CERTIFICATE

It is to certify the bona fide record of the work done in Object Oriented Programming Through Java Laboratory by Mr. T. Nikhil of Computer Science Engineering Department bearing ID No B161029 during the year 2020-2021.

Signature of the Examiner

Signature of the Branch Coordinator

Signature of the Lab Instructor

ID: B161029

Name: T. Nikhil

Class: AB-II 311

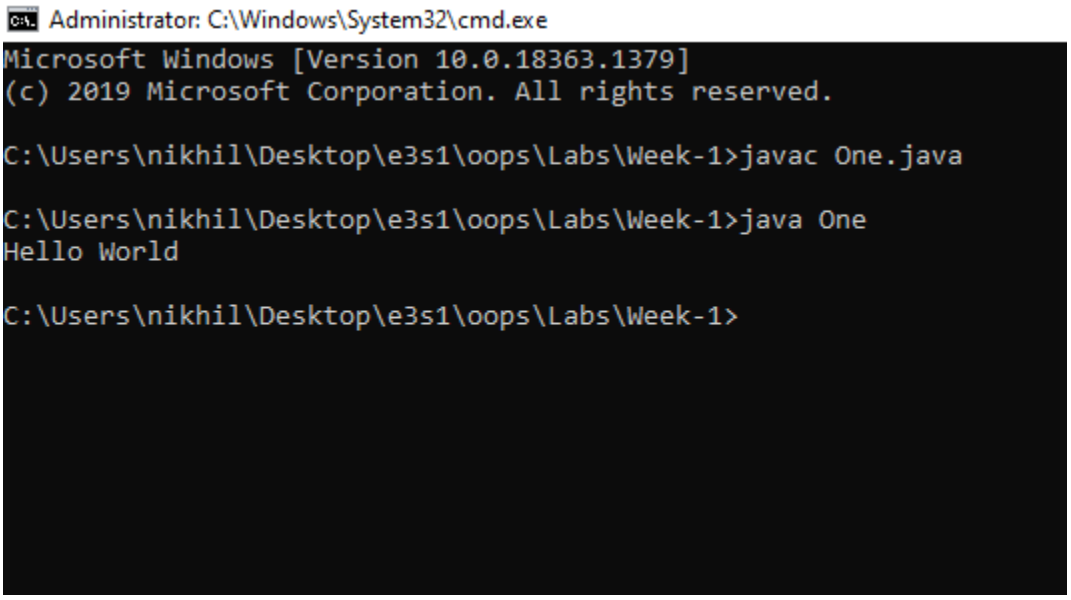
Week-1

1. Write a java program to print "Hello World"

Source Code:

```
class One
{
	public static void main(String args[])
	{
		System.out.println("Hello World");
	}
}
```

Output:



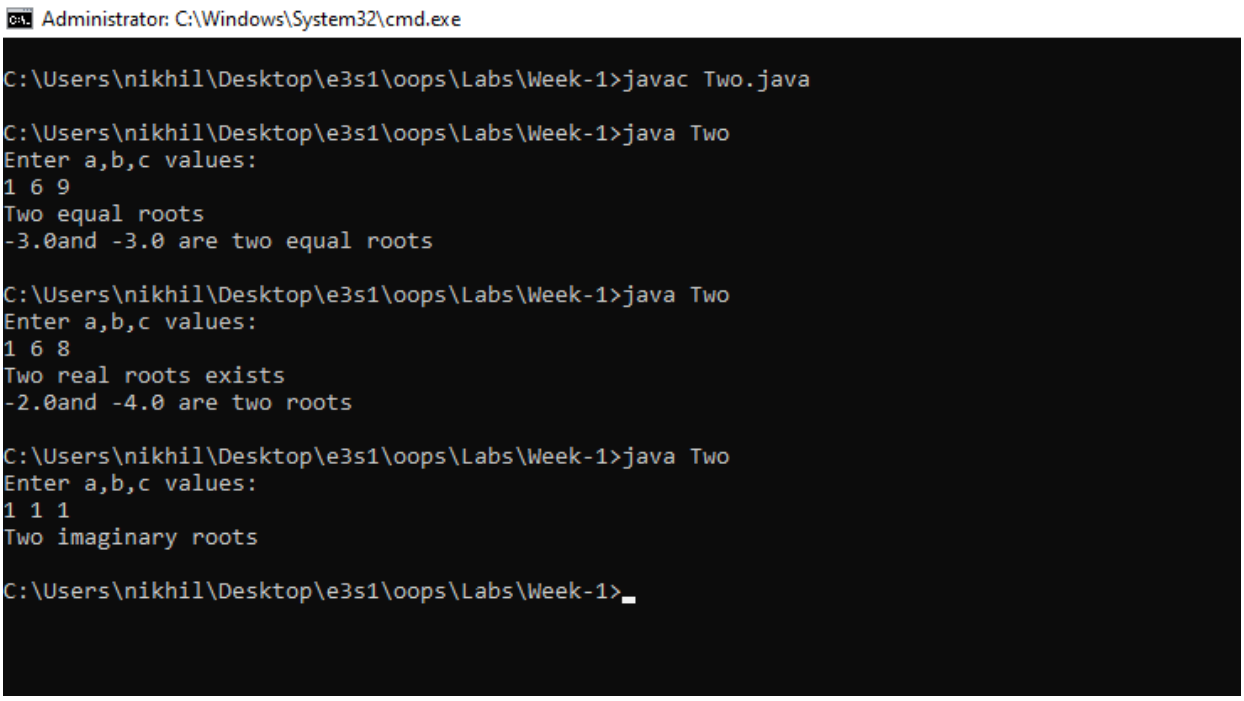
2. Write a java program that prints all real and imaginary solutions to the quadratic equation ax^2+bx+c=0. Read in a, b, c and use the quadratic formula.

Source Code:

```
//Write a java program that prints all real and imaginary solutions to the quadratic equation
ax^2+bx+c=0. Read in a,b,c
// and use the quadratic formula

import java.util.Scanner;
```

```
class Two
{
	public static void main(String args[])
	{
		Scanner sc=new Scanner(System.in);
		double a,b,c;
		System.out.println("Enter a,b,c values:");
		a=sc.nextDouble();
		b=sc.nextDouble();
		c=sc.nextDouble();
		double determinant=(b*b)-(4*a*c);
		if(determinant==0)
		{
			System.out.println("Two equal roots");
			double root1=-b/(2*a);
			System.out.println(root1+ "and " + root1 +" are two equal roots");
		}
		else if(determinant>0)
		{
			System.out.println("Two real roots exists");
			double root1=(-b+Math.sqrt(determinant))/(2*a);
			double root2=(-b-Math.sqrt(determinant))/(2*a);
			System.out.println(root1+ "and " + root2 +" are two roots");
		}
		else
		{
			System.out.println("Two imaginary roots");
		}
	}
}
```

Output:



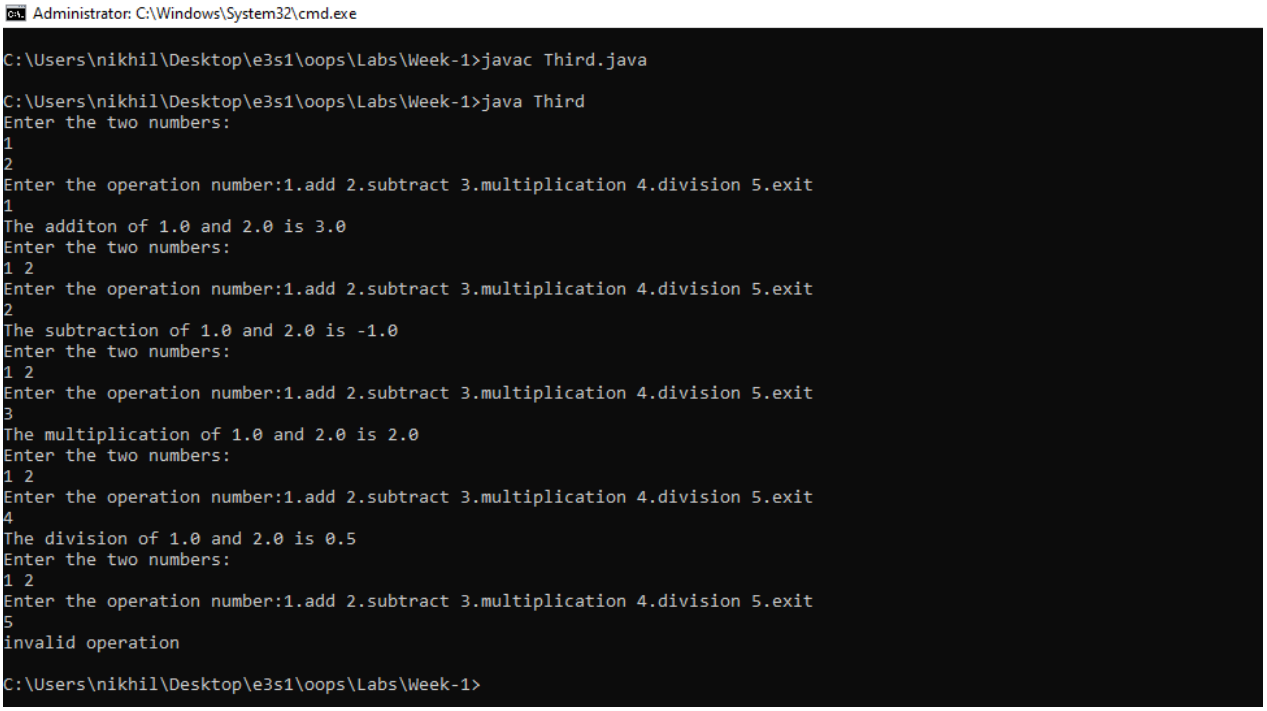
3. Write a Java program to implement calculator operations

Source Code:

```
//. Write a Java program to implement calculator operations

import java.util.Scanner;

class Third
{
	static void addition(double a,double b)
	{
		System.out.println("The additon of "+a+" and "+b+" is "+(a+b));
	}
	static void subtraction(double a,double b)
	{
		System.out.println("The subtraction of "+a+" and "+b+" is "+(a-b));
	}
	static void multiplication(double a,double b)
	{
		System.out.println("The multiplication of "+a+" and "+b+" is "+(a*b));
	}
	static void division(double a,double b)
	{
		if(b!=0)
		{
			System.out.println("The division of "+a+" and "+b+" is "+(a/b));
		}
		else
		{
			System.out.println("The division can't be performed");
		}
	}
	public static void main(String args[])
	{
		Scanner sc=new Scanner(System.in);
		int k;
		do
		{
			double a,b;

			System.out.println("Enter the two numbers:");
			a=sc.nextDouble();
			b=sc.nextDouble();
			System.out.println("Enter the operation number:1.add 2.subtract
3.multiplication 4.division 5.exit");
			k=sc.nextInt();
			switch(k)
			{
				case 1: addition(a,b);
					break;
				case 2: subtraction(a,b);
```

```
                                        break;
                                case 3:multiplication(a,b);
                                        break;
                                case 4:division(a,b);
                                        break;
                                default: System.out.println("invalid operation");
                        }
                }while(k!=5);
        }
}
```

Output:



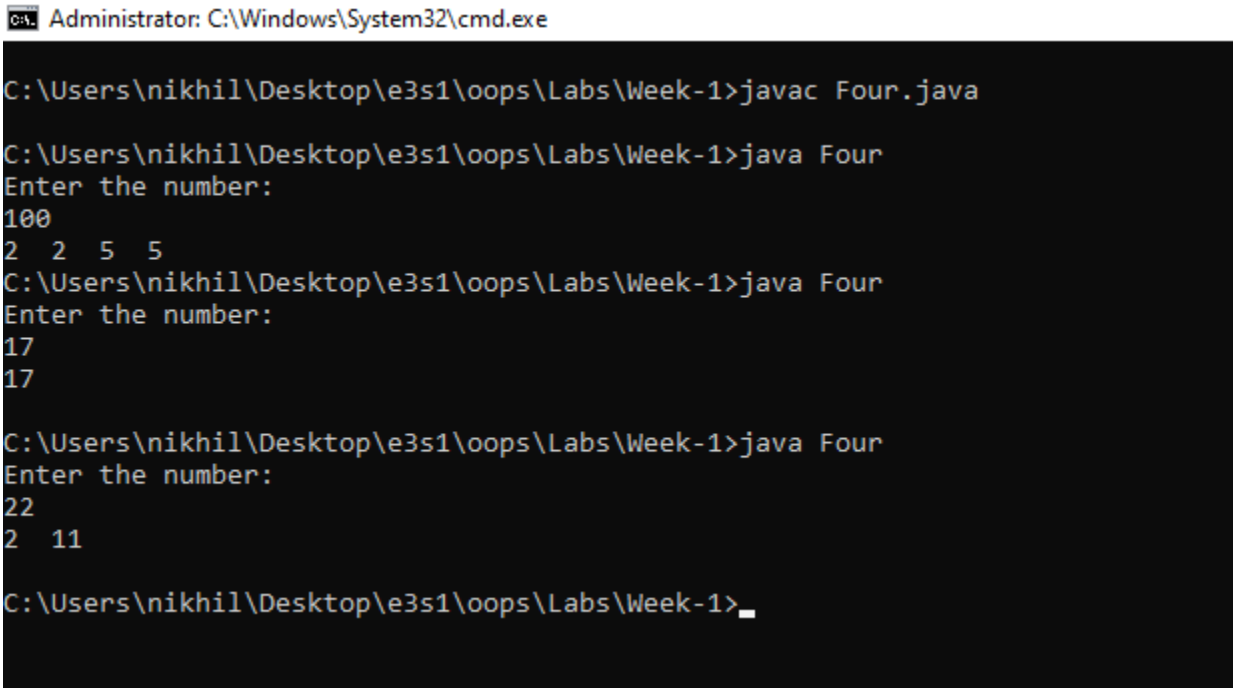
4. Write a java program to find prime factors of given number

Source Code

```
//Write a java program to find prime factors of given number

import java.util.Scanner;

class Four
{
        static void printPrimeFactor(int n)
        {
                //step-1
                while(n%2==0)
                {
                        System.out.print("2  ");
                        n=n/2;
                }
                //step-2
                int i;
```

```
                        for(i=3;i<=Math.sqrt(n);i+=2)
                        {
                                while(n%i==0)
                                {
                                        System.out.print(i+"  ");
                                        n=n/i;
                                }
                        }
                        //step-3
                        if(n>2)
                        {
                                System.out.println(n);
                        }
                }
                public static void main(String args[])
                {
                        Scanner sc=new Scanner(System.in);
                        int a,b,c;
                        System.out.println("Enter the number:");
                        a=sc.nextInt();
                        printPrimeFactor(a);
                }
}
```

Output:



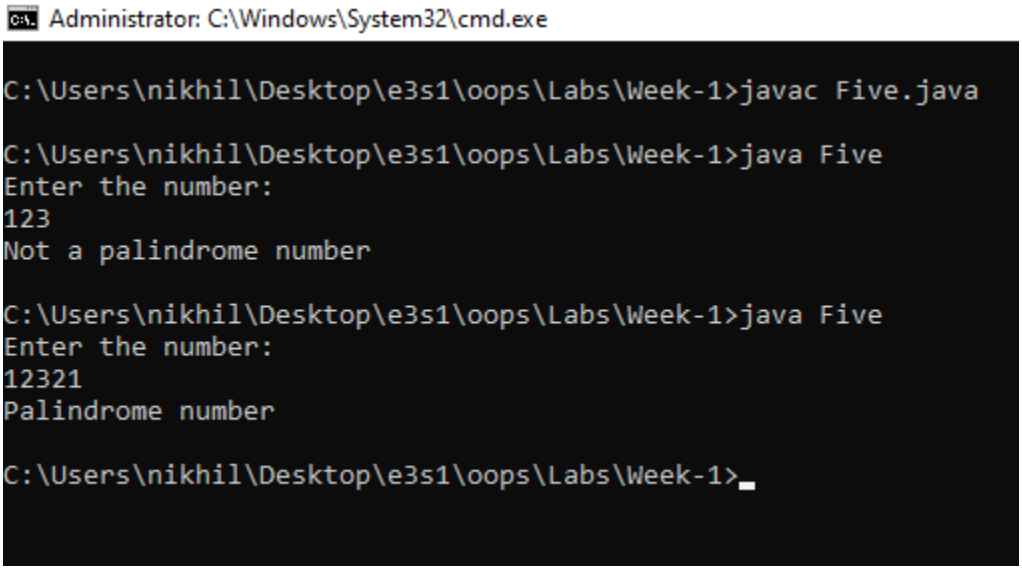
5. Write a java program to find whether given number is Palindrome or not

Source Code:

```
        //Write a java program to find whether given number is Palindrome or not
```

```
import java.util.Scanner;

class Five
{
	public static void main(String args[])
	{
		Scanner sc=new Scanner(System.in);
		int a,b,c=0,d;
		System.out.println("Enter the number:");
		a=sc.nextInt();
		b=a;
		while(b!=0)
		{
			d=b%10;
			c=c*10+d;
			b=b/10;
		}
		if(c==a)
			System.out.println("Palindrome number");
		else
			System.out.println("Not a palindrome number");
	}
}
```

Output:



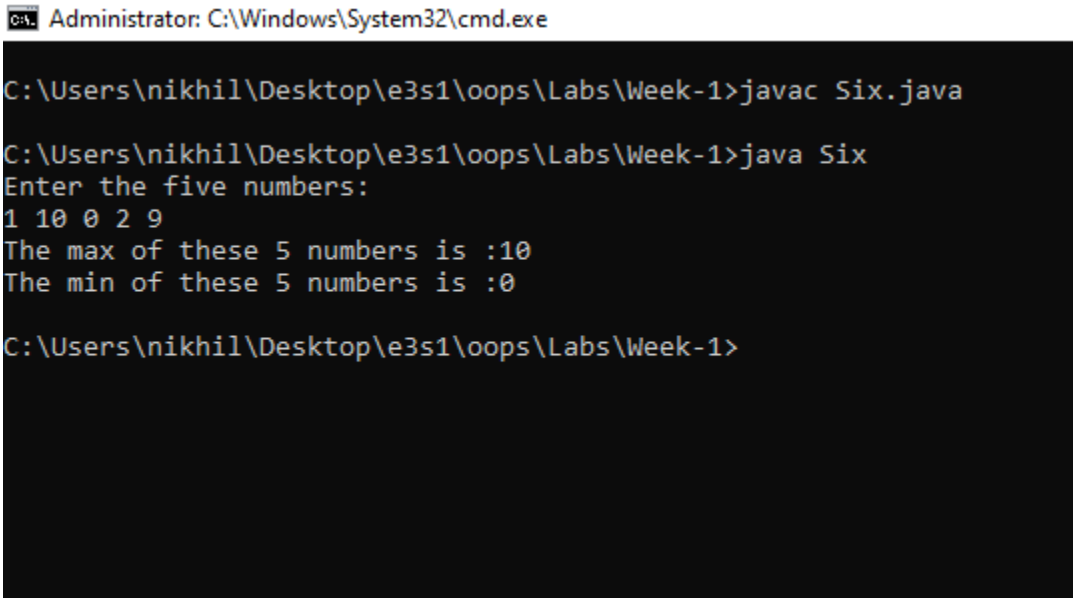
6. Write an application that declares 5 integers, determines and prints the largest and smallest in the group.

