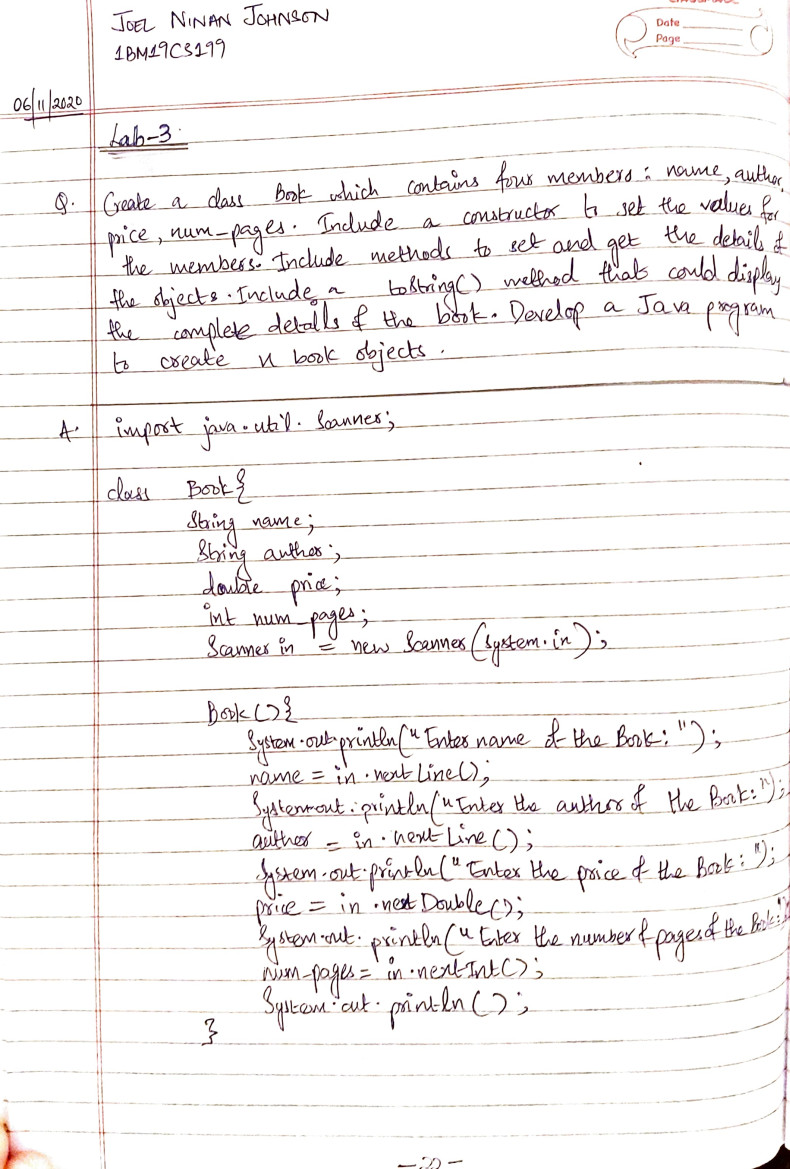
ob[x-1].display();

}

}

**OUTPUT:**





**LAB 4**

**Q.** Develop a Java program to create an abstract class named Shape that contains two integers and an empty method named printArea( ). Provide three classes named Rectangle, Triangle and Circle such that each one of the classes extends the class Shape. Each one of the classes contain only the method printArea( ) that prints the area of the given shape.

**PROGRAM:**

abstract class Shape {

int dim1; int dim2;

Shape(int x,int y){ dim1 = x; dim2 = y;

}

Shape(int rad){

dim1 = dim2 = rad;

}

abstract double printArea();

}

class Rectangle extends Shape {

Rectangle(int x,int y)

{

super(x,y);

}

double printArea(){

return (dim1\*dim2);

}

}

class Triangle extends Shape {

Triangle(int x,int y)

{

super(x,y);

}

double printArea(){ return (dim1\*dim2)/2;

}

}

class Circle extends Shape {

Circle(int rad)

{

super(rad);

}

double printArea(){ return (3.14\*dim1\*dim1);

}

}

class Lab4 { public static void main(String args[]){ Rectangle R = new Rectangle(10,8);

Triangle T = new Triangle(4,6);

Circle C = new Circle(2);

Shape ref;

ref = R;

System.out.println("Area of Rectangle: "+ref.printArea());

ref = T;

System.out.println("Area of Triangle: "+ref.printArea());

