**Experiment-01**

**Aim:**

Write a java to print the Fibonacci Series of a number using for loop.

**CO 1:**

Understand object-oriented concepts and design classes and objects to solve problems

**Procedure:**

class FibonacciF{

public static void main(String[] arg){

int num=12;

int n1=0;

int n2=1;

int i=0;

int n3=0;

System.out.println("Fibonacci: ");

System.out.print(n1);

System.out.print(" "+ n2);

for(i=2;i<num;i++){

n3=n1+n2;

System.out.print(" "+n3);

n1=n2;

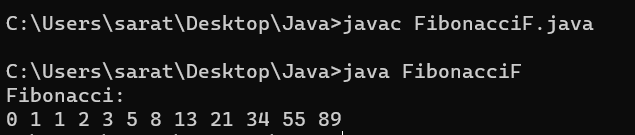
n2=n3;

}

}

}

**Output Screenshot:**

****

**Result:**

The output was successfully generated and the program was executed. Thus CO1 was attained.

**Experiment-02**

**Aim:**

Write a java program to check whether a number is palindrome or not

**CO 1:**

Understand object-oriented concepts and design classes and objects to solve problems

**Procedure:**

class PalindromeW{

public static void main(String[] args){

int num=121;

int temp=num;

int rev=0;

int rem=0;

while(temp!=0){

rem=temp%10;

rev=rev\*10+rem;

temp=temp/10;

}

if(rev==num){

System.out.println("Palindrome");

}

else{

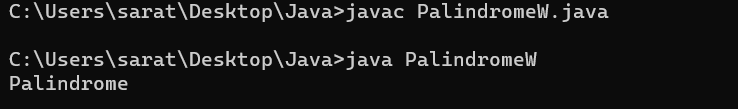
System.out.println("Not Palindrome");

}

}

}

**Output Screenshot:**

****

**Result:**

The output was successfully generated and the program was executed. Thus CO1 was attained.

**Experiment-03**

**Aim:**

Write a java program to print the Armstrong numbers of the limit.

**CO 1:**

Understand object-oriented concepts and design classes and objects to solve problems

**Procedure:**

class ArmstrongF{

public static void main(String[] arg){

int num=160;

int i=0;

for(i=0;i<num;i++){

int rem=0;

int cb=0;

int arm=0;

int temp=i;

while(temp!=0){

rem=temp%10;

cb=rem\*rem\*rem;

arm=arm+cb;

temp=temp/10;

}

if(arm==i)

{

System.out.println(arm + " is an armstrong number");

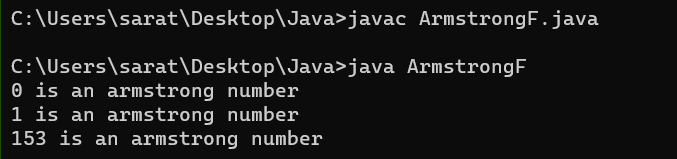
}

}

}

}

**Output Screenshot:**

****

**Result:**

The output was successfully generated and the program was executed. Thus CO1 was attained.

**Experiment-04**

**Aim:**

Write a java program to reverse a number.

**CO 1:**

Understand object-oriented concepts and design classes and objects to solve problems

**Procedure:**

class RevNumW{

public static void main(String[] args){

int num=145;

int rev=0;

int rem=0;

int temp=num;

while(temp!=0){

rem=temp%10;

rev=rev\*10+rem;

temp=temp/10;

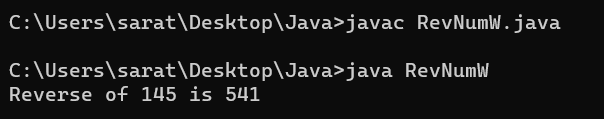
}

System.out.println("Reverse of "+ num + " is " + rev);

}

}

**Output Screenshot:**

****

**Result:**

The output was successfully generated and the program was executed. Thus CO1 was attained.

**Experiment-05**

**Aim:**

Write a java program to find the sum of digits of a number.

**CO 1:**

Understand object-oriented concepts and design classes and objects to solve problems

**Procedure:**

class SDigitW{

public static void main(String[] args){

int num=15;

int temp=num;

int rem=0;

int sum=0;

while(temp!=0){

rem=temp%10;

sum=sum+rem;

temp=temp/10;

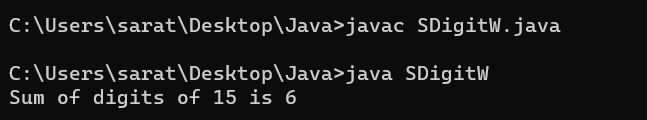
}

System.out.println("Sum of digits of " + num + " is " +sum);

}

}

**Output Screenshot:**

****

**Result:**

The output was successfully generated and the program was executed. Thus CO1 was attained.

**Experiment-06**

**Aim:**

Write a java program to add complex numbers.

**CO 1:**

Understand object-oriented concepts and design classes and objects to solve problems

**Procedure:**

import java.util.\*;

public class Complex{

public static void main(String[] args){

Scanner Snr = new Scanner(System.in);

System.out.println("Enter the real part of first imaginary number: ");

int r1 = Snr.nextInt();

System.out.println("Enter the coefficient of the first imaginary constant: ");

int i1 = Snr.nextInt();

System.out.println("Enter the real part of second imaginary number: ");

int r2 = Snr.nextInt();

System.out.println("Enter the coefficient of the second imaginary constant: ");

int i2 = Snr.nextInt();

System.out.println("The first imaginary number is "+r1+" + "+i1+"i");

System.out.println("The second imaginary number is "+r2+" + "+i2+"i");

int r3=r1+r2;

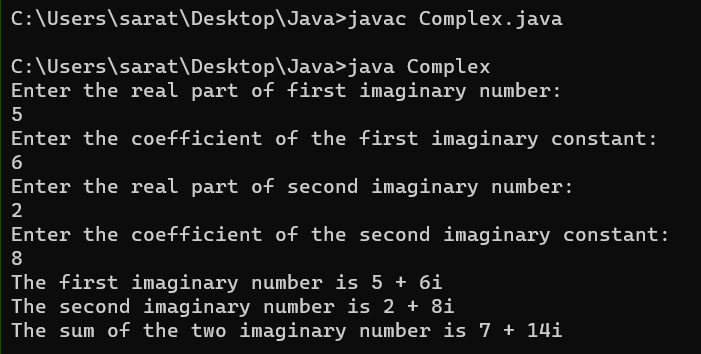
int i3=i1+i2;

System.out.println("The sum of the two imaginary number is "+ r3 +" + "+ i3 +"i");

}

}

**Output Screenshot:**

****

**Result:**

The output was successfully generated and the program was executed. Thus CO1 was attained.

**Experiment-07**

**Aim:**

Write a java program to perform matrix addition.