Source Code

```
// Write an application that declares 5 integers, determines and prints the largest and smallest in the group.

import java.util.Scanner;
```

```
class Six
{
	public static void main(String args[])
	{
		Scanner sc=new Scanner(System.in);
		int arr[]=new int [5];
		System.out.println("Enter the five numbers:");
		int i,min,max;
		for(i=0;i<5;i++)
			arr[i]=sc.nextInt();
		min=arr[0];
		max=arr[0];
		for(i=1;i<5;i++)
		{
			if(arr[i]>max)
				max=arr[i];
			if(arr[i]<min)
				min=arr[i];
		}
		System.out.println("The max of these 5 numbers is :"+max);
		System.out.println("The min of these 5 numbers is :"+min);
	}
}
```

Output:



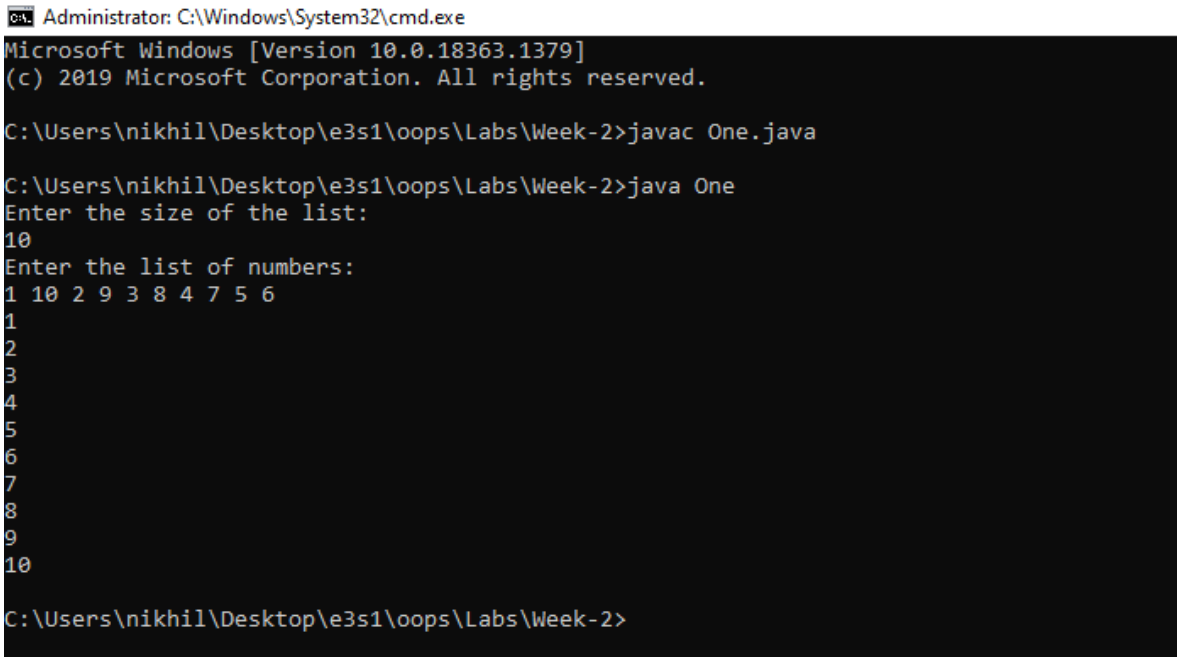
Week-2

1. Write a Java program to sort given list of numbers.

Source Code

```
// Write a Java program to sort given list of numbers.

import java.util.Scanner;

class One
{
	public static void main(String args[])
	{
		Scanner sc=new Scanner(System.in);
		int n,i,j;
		System.out.println("Enter the size of the list:");
		n=sc.nextInt();
		int arr[]=new int[n];
		System.out.println("Enter the list of numbers:");
		for(i=0;i<n;i++)
			arr[i]=sc.nextInt();
		//bubble sort
		for(i=0;i<n-1;i++)
		{
			for(j=0;j<n-i-1;j++)
			{
				if(arr[j]>arr[j+1])
					arr[j]=arr[j]+arr[j+1]-(arr[j+1]=arr[j]);
			}
		}
		for(int ele:arr)
		{
			System.out.println(ele);
		}
	}
}
```

Output:

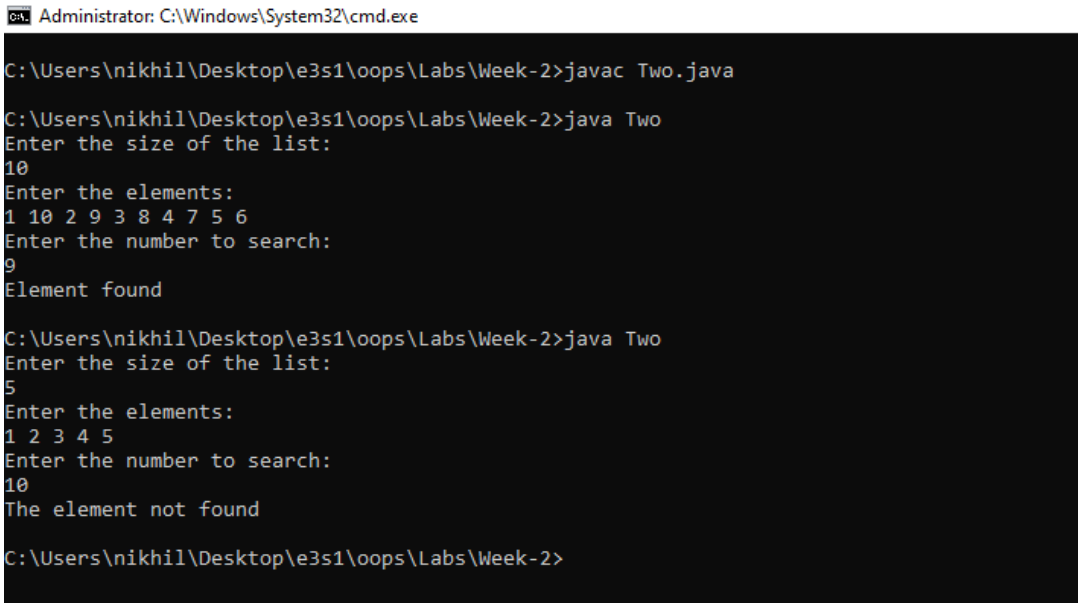


2. Write a Java program to implement linear search.

Source Code:

```
//Write a Java program to implement linear search.

import java.util.Scanner;

class Two
{
	public static void main(String args[])
	{
		Scanner sc=new Scanner(System.in);
		int n,i,j,flag=0;
		System.out.println("Enter the size of the list:");
		n=sc.nextInt();
		int arr[]=new int[n];
		System.out.println("Enter the elements:");
		for(i=0;i<n;i++)
			arr[i]=sc.nextInt();
		System.out.println("Enter the number to search:");
		j=sc.nextInt();
		for(int ele:arr)
		{
			if(ele==j)
			{
				System.out.println("Element found");
				flag=1;
				break;
			}
		}
		if(flag==0)
			System.out.println("The element not found");
	}
}
```

Output:



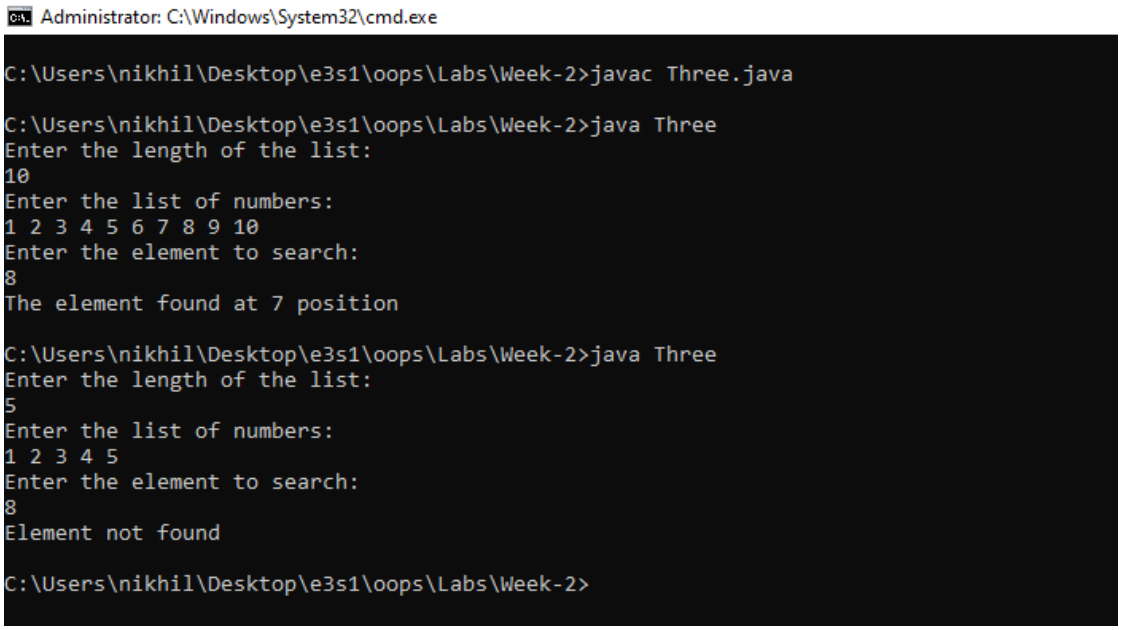
3. Write a Java program to implement binary search.

Source Code:

```
//Write a Java program to implement binary search.

import java.util.Scanner;

class Three
{
	static int binarySearch(int arr[],int n,int ele)
	{
		int start=0,end=n-1,mid;
		while(start<=end)
		{
			mid=(start+end)/2;
			if(arr[mid]==ele)
				return mid;
			if(arr[mid]>ele)
				end=mid-1;
			else
				start=mid+1;
		}
		return -1;
	}
	public static void main(String args[])
	{
		Scanner sc=new Scanner(System.in);
		int n,i,j;
		System.out.println("Enter the length of the list:");
		n=sc.nextInt();
```

```
                        System.out.println("Enter the list of numbers:");
                        int arr[]=new int[n];
                        for(i=0;i<n;i++)
                                arr[i]=sc.nextInt();
                        System.out.println("Enter the element to search:");
                        j=sc.nextInt();
                        j=binarySearch(arr,n,j);
                        if(j!=-1)
                                System.out.println("The element found at "+j+" position");
                        else
                                System.out.println("Element not found");
                }
}
```

Output:



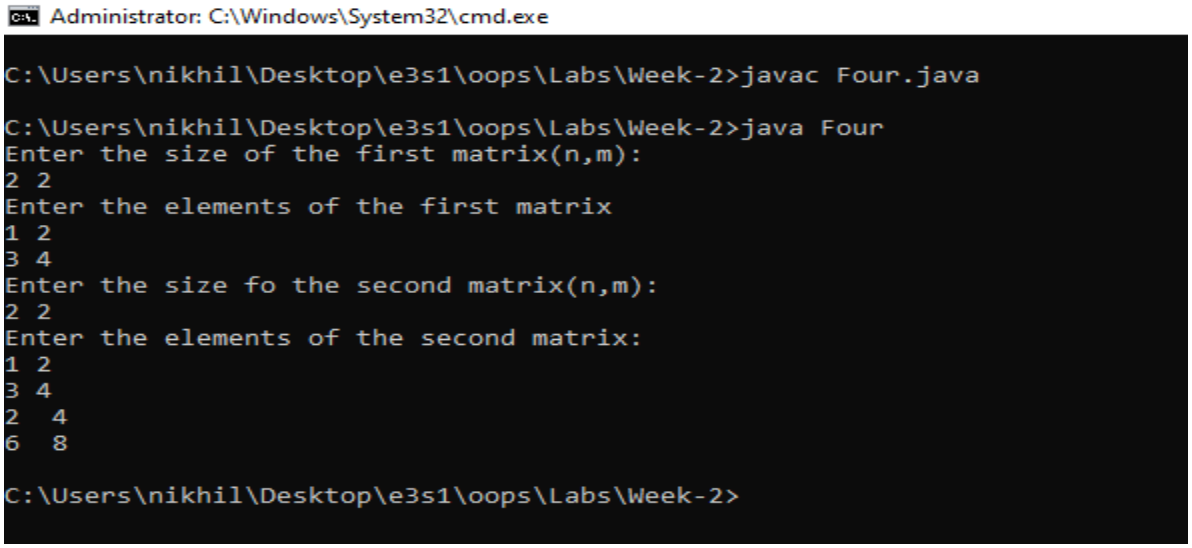
4. Write a java program to add two given matrices.

Source Code:

```
        //Write a java program to add two given matrices.

        import java.util.Scanner;

        class Four
        {
                public static void main(String args[])
                {
                        Scanner sc= new Scanner(System.in);
                        int n,m,a,b,i,j,k;
                        System.out.println("Enter the size of the first matrix(n,m):");
                        n=sc.nextInt();
                        m=sc.nextInt();
```

```
                        int arr[][]=new int [n][m];
                        System.out.println("Enter the elements of the first matrix");
                        for(i=0;i<n;i++)
                                for(j=0;j<n;j++)
                                        arr[i][j]=sc.nextInt();
                        System.out.println("Enter the size fo the second matrix(n,m):");
                        a=sc.nextInt();
                        b=sc.nextInt();
                        int brr[][]=new int[a][b];
                        System.out.println("Enter the elements of the second matrix:");
                        for(i=0;i<a;i++)
                                for(j=0;j<b;j++)
                                        brr[i][j]=sc.nextInt();
                        if(n==a && m==b)
                        {
                                for(i=0;i<n;i++)
                                        for(j=0;j<m;j++)
                                                arr[i][j]+=brr[i][j];
                                for(i=0;i<n;i++)
                                {
                                        for(j=0;j<m;j++)
                                        {
                                                System.out.print(arr[i][j]+"  ");
                                        }
                                        System.out.println();
                                }
                        }
                        else
                        {
                                System.out.println("Addition can't be performed as size of the matrix are
        different");
                        }
                }
}
```

Output:



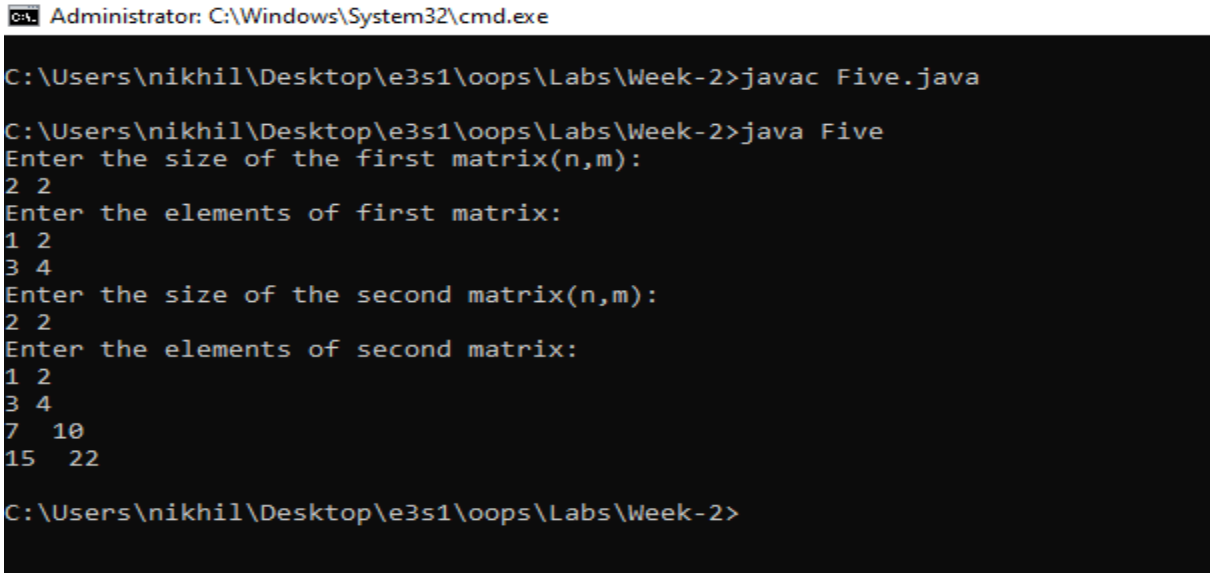
5. Write a java program to multiply two given matrices.

Source Code:

```
//Write a java program to multiply two given matrices.

import java.util.Scanner;

class Five
{
	public static void main(String args[])
	{
		Scanner sc=new Scanner(System.in);
		System.out.println("Enter the size of the first matrix(n,m):");
		int a,b,c,d,i,j,k;
		a=sc.nextInt();
		b=sc.nextInt();
		System.out.println("Enter the elements of first matrix:");
		int arr[][]=new int[a][b];
		for(i=0;i<a;i++)
		{
			for(j=0;j<b;j++)
			{
				arr[i][j]=sc.nextInt();
			}
		}
		System.out.println("Enter the size of the second matrix(n,m):");
		c=sc.nextInt();
		d=sc.nextInt();
		System.out.println("Enter the elements of second matrix:");
		int brr[][]=new int[c][d];
		for(i=0;i<c;i++)
		{
			for(j=0;j<d;j++)
			{
				brr[i][j]=sc.nextInt();
			}
		}
		if(b==c)
		{
			int crr[][]=new int[a][d];
			for(i=0;i<a;i++)
			{
				for(j=0;j<d;j++)
				{
					crr[i][j]=0;
					for(k=0;k<c;k++)
					{
						crr[i][j]+=arr[i][k]*brr[k][j];
					}
				}
			}
			for(i=0;i<a;i++)
			{
```

```
                                        for(j=0;j<d;j++)
                                        {
                                                System.out.print(crr[i][j]+"  ");
                                        }
                                        System.out.println();
                                }
                        }
                        else
                        {
                                System.out.println("The multiplication can't be performed");
                        }
                }
}
```

Output:



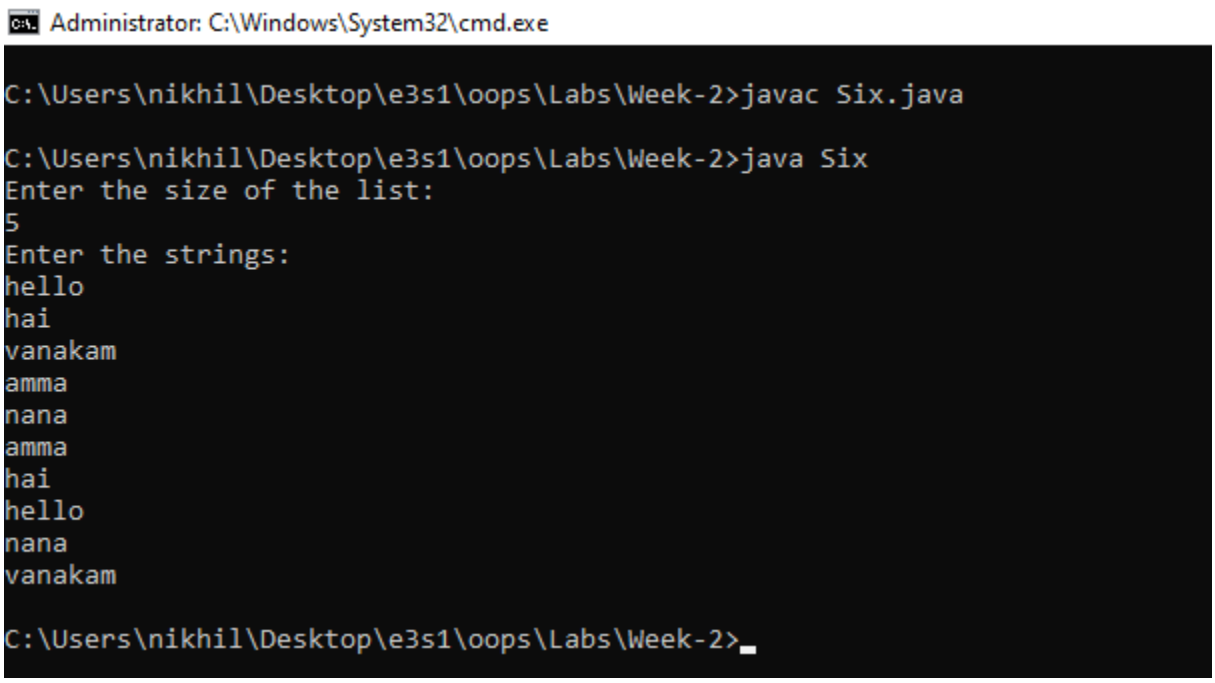
6. Write a java program for sorting a given list of names.

Source code:

```
        //Write a java program for sorting a given list of names.

        import java.util.Scanner;

        class Six
        {
                public static void main(String args[])
                {
                        Scanner sc=new Scanner(System.in);
                        int n,i,j;
                        System.out.println("Enter the size of the list:");
                        n=sc.nextInt();
                        String s[]=new String[n];
                        System.out.println("Enter the strings:");
                        for(i=0;i<n;i++)
```

```
                        s[i]=sc.next();
                for(i=0;i<n-1;i++)
                {
                        for(j=0;j<n-i-1;j++)
                        {
                                int k=s[j].compareTo(s[j+1]);
                                if(k>0)
                                {
                                        String temp=s[j];
                                        s[j]=s[j+1];
                                        s[j+1]=temp;
                                }
                        }
                }
                for(i=0;i<n;i++)
                {
                        System.out.println(s[i]);
                }
        }
}
```

Output:



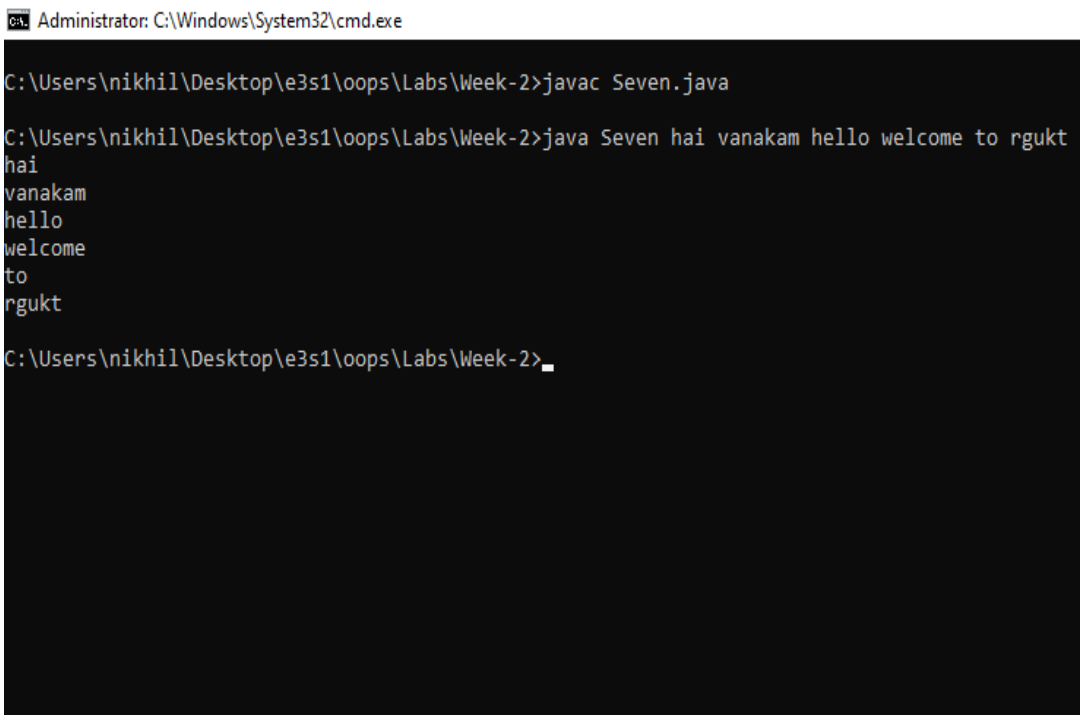
7. Write a Java program to give an example for command line arguments

Source Code:

```
// Write a Java program to give an example for command line arguments

import java.util.Scanner;
```

```
class Seven
{
	public static void main(String args[])
	{
		Scanner sc=new Scanner(System.in);
		int n=args.length,i;
		for(i=0;i<n;i++)
		{
			System.out.println(args[i]);
		}
	}
}
```

Output:



Week-3

1. Write a program to display details of the required employee based on his Id. The details of employee include, Emp_name, Emp_age, Emp_gender, Emp_designation, Emp_salary, Emp_Address etc.,