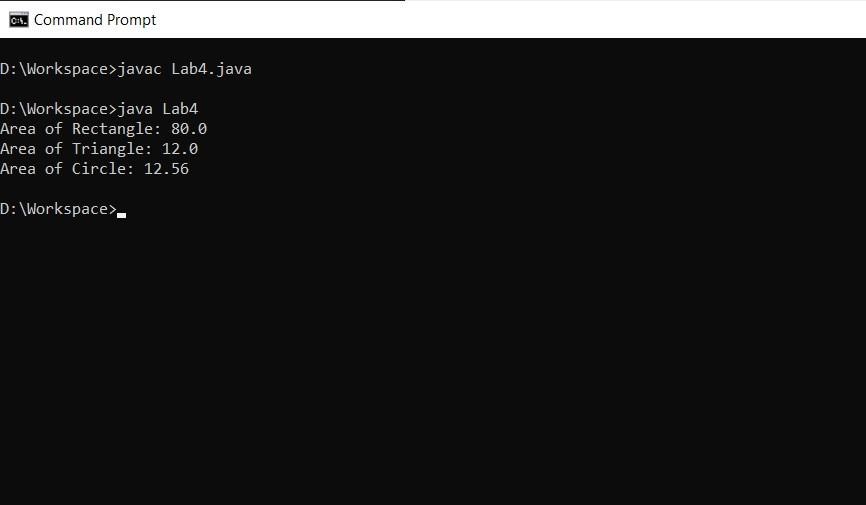
ref = C;

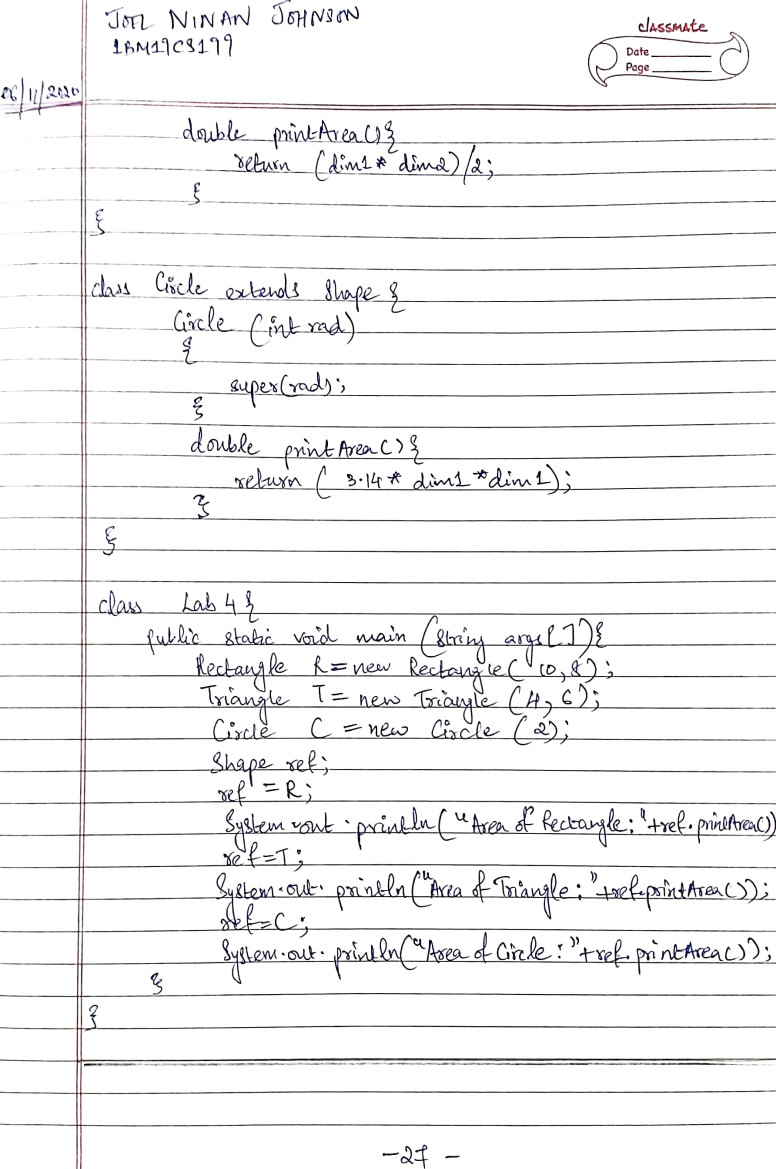
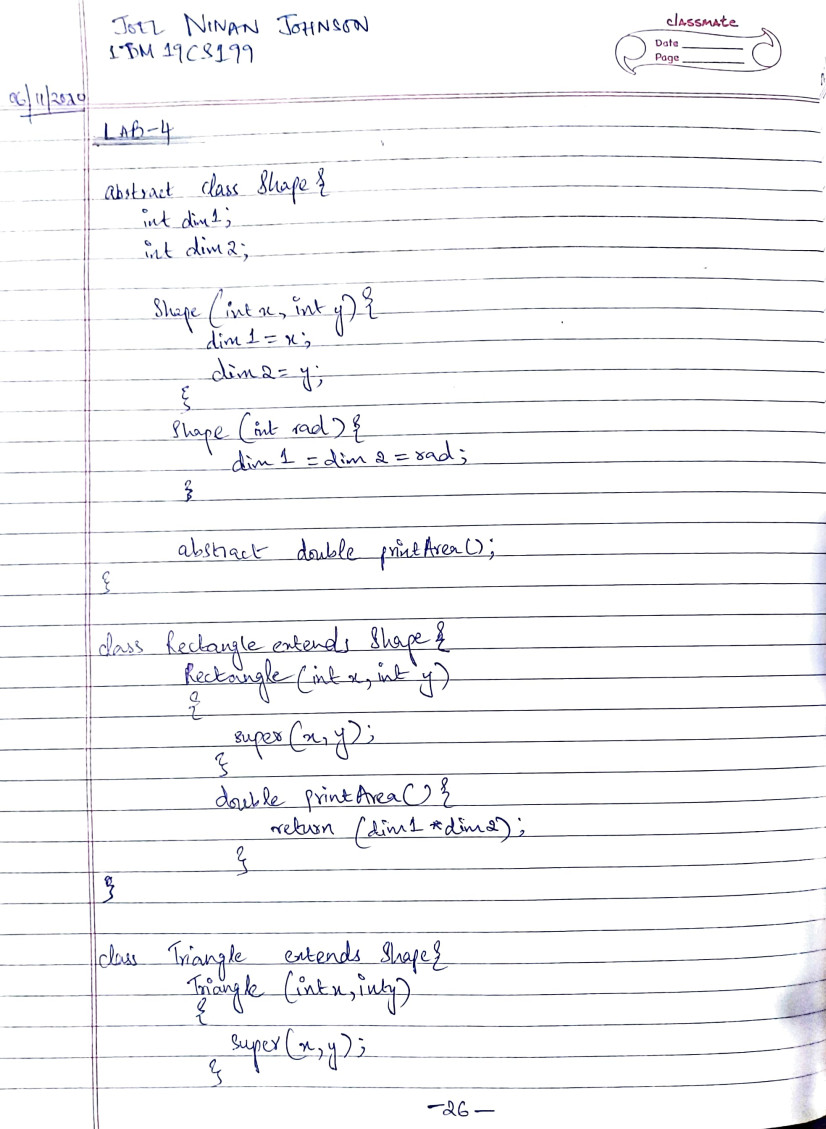
System.out.println("Area of Circle: "+ref.printArea());