**CO 1:**

Understand object-oriented concepts and design classes and objects to solve problems

**Procedure:**

import java.util.\*;

public class Matrixadd{

public static void main(String[] args){

Scanner Snr = new Scanner(System.in);

System.out.println("Enter the dimensions of both matrix: ");

int rw = Snr.nextInt();

int cl = rw;

int Mtx1[][] = new int[rw][cl];

int Mtx2[][] = new int[rw][cl];

int Mtx3[][] = new int[rw][cl];

System.out.println("Enter the elements in first matrix: ");

for(int i=0;i<rw;i++){

for(int j=0;j<cl;j++){

Mtx1[i][j] = Snr.nextInt();

}

}

System.out.println("The First matrix: ");

for(int i=0;i<rw;i++){

for(int j=0;j<cl;j++){

System.out.print(Mtx1[i][j] + " ");

}

System.out.println(" ");

}

System.out.println("Enter the elements in second matrix: ");

for(int i=0;i<rw;i++){

for(int j=0;j<cl;j++){

Mtx2[i][j] = Snr.nextInt();

}

}

System.out.println("The Second matrix: ");

for(int i=0;i<rw;i++){

for(int j=0;j<cl;j++){

System.out.print(Mtx2[i][j] + " ");

}

System.out.println(" ");

}

System.out.println("The sum of the matrices: ");

for(int i=0;i<rw;i++){

for(int j=0;j<cl;j++){

Mtx3[i][j] = Mtx1[i][j] + Mtx2[i][j];

}

}

for(int i=0;i<rw;i++){

for(int j=0;j<cl;j++){

System.out.print(Mtx3[i][j] + " ");

}

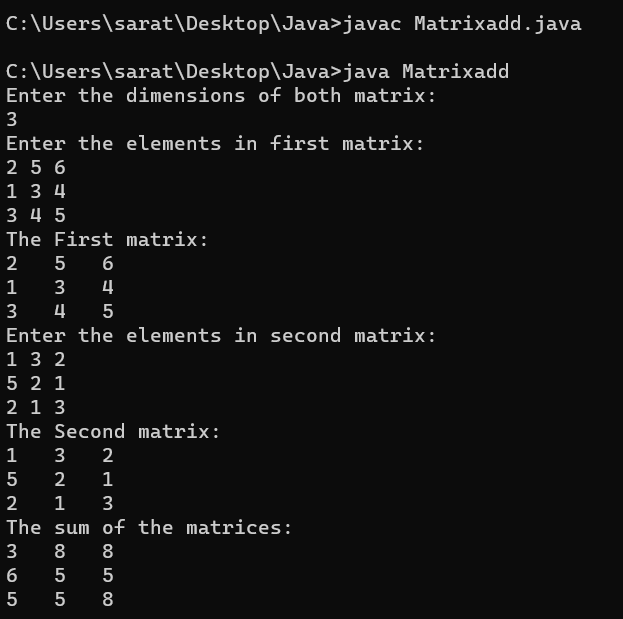
System.out.println(" ");

}

}

}

**Output Screenshot:**

****

**Result:**

The output was successfully generated and the program was executed. Thus CO1 was attained.

**Experiment-08**

**Aim:**

Write a java Program to check whether a matrix is symmetric or not.

**CO 1:**

Understand object-oriented concepts and design classes and objects to solve problems

**Procedure:**

import java.util.\*;

public class SymmetricMatrix{

public static void main(String[] args){

Scanner Snr = new Scanner(System.in);

System.out.println("Enter the dimension of the matrix :");

int sz = Snr.nextInt();

int Arr[][] = new int[sz][sz];

int Arr1[][] = new int[sz][sz];

System.out.println("Enter the elements in matrix :");

for(int i=0;i<sz;i++){

for(int j=0;j<sz;j++){

Arr[i][j] = Snr.nextInt();

}

}

System.out.println("The matrix: ");

for(int i=0;i<sz;i++){

for(int j=0;j<sz;j++){

System.out.print(Arr[i][j] + " ");

}

System.out.println(" ");

}

System.out.println("The Transpose of the matrix: ");

for(int i=0;i<sz;i++){

for(int j=0;j<sz;j++){

Arr1[i][j]=Arr[j][i];

}

}

for(int i=0;i<sz;i++){

for(int j=0;j<sz;j++){

System.out.print(Arr1[i][j] + " ");

}

System.out.println(" ");

}

int flag=0;

for(int i=0;i<sz;i++){

for(int j=0;j<sz;j++){

if(Arr[i][j] != Arr1[i][j]){

flag=1;

break;

}

}

}

if(flag==1){

System.out.println("The matrix is not symmetric");

}

else{

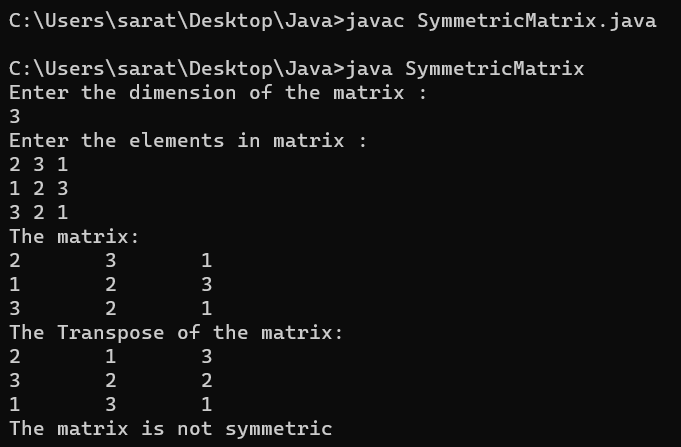
System.out.println("The matrix is symmetric");

}

}

}

**Output Screenshot:**

****

**Result:**

The output was successfully generated and the program was executed. Thus CO1 was attained.

**Experiment-09**

**Aim:**

Write a java program to find transpose of a matrix.

**CO 1:**

Understand object-oriented concepts and design classes and objects to solve problems

**Procedure:**

import java.util.\*;

public class Transpose{

public static void main(String[] args){

Scanner Snr = new Scanner(System.in);

System.out.println("Enter the dimension of the matrix :");

int sz = Snr.nextInt();

int Arr[][] = new int[sz][sz];

int Arr1[][] = new int[sz][sz];

System.out.println("Enter the elements in matrix :");

for(int i=0;i<sz;i++){

for(int j=0;j<sz;j++){

Arr[i][j] = Snr.nextInt();

}

}

System.out.println("The matrix: ");

for(int i=0;i<sz;i++){

for(int j=0;j<sz;j++){

System.out.print(Arr[i][j] + " ");

}

System.out.println(" ");

}

System.out.println("The Transpose of the matrix: ");

for(int i=0;i<sz;i++){

for(int j=0;j<sz;j++){

Arr1[i][j]=Arr[j][i];

}

}

for(int i=0;i<sz;i++){

for(int j=0;j<sz;j++){

System.out.print(Arr1[i][j] + " ");

}

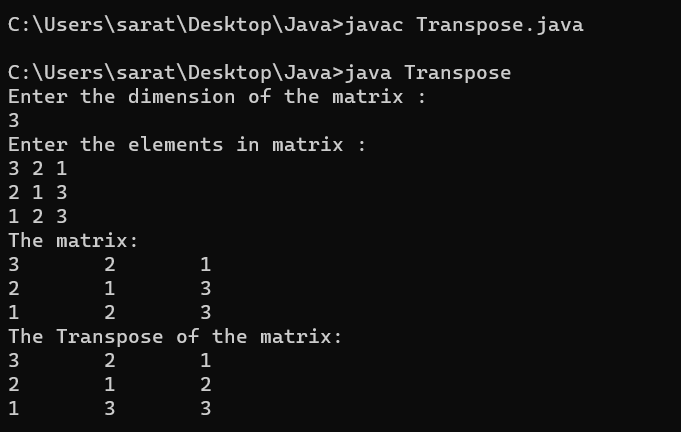
System.out.println(" ");

}

}

}

**Output Screenshot:**

****

**Result:**

The output was successfully generated and the program was executed. Thus CO1 was attained.