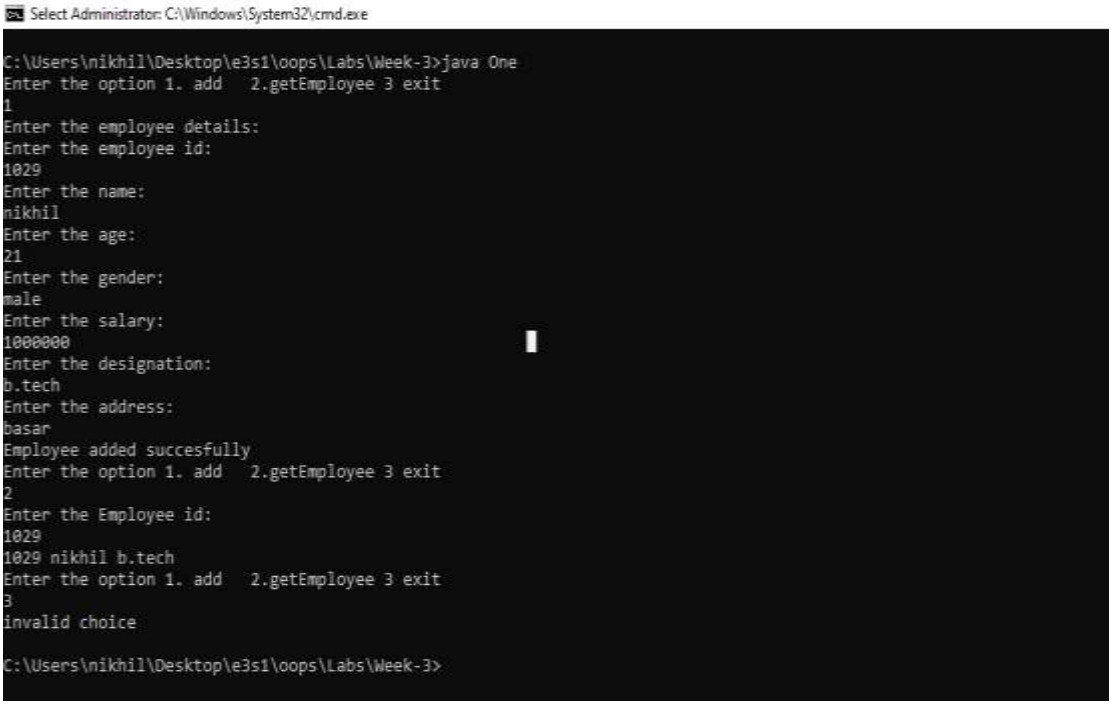
Source Code:

```
// Write a program to display details of the required employee based on his Id. The
//details of employee includes, Emp_name, Emp_age, Emp_gender, Emp_designation,
//Emp_salary, Emp_Address etc.,

import java.util.*;

class Employee
{
	int emp_id,emp_age,emp_salary;
	String emp_name,emp_gender,emp_designation,emp_address;
	Employee()
	{
	}
	Employee(int id,String name,int age,String gender,String designation,int salary,String
address)
	{
		this.emp_id=id;
		this.emp_name=name;
		this.emp_age=age;
		this.emp_gender=gender;
		this.emp_designation=designation;
		this.emp_salary=salary;
		this.emp_address=address;
	}

}
class List
{
	Vector <Employee> v=new Vector<Employee>();
	void add_Employee(Employee e)
	{
		v.add(e);
	}
	Employee get_Employee(int id)
	{
		for(Employee e :v)
		{
			if(e.emp_id==id)
			{
				return e;
			}
		}
		return new Employee();
```

```
		}
	}

	class One
	{

		public static void main(String args[])
		{
			List LL=new List();
			Scanner sc=new Scanner(System.in);
			Employee ee;
			int id,age,salary,k;
			String name,gender,designation,address;
			do
			{
				System.out.println("Enter the option 1. add   2.getEmployee 3 exit");
				k=Integer.parseInt(sc.nextLine());
				switch(k)
				{
					case 1: System.out.println("Enter the employee details:");
						System.out.println("Enter the employee id:");
						id=Integer.parseInt(sc.nextLine());
						System.out.println("Enter the name:");
						name=sc.nextLine();
						System.out.println("Enter the age:");
						age=Integer.parseInt(sc.nextLine());
						System.out.println("Enter the gender:");
						gender=sc.nextLine();
						System.out.println("Enter the salary:");
						salary=Integer.parseInt(sc.nextLine());
						System.out.println("Enter the designation:");
						designation=sc.nextLine();
						System.out.println("Enter the address:");
						address=sc.nextLine();

						ee=new
	Employee(id,name,age,gender,designation,salary,address);
						LL.add_Employee(ee);
						System.out.println("Employee added succesfully");
						break;

					case 2:
						System.out.println("Enter the Employee id:");
						id=Integer.parseInt(sc.nextLine());
						ee=LL.get_Employee(id);
						if(ee.emp_id==0)
						{
							System.out.println("No employee found");
						}
						else
						{
							System.out.println(ee.emp_id+"
	"+ee.emp_name+" "+ee.emp_designation);
```

```
                                        }
                                        break;
                                default :
                                        System.out.println("invalid choice");
                                }
                        }while(k!=3);
                }
        }
```

Output:



2. A mail-order house sells five products whose retail prices are as follows : Product 1 : Rs. 99.90 , Product 2 : Rs. 20.20 , Product 3 : Rs. 6.87 , Product 4 : Rs. 45.50 and Product 5 : Rs. 40.49 . Each product has Prdouct_Id, Product_Name, Product_Quantity, Product_Price. Write an application that reads a series of pairs of numbers as follows :
   a) product Id
   b) quantity sold your program use a switch statement to determine the retail price for each product. it should calculate and display the total retail value of all products sold.

Source Code:

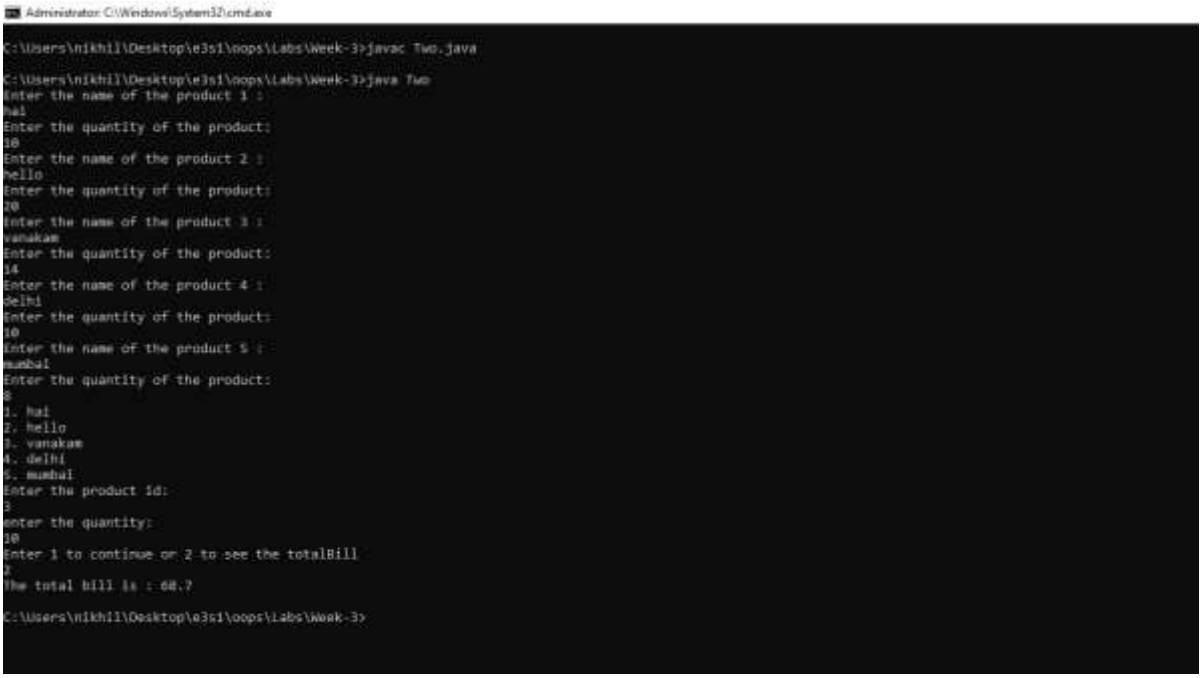
```
        //A mail-order house sells five products whose retail prices are as follows : Product 1 :
        //Rs. 99.90 , Product 2 : Rs. 20.20 , Product 3 : Rs. 6.87 , Product 4 : Rs. 45.50 and
        //Product 5 : Rs. 40.49 . Each product has Prdouct_Id, Product_Name,
        //Product_Quantity, Product_Price. Write an application that reads a series of pairs of
        //numbers as follows :
        //a) product Id
        //b) quantity sold your program use a switch statement to determine the retail price for
        //each product. it should calculate and display the total retail value of all products sold.
```

```
import java.util.*;


class Product
{
	int pId,pQuantity;
	String pName;
	double pPrice;
	Product(int id,String name,int quantity,double price)
	{
		this.pId=id;
		this.pPrice=price;
		this.pName=name;
		this.pQuantity=quantity;
	}
	int decrementQuantity(int q)
	{
		if(this.pQuantity<q)
		{
			System.out.println("The required quantity is not available");
			return -1;
		}
		else
		{
			this.pQuantity-=q;
			return 1;
		}
	}
}
class Two
{
	public static void main(String args[])
	{
		Product P[]=new Product[5];
		double Pprice[]=new double[5],price,totalPrice=0,k;
		Pprice[0]=99.90;
		Pprice[1]=20.20;
		Pprice[2]=6.87;
		Pprice[3]=45.50;
		Pprice[4]=40.49;
		String name;
		int quantity;
		Scanner sc=new Scanner(System.in);
		int i,j;
		//seller journey
		for(i=0;i<5;i++)
		{
			System.out.println("Enter the name of the product "+(i+1)+" :");
			name=sc.nextLine();
			System.out.println("Enter the quantity of the product:");
			quantity=sc.nextInt();
			P[i]=new Product(i+1,name,quantity,Pprice[i]);
```

```
            name=sc.nextLine();
        }

        //customer journey
        do
        {
            for(i=0;i<5;i++)
            {
                System.out.println((i+1) + ". "+P[i].pName);
            }
            System.out.println("Enter the product id:");
            j=sc.nextInt();
            System.out.println("enter the quantity:");
            quantity=sc.nextInt();
            switch(j)
            {
                case 1:
                    k=P[0].decrementQuantity(quantity);
                    if(k==1)
                    {
                        totalPrice+=quantity*P[0].pPrice;
                    }
                    break;
                case 2:
                    k=P[1].decrementQuantity(quantity);
                    if(k==1)
                    {
                        totalPrice+=quantity*P[1].pPrice;
                    }
                    break;
                case 3:
                    k=P[2].decrementQuantity(quantity);
                    if(k==1)
                    {
                        totalPrice+=quantity*P[2].pPrice;
                    }
                    break;
                case 4:
                    k=P[3].decrementQuantity(quantity);
                    if(k==1)
                    {
                        totalPrice+=quantity*P[3].pPrice;
                    }
                    break;
                case 5:
                    k=P[4].decrementQuantity(quantity);
                    if(k==1)
                    {
                        totalPrice+=quantity*P[4].pPrice;
                    }
                    break;
                default:
                    System.out.println("invalid choice");
```

```
                                        }
                                        System.out.println("Enter 1 to continue or 2 to see the totalBill");
                                        k=sc.nextInt();
                                }while(k!=2);

                                System.out.println("The total bill is : "+ totalPrice);
                        }
}
```

Output:



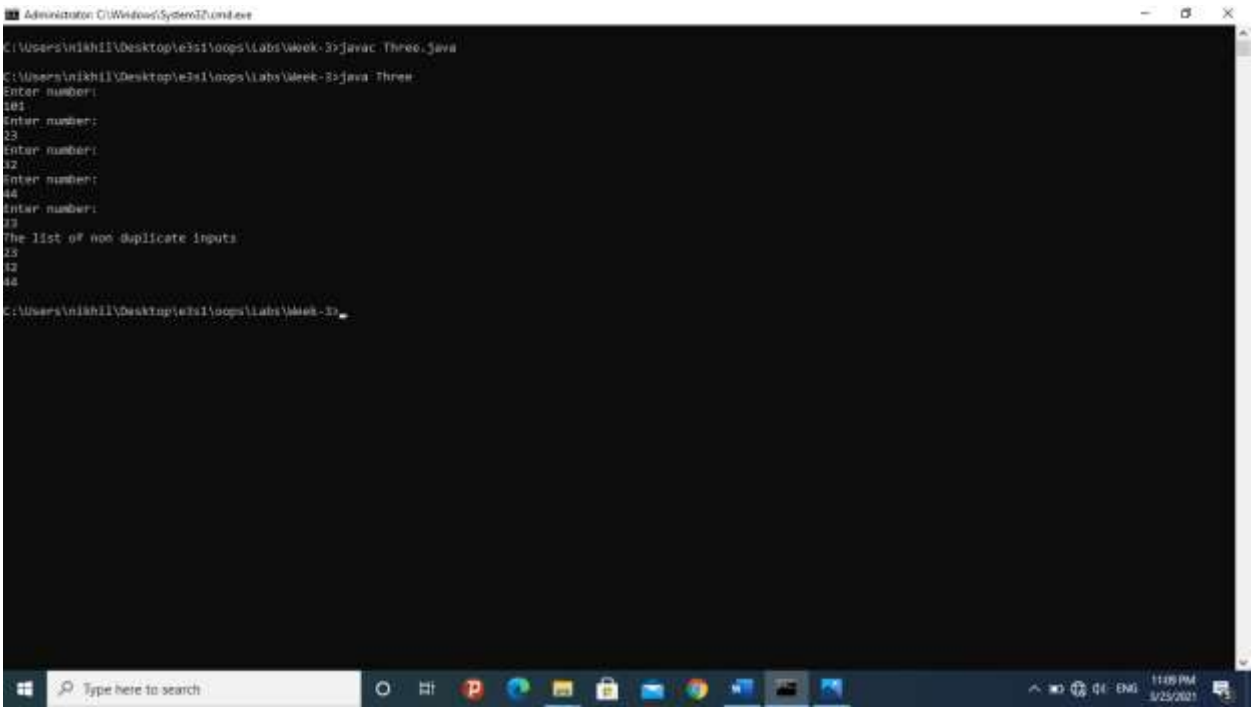
3. Write java program that inputs 5 numbers, each between 10 and 100 inclusive. As each number is read display it only if it's not a duplicate of any number already read display the complete set of unique values input after the user enters each new value

Source Code:

```
// Write java program that inputs 5 numbers, each between 10 and 100 inclusive. As
//each number is read display it only if it's not a duplicate of any number already read
//display the complete set of unique values input after the user enters each new value


import java.util.Scanner;

class Three
{
        public static void main(String args[])
        {
                int arr[]=new int[5];
                int start=0,i,j,k;
                Scanner sc=new Scanner(System.in);
                for(i=0;i<5;i++)
```

```
                        {
                                System.out.println("Enter number:");
                                j=sc.nextInt();
                                for(k=0;k<start;k++)
                                {
                                        if(arr[k]==j)
                                        {
                                                j=-1;
                                                break;
                                        }
                                }
                                if(j!=-1 && j>=10 && j<=100)
                                {
                                        arr[start]=j;
                                        start+=1;
                                }
                        }
                        System.out.println("The list of non duplicate inputs");
                        for(i=0;i<start;i++)
                                System.out.println(arr[i]);
                }
}
```

Output:

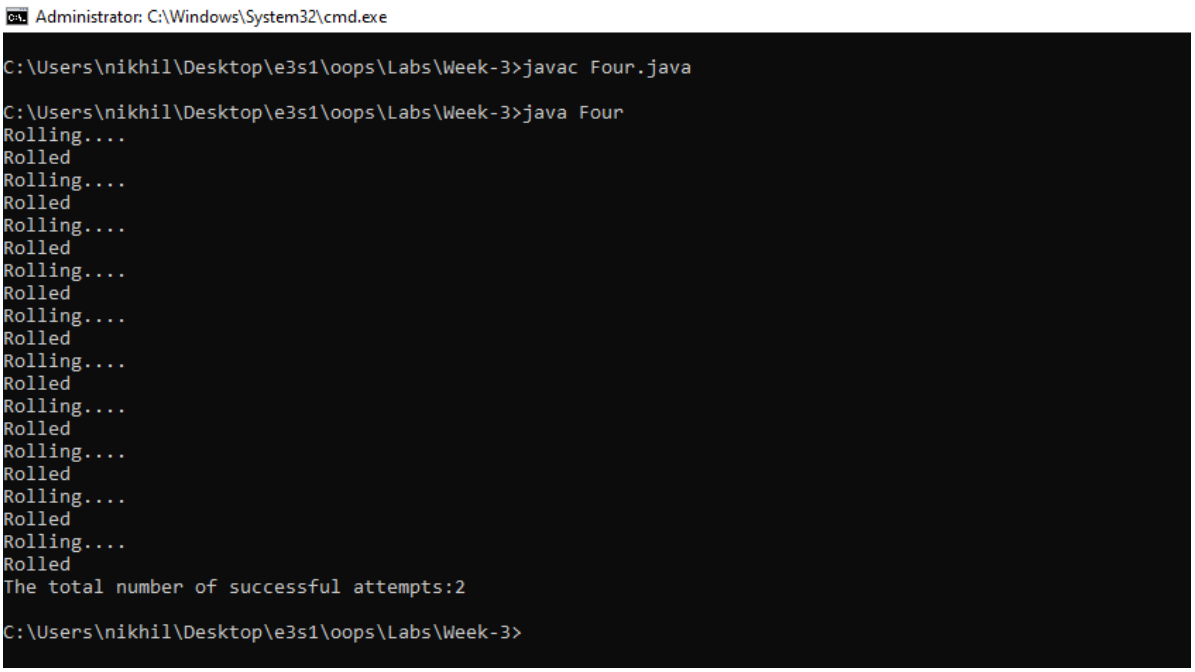


4. Write a java program : rolling a pair of dices 10 times [each attempt should be delayed by 10000 ms] and count number Successful attempts. successful attempt : If the pair of Dice results in same values.

Source Code:

```
//Write a java program : rolling a pair of dices 10 times [ each attempt should be delayed
//by 10000 ms ] and count number Successful attempts. successful attempt : If the pair of
//Dice results in same values.


import java.util.*;


class Four
{
	public static void main(String args[]) throws Exception
	{
		Random rand=new Random();
		int a,b,count=0;
		int success=0;
		while(count!=10)
		{
			System.out.println("Rolling....");
			Thread.sleep(10000);
			a=rand.nextInt(6)+1;
			b=rand.nextInt(6)+1;
			if(a==b)
			{
				success++;
			}
			count++;
			System.out.println("Rolled");
		}
		System.out.println("The total number of successful attempts:"+success);
	}
}
```

Outptut:



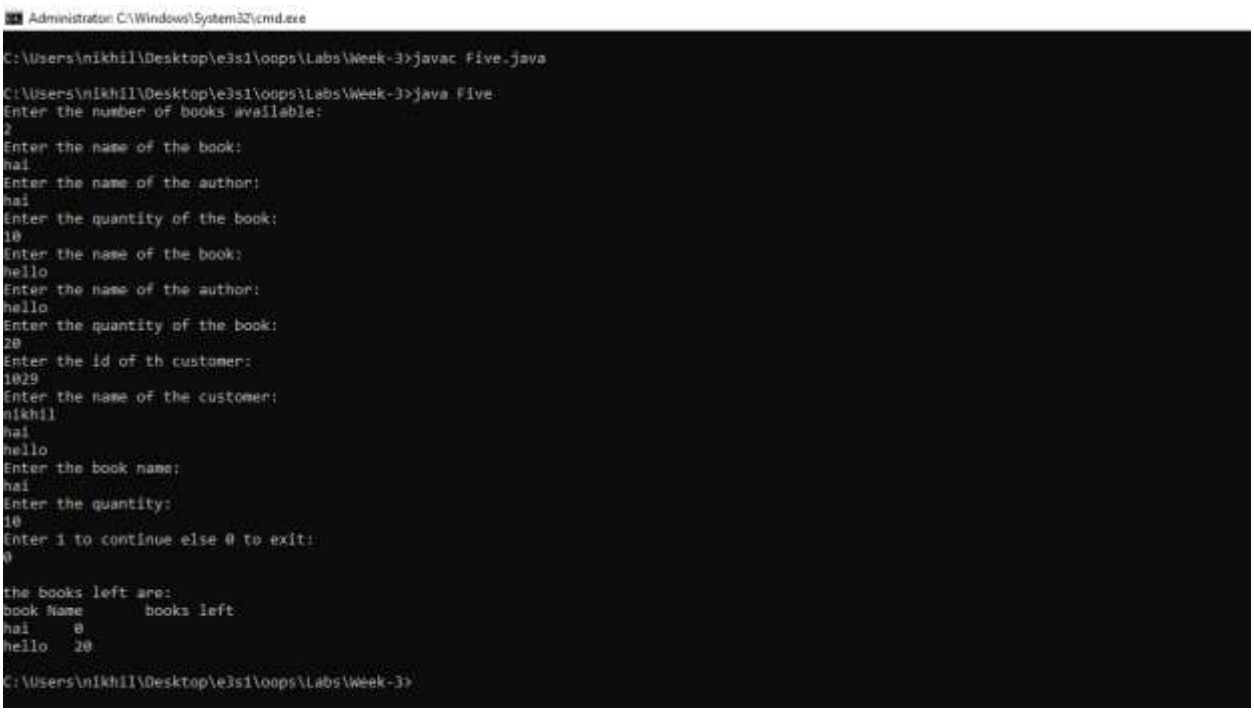
5. Implement the following case study using OOP concepts in Java. E-Book stall :
Every book has Properties which includes : Book _Name, Book_Author, Book_Count ;
Every Customer is having properties as : Customer_Id, Customer_Name,
Customer_Address and he can buy Books from E-Book stall. Write a Program which will display the text book name and the remaining count of text books when a customer buys a text book.