}

}

**OUTPUT:**





**LAB 5**

**Q.** Develop a Java program to create a class Bank that maintains two kinds of account for its customers, one called savings account and the other current account. The savings account provides compound interest and withdrawal facilities but no cheque book facility. The current account provides cheque book facility but no interest. Current account holders should also maintain a minimum balance and if the balance falls below this level, a service charge is imposed. Create a class Account that stores customer name, account number and type of account. From this derive the classes Curr-acct and Sav-acct to make them more specific to their requirements. Include the necessary methods in order to achieve the following tasks:

1. Accept deposit from customer and update the balance.
2. Display the balance.
3. Compute and deposit interest
4. Permit withdrawal and update the balance Check for the minimum balance, impose penalty if necessary and update the balance.

**PROGRAM:**

import java.util.\*; import java.lang.\*;

class Account{

String name; int acct\_num; char acct\_type; double balance = 0;

Scanner in = new

Scanner(System.in);

void input\_details()

{

System.out.println("Ent er Your Name: "); name =

in.nextLine();

System.out.println("Ent er you Account Number: ");

acct\_num =

in.nextInt();

System.out.println("Ple ase enter your Account type

[C/S] : ");

acct\_type =

in.next().charAt(0);

}

void deposit(){

System.out.println("Ent er the amount to deposit: "); double dep =

in.nextDouble();

balance =+ dep;

System.out.println("Am ount Successfully Deposited!!");

System.out.println("Acc ount Balance: "+balance); System.out.println();

}

void view\_balance(){

System.out.println("Acc ount Balance: "+balance);

System.out.println();

}

}

class Current extends Account{

Double cheq\_amnt;

void issue\_cheque(){

System.out.println("Ent er the Amount to issue the Cheque: ");

cheq\_amnt =

in.nextDouble();

if(cheq\_amnt>balance)

{

System.out.println("Un able to Issue

Cheque!!Entered Amount

Unavailable in Account Balance!!");

System.out.println();

}

else{ balance =

balance - cheq\_amnt;