**Experiment-10**

**Aim:**

Write a java program to find the largest element in an array.

**CO 1:**

Understand object-oriented concepts and design classes and objects to solve problems

**Procedure:**

import java.util.\*;

public class Largest{

public static void main(String[] args){

Scanner Snr = new Scanner(System.in);

System.out.println("Enter the size of array: ");

int sz = Snr.nextInt();

int Arr[] = new int[sz];

System.out.println("Enter the elements in array: ");

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

{

Arr[i] = Snr.nextInt();

}

System.out.print("The array: ");

System.out.print("[ ");

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

{

System.out.print(Arr[i]+" ");

}

System.out.println("]");

int large=0;

for(int i=0;i<sz;i++){

for(int j=0;j<sz;j++){

if(Arr[i]>=Arr[j]){

large=Arr[i];

}

else

{

break;

}

}

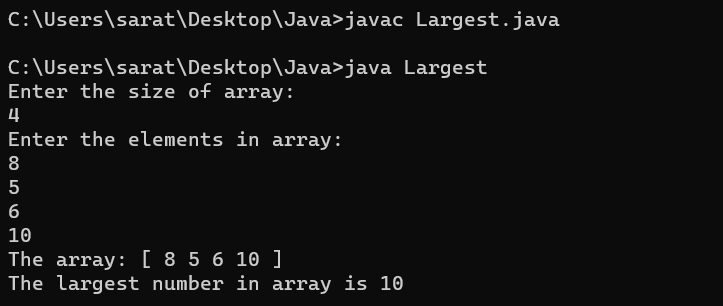
}

System.out.println("The largest number in array is "+large);

}

}

**Output Screenshot:**

****

**Result:**

The output was successfully generated and the program was executed. Thus CO1 was attained.

**Experiment-11**

**Aim:**

Write a java program to find the smallest element in an array.

**CO 1:**

Understand object-oriented concepts and design classes and objects to solve problems

**Procedure:**

import java.util.\*;

public class Smallest{

public static void main(String[] args){

Scanner Snr = new Scanner(System.in);

System.out.println("Enter the size of array: ");

int sz = Snr.nextInt();

int Arr[] = new int[sz];

System.out.println("Enter the elements in array: ");

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

{

Arr[i] = Snr.nextInt();

}

System.out.print("The array: ");

System.out.print("[ ");

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

{

System.out.print(Arr[i]+" ");

}

System.out.println("]");

int small=0;

for(int i=0;i<sz;i++){

for(int j=0;j<sz;j++){

if(Arr[i]<=Arr[j]){

small=Arr[i];

}

else

{

break;

}

}

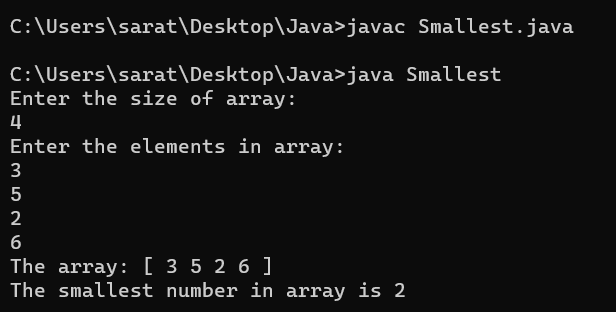
}

System.out.println("The largest number in array is "+small);

}

}

**Output Screenshot:**

****

**Result:**

The output was successfully generated and the program was executed. Thus CO1 was attained.

**Experiment-12**

**Aim:**

Write a java program to search for an element in an array (Linear Search)

**CO 2:**

Implement arrays and strings.

**Procedure:**

public class linearSearch {

public static void main(String[] args) {

int[] arr = { 1, 3, 5, 7, 9 };

int key = 7;

int position = -1;

for (int i = 0; i < arr.length; i++) {

if (arr[i] == key) {

position = i;

break;

}

}

if (position != -1) {

System.out.println(key + " found at position " + position);

} else {

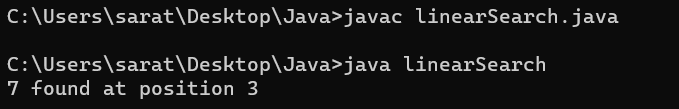
System.out.println(key + " not found in the array");

}

}

}

**Output Screenshot:**

****

**Result:**

The output was successfully generated and the program was executed. Thus CO2 was attained.

**Experiment-13**

**Aim:**

Write a java program to search for an element in an array (Binary Search)

**CO 2:**

Implement arrays and strings.

**Procedure:**

import java.util.Scanner;

public class binarysearch {

public static int binarySearch(int[] arr, int x) {

int left = 0;

int right = arr.length - 1;

while (left <= right) {

int mid = left + (right - left) / 2;

if (arr[mid] == x) {

return mid;

} else if (arr[mid] < x) {

left = mid + 1;

} else {

right = mid - 1;

}

}

return -1;

}

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the size of the array: ");

int n = sc.nextInt();

int[] arr = new int[n];

System.out.println("Enter the elements of the array in sorted order:");

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

arr[i] = sc.nextInt();

}

System.out.print("Enter the element to search for: ");

int x = sc.nextInt();

int pos = binarySearch(arr, x);

if (pos == -1) {

System.out.println("Element not found");

} else {

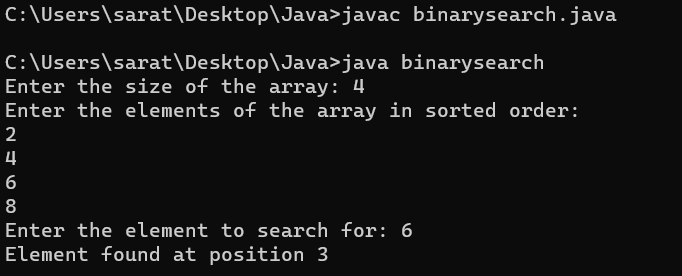
System.out.println("Element found at position " + (pos + 1));

}

}

}

**Output Screenshot:**

****

**Result:**

The output was successfully generated and the program was executed. Thus CO2 was attained.

**Experiment-14**

**Aim:**

Write a java program to sort strings

**CO 2:**

Implement arrays and strings.

**Procedure:**

import java.util.\*;

public class stringSort{

public static void main(String[] args){

Scanner input= new Scanner(System.in);

System.out.println("Enter the string 1:");

String str1 = input.nextLine();

System.out.println("Enter the string 2:");

String str2 = input.nextLine();

System.out.println("Enter the string 3:");

String str3 = input.nextLine();

System.out.println("Enter the string 4:");

String str4 = input.nextLine();

System.out.println(str1.compareTo(str2));

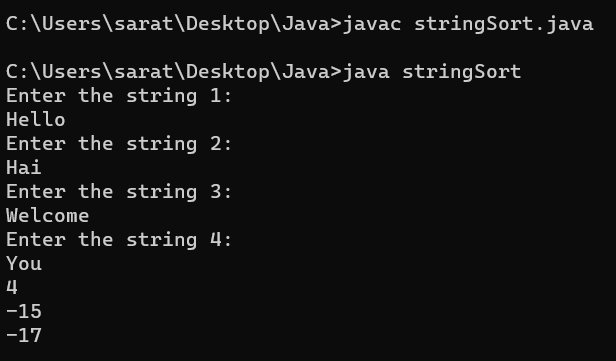
System.out.println(str1.compareTo(str3));

System.out.println(str1.compareTo(str4));

}

}

**Output Screenshot:**

****

**Result:**

The output was successfully generated and the program was executed. Thus CO2 was attained.