Source Code:

```
//Implement the following case study using OOP concepts in Java. E-Book stall :
//Every book has Properties which includes : Book _Name, Book_Author, Book_Count ;
//Every Customer is having properties as : Customer_Id, Customer_Name,
//Customer_Address and he can buy Books from E-Book stall. Write a Program which will
//display the text book name and the remaining count of text books when a customer
//buys a text book.

import java.util.*;

class Book
{
	int bQuantity;
	String bName,bAuthor;
	Book(int quantity,String name,String author)
	{
		this.bQuantity=quantity;
		this.bName=name;
		this.bAuthor=author;
	}
}
class Customer
{
	int custId;
	String custName;
	Customer(int id,String name)
	{
		this.custId=id;
		this.custName=name;
	}
}
class Five
{
	public static void main(String args[])
	{
		Vector<Book> list= new Vector<Book>();
		//Seller choice

		System.out.println("Enter the number of books available:");
		Scanner sc=new Scanner(System.in);
		int k,i,q;
		String name,author;
		k=Integer.parseInt(sc.nextLine());
```

```
                        for(i=0;i<k;i++)
                        {
                                System.out.println("Enter the name of the book:");
                                name=sc.nextLine();
                                System.out.println("Enter the name of the author:");
                                author=sc.nextLine();
                                System.out.println("Enter the quantity of the book:");
                                q=Integer.parseInt(sc.nextLine());
                                Book b=new Book(q,name,author);
                                list.add(b);
                        }
                        //customer choice:
                        System.out.println("Enter the id of th customer:");
                        k=Integer.parseInt(sc.nextLine());
                        System.out.println("Enter the name of the customer:");
                        name=sc.nextLine();
                        Customer c=new Customer(k,name);

                        //Selling arena
                        do
                        {
                                for(Book b:list)
                                {
                                        System.out.println(b.bName);
                                }
                                System.out.println("Enter the book name:");
                                name=sc.nextLine();
                                System.out.println("Enter the quantity:");
                                k=Integer.parseInt(sc.nextLine());
                                for(Book b:list)
                                {
                                        if(name.compareTo(b.bName)==0)
                                        {
                                                if(b.bQuantity>=k)
                                                        b.bQuantity-=k;
                                                else
                                                        System.out.println("the quantity required is
        not available:");
                                        }
                                }
                                System.out.println("Enter 1 to continue else 0 to exit:");
                                k=Integer.parseInt(sc.nextLine());
                        }while(k!=0);
                        System.out.println();
                        System.out.println("the books left are:");
                        System.out.println("book Name \t"+ "books left");
                        for(Book b:list)
                        {
                                System.out.println(b.bName+"\t"+b.bQuantity);
                        }
                }
}
```

Output:


Administrator: C:\Windows\System32\cmd.exe
C:\Users\nikhil\Desktop\e3s1\oops\Labs\Week-3>javac Five.java
C:\Users\nikhil\Desktop\e3s1\oops\Labs\Week-3>java Five
Enter the number of books available:
2
Enter the name of the book:
hai
Enter the name of the author:
hai
Enter the quantity of the book:
10
Enter the name of the book:
hello
Enter the name of the author:
hello
Enter the quantity of the book:
20
Enter the id of th customer:
1029
Enter the name of the customer:
nikhil
hai
hello
Enter the book name:
hai
Enter the quantity:
10
Enter 1 to continue else 0 to exit:
0
the books left are:
book Name books left
hai 0
hello 20
C:\Users\nikhil\Desktop\e3s1\oops\Labs\Week-3>

Week-4

1. Write an application that uses String method compareTo to compare two strings defined by the user

Source Code:

```
//Write an application that uses String method compareTo to compare two strings
//defined by the user

import java.util.*;

class One
{
	public static void main(String args[])
	{
		Scanner sc=new Scanner(System.in);
		System.out.println("Enter the first string:");
		String first=sc.nextLine();
		System.out.println("Enter the second string:");
		String second=sc.nextLine();
		if(first.compareTo(second)==0)
			System.out.println("the two strings are equal");
		else
			System.out.println("The two strings are not equal");
	}
}
```

Output:

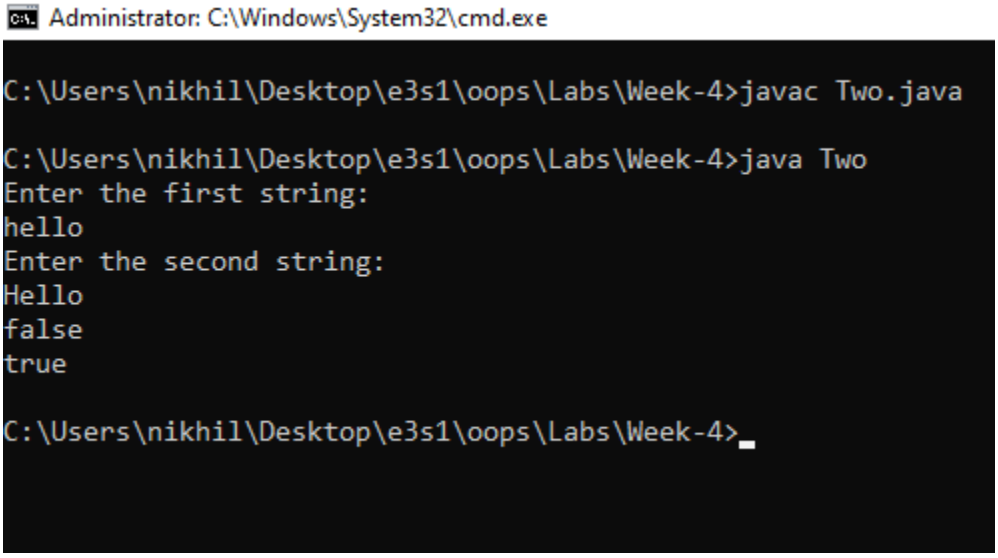
```
Select Administrator: C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.18363.1379]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\nikhil\Desktop\e3s1\oops\Labs\Week-4>javac One.java

C:\Users\nikhil\Desktop\e3s1\oops\Labs\Week-4>java One
Enter the first string:
hello world
Enter the second string:
hello world
the two strings are equal

C:\Users\nikhil\Desktop\e3s1\oops\Labs\Week-4>java One
Enter the first string:
hai
Enter the second string:
vanakam
The two strings are not equal

C:\Users\nikhil\Desktop\e3s1\oops\Labs\Week-4>
```

2. Write an application that uses String method equals and equalsIgnoreCase to tests any two string objects for equality

Source Code:

```
//. Write an application that uses String method equals and equalsIgnoreCase to tests sany
two string objects for equality

import java.util.*;

class Two
{
        public static void main(String args[])
        {
                Scanner sc=new Scanner(System.in);
                String first,second;
                System.out.println("Enter the first string:");
                first=sc.nextLine();
                System.out.println("Enter the second string:");
                second=sc.nextLine();
                System.out.println(first.equals(second));
                System.out.println(first.equalsIgnoreCase(second));
        }
}
```

Output:



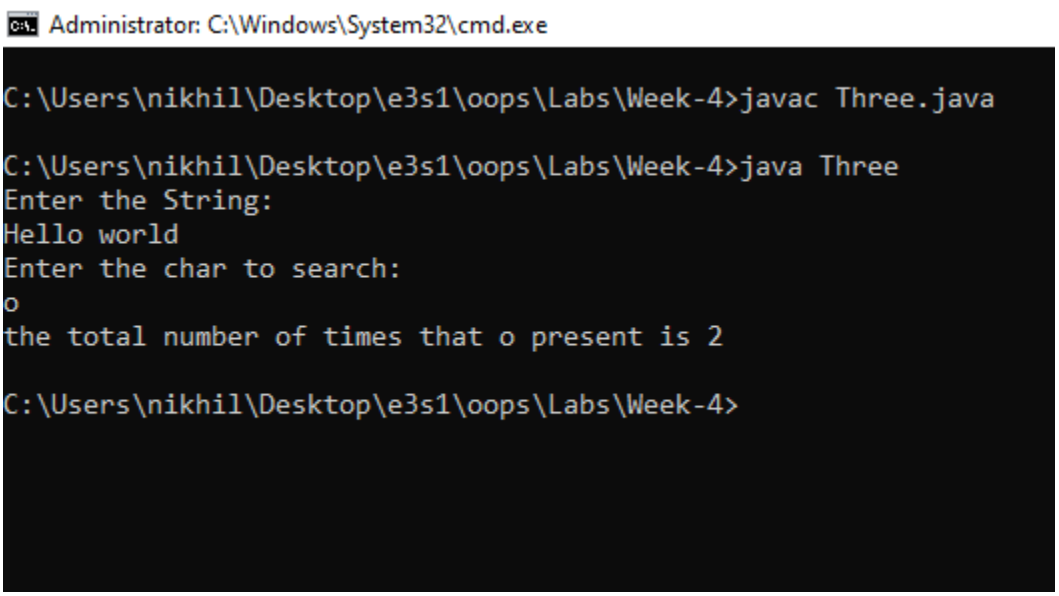
3. Write an application that uses String method indexOf to determine the total number of occurrences of any given alphabet in a defined text

Source Code:

```
//Write an application that uses String method indexOf to determine the total
//number of occurrences of any given alphabet in a defined text
```

```
import java.util.*;

class Three
{
	public static void main(String args[])
	{
		System.out.println("Enter the String:");
		Scanner sc=new Scanner(System.in);
		String a=sc.nextLine();
		System.out.println("Enter the char to search:");
		String c=sc.nextLine();
		int k=a.indexOf(c,0);
		if(k!=-1)
		{
			int count=0;
			while(k!=-1)
			{
				count++;
				k=a.indexOf(c,k+1);
			}
			System.out.println("the total number of times that "+ c + " present is
"+ count);
		}
	}
}
```

Output:



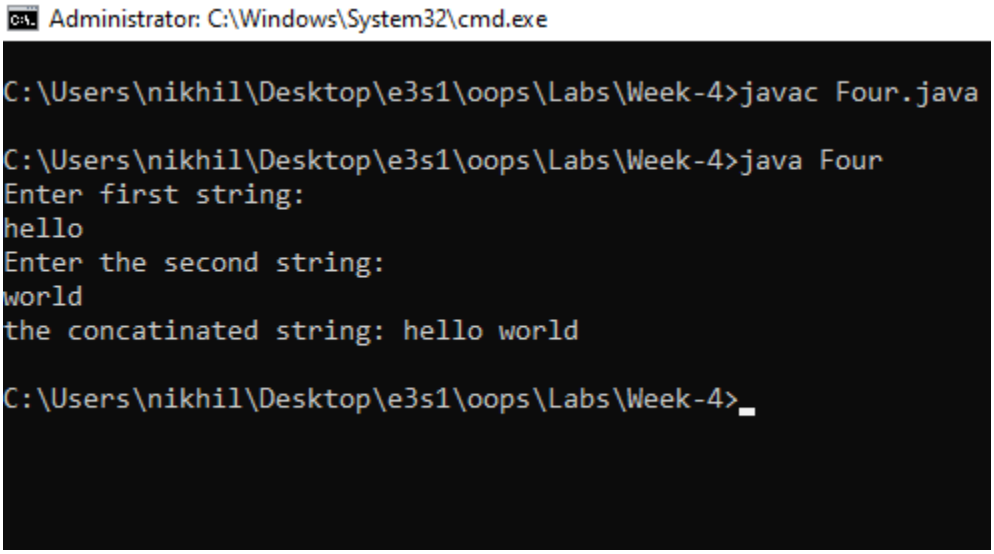
4. Write an application that uses String method concat to concatenate two defined strings

Source Code:

```
//Write an application that uses String method concat to concatenate two defined strings
```

```
import java.util.*;

class Four
{
	public static void main(String args[])
	{
		Scanner sc=new Scanner(System.in);
		String a,b;
		System.out.println("Enter first string:");
		a=sc.nextLine();
		System.out.println("Enter the second string:");
		b=sc.nextLine();
		a=a.concat(b);
		System.out.println("the concatinated string: "+a);
	}
}
```

Output:



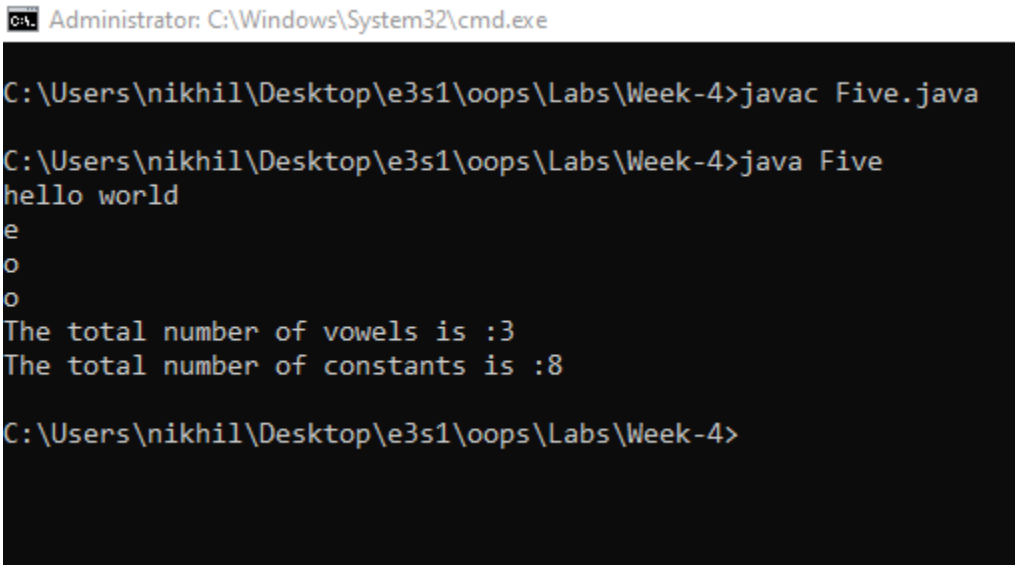
5. Write a Java program to print all vowels in given string and count number of vowels and consonants present in given string

Source Code:

```
//Write a Java program to print all vowels in given string and count number of vowels and
consonants present in given string

import java.util.*;

class Five
{
	public static void main(String args[])
	{
		String a;
		int i,j,vowels=0,constants=0;
		Scanner sc=new Scanner(System.in);
```

```
                        a=sc.nextLine();
                        for(i=0;i<a.length();i++)
                        {
                                char c=a.charAt(i);
                                if(c=='a' || c=='A'|| c=='e' || c=='E'|| c=='i' || c=='I'|| c=='o' ||
        c=='O'|| c=='u' || c=='U')
                                {
                                        System.out.println(c);
                                        vowels++;
                                }
                                else
                                        constants++;
                        }
                        System.out.println("The total number of vowels is :"+ vowels);
                        System.out.println("The total number of constants is :"+constants);
                }
}
```

Output:



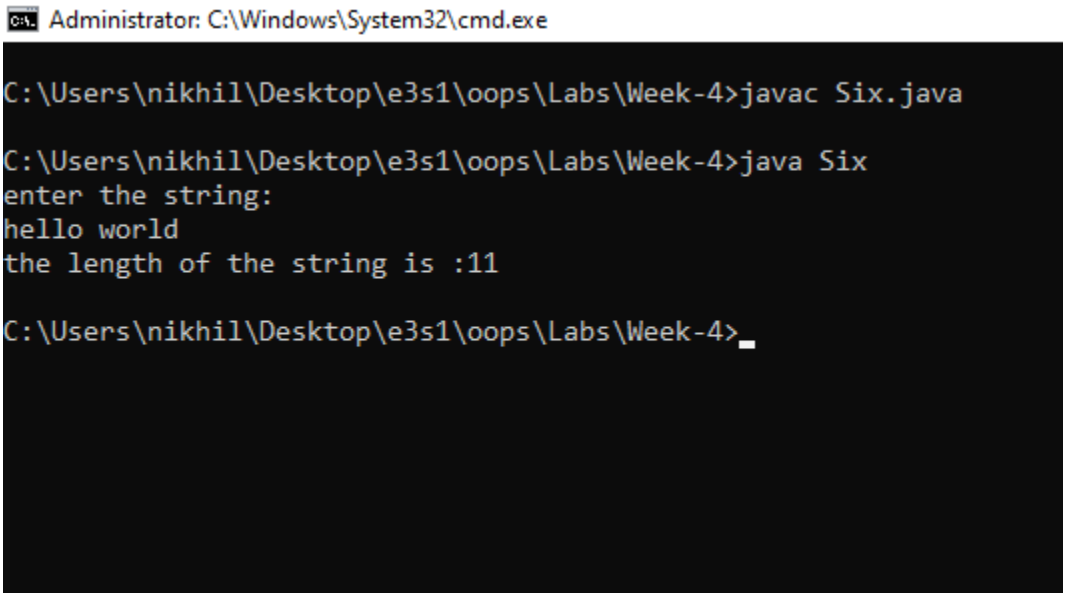
6. Write an application that finds the length of a given string.

Source Code:

```
        //Write an application that finds the length of a given string.

        import java.util.*;

        class Six
        {
                public static void main(String args[])
                {
                        String s;
                        Scanner sc=new Scanner(System.in);
                        System.out.println("enter the string:");
```

```
                        s=sc.nextLine();
                        char arr[]=s.toCharArray();
                        int count=0;
                        for(char c:arr)
                        {
                                count++;
                        }
                        System.out.println("the length of the string is :"+count);
                }
        }
```

Output:



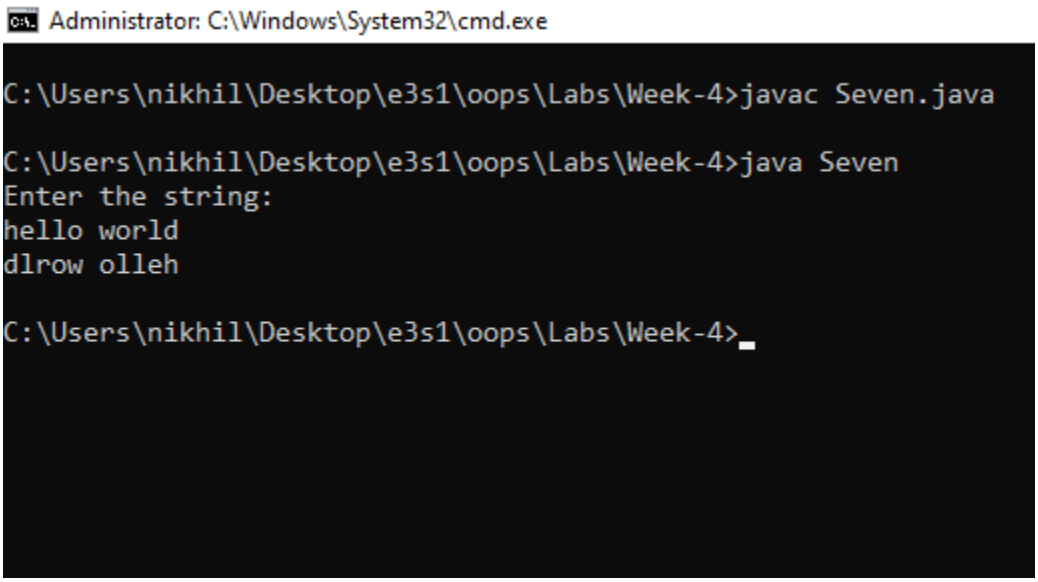
7. Write an application that uses String method charAt to reverse the string.

Source Code:

```
        //Write an application that uses String method charAt to reverse the string.

        import java.util.*;

        class Seven
        {
                public static void main(String args[])
                {
                        Scanner sc=new Scanner(System.in);
                        System.out.println("Enter the string:");
                        String s=sc.nextLine();
                        int n=s.length(),i;
                        String a="";
                        for(i=n-1;i>=0;i--)
                        {
                                a+=s.charAt(i);
                        }
```

```
                System.out.println(a);
            }
}
```

Output:



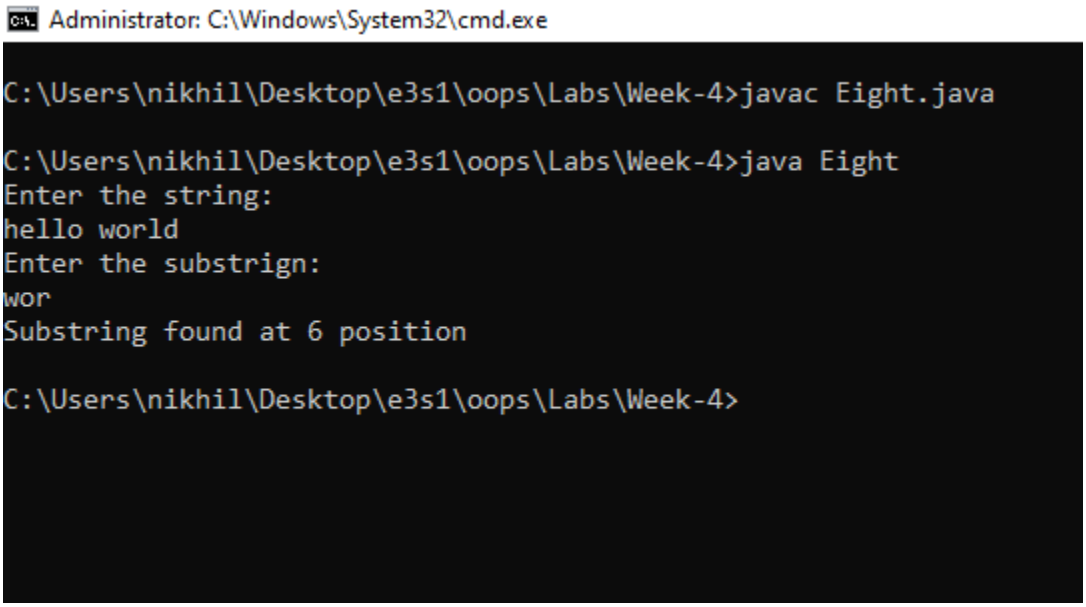
8. Write an application that finds the substring from any given string using substring method and startsWith & endsWith methods.