System.out.println("Cheque For Rs."+cheq\_amnt+"only

Issued Successfully!!");

System.out.println();

}

}

void check\_balance(){

if(balance<5000

&& balance>0)

{

System.out.println("Cur rent Account Balance is below the Minimum Required Balance!!");

balance =

balance - 1000; System.out.println("Ser vice Charge of Rs. 1000 deducted from Account Balance!!");

System.out.println();

}

view\_balance();

}

}

class Savings extends Account{

Double

withdraw\_amnt,CI;

void withdraw(){

System.out.println("Ent er the Amount to Withdraw:

");

withdraw\_amnt =

in.nextDouble();

if(withdraw\_amnt>bala

nce){

System.out.println("Ent ered Amount Unavailable in Account Balance!!");

System.out.println();

}

else{ balance =

balance - withdraw\_amnt;

System.out.println("Suc cessfully Withdrawn Rs."+withdraw\_amnt+" from Account!!");

System.out.println();

}

}

void compute\_CI(){

System.out.println("Ent er time period(years): "); int time = in.nextInt();

CI =

(balance\*Math.pow(1+(0.02/

12),12\*time)) - balance;

System.out.println("Co

mpound Interest for "+time+" years compounded monthly at a rate of 2% : Rs."+CI); balance =

balance + CI;

System.out.println("Inte rest Has Been Successfully Deposited!!");

System.out.println();

}

}

class Lab5{

public static void main(String args[]){

int choice;

Scanner in = new

Scanner(System.in);

Account A = new

Account();

A.input\_details();

if(A.acct\_type=='c' ||

A.acct\_type=='C')

{

Current Ac

= new Current();

System.out.println("\*\*\*\* \*\*\*CURRENT

ACCOUNT\*\*\*\*\*\*\*");

do{

System.out.println();

System.out.println("-----

-----MENU------------");

System.out.println("[1] DEPOSIT AMOUNT");

System.out.println("[2] VIEW BALANCE");

System.out.println("[3]I SSUE CHEQUE");

System.out.println("[4] EXIT");

System.out.println();

System.out.println("Ent

er your choice:");

choice = in.nextInt();

System.out.println();

switch(choice)

{ case 1:

Ac.deposit();

break;

case 2:

Ac.check\_balance();

break;

case 3:

Ac.issue\_cheque();

break;

case 4:

System.exit(0);

break;

default:

System.out.println("Invalid Input!!!");

}

}while(choice <= 4 &&

choice >= 1);

}

else

if(A.acct\_type=='s' ||

A.acct\_type=='S')

{

Savings As

= new Savings();

System.out.println("\*\*\*\*

\*\*\*SAVINGS

ACCOUNT\*\*\*\*\*\*\*");

do{

System.out.println();

System.out.println("-----

-----MENU------------");

System.out.println("[1] DEPOSIT AMOUNT");

System.out.println("[2] VIEW BALANCE");

System.out.println("[3]WITHDRAW");

System.out.println("[4]COMPUTE COMPOUND INTEREST");

System.out.println("[5]EXIT");

System.out.println();

System.out.println("Ent

er your choice:");

choice = in.nextInt();

System.out.println();

switch(choice)

{

case 1:

As.deposit();

break;

case 2:

As.view\_balance();

break;

case 3:

As.withdraw();

break;

case 4:

As.compute\_CI();

break;

case 5:

System.exit(0);

break;

default:

System.out.println("Invalid Input!!!");

}

}while(choice <= 5 &&

choice >= 1);