**Experiment-15**

**Aim:**

Write a java program to perform String Manipulations.

**CO 2:**

Implement arrays and strings.

**Procedure:**

import java.util.\*;

public class stringMani{

public static void main(String[] args){

Scanner sc = new Scanner(System.in);

System.out.print("Enter the string 1: ");

String var1 = sc.nextLine();

System.out.print("Enter the string 2: ");

String var2 = sc.nextLine();

System.out.println("Concate :" + var1.concat(var2));

System.out.println("Equals :" + var1.equals(var2));

System.out.println("Length :" + var1.length());

System.out.println("To uppercase :" + var1.toUpperCase());

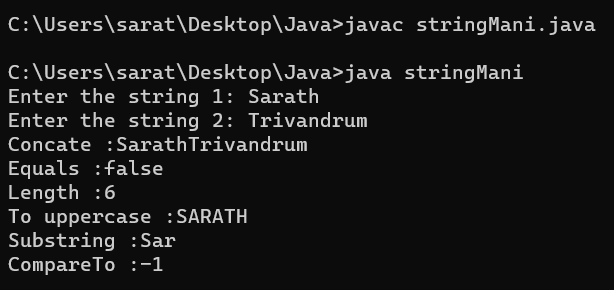
System.out.println("Substring :" + var1.substring(0,3));

System.out.println("CompareTo :" + var1.compareTo(var2));

}

}

**Output Screenshot:**

****

**Result:**

The output was successfully generated and the program was executed. Thus CO2 was attained.

**Experiment-16**

**Aim:**

Write a program to create a class for employee having attributes eno, ename, esalary. Read n employee information and search for an employee given eno using the concept of array of objects.

**CO 1:**

Understand object-oriented concepts and design classes and objects to solve problems

**Procedure:**

import java.util.Scanner;

public class Emp{

int eno;

String ename;

int esalary;

public void get()

{

Scanner nw = new Scanner(System.in);

System.out.println("Enter employee number: ");

eno = nw.nextInt();

System.out.println("Enter employee name: ");

ename = nw.next();

System.out.println("Enter employee salary: ");

esalary = nw.nextInt();

}

public void display()

{

System.out.println("Employee number is "+eno);

System.out.println("Employee name is "+ename);

System.out.println("Employee salary is "+esalary);

}

public static void main(String[] args){

int i;

Scanner nw=new Scanner(System.in);

System.out.println("Enter the limit of array");

int n=nw.nextInt();

Emp e[]=new Emp[n];

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

{

e[i]=new Emp();

e[i].get();

}

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

{

e[i].display();

}

System.out.println("Enter the eno:");

int val=nw.nextInt();

int flag=0;

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

{

if(e[i].eno==val)

{

e[i].display();

flag=1;

}

}

if(flag==0)

{

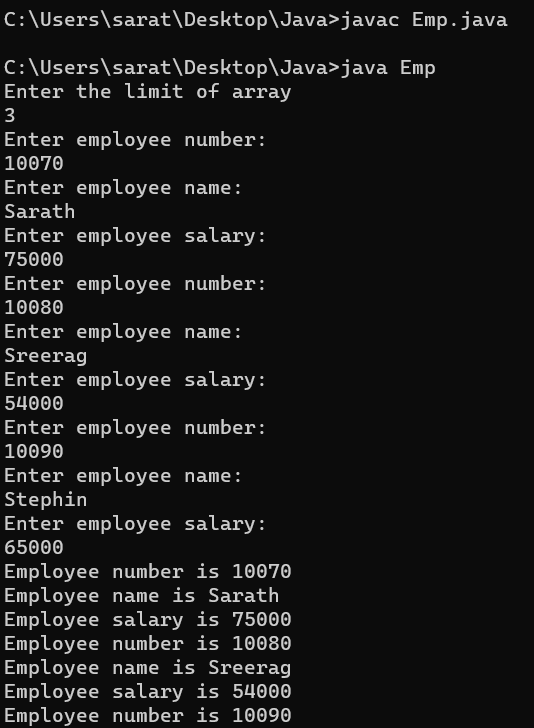
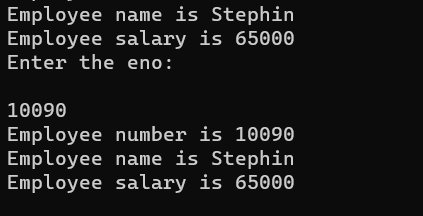
System.out.println("Not Found");

}

}

}

**Output Screenshot:**

** **

**Result:**

The output was successfully generated and the program was executed. Thus CO1 was attained.

**Experiment-17**

**Aim:**

Define a class ‘product’ with data members pcode, pname and price. Create 3 objects of the class and find the product having the lowest price.

**CO 1:**

Understand object-oriented concepts and design classes and objects to solve problems.

**Procedure:**

import java.util.\*;

class product{

int pcode;

String pname;

int price;

public void get() {

Scanner sc = new Scanner(System.in);

System.out.println("Enter the product code:");

pcode = sc.nextInt();

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

pname = sc.next();

System.out.println("Enter the product price:");

price = sc.nextInt();

}

public void display(){

System.out.println("Product code is: "+pcode);

System.out.println("Product name is: "+pname);

System.out.println("Product price is: "+price);

}

public static void main(String[] args){

product ob1 = new product();

product ob2 = new product();

product ob3 = new product();

ob1.get();

ob1.display();

ob2.get();

ob2.display();

ob3.get();

ob3.display();

if(ob1.price<ob2.price && ob1.price<ob3.price)

{

System.out.println("first product has lowest price");

}

else if(ob2.price<ob1.price && ob2.price<ob3.price)

{

System.out.println("second product has lowest price");

}

else

{

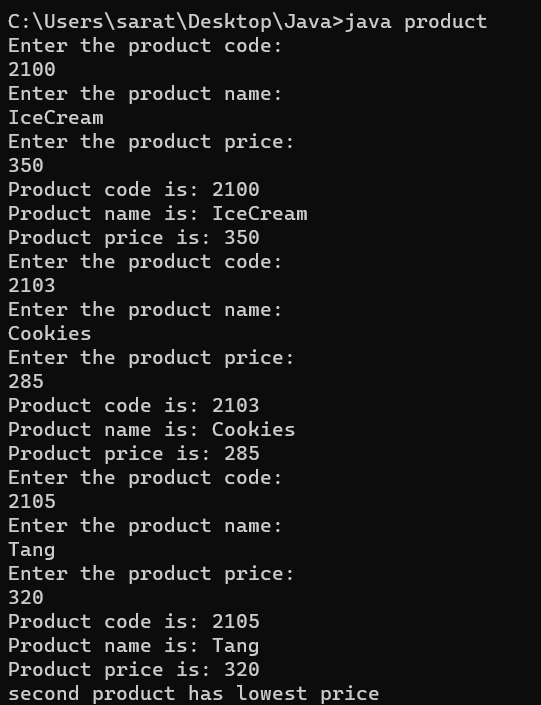
System.out.println("Third product has lowest price");

}

}

}

**Output Screenshot:**

****

**Result:**

The output was successfully generated and the program was executed. Thus CO1 was attained.