Source Code:

```
//Write an application that finds the substring from any given string using substring method
and startsWith & endsWith methods.

import java.util.*;

class Eight
{
        public static void main(String args[])
        {
                String s,sub;
                Scanner sc=new Scanner(System.in);
                System.out.println("Enter the string:");
                s=sc.nextLine();
                System.out.println("Enter the substrign:");
                sub=sc.nextLine();

                if(s.startsWith(sub))
                        System.out.println("substring is at the start of the given string");

                else if(s.endsWith(sub))
                        System.out.println("Substring is at the end of the given string:");
                else
                {
                        int k=s.indexOf(sub);
```

```
                        if(k==-1)
                                System.out.println("substring not found in the string");
                        else
                        {
                                String a=s.substring(k,k+sub.length());
                                System.out.println("Substring found at "+k+" position");
                        }
                }
        }
}
```

Output:

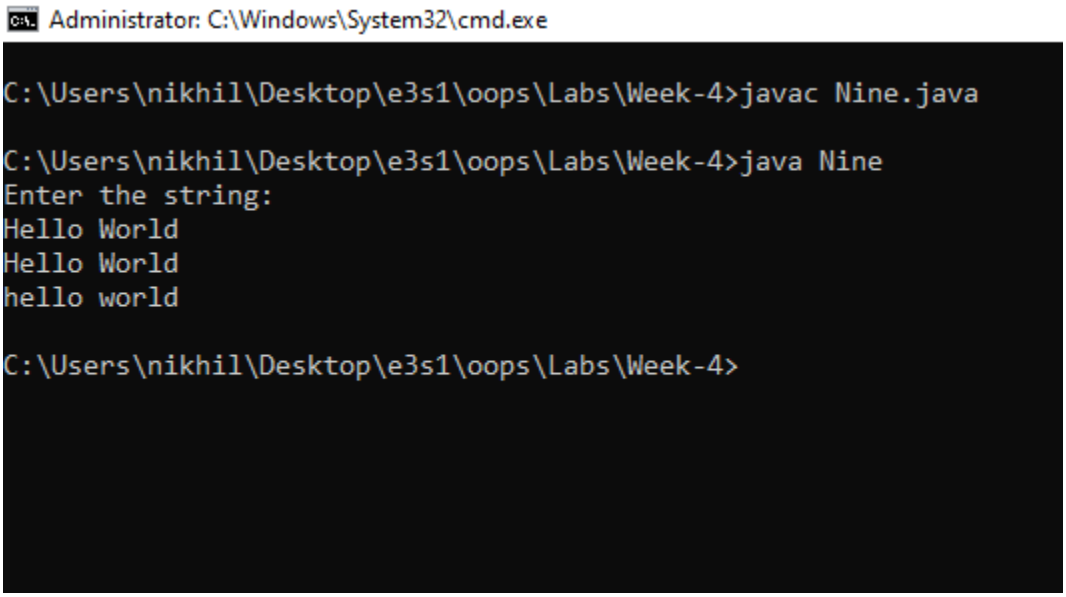


9. Write an application that changes any given string with uppercase letters, displays it, changes it back to lowercase letters and displays it.

Source Code:

```
        //Write an application that changes any given string with uppercase letters, displays it,
        changes it back to lowercase letters and displays it.

        import java.util.*;

        class Nine
        {
                public static void main(String args[])
                {
                        Scanner sc=new Scanner(System.in);
                        System.out.println("Enter the string:");
                        String s=sc.nextLine();
                        System.out.println(s);
                        System.out.println(s.toLowerCase());
                }
}
```

Output:


Administrator: C:\Windows\System32\cmd.exe
C:\Users\nikhil\Desktop\e3s1\oops\Labs\Week-4>javac Nine.java
C:\Users\nikhil\Desktop\e3s1\oops\Labs\Week-4>java Nine
Enter the string:
Hello World
Hello World
hello world
C:\Users\nikhil\Desktop\e3s1\oops\Labs\Week-4>

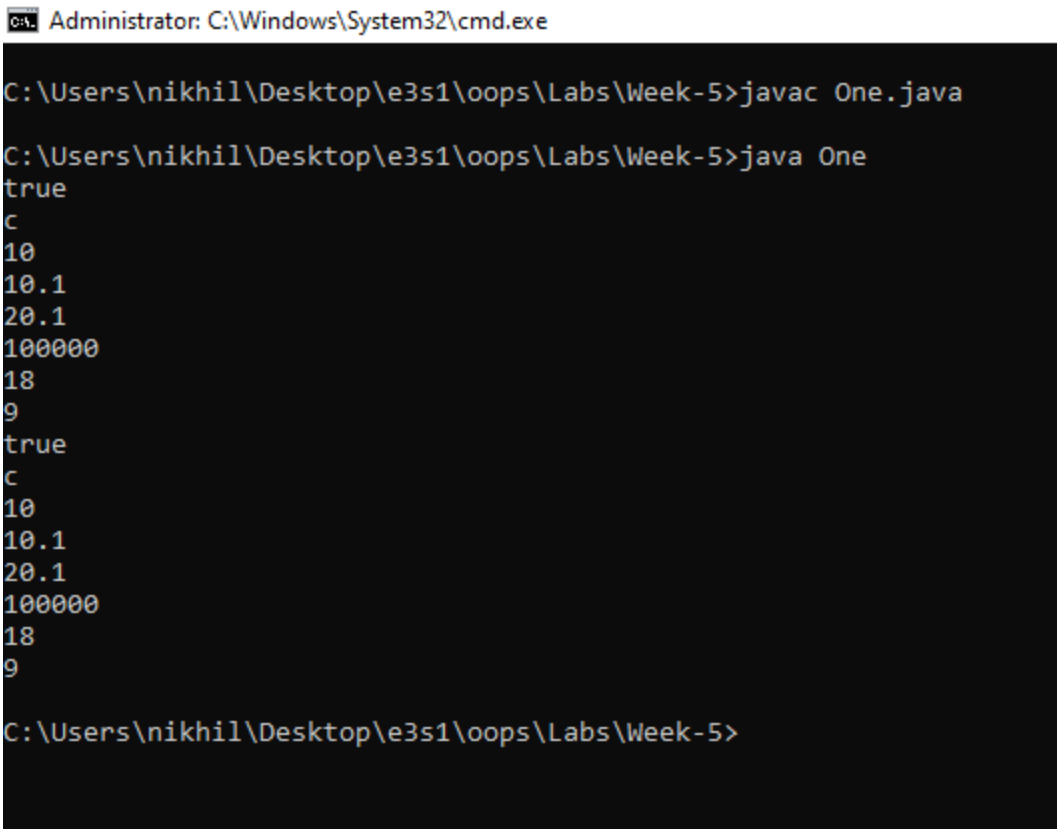
Week-5

1. Write a Java Program to implement Wrapper classes and their methods

Source Code:

```
//Write a Java Program to implement Wrapper classes and their methods

import java.util.*;

class One
{
	public static void main(String args[])
	{
		Scanner sc=new Scanner(System.in);
		Boolean b=Boolean.valueOf(sc.next());
		Character c=Character.valueOf(sc.next().charAt(0));
		Integer i=Integer.valueOf(sc.next());
		Double d=Double.valueOf(sc.next());
		Float f=Float.valueOf(sc.next());
		Long L=Long.valueOf(sc.next());
		Short S=Short.valueOf(sc.next());
		Byte B=Byte.valueOf(sc.next());


		System.out.println(b);
		System.out.println(c);
		System.out.println(i);
		System.out.println(d);
		System.out.println(f);
		System.out.println(L);
		System.out.println(S);
		System.out.println(B);
	}
}
```

Ouptut:



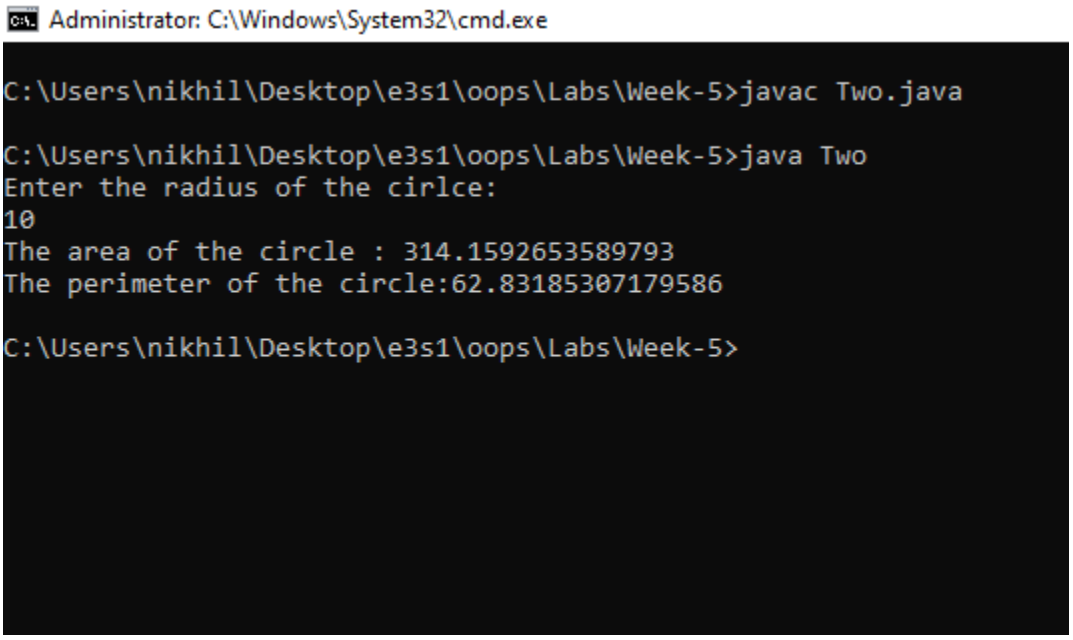
2. Write an application that prompts the user for the radius of a circle and uses a method called circleArea to calculate the area of the circle and uses a method circlePerimeter to calculate the perimeter of the circle

Source Code:

```
//Write an application that prompts the user for the radius of a circle and uses a method
//called circleArea to calculate the area of the circle and uses a method circlePerimeter to
//calculate the perimeter of the circle


import java.util.*;

class Two
{
	static double Area(double radius)
	{
		return Math.PI*radius*radius;
	}
	static double  Perimeter(double radius)
	{
		return Math.PI*2*radius;
	}
	public static void main(String args[])
	{
		Scanner sc=new Scanner(System.in);
		System.out.println("Enter the radius of the cirlce:");
		double radi=sc.nextDouble();
```

```
                System.out.println("The area of the circle : " + Area(radi));
                System.out.println("The perimeter of the circle:"+ Perimeter(radi));
        }
}
```

Output:



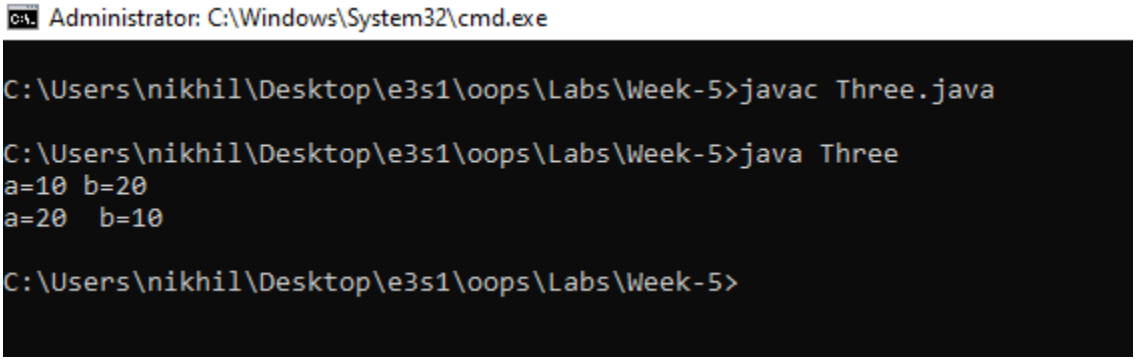
3. Write a JAVA program for the following a. Call by value b. Call by object

Source Code:

```
//.

import java.util.*;

class Swap
{
        int a,b;
        Swap(int a,int b)
        {
                this.a=a;
                this.b=b;
        }
}
class Three
{
        static void swap(int a,int b)
        {
                a=a+b-(b=a);
        }
        static void swap(Swap s)
        {
```

```
                s.a=s.a+s.b-(s.b=s.a);
        }
        public static void main(String args[])
        {
                int a=10,b=20;
                Swap s=new Swap(a,b);
                swap(a,b);
                System.out.println("a="+a+ " b="+b);
                swap(s);
                System.out.println("a="+s.a+"  b="+s.b);
        }
}
```

Output:



4. Create a class Account with an instance variable balance (double). It should contain a constructor that initializes the balance, ensure that the initial balance is greater than 0.0. Acct details: Acct_Name, Acct_acctno, Acct_Bal, Acct_Address.
Create two methods namely credit and debit, getBalance. The Credit adds the amount (passed as parameter) to balance and does not return any data. Debit method withdraws money from an Account. GetBalance displays the amount. Ensure that the debit amount does not exceed the Account's balance. In that case the balance should be left unchanged and the method should print a message indicating "Debit amount exceeded account balance

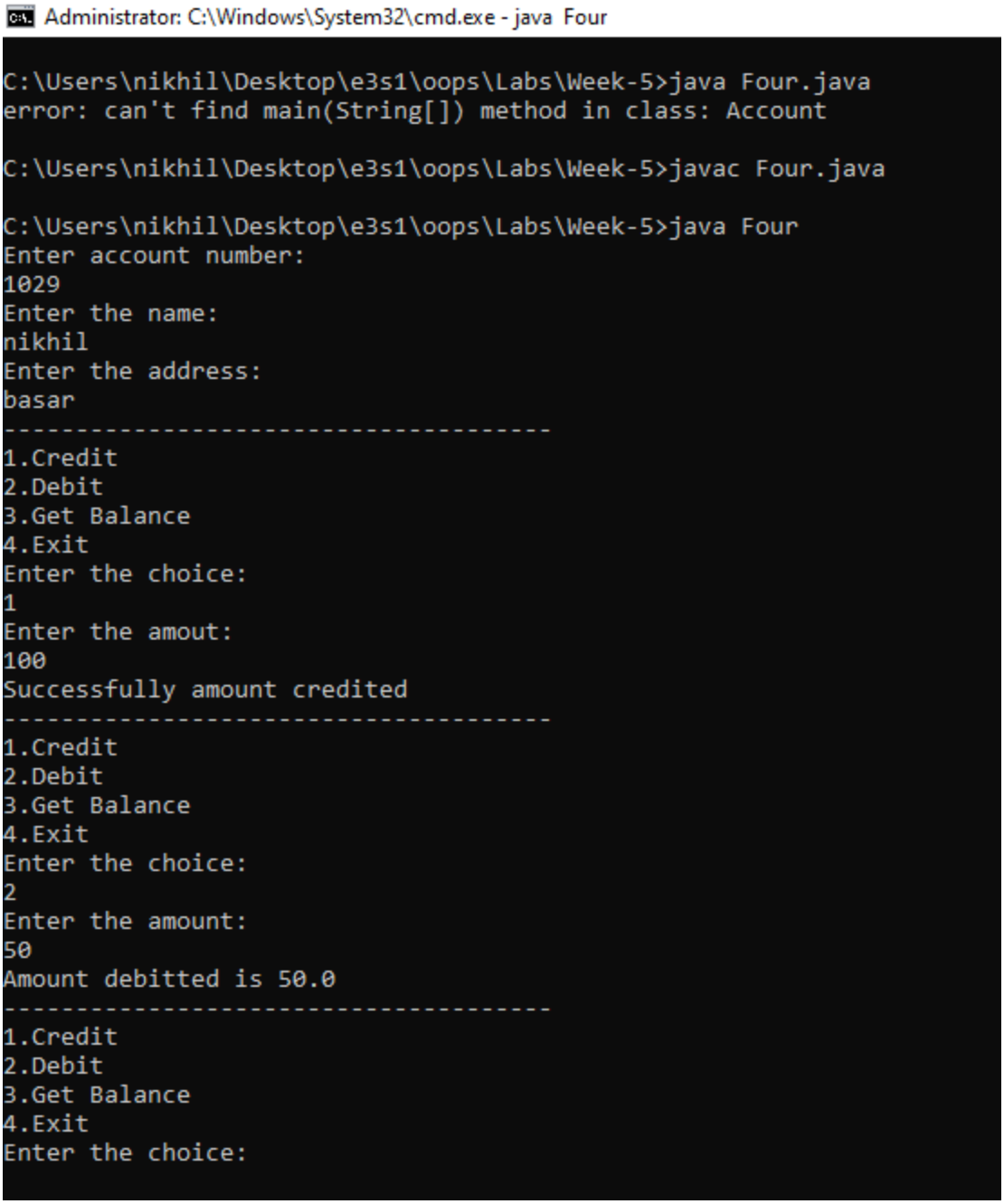
Source Code:

```
        //Create a class Account with an instance variable balance (double). It should contain a
        //constructor that initializes the balance, ensure that the initial balance is greater than 0.0.
        //Acct details: Acct_Name, Acct_acctno, Acct_Bal, Acct_Address.
        //Create two methods namely credit and debit, getBalance. The Credit adds the amount
        //(passed as parameter) to balance and does not return any data. Debit method withdraws
        //money from an Account. GetBalance displays the amount. Ensure that the debit amount
        //does not exceed the Account's balance. In that case the balance should be left unchanged
        //and the method should print a message indicating "Debit amount exceeded account
        //balance


        import java.util.Scanner;
```

```
class Account
{
	private String  Acct_Name,Acct_Acctno,Acct_Address;
	private double Acct_Bal=0;
	Account(String name,String acctno,String address)
	{
		this.Acct_Name=name;
		this.Acct_Acctno=acctno;
		this.Acct_Address=address;
		this.Acct_Bal=0;
	}
	void credit(double balance)
	{
		if(balance>0)
		{
			this.Acct_Bal+=balance;
			System.out.println("Successfully amount credited");
		}
		else
			System.out.println("invalid amount");
	}
	void debit(double balance)
	{
		if(balance>this.Acct_Bal)
			System.out.println("Debit amount exceeded account balance");
		else
		{
			Acct_Bal-=balance;
			System.out.println("Amount debitted is "+ balance);
		}
	}
	double getBalance()
	{
		return this.Acct_Bal;
	}
}
class Four
{
	public static void main(String args[])
	{
		Scanner sc=new Scanner(System.in);
		String name,address,acctno;
		int k;
		double bal;
		System.out.println("Enter account number:");
		acctno=sc.nextLine();
		System.out.println("Enter the name:");
		name=sc.nextLine();
		System.out.println("Enter the address:");
		address=sc.nextLine();
		Account ac=new Account(name,acctno,address);
		do
```

```
                {
                        System.out.println("-------------------------------------");
                        System.out.println("1.Credit");
                        System.out.println("2.Debit");
                        System.out.println("3.Get Balance");
                        System.out.println("4.Exit");
                        System.out.println("Enter the choice:");
                        k=sc.nextInt();
                        switch(k)
                        {
                                case 1:
                                        System.out.println("Enter the amout:");
                                        bal=sc.nextDouble();
                                        ac.credit(bal);
                                        break;
                                case 2:
                                        System.out.println("Enter the amount:");
                                        bal=sc.nextDouble();
                                        ac.debit(bal);
                                        break;
                                case 3:
                                        bal=ac.getBalance();
                                        System.out.println("The balance amount is : "+bal);
                                        break;
                                case 4:
                                        System.out.println("Thank you for using our Bank");
                                        break;
                                default:
                                        if(k!=4)
                                                System.out.println("invalid choice");
                        }
                }while(k!=4);
        }
}
```

Output:



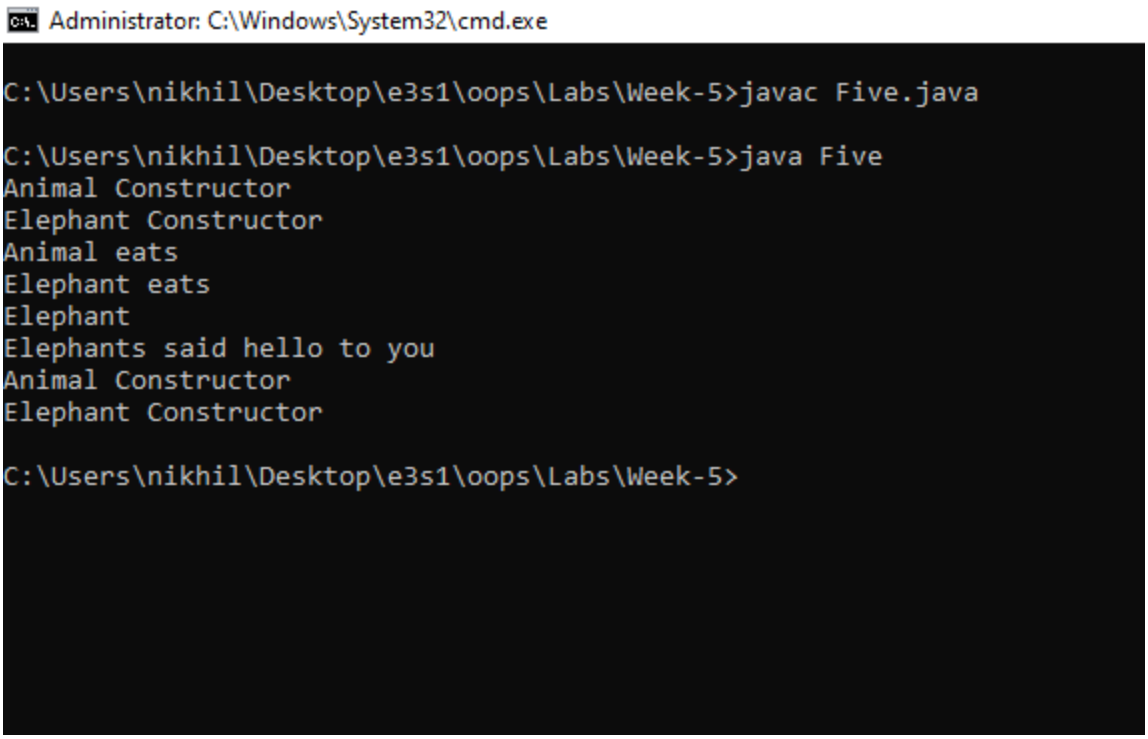
5. Write Java program for the following
   a. Example for this operator and the use of this keyword.
   b. Example for super keyword.
   c. Example for static variables and method

Source Code:

```
//Write Java program for the following
//a. Example for this operator and the use of this keyword.
//b. Example for super keyword.
//c. Example for static variables and method


import java.util.*;
```

```
class Animal
{
	Animal()
	{
		System.out.println("Animal Constructor");
	}
	void eat()
	{
		System.out.println("Animal eats");
	}
}
class Elephant extends Animal
{
	String name;
	static String type="Elephant";

	Elephant()
	{

		System.out.println("Elephant Constructor");
	}
	Elephant(int id)
	{
		this();
	}
	void eats()
	{

		super.eat();
		System.out.println("Elephant eats");
		System.out.println(Elephant.type);
	}
	static void game()
	{
		System.out.println("Elephants said hello to you");
	}
}

class Five
{
	public static void main(String args[])
	{
		Elephant E=new Elephant();
		E.eats();
		Elephant.game();
		Elephant e1=new Elephant(1029);
	}
}
```