}

else

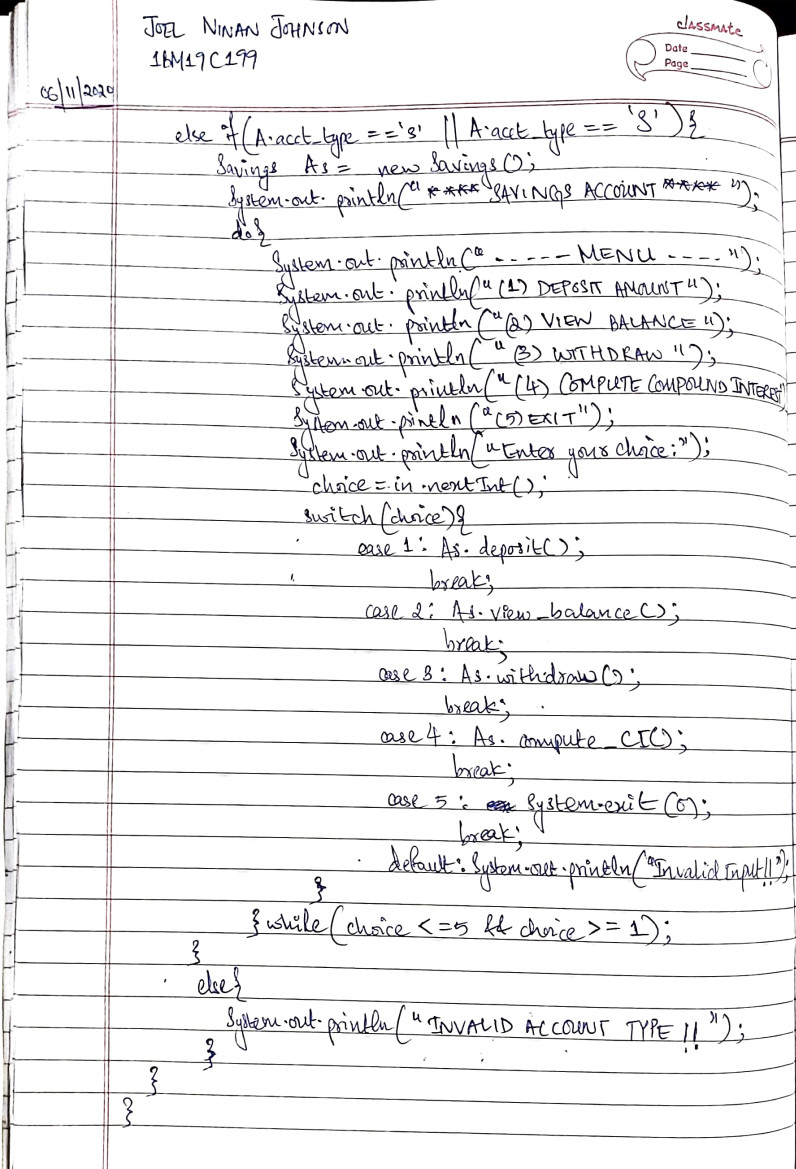
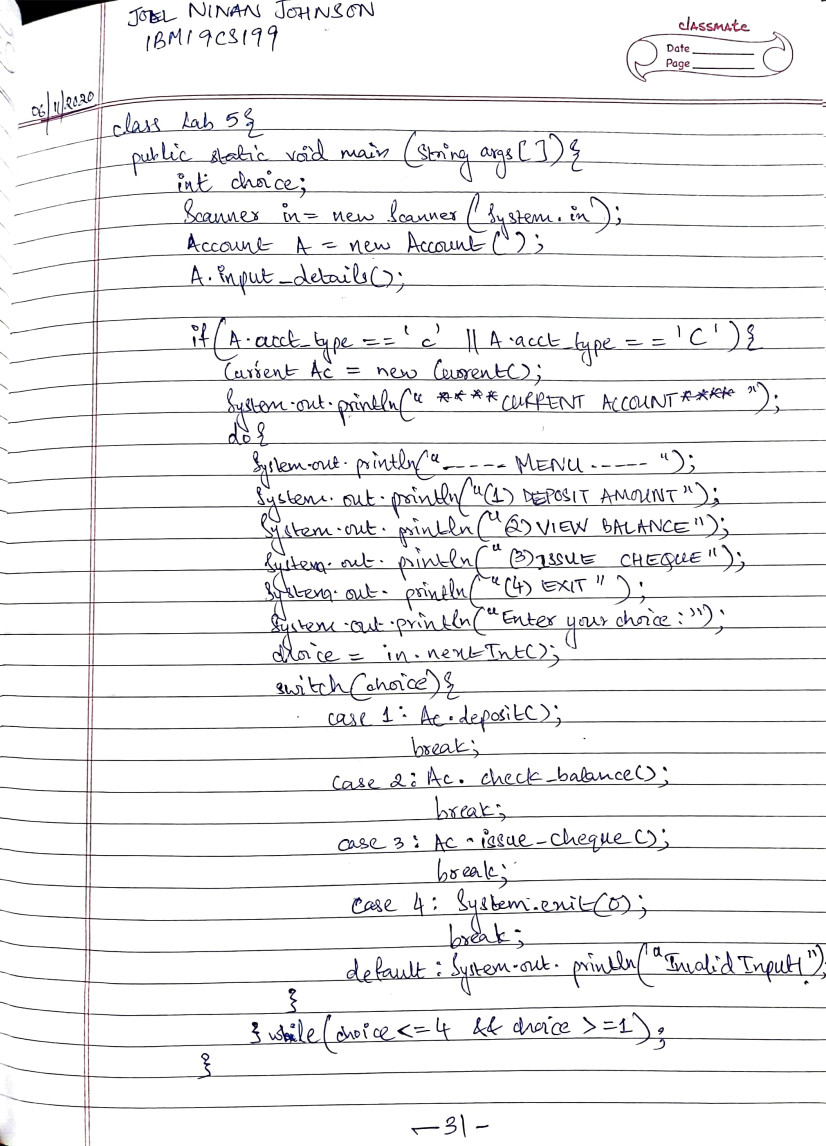