**Experiment-18**

**Aim:**

Define a class item having data members item no, Name and cost. Write java program to accept data of 4 items and display items having cost more than 5000 using array of objects

**CO 1:**

Understand object-oriented concepts and design classes and objects to solve problems.

**Procedure:**

import java.util.Scanner;

class Item {

int itemNo;

String name;

int cost;

public Item(int itemNo, String name, int cost) {

this.itemNo = itemNo;

this.name = name;

this.cost = cost;

}

}

public class item2{

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

Item[] items = new Item[4];

for (int i = 0; i < 4; i++) {

System.out.println("Enter details for item " + (i + 1) + ":");

System.out.print("Item No.: ");

int itemNo = scanner.nextInt();

System.out.print("Name: ");

String name = scanner.next();

System.out.print("Cost: ");

int cost = scanner.nextInt();

items[i] = new Item(itemNo, name, cost);

}

System.out.println("Items with cost greater than 5000:");

for (Item item : items) {

if (item.cost > 5000) {

System.out.println(item.itemNo + " " + item.name + " " + item.cost);

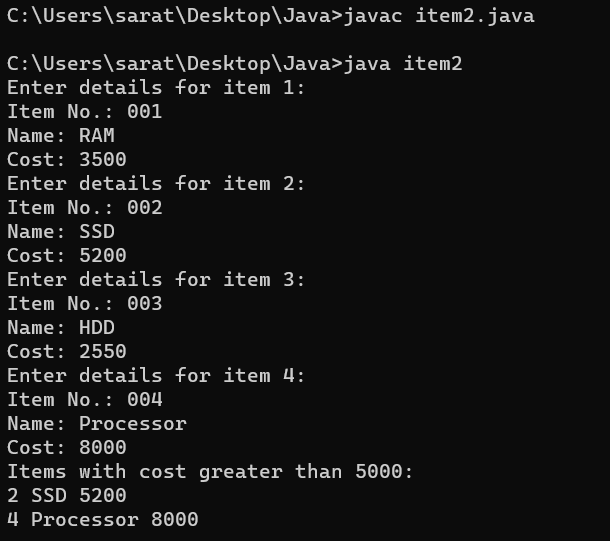
}

}

}

}

**Output Screenshot:**

****

**Result:**

The output was successfully generated and the program was executed. Thus CO1 was attained.

**Experiment-19**

**Aim:**

Define a class employee having data members Emp number, name and salary. Java program to accept data of 6 employes and display a list of employees having salary between 10000 to 15000 using array of objects Hint: Use if (salary>=10000 && salary<=15000)

**CO 1:**

Understand object-oriented concepts and design classes and objects to solve problems.

**Procedure:**

import java.util.Scanner;

class Employee {

int empNumber;

String name;

int salary;

public Employee(int empNumber, String name, int salary) {

this.empNumber = empNumber;

this.name = name;

this.salary = salary;

}

}

public class emp1 {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

Employee[] employees = new Employee[6];

for (int i = 0; i < 6; i++) {

System.out.println("Enter details for employee " + (i + 1) + ":");

System.out.print("Emp Number: ");

int empNumber = scanner.nextInt();

System.out.print("Name: ");

String name = scanner.next();

System.out.print("Salary: ");

int salary = scanner.nextInt();

employees[i] = new Employee(empNumber, name, salary);

}

System.out.println("Employees with salary between 10000 and 15000:");

for (Employee employee : employees) {

if (employee.salary >= 10000 && employee.salary <= 15000) {

System.out.println(employee.empNumber + " " + employee.name + " " + employee.salary);

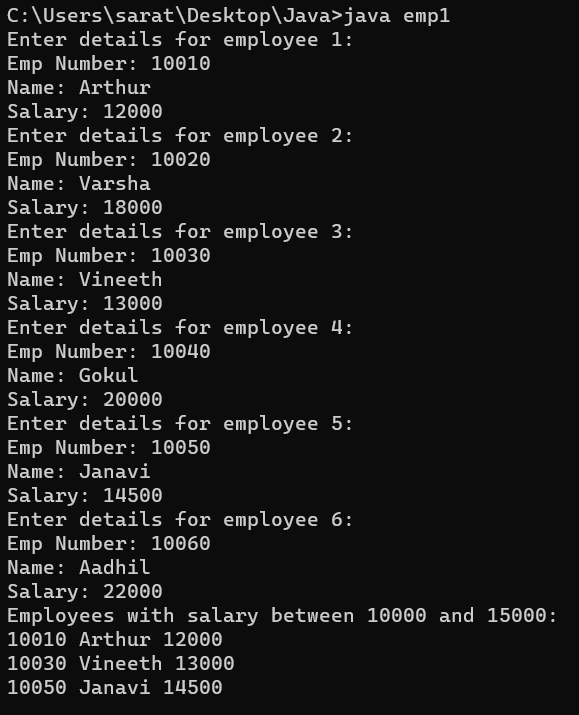
}

}

}

}

**Output Screenshot:**

****

**Result:**

The output was successfully generated and the program was executed. Thus CO1 was attained.

**Experiment-20**

**Aim:**

Define a class account having data members account number name and balance amount, Write java program to accept data of 6 bank accounts & display information of account having balance less than 500 using array of objects.

**CO 1:**

Understand object-oriented concepts and design classes and objects to solve problems.

**Procedure:**

import java.util.Scanner;

class Account {

int accountNumber;

String name;

int balance;

public Account(int accountNumber, String name, int balance) {

this.accountNumber = accountNumber;

this.name = name;

this.balance = balance;

}

}

public class acc {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

Account[] accounts = new Account[6];

for (int i = 0; i < 6; i++) {

System.out.println("Enter details for account " + (i + 1) + ":");

System.out.print("Account Number: ");

int accountNumber = scanner.nextInt();

System.out.print("Name: ");

String name = scanner.next();

System.out.print("Balance: ");

int balance = scanner.nextInt();

accounts[i] = new Account(accountNumber, name, balance);

}

System.out.println("Accounts with balance less than 500:");

for (Account account : accounts) {

if (account.balance < 500) {

System.out.println(account.accountNumber + " " + account.name + " " + account.balance);

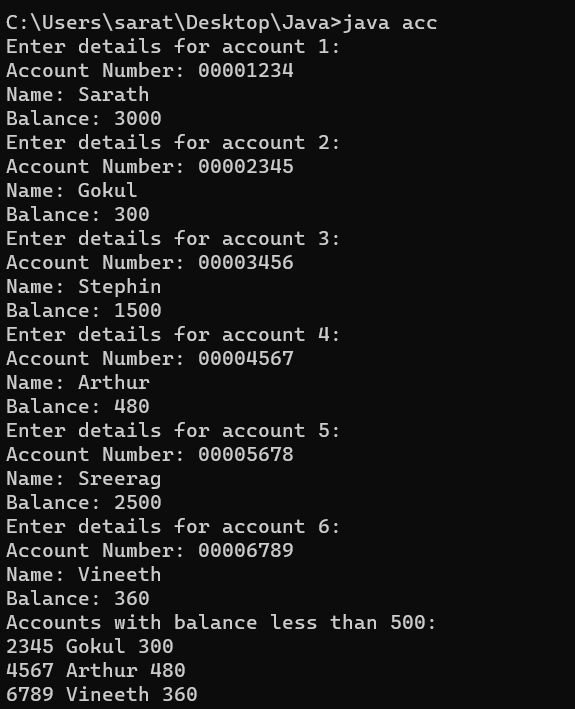
}

}

}

}

**Output Screenshot:**

****

**Result:**

The output was successfully generated and the program was executed. Thus CO1 was attained.