Output:


Administrator: C:\Windows\System32\cmd.exe
C:\Users\nikhil\Desktop\e3s1\oops\Labs\Week-5>javac Five.java
C:\Users\nikhil\Desktop\e3s1\oops\Labs\Week-5>java Five
Animal Constructor
Elephant Constructor
Animal eats
Elephant eats
Elephant
Elephants said hello to you
Animal Constructor
Elephant Constructor
C:\Users\nikhil\Desktop\e3s1\oops\Labs\Week-5>

Week-6

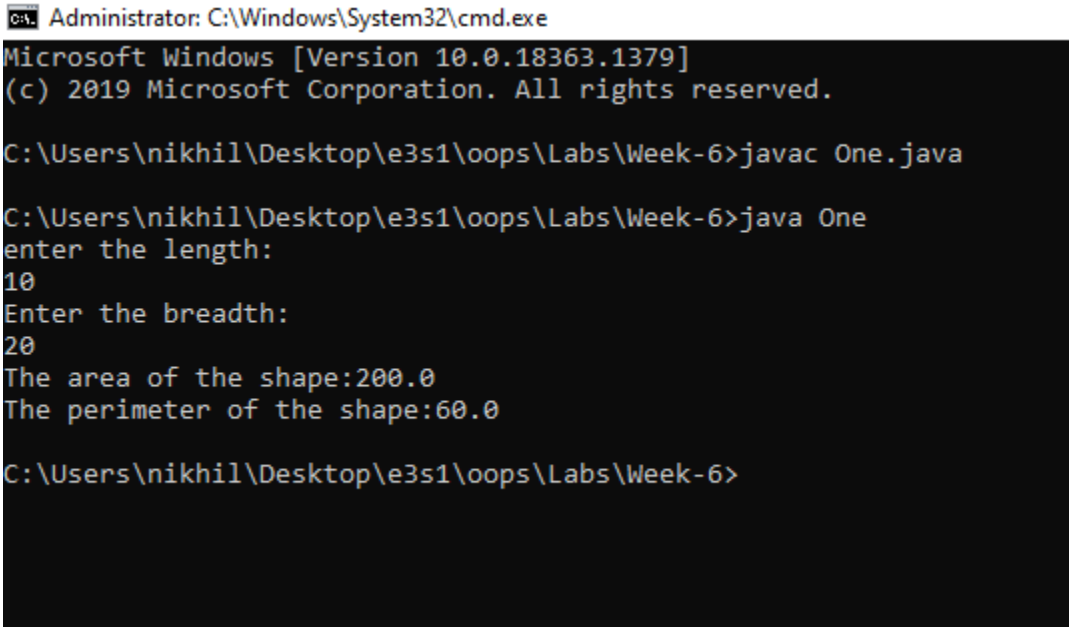
1. Write a Java program to find Area and Circle of different shapes using polymorphism concept

Source Code:

```
 //Write a Java program to find Area and Circle of different shapes using polymorphism
concept

import java.util.*;

class Shape
{
	int length,breadth,radius;
	double area()
	{
		return this.length*this.breadth;
	}
	double perimeter()
	{
		return 2*(this.length+this.breadth);
	}
}
class Polygon extends Shape
{

	Polygon(int length,int breadth)
	{
		super.length=length;
		super.breadth=breadth;
	}
	double area()
	{
		return this.length*this.breadth;
	}
	double perimeter()
	{
		return 2*(this.length+this.breadth);
	}
}
class One
{
	public static void main(String args[])
	{
		Scanner sc=new Scanner(System.in);
		int len,wid;
		System.out.println("enter the length:");
		len=sc.nextInt();
		System.out.println("Enter the breadth:");
		wid=sc.nextInt();
		Polygon p=new Polygon(len,wid);
```

```
                System.out.println("The area of the shape:"+p.area());
                System.out.println("The perimeter of the shape:"+p.perimeter());
        }
}
```

Output:



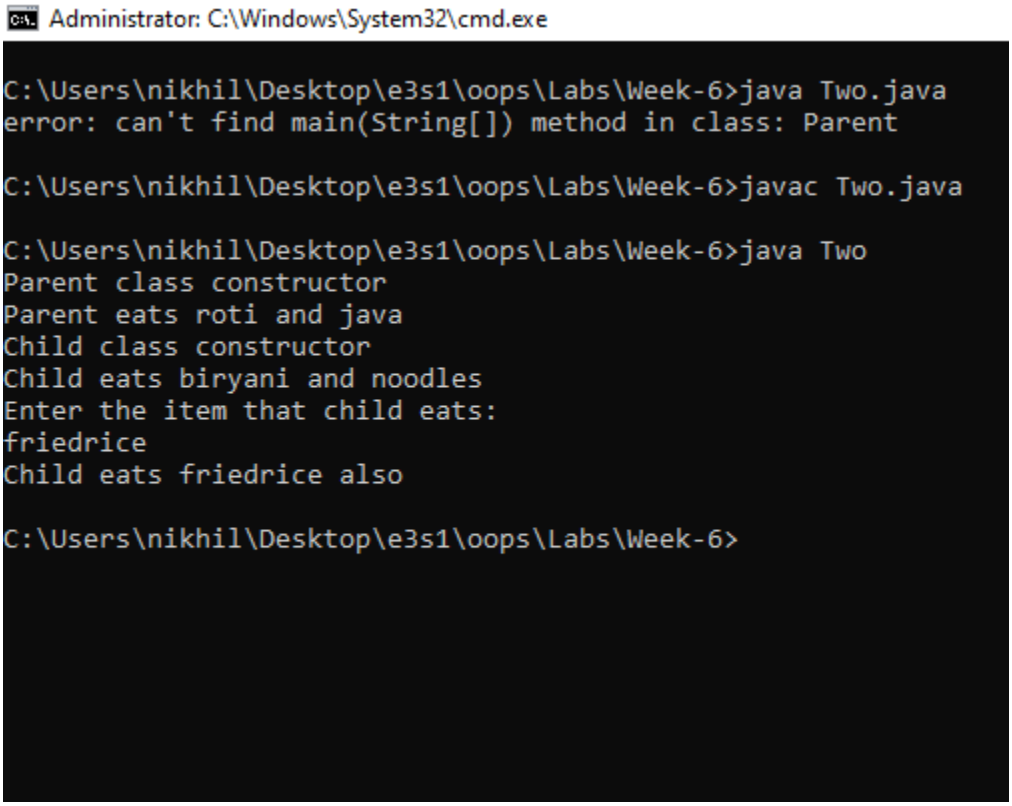
2. Write a Java program which can give example of Method overloading and overriding

Source Code:

```
// Write a Java program which can give example of Method overloading and overriding

import java.util.*;

class Parent
{
        Parent()
        {
                System.out.println("Parent class constructor");
        }
        void Eats()
        {
                System.out.println("Parent eats roti and java");
        }
}
class Child
{
        Child()
        {
                System.out.println("Child class constructor");
        }
        void Eats()
```

```
		{
			System.out.println("Child eats biryani and noodles");
		}
		void Eats(String item)
		{
			System.out.println("Child eats "+item+" also");
		}
	}
	class Two
	{
		public static void main(String args[])
		{
			Scanner sc=new Scanner(System.in);
			Parent P=new Parent();
			P.Eats();
			Child C=new Child();
			C.Eats();
			System.out.println("Enter the item that child eats:");
			String item=sc.nextLine();
			C.Eats(item);
		}
}
```

Output:



3. Write an application to create a super class Employee with information first name & last name and methods getFirstName(), getLastName() derive the sub-classes ContractEmployee and RegularEmployee with the information about department, designation & method displayFullName() , getDepartment(), getDesig() to print the salary and to set department name & designation of the corresponding sub-class objects respectively.

Source Code:

```
//Write an application to create a super class Employee with information first name &
//last name and methods getFirstName(), getLastName() derive the sub-classes
//ContractEmployee and RegularEmployee with the information about department,
//designation & method displayFullName() , getDepartment(), getDesig() to print the
//salary and to set department name & designation of the corresponding sub-class objects
//respectively.


import java.util.*;

class Employee
{
	private String f_name,l_name;
	Employee(String fname,String lname)
	{
		this.f_name=fname;
		this.l_name=lname;
	}
	String getFirstName()
	{
		return this.f_name;
	}
	String getLastName()
	{
		return this.l_name;
	}
}
class ContractEmployee extends Employee
{
	private String designation,department;
	private double salary;
	ContractEmployee(String fname,String lname,String designation,String
department,double salary)
	{
		super(fname,lname);
		this.designation=designation;
		this.department=department;
		this.salary=salary;
	}
	String displayFullName()
	{
		return super.getFirstName()+super.getLastName();
```

```
		}
		String getDepartment()
		{
			return this.department;
		}
		String getDesignation()
		{
			return this.designation;
		}
		double getSalary()
		{
			return this.salary;
		}
		void setDesignation(String designation)
		{
			this.designation=designation;
		}
		void setDepartment(String department)
		{
			this.department=department;
		}
	}
	class RegularEmployee extends Employee
	{
		private String designation,department;
		private double salary;
		RegularEmployee(String fname,String lname,String department,String
	designation,double salary)
		{
			super(fname,lname);
			this.designation=designation;
			this.department=department;
			this.salary=salary;
		}
		String displayFullName()
		{
			return super.getFirstName()+super.getLastName();
		}
		String getDepartment()
		{
			return this.department;
		}
		String getDesignation()
		{
			return this.designation;
		}
		double getSalary()
		{
			return this.salary;
		}
		void setDesignation(String designation)
		{
			this.designation=designation;
```

```
	}
	void setDepartment(String department)
	{
		this.department=department;
	}
}
class Three
{
	public static void main(String args[])
	{
		Scanner sc=new Scanner(System.in);
		String fname,lname,depart,design;
		double salary;
		System.out.println("-----Regular Employee---------");
		System.out.println("Enter the details one by one first name, last name,
department designation,salary");
		fname=sc.nextLine();
		lname=sc.nextLine();
		depart=sc.nextLine();
		design=sc.nextLine();
		salary=sc.nextDouble();
		RegularEmployee r=new
RegularEmployee(fname,lname,depart,design,salary);
		int k;
		do
		{
			System.out.println("Display Features");
			System.out.println("1. Name   2. Department  3. Designation 4.
Salary");
			System.out.println("Set Features:");
			System.out.println("5. Department 6. Designation 7.exit");
			System.out.println("Enter the choices:");
			k=sc.nextInt();
			sc.nextLine();
			switch(k)
			{
				case 1: System.out.println(r.displayFullName());
					break;
				case 2: System.out.println(r.getDepartment());
					break;
				case 3: System.out.println(r.getDesignation());
					break;
				case 4: System.out.println(r.getSalary());
					break;
				case 5: System.out.println("Enter the department:");
					depart=sc.nextLine();
					r.setDepartment(depart);
					break;
				case 6: System.out.println("Enter the designation:");
					design=sc.nextLine();
					r.setDesignation(design);
					break;
			}
```

```
                        }while(k!=7);
                        System.out.println();
                        System.out.println("-----Contract Employee---------");
                        System.out.println("Enter the details one by one first name, last name,
        department designation,salary");
                        fname=sc.nextLine();
                        lname=sc.nextLine();
                        depart=sc.nextLine();
                        design=sc.nextLine();
                        salary=sc.nextDouble();
                        ContractEmployee c=new
        ContractEmployee(fname,lname,depart,design,salary);

                        do
                        {
                                System.out.println("Display Features");
                                System.out.println("1. Name   2. Department  3. Designation 4.
        Salary");
                                System.out.println("Set Features:");
                                System.out.println("5. Department 6. Designation 7.exit");
                                System.out.println("Enter the choices:");
                                k=sc.nextInt();
                                switch(k)
                                {
                                        case 1: System.out.println(c.displayFullName());
                                                break;
                                        case 2: System.out.println(c.getDepartment());
                                                break;
                                        case 3: System.out.println(c.getDesignation());
                                                break;
                                        case 4: System.out.println(c.getSalary());
                                                break;
                                        case 5: System.out.println("Enter the department:");
                                                depart=sc.nextLine();
                                                c.setDepartment(depart);
                                                break;
                                        case 6: System.out.println("Enter the designation:");
                                                design=sc.nextLine();
                                                c.setDesignation(design);
                                                break;
                                }
                        }while(k!=7);
                }
}
```

Output:

```
Administrator: C:\Windows\System32\cmd.exe - java  Three

C:\Users\nikhil\Desktop\e3s1\oops\Labs\Week-6>javac Three.java

C:\Users\nikhil\Desktop\e3s1\oops\Labs\Week-6>java Three
-----Regular Employee---------
Enter the details one by one first name, last name, department designation,salary
nikhil
thangellapally
cse
btech
100
Display Features
1. Name   2. Department  3. Designation 4. Salary
Set Features:
5. Department 6. Designation 7.exit
Enter the choices:
6
Enter the designation:
mtech
Display Features
1. Name   2. Department  3. Designation 4. Salary
Set Features:
5. Department 6. Designation 7.exit
Enter the choices:
3
mtech
Display Features
1. Name   2. Department  3. Designation 4. Salary
Set Features:
5. Department 6. Designation 7.exit
Enter the choices:
7

-----Contract Employee---------
Enter the details one by one first name, last name, department designation,salary
pranay
thangellapally
cse
mtech
10000
Display Features
1. Name   2. Department  3. Designation 4. Salary
Set Features:
5. Department 6. Designation 7.exit
```

4. Derive sub-classes of ContractEmployee namely HourlyEmployee & WeeklyEmployee with information number of hours & wages per hour, number of weeks & wages per week respectively & method calculateWages() to calculate their monthly salary. Also override getDesig () method depending on the type of contract employee.

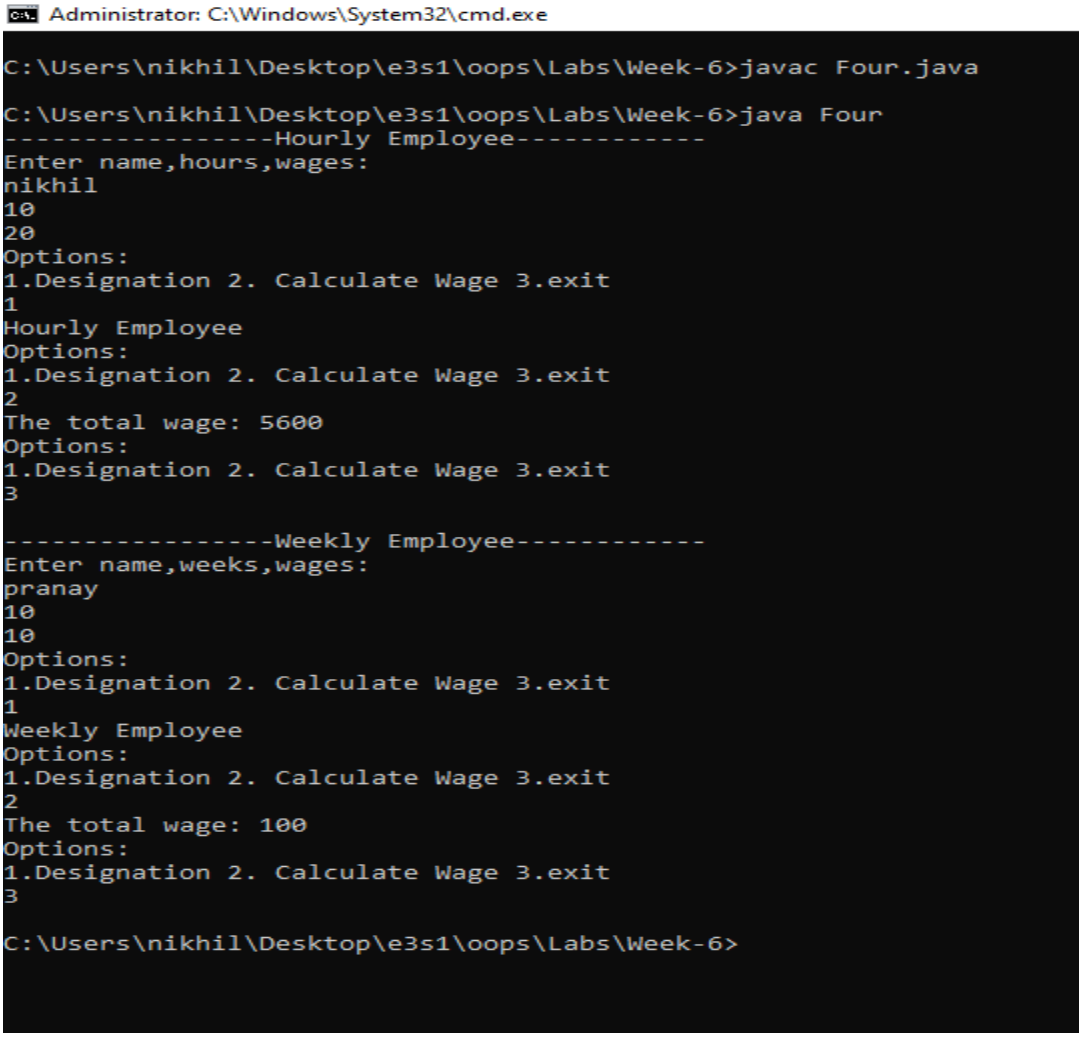
Source Code:

```
// Derive sub-classes of ContractEmployee namely HourlyEmployee & WeeklyEmployee
//with information number of hours & wages per hour, number of weeks & wages per
//week respectively & method calculateWages() to calculate their monthly salary. Also
//override getDesig () method depending on the type of contract employee.

import java.util.*;

class ContractEmployee
{
        String name,design;
```

```
	ContractEmployee(String name)
	{
		this.name=name;
		this.design="contractEmployee";
	}
	String getDesign()
	{
		return this.design;
	}
}
class HourlyEmployee extends ContractEmployee
{
	int hours,wages;
	HourlyEmployee(String name,int hours,int wages)
	{
		super(name);
		this.hours=hours;
		this.wages=wages;
		super.design="Hourly Employee";
	}
	String getDesign()
	{
		return super.design;
	}
	int calculateWages()
	{
		return this.hours*this.wages*28;
		//assuming number of working days:28
	}
}
class WeeklyEmployee extends ContractEmployee
{
	int weeks,wages;
	WeeklyEmployee(String name,int weeks,int wages)
	{
		super(name);
		this.weeks=weeks;
		this.wages=wages;
		super.design="Weekly Employee";
	}
	String getDesign()
	{
		return super.design;
	}
	int calculateWages()
	{
		return this.weeks*this.wages;
	}
}
class Four
{
	public static void main(String args[])
	{
```

```
                        Scanner sc=new Scanner(System.in);
                        String name;
                        System.out.println("----------------Hourly Employee----------");
                        int hours,wages,weeks,sal,k;
                        System.out.println("Enter name,hours,wages:");
                        name=sc.nextLine();
                        hours=sc.nextInt();
                        wages=sc.nextInt();
                        HourlyEmployee h=new HourlyEmployee(name,hours,wages);
                        do
                        {
                                System.out.println("Options:");
                                System.out.println("1.Designation 2. Calculate Wage 3.exit");
                                k=sc.nextInt();
                                switch(k)
                                {
                                        case 1: System.out.println(h.getDesign());
                                                break;
                                        case 2: System.out.println("The total wage:
        "+h.calculateWages());
                                                break;
                                }
                        }while(k!=3);
                        System.out.println();
                        System.out.println("----------------Weekly Employee----------");
                        System.out.println("Enter name,weeks,wages:");
                        sc.nextLine();
                        name=sc.nextLine();
                        weeks=sc.nextInt();
                        wages=sc.nextInt();
                        WeeklyEmployee w=new WeeklyEmployee(name,hours,wages);
                        do
                        {
                                System.out.println("Options:");
                                System.out.println("1.Designation 2. Calculate Wage 3.exit");
                                k=sc.nextInt();
                                switch(k)
                                {
                                        case 1: System.out.println(w.getDesign());
                                                break;
                                        case 2: System.out.println("The total wage:
        "+w.calculateWages());
                                                break;
                                }
                        }while(k!=3);
                }
}
```

Output:



5. Write an application to create a super class Vehicle with information vehicle number, insurance number, color and methods getConsumption() displayConsumption(). Derive the sub-classes TwoWheeler and FourWheeler with method maintenance() and average() to print the maintenance And average of vehicle.