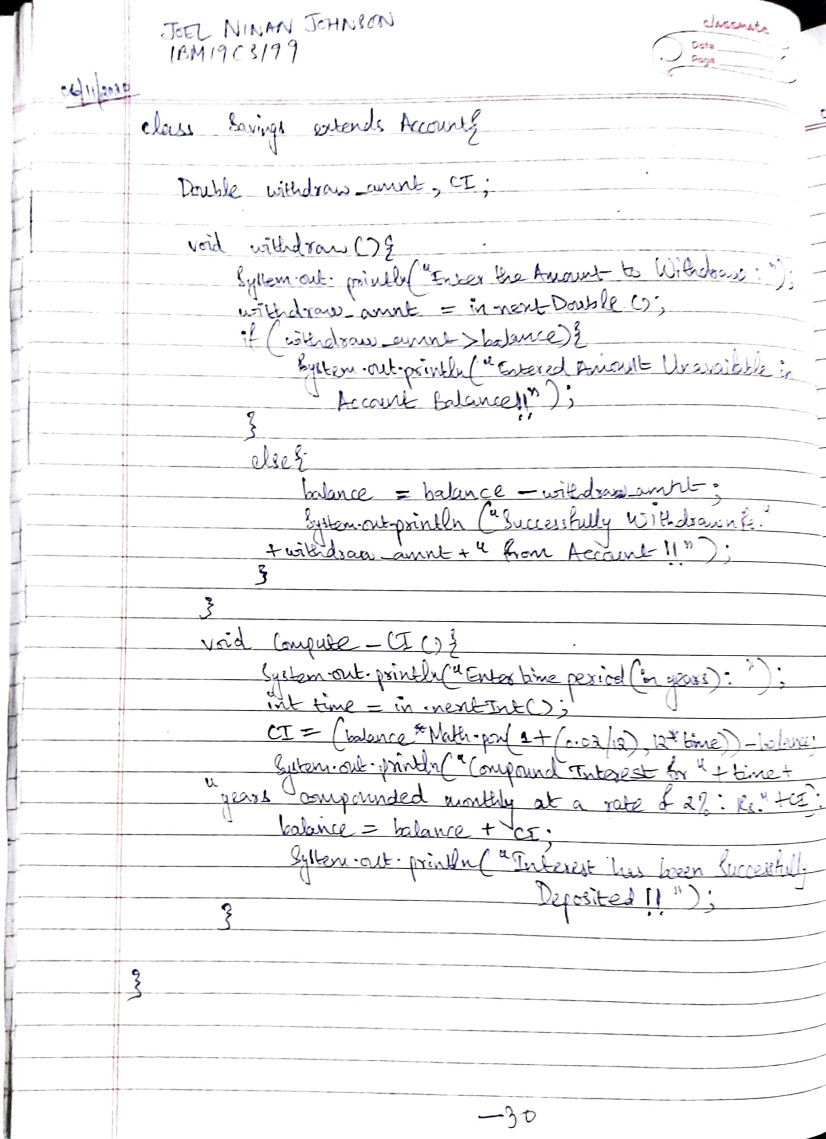