**Experiment-21**

**Aim:**

Create a class ‘Employee’ with data members Empid, Name, Salary, Address and constructors to initialize the data members. Create another class ‘Teacher’ that inherit the properties of class employee and contain its own data members department, Subjects taught and constructors to initialize these data members and also include display function to display all the data members. Use array of objects to display details of N teachers.

**CO 3:**

Implement object-oriented concepts like inheritance, overloading and interfaces

**Procedure:**

import java.util.Scanner;

class Employee {

int empId;

String name;

double salary;

String address;

Employee(int empId, String name, double salary, String address) {

this.empId = empId;

this.name = name;

this.salary = salary;

this.address = address;

}

}

class Teacher extends Employee {

String department;

String subjectsTaught;

Teacher(int empId, String name, double salary, String address, String department, String subjectsTaught) {

super(empId, name, salary, address);

this.department = department;

this.subjectsTaught = subjectsTaught;

}

void display() {

System.out.println("Emp Id: " + empId);

System.out.println("Name: " + name);

System.out.println("Salary: " + salary);

System.out.println("Address: " + address);

System.out.println("Department: " + department);

System.out.println("Subjects taught: " + subjectsTaught);

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

}

}

public class emp3 {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the number of teachers: ");

int n = sc.nextInt();

Teacher[] teachers = new Teacher[n];

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

System.out.println("Enter details of teacher " + (i + 1));

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

int empId = sc.nextInt();

sc.nextLine();

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

String name = sc.nextLine();

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

double salary = sc.nextDouble();

sc.nextLine();

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

String address = sc.nextLine();

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

String department = sc.nextLine();

System.out.print("Enter subjects taught: ");

String subjectsTaught = sc.nextLine();

teachers[i] = new Teacher(empId, name, salary, address, department, subjectsTaught);

}

System.out.println("\nDetails of all teachers:");

for (Teacher t : teachers) {

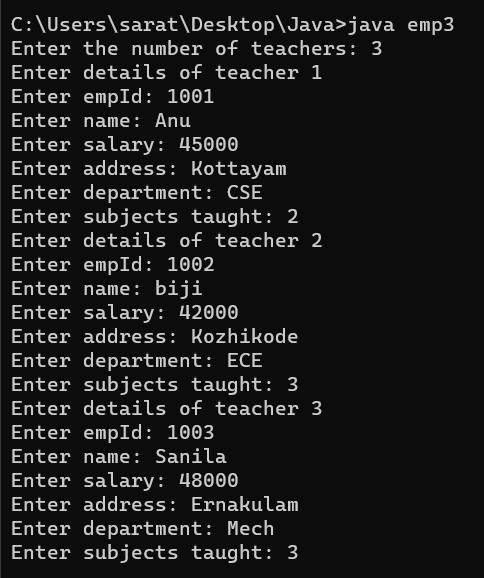
t.display();

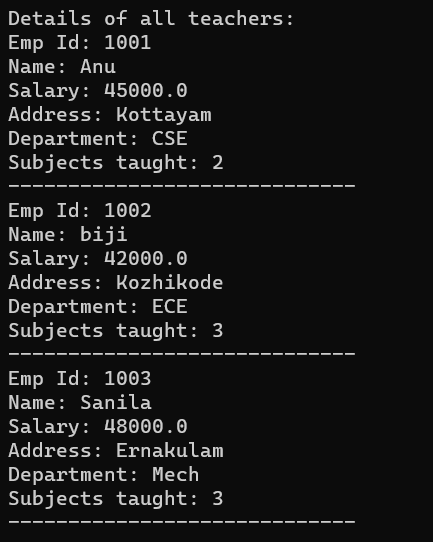
}

}

}

**Output Screenshot:**

****

****

**Result:**

The output was successfully generated and the program was executed. Thus CO3 was attained.

**Experiment-22**

**Aim:**

Create a class ‘Person’ with data members Name, Gender, Address, Age and a constructor to initialize the data members and another class ‘Employee’ that inherits the properties of class Person and also contains its own data members like Empid, Company\_name, Qualification, Salary and its own constructor. Create another class ‘Teacher’ that inherits the properties of class Employee and contains its own data members like Subject, Department, Teacherid and also contain constructors and methods to display the data members. Use array of objects to display details of N teachers.

**CO 3:**

Implement object-oriented concepts like inheritance, overloading and interfaces

**Procedure:**

import java.util.Scanner;

class Person {

String name;

String gender;

String address;

int age;

Person(String name, String gender, String address, int age) {

this.name = name;

this.gender = gender;

this.address = address;

this.age = age;

}

}

class Employee extends Person {

int empId;

String companyName;

String qualification;

double salary;

Employee(String name, String gender, String address, int age, int empId, String companyName, String qualification, double salary) {

super(name, gender, address, age);

this.empId = empId;

this.companyName = companyName;

this.qualification = qualification;

this.salary = salary;

}

}

class Teacher extends Employee {

String subject;

String department;

int teacherId;

Teacher(String name, String gender, String address, int age, int empId, String companyName, String qualification, double salary, String subject, String department, int teacherId) {

super(name, gender, address, age, empId, companyName, qualification, salary);

this.subject = subject;

this.department = department;

this.teacherId = teacherId;

}

void display() {

System.out.println("Name: " + name);

System.out.println("Gender: " + gender);

System.out.println("Address: " + address);

System.out.println("Age: " + age);

System.out.println("Emp Id: " + empId);

System.out.println("Company Name: " + companyName);

System.out.println("Qualification: " + qualification);

System.out.println("Salary: " + salary);

System.out.println("Subject: " + subject);

System.out.println("Department: " + department);

System.out.println("Teacher Id: " + teacherId);

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

}

}

public class persQn {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the number of teachers: ");

int n = sc.nextInt();

Teacher[] teachers = new Teacher[n];

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

System.out.println("Enter details of teacher " + (i + 1));

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

String name = sc.next();

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

String gender = sc.next();

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

String address = sc.next();

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

int age = sc.nextInt();

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

int empId = sc.nextInt();

sc.nextLine();

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

String companyName = sc.nextLine();

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

String qualification = sc.nextLine();

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

double salary = sc.nextDouble();

sc.nextLine();

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

String subject = sc.nextLine();

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

String department = sc.nextLine();

System.out.print("Enter teacher id: ");

int teacherId = sc.nextInt();

teachers[i] = new Teacher(name, gender, address, age, empId, companyName, qualification, salary, subject, department, teacherId);

}

System.out.println("\nDetails of all teachers:");

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

System.out.println("Details of teacher " + (i + 1));

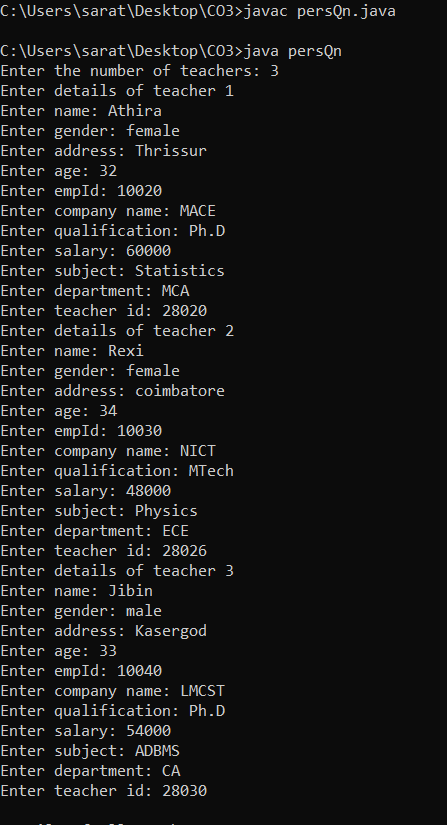
teachers[i].display();

}

}

}

**Output Screenshot:**



**Result:**

The output was successfully generated and the program was executed. Thus CO3 was attained.