Source Code:

```
//Write an application to create a super class Vehicle with information vehicle
//number,insurance number,color and methods getConsumption() displayConsumption().
//Derive the sub-classes TwoWheeler and FourWheeler with method maintenance() and
//average() to print the maintenance And average of vehicle.


import java.util.*;

class Vehicle
{
        int vehicleNo,insuranceNo,consumption;
        String color;
        Vehicle(int no,int incno,int consum,String color)
        {
```

```
                this.vehicleNo=no;
                this.insuranceNo=incno;
                this.consumption=consum;
                this.color=color;
        }
        int getConsumption()
        {
                return this.consumption;
        }
}
class TwoWheeler extends Vehicle
{
        int maintenance,average;
        TwoWheeler(int no,int inco,int consum,String color,int maintenance,int average);
        {
                super(no,inco,consum,color);
                this.maintenance=maintenance;
                this.average=average;
        }
        int getMaintenance()
        {
                return this.maintenance;
        }
        int getAverage()
        {
                return this.average;
        }
}
class FourWheeler extends Vehicle
{
        int maintenance,average;
        FourWheeler(int no,int inco,int consum,String color,int maintenance,int average);
        {
                super(no,inco,consum,color);
                this.maintenance=maintenance;
                this.average=average;
        }
        int getMaintenance()
        {
                return this.maintenance;
        }
        int getAverage()
        {
                return this.average;
        }
}
class Five
{
        public static void main(String args[])
        {
                Scanner sc=new Scanner(System.in);
                System.out.println("----------TwoWheeler----------");
```

```
                System.out.println("Enter the
color,vehicle_no,insurance_no,consumption,maintenance,average");
                int vno,incno,consump,maintenance,average,k;
                String color;
                color=sc.nextLine();
                vno=sc.nextInt();
                incno=sc.nextInt();
                consump=sc.nextInt();
                maintenance=sc.nextInt();
                average=sc.nextInt();
                TwoWheeler T=new
TwoWheeler(vno,incno,consump,color,maintenance,average);
                do
                {
                        System.out.println("1.consumption 2. maintenance 3. average 4.
exit");
                        k=sc.nextInt();
                        switch(k)
                        {
                                case 1: System.out.println(T.getConsumption());
                                        break;
                                case 2: System.out.println(T.getMaintenance());
                                        break;
                                case 3: System.out.println(T.getAverage());
                                        break;
                                default:
                                        if(k!=4)
                                                System.out.println("invalid choice:");
                        }
                }while(k!=4);

                System.out.println();
                System.out.println("-----------FourWheeler----------------");
                System.out.println("Enter the
color,vehicle_no,insurance_no,consumption,maintenance,average");
                sc.nextLine();
                color=sc.nextLine();
                vno=sc.nextInt();
                incno=sc.nextInt();
                consump=sc.nextInt();
                maintenance=sc.nextInt();
                average=sc.nextInt();
                FourWheeler F=new
FourWheeler(vno,incno,consump,color,maintenance,average);
                do
                {
                        System.out.println("1.consumption 2. maintenance 3. average 4.
exit");
                        k=sc.nextInt();
                        switch(k)
                        {
                                case 1: System.out.println(F.getConsumption());
                                        break;
```

```
                                        case 2: System.out.println(F.getMaintenance());
                                                break;
                                        case 3: System.out.println(F.getAverage());
                                                break;
                                        default:
                                                if(k!=4)
                                                        System.out.println("invalid choice:");
                                }
                        }while(k!=4);
                }
}
```

Output:

Administrator: C:\Windows\System32\cmd.exe

```
C:\Users\nikhil\Desktop\e3s1\oops\Labs\Week-6>javac Five.java

C:\Users\nikhil\Desktop\e3s1\oops\Labs\Week-6>java Five
----------TwoWheeler----------------
Enter the color,vehicle_no,insurance_no,consumption,maintenance,average
white
1
12
123
1234
12345
1.consumption 2. maintenance 3. average 4. exit
1
123
1.consumption 2. maintenance 3. average 4. exit
2
1234
1.consumption 2. maintenance 3. average 4. exit
3
12345
1.consumption 2. maintenance 3. average 4. exit
4

----------FourWheeler----------------
Enter the color,vehicle_no,insurance_no,consumption,maintenance,average
black
2
21
212
21234
212345
1.consumption 2. maintenance 3. average 4. exit
1
212
1.consumption 2. maintenance 3. average 4. exit
2
21234
1.consumption 2. maintenance 3. average 4. exit
4

C:\Users\nikhil\Desktop\e3s1\oops\Labs\Week-6>
```

6. Extend the above TwoWheeler class with methods getType() and getName() which gives the information about the type and the name of the company.Create sub-classes Geared and NonGeared with method average() to print the average of a geared and nongeared two wheeler.

Source Code

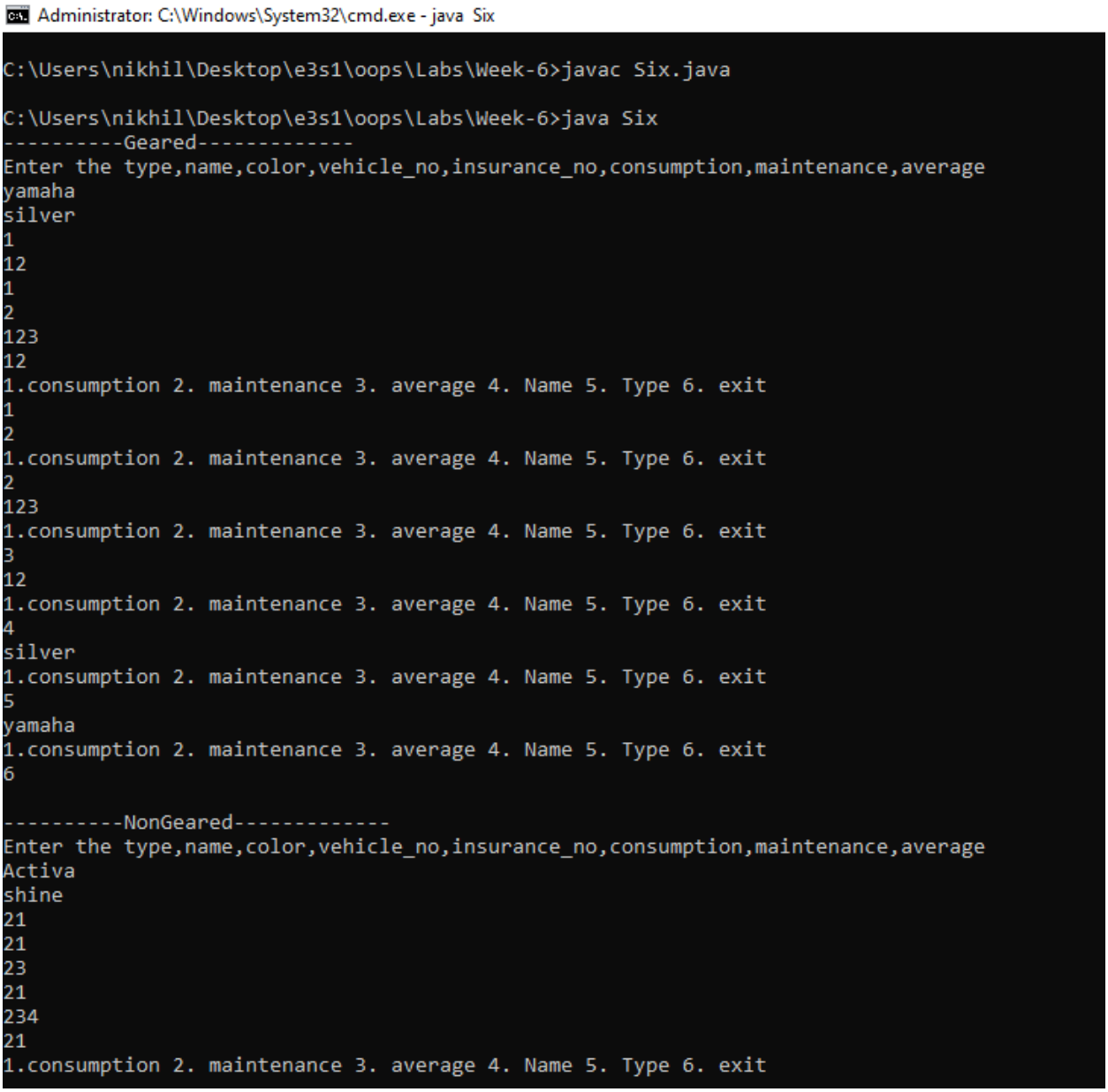
```
// Extend the above TwoWheeler class with methods getType() and getName() which
//gives the information about the type and the name of the company.Create sub-classes
//Geared and NonGeared with method average() to print the average of a geared and
nongeared two wheeler


import java.util.*;
class Vehicle
{
	int vehicleNo,insuranceNo,consumption;
	String color;
	Vehicle(int no,int incno,int consum,String color)
	{
		this.vehicleNo=no;
		this.insuranceNo=incno;
		this.consumption=consum;
		this.color=color;
	}
	int getConsumption()
	{
		return this.consumption;
	}
}
class TwoWheeler extends Vehicle
{
	int maintenance;
	String name,type;
	TwoWheeler(int no,int inco,int consum,String color,int maintenance,String
name,String type)
	{
		super(no,inco,consum,color);
		this.name=name;
		this.maintenance=maintenance;
		this.type=type;
	}
	int getMaintenance()
	{
		return this.maintenance;
	}
	String getName()
	{
		return this.name;
	}
	String getType()
	{
```

```
                return this.type;
        }
}
class Geared extends TwoWheeler
{
        int average;
        Geared(int no,int inco,int consum,String color,int maintenance,int average,String
name,String type)
        {
                super(no,inco,consum,color,maintenance,name,type);
                this.average=average;
        }
        int getAverage()
        {
                return this.average;
        }
}
class NonGeared extends TwoWheeler
{
        int average;
        NonGeared(int no,int inco,int consum,String color,int maintenance,int average,String
name,String type)
        {
                super(no,inco,consum,color,maintenance,name,type);
                this.average=average;
        }
        int getAverage()
        {
                return this.average;
        }
}
class Six
{
        public static void main(String args[])
        {
                Scanner sc=new Scanner(System.in);
                System.out.println("----------Geared----------");
                System.out.println("Enter the
type,name,color,vehicle_no,insurance_no,consumption,maintenance,average");
                int vno,incno,consump,maintenance,average,k;
                String color,type,name;
                type=sc.nextLine();
                name=sc.nextLine();
                color=sc.nextLine();
                vno=sc.nextInt();
                incno=sc.nextInt();
                consump=sc.nextInt();
                maintenance=sc.nextInt();
                average=sc.nextInt();
                Geared G=new
Geared(vno,incno,consump,color,maintenance,average,name,type);
                do
                {
```

```
                        System.out.println("1.consumption 2. maintenance 3. average 4.
Name 5. Type 6. exit");
                        k=sc.nextInt();
                        switch(k)
                        {
                                case 1: System.out.println(G.getConsumption());
                                        break;
                                case 2: System.out.println(G.getMaintenance());
                                        break;
                                case 3: System.out.println(G.getAverage());
                                        break;
                                case 4: System.out.println(G.getName());
                                        break;
                                case 5: System.out.println(G.getType());
                                        break;
                                default:
                                        if(k!=6)
                                                System.out.println("invalid choice:");
                        }
                }while(k!=6);

                System.out.println();
                System.out.println("----------Geared----------");
                System.out.println("Enter the
type,name,color,vehicle_no,insurance_no,consumption,maintenance,average");
                sc.nextLine();
                type=sc.nextLine();
                name=sc.nextLine();
                color=sc.nextLine();
                vno=sc.nextInt();
                incno=sc.nextInt();
                consump=sc.nextInt();
                maintenance=sc.nextInt();
                average=sc.nextInt();
                NonGeared N=new
NonGeared(vno,incno,consump,color,maintenance,average,name,type);
                do
                {
                        System.out.println("1.consumption 2. maintenance 3. average 4.
Name 5. Type 6. exit");
                        k=sc.nextInt();
                        switch(k)
                        {
                                case 1: System.out.println(N.getConsumption());
                                        break;
                                case 2: System.out.println(N.getMaintenance());
                                        break;
                                case 3: System.out.println(N.getAverage());
                                        break;
                                case 4: System.out.println(N.getName());
                                        break;
                                case 5: System.out.println(N.getType());
                                        break;
```

```
                                        default:
                                                if(k!=6)
                                                        System.out.println("invalid choice:");
                                }
                        }while(k!=6);
                }
}
```

Output:



Week-7

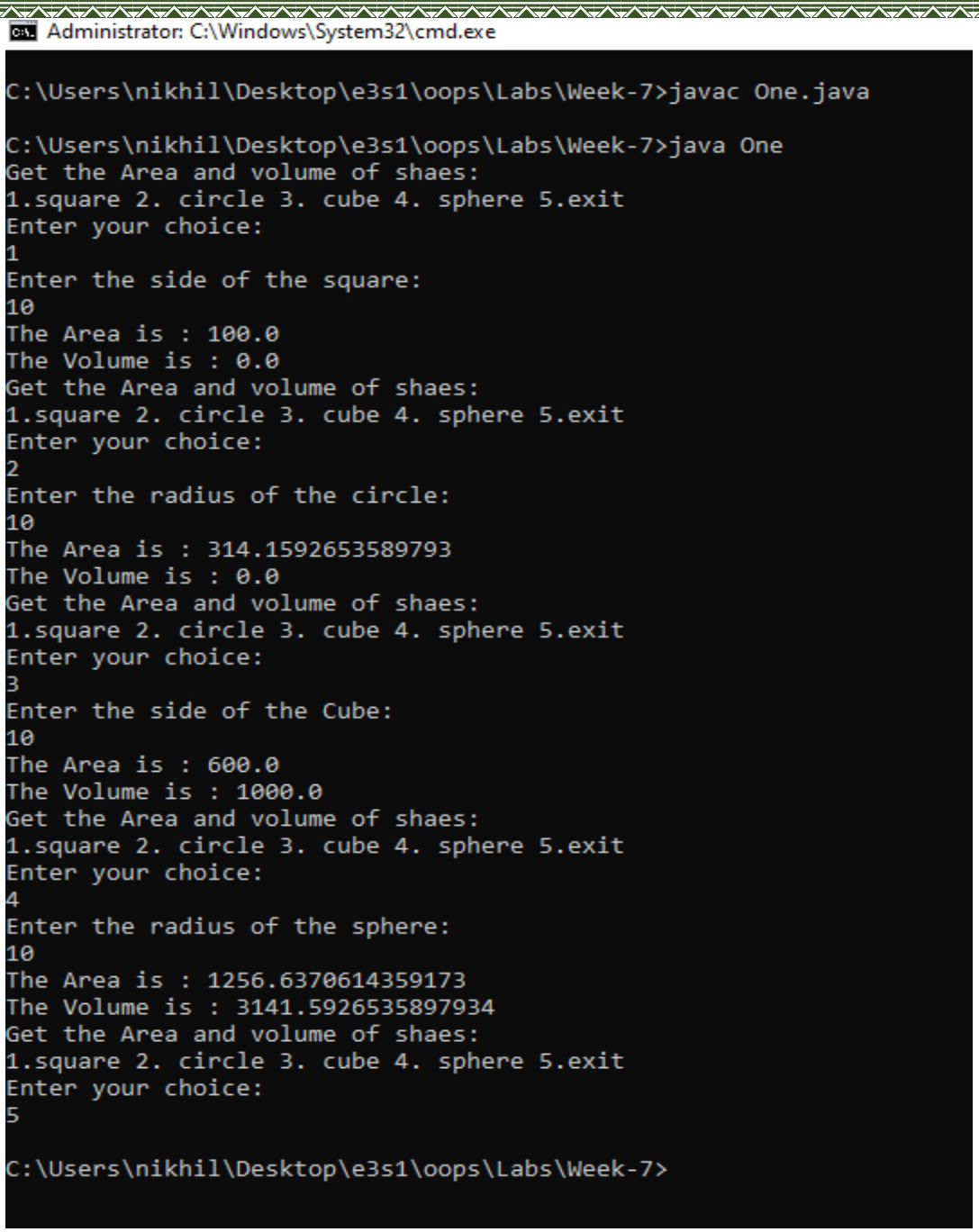
1. Create an abstract class Shape which calculate the area and volume of 2-d and 3-d shapes with methods getArea() and getVolume(). Reuse this class to calculate the area and volume of square, circle, cube and sphere

Source Code:

```
// Create an abstract class Shape which calculate the area and volume of 2-d and 3-d
//shapes with methods getArea() and getVolume(). Reuse this class to calculate the area
//and volume of square ,circle ,cube and sphere

import java.util.*;

abstract class Shape
{
	abstract double getArea();
	abstract double getVolume();
}
class Square extends Shape
{
	double side;
	Square(double side)
	{
		this.side=side;
	}
	double getArea()
	{
		return this.side*this.side;
	}
	double getVolume()
	{
		return 0.0;
	}
}
class Circle extends Shape
{
	double radius;
	Circle(double radius)
	{
		this.radius=radius;
	}
	double getArea()
	{
		return this.radius*this.radius*Math.PI;
	}
	double getVolume()
	{
		return 0.0;
	}
}
class Cube extends Shape
{
	double side;
	Cube(double side)
	{
		this.side=side;
	}
	double getArea()
	{
		return 6*this.side*this.side;
	}
	double getVolume()
	{
		return this.side*this.side*this.side;
	}
}
```

```
class Sphere extends Shape
{
	double radius;
	Sphere(double radius)
	{
		this.radius=radius;
	}
	double getArea()
	{
		return 4*Math.PI*this.radius*this.radius;
	}
	double getVolume()
	{
		return 4/3*Math.PI*this.radius*this.radius*this.radius;
	}
}
class One
{
	public static void main(String args[])
	{
		int k;
		Scanner sc=new Scanner(System.in);
		double side,radius;
		do
		{
			System.out.println("Get the Area and volume of shaes:");
			System.out.println("1.square 2. circle 3. cube 4. sphere 5.exit");
			System.out.println("Enter your choice:");
			k=sc.nextInt();
			switch(k)
			{
				case 1: System.out.println("Enter the side of the square:");
					side=sc.nextDouble();
					Square S=new Square(side);
					System.out.println("The Area is : "+S.getArea());
					System.out.println("The Volume is : "+S.getVolume());
					break;
				case 2: System.out.println("Enter the radius of the circle:");
					radius=sc.nextDouble();
					Circle C=new Circle(radius);
					System.out.println("The Area is : "+C.getArea());
					System.out.println("The Volume is : "+C.getVolume());
					break;
				case 3: System.out.println("Enter the side of the Cube:");
					side=sc.nextDouble();
					Cube Cu=new Cube(side);
					System.out.println("The Area is : "+Cu.getArea());
					System.out.println("The Volume is : "+Cu.getVolume());
					break;
				case 4: System.out.println("Enter the radius of the sphere:");
					radius=sc.nextDouble();
					Sphere Sp=new Sphere(radius);
					System.out.println("The Area is : "+Sp.getArea());
					System.out.println("The Volume is : "+Sp.getVolume());
					break;
				default: if(k!=5)
						System.out.println("Invalid choice:");
			}
		}while(k!=5);
	}
}
```

Output:



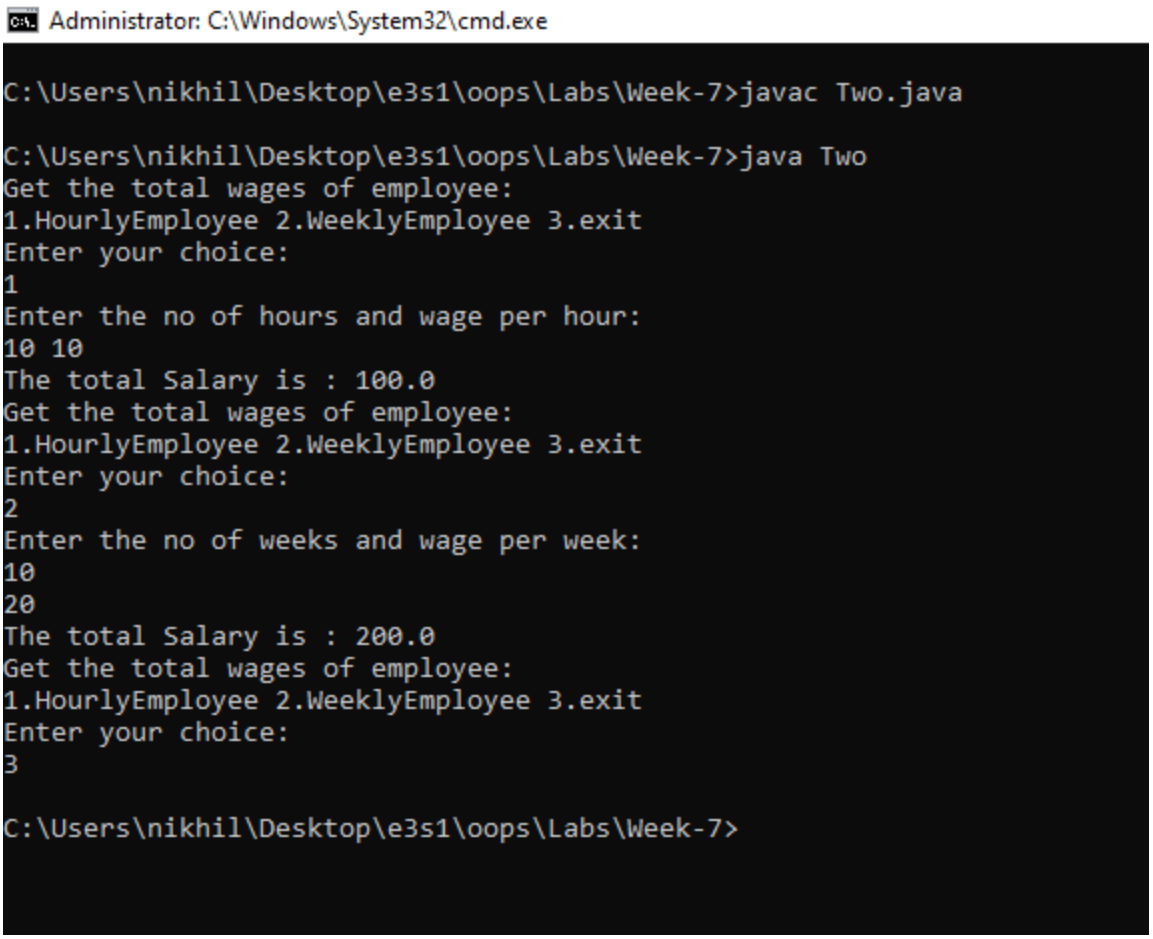
2. Create an abstract class Employee with methods getAmount() which displays the amount paid to employee. Reuse this class to calculate the amount to be paid to WeeklyEmployeed and HourlyEmployee according to no. of hours and total hours for HourlyEmployee and no. of weeks and total weeks for WeeklyEmployee.