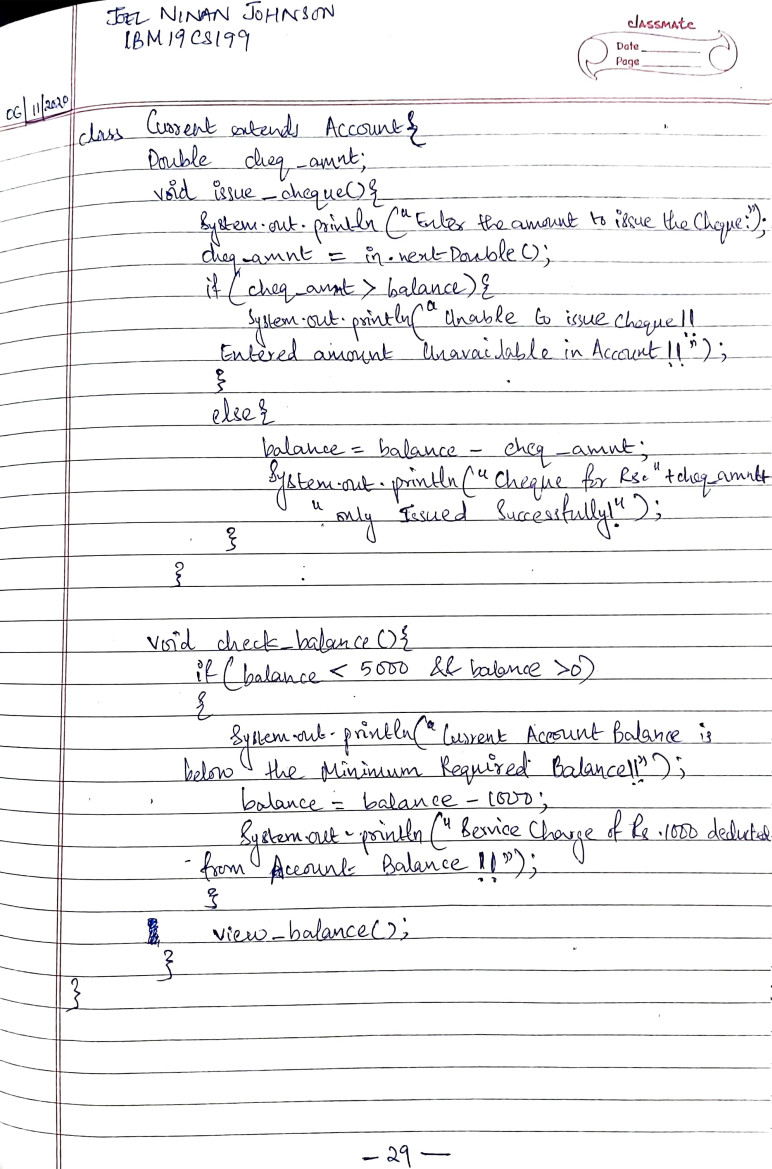
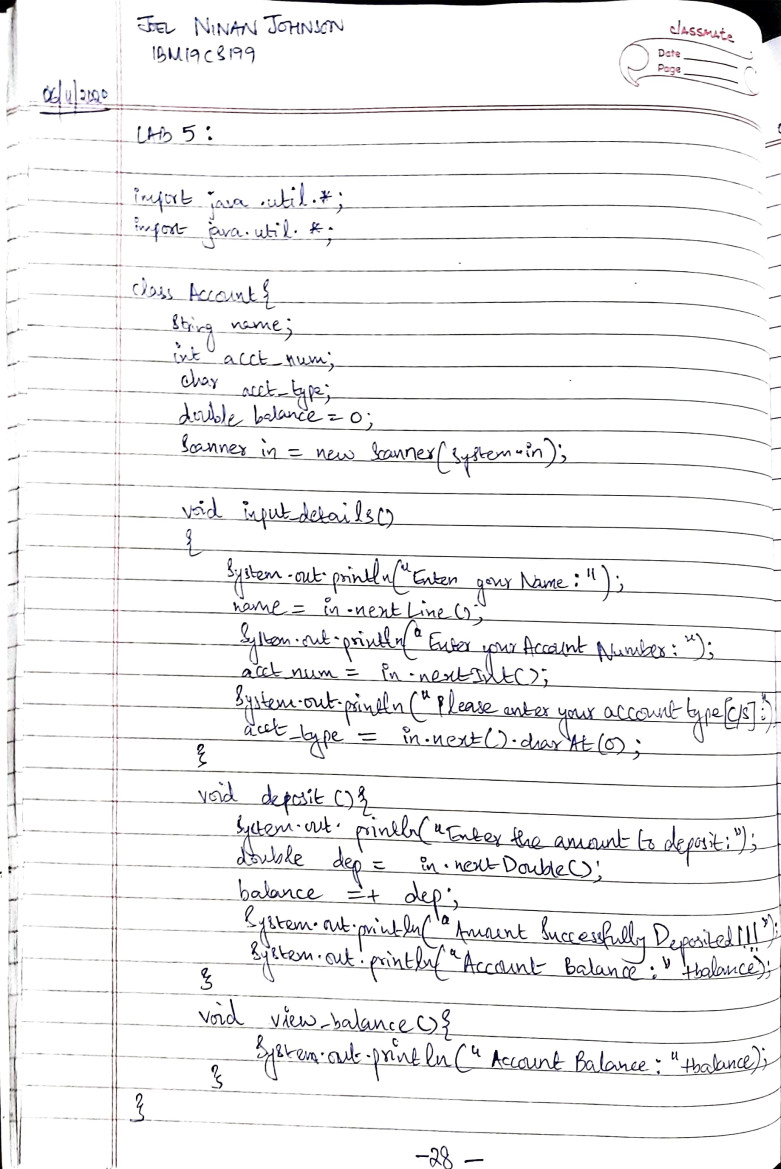
{