**Experiment-23**

**Aim:**

Using overloaded function,Write a java program to perform different types of sum such as

a)sum of two numbers

b)sum of digits of a number

c)sum of 2 strings

**CO 3:**

Implement object-oriented concepts like inheritance, overloading and interfaces

**Procedure:**

import java.util.Scanner;

public class sum {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the first number: ");

int num1 = sc.nextInt();

System.out.print("Enter the second number: ");

int num2 = sc.nextInt();

System.out.println("The sum of two numbers is: " + sum(num1, num2));

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

int num = sc.nextInt();

System.out.println("The sum of digits in the number is: " + sum(num));

System.out.print("Enter the first string: ");

String str1 = sc.next();

System.out.print("Enter the second string: ");

String str2 = sc.next();

System.out.println("The sum of two strings is: " + sum(str1, str2));

sc.close();

}

public static int sum(int num1, int num2) {

return num1 + num2;

}

public static int sum(int num) {

int sum = 0;

while (num > 0) {

sum += num % 10;

num /= 10;

}

return sum;

}

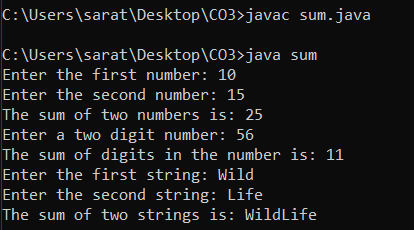
public static String sum(String str1, String str2) {

return str1 + str2;

}

}

**Output Screenshot:**



**Result:**

The output was successfully generated and the program was executed. Thus CO3 was attained.

**Experiment-24**

**Aim:**

Area of different shapes using overloaded functions

**CO 3:**

Implement object-oriented concepts like inheritance, overloading and interfaces

**Procedure:**

import java.util.\*;

public class AreaShape {

static double Triangle(double b, double h) {

return 0.5\*(b\*h);

}

static int Rectangle(int l, int w ) {

return l\*w;

}

static double Circle(double r) {

return Math.PI\*(r\*r);

}

static float Square(float a) {

return a\*a;

}

static int Parallelogram(int b1, int h1) {

return b1\*h1;

}

static double Trapezuim(double a2, double b2, double h2) {

return 0.5\*(a2+b2)\*h2;

}

static double Ellipse(double a3, double b3) {

return Math.PI\*(a3\*b3);

}

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

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

System.out.println("\*\*\*\*\*\*AREA OF SHAPES\*\*\*\*\*\*" );

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

System.out.println("<<< ENTER THE REQUIREMENTS >>>" );

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

System.out.print("\n");

System.out.println("<<<<<<<<<< TRIANGLE >>>>>>>>>>>");

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

System.out.print("Enter the base of the triangle: ");

double b = sc.nextDouble();

System.out.print("Enter the height of the triangle: ");

double h = sc.nextDouble();

double triangleArea = Triangle(b, h);

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

System.out.println("Area of the triangle: " + triangleArea);

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

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

System.out.print("\n");

System.out.println("<<<<<<<<<< RECTANGLE >>>>>>>>>>>");

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

System.out.print("Enter the Length of the Rectangle: ");

int l = sc.nextInt();

System.out.print("Enter the Width of the Rectangle: ");

int w = sc.nextInt();

int rectangleArea = Rectangle(l, w);

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

System.out.println("Area of the rectangle: " + rectangleArea);

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

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

System.out.print("\n");

System.out.println("<<<<<<<<<< CIRCLE >>>>>>>>>>>");

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

System.out.print("Enter the radius of the circle: ");

double r = sc.nextDouble();

double circleArea = Circle(r);

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

System.out.println("Area of the Circle: " + circleArea);

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

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

System.out.print("\n");

System.out.println("<<<<<<<<<< SQUARE >>>>>>>>>>>");

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

System.out.print("Enter the length of the Square: ");

float a = sc.nextFloat();

float squareArea = Square(a);

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

System.out.println("Area of the Square: " + squareArea);

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

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

System.out.print("\n");

System.out.println("<<<<<<<<<< PARALLELOGRAM >>>>>>>>>>>");

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

System.out.print("Enter the base of the Parallelogram: ");

int b1 = sc.nextInt();

System.out.print("Enter the Vertical Height of the Parallelogram: ");

int h1 = sc.nextInt();

int parallelogramArea = Parallelogram(b1, h1);

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

System.out.println("Area of the Parallelogram: " + parallelogramArea);

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

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

System.out.print("\n");

System.out.println("<<<<<<<<<< TRAPEZUIM >>>>>>>>>>>");

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

System.out.print("Enter the parallel side length 1 of the Trapezium: ");

double a2 = sc.nextDouble();

System.out.print("Enter the parallel side length 2 of the Trapezium: ");

double b2 = sc.nextDouble();

System.out.print("Enter the Height of the Trapezuim: ");

double h2 = sc.nextDouble();

double trapezuimArea = Trapezuim(a2, b2, h2);

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

System.out.println("Area of the Trapezium: " + trapezuimArea);

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

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

System.out.print("\n");

System.out.println("<<<<<<<<<< ELLIPSE >>>>>>>>>>>");

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

System.out.print("Enter the minor axis of the Ellipse: ");

double a3 = sc.nextDouble();

System.out.print("Enter the major axis of the Ellipse: ");

double b3 = sc.nextDouble();

double ellipseArea = Ellipse(a3, b3);

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

System.out.println("Area of the Ellipse: " + ellipseArea);

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

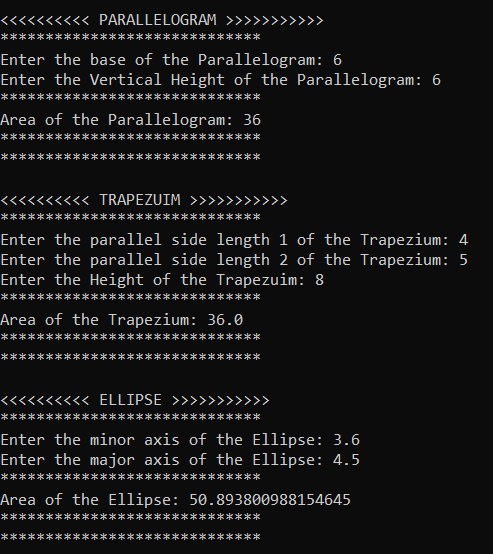
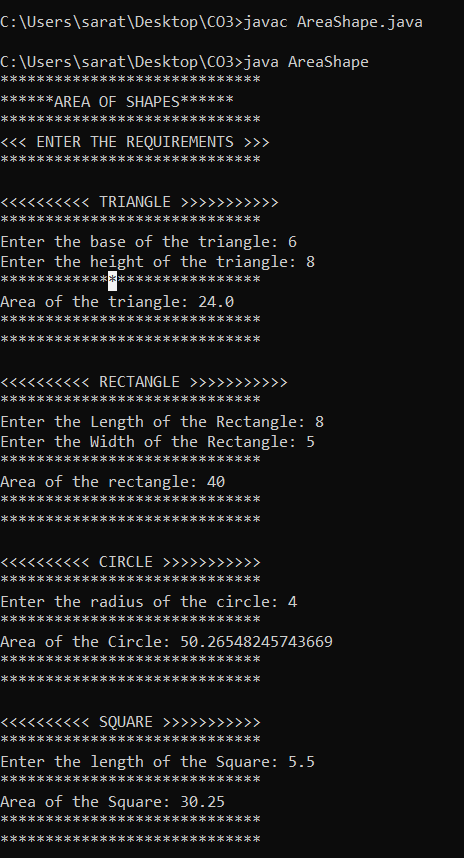
System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

System.out.print("\n");

}

}

**Output Screenshot:**



**Result:**

The output was successfully generated and the program was executed. Thus CO3 was attained.

**Experiment-25**

**Aim:**

Write a java program to define a class ‘student’ having data members rollno, name. Derive a class ‘marks’ from ‘student’ having data members m1, m2, m3, total and percentage. Accept and display data of one student.

**CO 3:**

Implement object-oriented concepts like inheritance, overloading and interfaces

**Procedure:**

import java.util.Scanner;

class Student {

int rollno;

String name;

}

class Marks extends Student {

int m1, m2, m3, total;

double percentage;

void calculateTotal() {

total = m1 + m2 + m3;

}

void calculatePercentage() {

percentage = (double) total / 3.0;

}

void display() {

System.out.println("Roll No: " + rollno);

System.out.println("Name: " + name);

System.out.println("Marks in Subject 1: " + m1);

System.out.println("Marks in Subject 2: " + m2);

System.out.println("Marks in Subject 3: " + m3);

System.out.println("Total Marks: " + total);

System.out.println("Percentage: " + percentage + "%");

}

}

public class stu2 {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

Marks student = new Marks();

System.out.print("Enter Roll No: ");

student.rollno = sc.nextInt();

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

sc.nextLine();

student.name = sc.nextLine();

System.out.print("Enter Marks in Subject 1: ");

student.m1 = sc.nextInt();

System.out.print("Enter Marks in Subject 2: ");

student.m2 = sc.nextInt();

System.out.print("Enter Marks in Subject 3: ");

student.m3 = sc.nextInt();

student.calculateTotal();

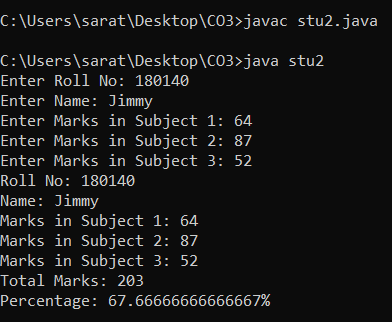
student.calculatePercentage();

student.display();

}

}

**Output Screenshot:**



**Result:**

The output was successfully generated and the program was executed. Thus CO3 was attained.

**Experiment-26**

**Aim:**

Write a java program to define a class ‘employee’ having data members emp\_id, emp\_name and emp\_designation. Derive a class ‘salary’ from ‘employee’ having data members basic, HRA, DA, gross\_salary. Accept and display data of one employee.

DA=basic\*35/100

HRA=basic\*15/100

GS=basic+DA+HRA

**CO 3:**

Implement object-oriented concepts like inheritance, overloading and interfaces

**Procedure:**

import java.util.Scanner;

class Employee {

int emp\_id;

String emp\_name;

String emp\_designation;

void acceptData() {

Scanner sc = new Scanner(System.in);

System.out.print("Enter employee ID: ");

emp\_id = sc.nextInt();

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

emp\_name = sc.next();

System.out.print("Enter employee designation: ");

emp\_designation = sc.next();

}

void displayData() {

System.out.println("Employee ID: " + emp\_id);

System.out.println("Employee name: " + emp\_name);

System.out.println("Employee designation: " + emp\_designation);

}

}

class Salary extends Employee {

double basic;

double HRA;

double DA;

double gross\_salary;

void calculateSalary() {

Scanner sc = new Scanner(System.in);

System.out.print("Enter basic salary: ");

basic = sc.nextDouble();

DA = basic \* 35 / 100;

HRA = basic \* 15 / 100;

gross\_salary = basic + DA + HRA;

}

void displaySalary() {

System.out.println("Basic salary: " + basic);

System.out.println("DA: " + DA);

System.out.println("HRA: " + HRA);

System.out.println("Gross salary: " + gross\_salary);

}

}

public class emp4 {

public static void main(String[] args) {

Salary s = new Salary();

s.acceptData();

s.calculateSalary();

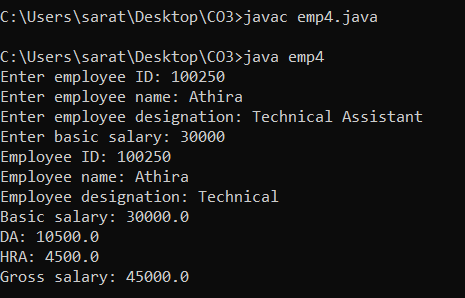
s.displayData();

s.displaySalary();

}

}

**Output Screenshot:**



**Result:**

The output was successfully generated and the program was executed. Thus CO3 was attained.

**Experiment-27**

**Aim:**

Declare a class ‘box’ having data members length, width and height. Derive a class ‘cupboard’ from ‘box’ having data members no of shelves. Wap to accept and display this data for one cupboard object.

**CO 3:**

Implement object-oriented concepts like inheritance, overloading and interfaces

**Procedure:**

import java.util.Scanner;

class Box {

double length;

double width;

double height;

}

class Cupboard extends Box {

int no\_of\_shelves;

void acceptData() {

Scanner sc = new Scanner(System.in);

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

length = sc.nextDouble();

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

width = sc.nextDouble();

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

height = sc.nextDouble();

System.out.print("Enter number of shelves: ");

no\_of\_shelves = sc.nextInt();

}

void displayData() {

System.out.println("Length: " + length);

System.out.println("Width: " + width);

System.out.println("Height: " + height);

System.out.println("Number of shelves: " + no\_of\_shelves);

}

}

public class cupbd {

public static void main(String[] args) {

Cupboard c = new Cupboard();

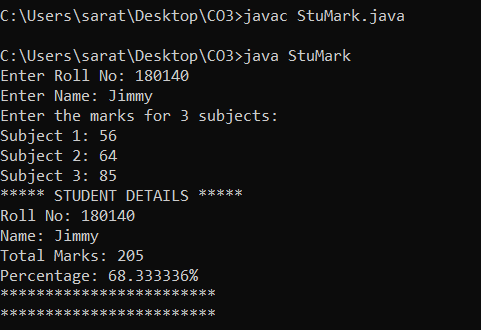
c.acceptData();

c.displayData();

}

}

**Output Screenshot:**



**Result:**

The output was successfully generated and the program was executed. Thus CO3 was attained.