System.out.println("INV ALID ACCOUNT TYPE!!!");

}

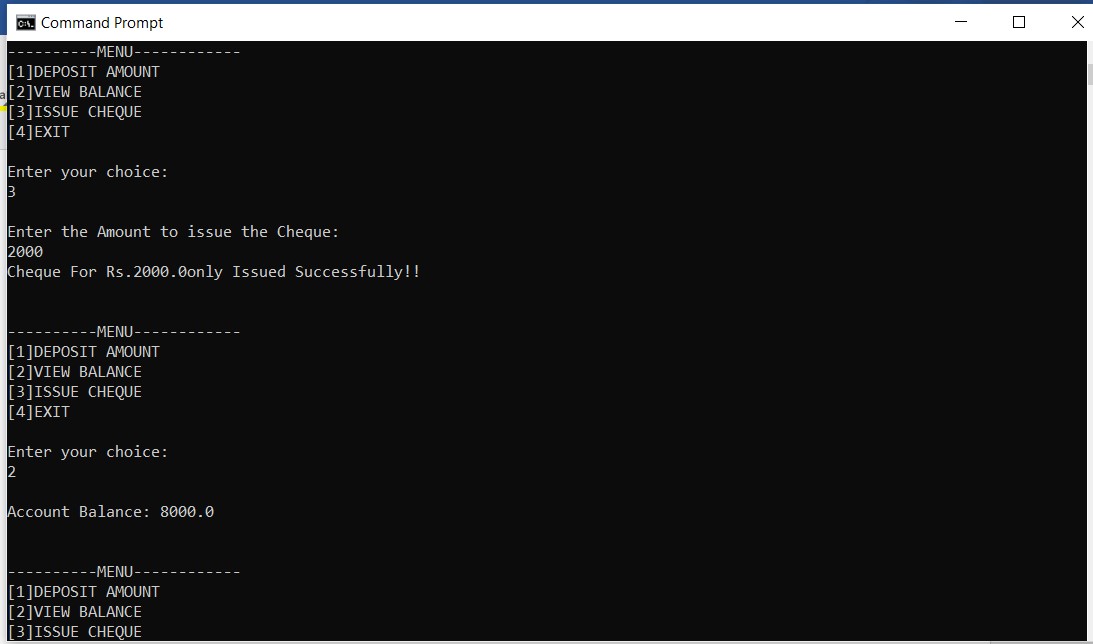
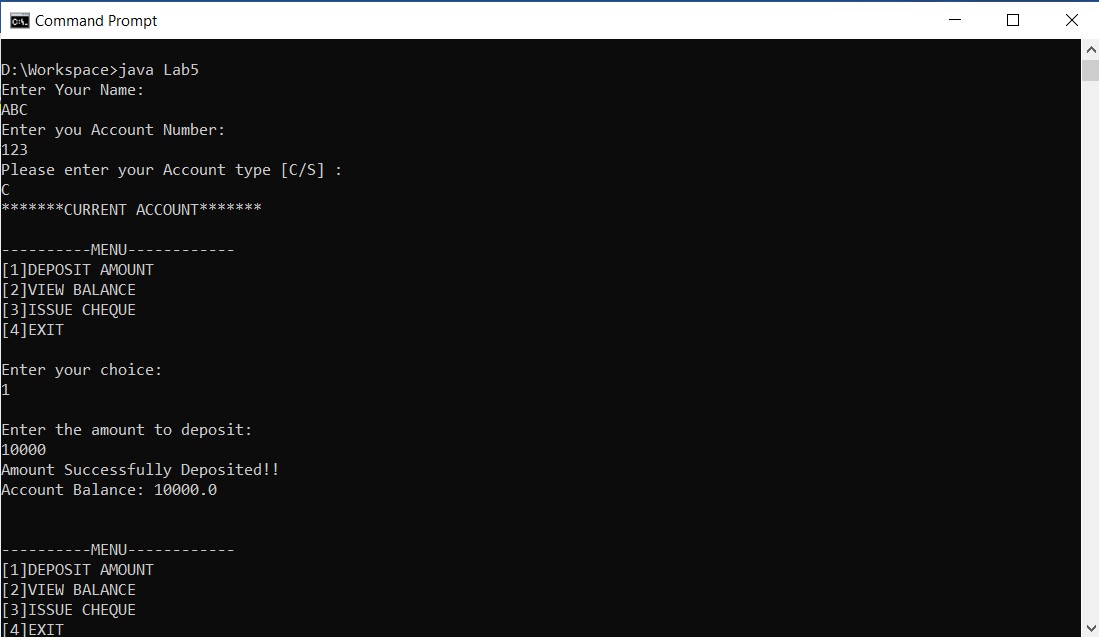
}

}



**OUTPUT:**

(Current Account)



(Savings Account)