Source Code:

```
/*
Create an abstract class Employee with methods getAmount() which displays the
amount paid to employee. Reuse this class to calculate the amount to be paid to
WeeklyEmployeed and HourlyEmployee according to no. of hours and total hours for
HourlyEmployee and no. of weeks and total weeks for WeeklyEmployee.
*/

import java.util.Scanner;
```

```
abstract class Employee
{
	abstract double getAmount();
}
class HourlyEmployee
{
	int noOfHours;
	double wagePerHour;
	HourlyEmployee(int noOfHours,double wagePerHour)
	{
		this.noOfHours=noOfHours;
		this.wagePerHour=wagePerHour;
	}
	double getAmount()
	{
		return noOfHours*wagePerHour;
	}
}
class WeeklyEmployee
{
	int noOfWeeks;
	double wagePerWeek;
	WeeklyEmployee(int noOfWeeks,double wagePerWeek)
	{
		this.noOfWeeks=noOfWeeks;
		this.wagePerWeek=wagePerWeek;
	}
	double getAmount()
	{
		return noOfWeeks*wagePerWeek;
	}
}
class Two
{
	public static void main(String args[])
	{
		int k,noOfWeeks,noOfHours;
		Scanner sc=new Scanner(System.in);
		double wagePerWeek, wagePerHour;
		do
		{
			System.out.println("Get the total wages of employee:");
			System.out.println("1.HourlyEmployee 2.WeeklyEmployee 3.exit");
			System.out.println("Enter your choice:");
			k=sc.nextInt();
			switch(k)
			{
				case 1: System.out.println("Enter the no of hours and wage per hour:");
					noOfHours=sc.nextInt();
					wagePerHour=sc.nextDouble();
					HourlyEmployee he=new HourlyEmployee(noOfHours,wagePerHour);
					System.out.println("The total Salary is : "+he.getAmount());
					break;
				case 2: System.out.println("Enter the no of weeks and wage per week:");
					noOfWeeks=sc.nextInt();
					wagePerWeek=sc.nextDouble();
					WeeklyEmployee we=new
WeeklyEmployee(noOfWeeks,wagePerWeek);
					System.out.println("The total Salary is : "+we.getAmount());
					break;
				default:
					if(k!=3)
						System.out.println("invalid choice");
			}
		}while(k!=3);
	}
}
```

Output:



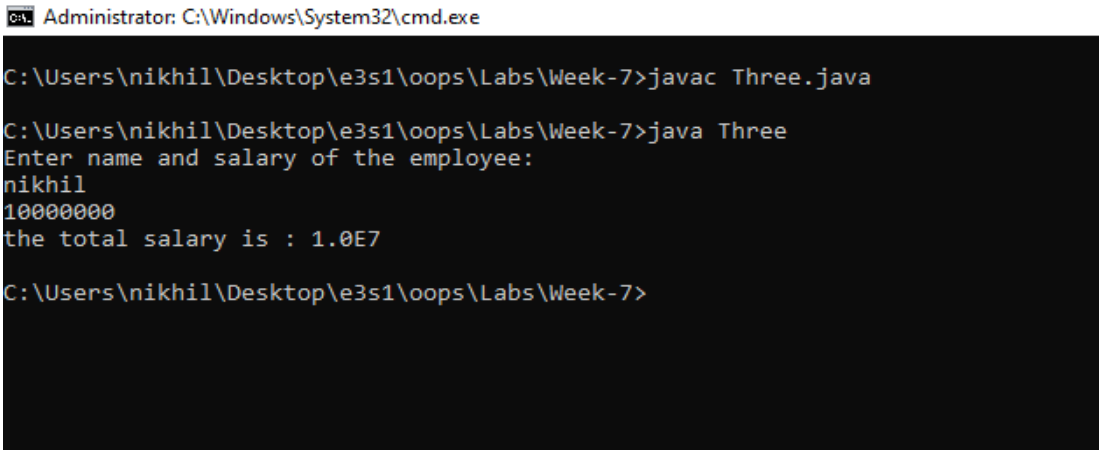
3. Create an Interface payable with method getAmount(). Calculate the amount to be paid to Invoice and Employee by implementing Interface.

Source Code:

```
/*
Create an Interface payable with method getAmount ().Calculate the amount to be
paid to Invoice and Employee by implementing Interface.
*/

import java.util.*;

interface Pay
{
        double getAmount();
}
class Employee implements Pay
{
        String name;
        double salary;
        Employee(String name,double salary)
        {
                this.name=name;
                this.salary=salary;
        }
        public double getAmount()
        {
                return this.salary;
        }
}
class Three
{
        public static void main(String args[])
        {
                Scanner sc=new Scanner(System.in);
```

```
                System.out.println("Enter name and salary of the employee:");
                String name=sc.nextLine();
                double salary=sc.nextDouble();
                Employee E=new Employee(name,salary);
                System.out.println("the total salary is : "+E.getAmount());
        }
}
```

Output:



4. Create an Interface Vehicle with method getColor(),getNumber(), getConsumption() calculate the fuel consumed, name and color for TwoWheeler and Four Wheeler By implementing interface Vehicle.

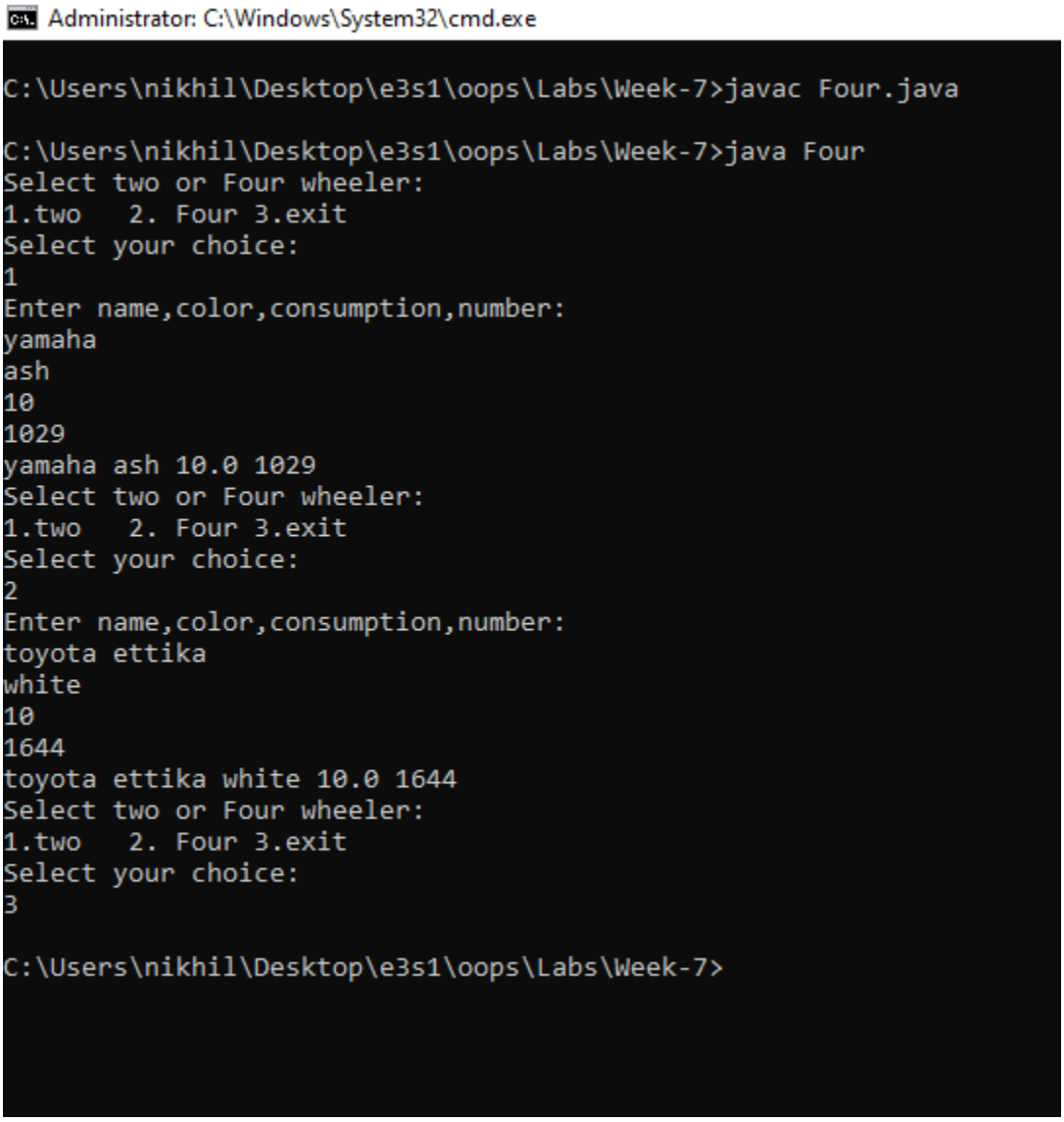
Source Code:

```
/*
Create an Interface Vehicle with method getColor(),getNumber(), getConsumption()
calculate the fuel consumed, name and color for TwoWheeler and Four Wheeler By
implementing interface Vehicle.
*/
import java.util.*;

interface Vehicle
{
        String getColor();
        double getConsumption();
        int getNumber();
}
class TwoWheeler implements Vehicle
{
        String name,color;
        double consumption;
        int number;
        TwoWheeler(String name,String color,double consumption,int number)
        {
                this.name=name;
                this.color=color;
                this.consumption=consumption;
                this.number=number;
        }
        public String getColor()
        {
                return this.color;
        }
        public double getConsumption()
        {
                return this.consumption;
        }
        public int getNumber()
```

```
		{
			return this.number;
		}
		public void Show()
		{
			System.out.println(this.name+ " "+this.getColor()+" "+this.getConsumption()+" "+this.getNumber());
		}
}
class FourWheeler implements Vehicle
{
		String name,color;
		double consumption;
		int number;
		FourWheeler(String name,String color,double consumption,int number)
		{
			this.name=name;
			this.color=color;
			this.consumption=consumption;
			this.number=number;
		}
		public String getColor()
		{
			return this.color;
		}
		public double getConsumption()
		{
			return this.consumption;
		}
		public int getNumber()
		{
			return this.number;
		}
		public void Show()
		{
			System.out.println(this.name+ " "+this.getColor()+" "+this.getConsumption()+" "+this.getNumber());
		}
}
class Four
{
		public static void main(String args[])
		{
			int k,number;
			double consumption;
			String name,color;
			Scanner sc=new Scanner(System.in);
			do
			{
				System.out.println("Select two or Four wheeler:");
				System.out.println("1.two		2. Four		3.exit");
				System.out.println("Select your choice:");
				k=sc.nextInt();
				switch(k)
				{
					case 1: System.out.println("Enter name,color,consumption,number:");
						sc.nextLine();
						name=sc.nextLine();
						color=sc.nextLine();
						consumption=sc.nextDouble();
						number=sc.nextInt();
						TwoWheeler two = new TwoWheeler(name,color,consumption,number);
						two.Show();
						break;
					case 2:  System.out.println("Enter name,color,consumption,number:");
						sc.nextLine();
						name=sc.nextLine();
						color=sc.nextLine();
						consumption=sc.nextDouble();
						number=sc.nextInt();
						FourWheeler four = new FourWheeler(name,color,consumption,number);
						four.Show();
						break;
					default:	if(k!=3)
								System.out.println("invalid choice");
```

```
                                        }
                                }while(k!=3);
                        }
}
```

Output:



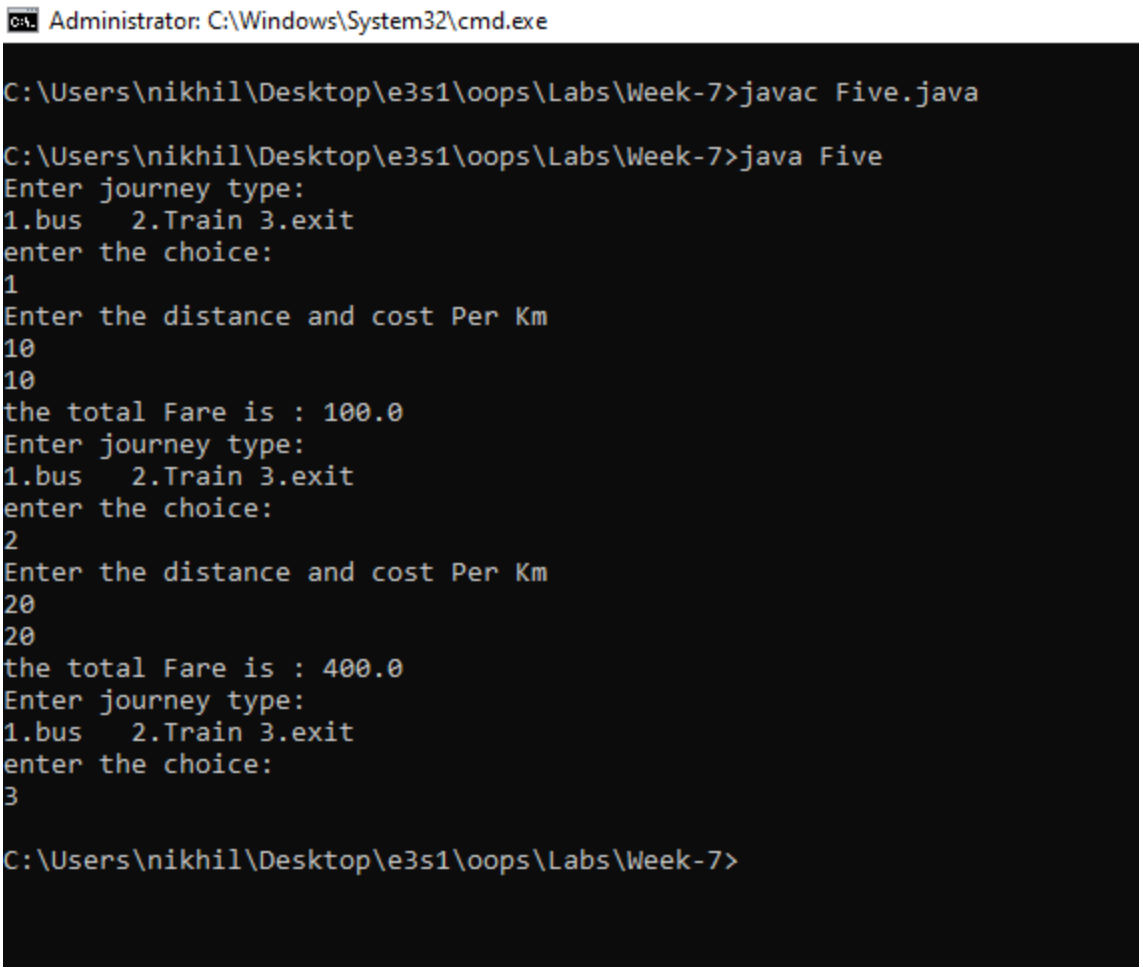
5. Create an Interface Fare with method getAmount() to get the amount paid for fare of travelling. Calculate the fare paid by bus and train implementing interface Fare.

Source Code:

```
/*
Create an Interface Fare with method getAmount() to get the amount paid for fare of
travelling. Calculate the fare paid by bus and train implementing interface Fare.
*/

import java.util.*;

interface Fare
{
        double getAmount();
}
class Bus implements Fare
```

```
{
	double distance,costPerKm;
	Bus(double distance,double costPerKm)
	{
		this.distance=distance;
		this.costPerKm=costPerKm;
	}
	public double getAmount()
	{
		return this.distance*this.costPerKm;
	}
}
class Train implements Fare
{
	double distance,costPerKm;
	Train(double distance,double costPerKm)
	{
		this.distance=distance;
		this.costPerKm=costPerKm;
	}
	public double getAmount()
	{
		return this.distance*this.costPerKm;
	}
}
class Five
{
	public static void main(String args[])
	{
		int k;
		Scanner sc=new Scanner(System.in);
		double distance,costPerKm;
		do
		{
			System.out.println("Enter journey type:");
			System.out.println("1.bus		2.Train		3.exit");
			System.out.println("enter the choice:		");
			k=sc.nextInt();
			switch(k)
			{
				case 1:	System.out.println("Enter the distance and cost Per Km");
					distance=sc.nextDouble();
					costPerKm=sc.nextDouble();
					Bus b=new Bus(distance,costPerKm);
					System.out.println("the total Fare is : "+b.getAmount());
					break;
				case 2:	System.out.println("Enter the distance and cost Per Km");
					distance=sc.nextDouble();
					costPerKm=sc.nextDouble();
					Train t=new Train(distance,costPerKm);
					System.out.println("the total Fare is : "+t.getAmount());
					break;
				default:	if(k!=3)
							System.out.println("invalid option");
			}
		}while(k!=3);
	}
}
```

Output:



6. Create an Interface StudentFee with method getAmount(),getFirstName(),getLastName(), getAddress(), getContact(). Calculate the amount paid by the Hostler and NonHostler student by implementing interface Student Fee

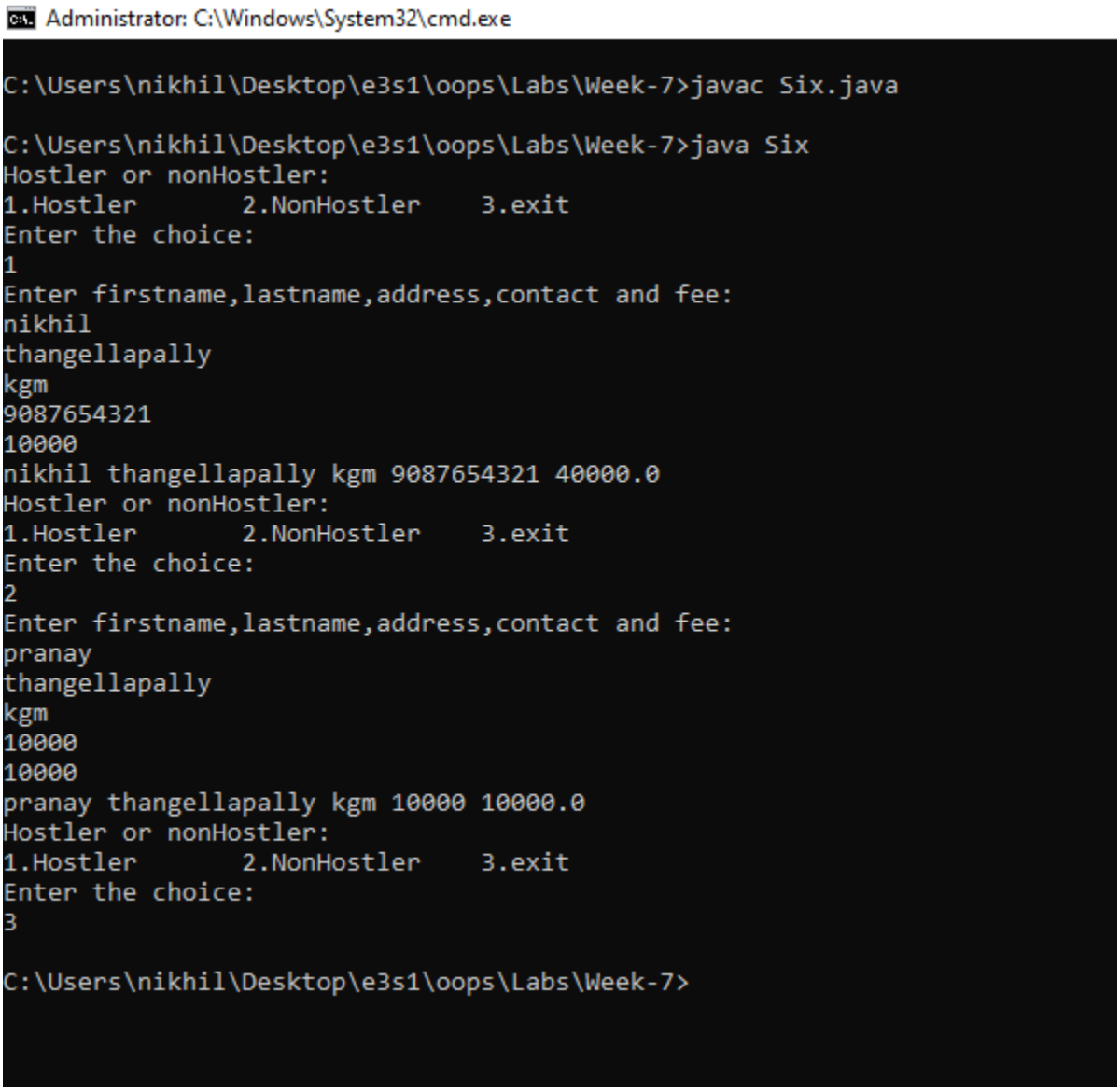
Source Code:

```
/*
.Create an Interface StudentFee with method
getAmount(),getFirstName(),getLastName(), getAddress(), getContact(). Calculate the
amount paid by the Hostler and NonHostler student by implementing interface Student
Fee
*/
import java.util.*;

interface StudentFee
{
        double getAmount();
        String getFirstName();
        String getLastName();
        String getAddress();
        String getContact();
}
class Hostler implements StudentFee
{
        String fName,lName,address,contact;
        double fee;
        Hostler(String fName,String lName,String address,String contact,double fee)
        {
                this.fName=fName;
                this.lName=lName;
```

```
                this.address=address;
                this.contact=contact;
                this.fee=fee+30000;
        }
        public double getAmount()
        {return this.fee;}
        public String getFirstName()
        {return this.fName;}
        public String getLastName()
        {return this.lName;}
        public String getAddress()
        {return this.address;}
        public String getContact()
        {return this.contact;}
        public void Show()
        {
                System.out.println(this.getFirstName()+" "+this.getLastName()+" "+this.getAddress()+"
"+this.getContact()+" "+this.getAmount());
        }
}
class NonHostler implements StudentFee
{
        String fName,lName,address,contact;
        double fee;
        NonHostler(String fName,String lName,String address,String contact,double fee)
        {
                this.fName=fName;
                this.lName=lName;
                this.address=address;
                this.contact=contact;
                this.fee=fee;
        }
        public double getAmount()
        {return this.fee;}
        public String getFirstName()
        {return this.fName;}
        public String getLastName()
        {return this.lName;}
        public String getAddress()
        {return this.address;}
        public String getContact()
        {return this.contact;}
        public void Show()
        {
                System.out.println(this.getFirstName()+" "+this.getLastName()+" "+this.getAddress()+"
"+this.getContact()+" "+this.getAmount());
        }
}

class Six
{
        public static void main(String args[])
        {
                int k;
                String fName,lName,address,contact;
                double fee;
                Scanner sc=new Scanner(System.in);
                do
                {
                        System.out.println("Hostler or nonHostler:");
                        System.out.println("1.Hostler      2.NonHostler            3.exit");
                        System.out.println("Enter the choice:");
                        k=sc.nextInt();
                        switch(k)
                        {
                                case 1:
                                        System.out.println("Enter firstname,lastname,address,contact and fee:");
                                        sc.nextLine();
                                        fName=sc.nextLine();
                                        lName=sc.nextLine();
                                        address=sc.nextLine();
                                        contact=sc.nextLine();
                                        fee=sc.nextInt();
```

```
                                        Hostler h=new Hostler(fName,lName,address,contact,fee);
                                        h.Show();
                                        break;
                                case 2:  System.out.println("Enter firstname,lastname,address,contact and fee:");
                                        sc.nextLine();
                                        fName=sc.nextLine();
                                        lName=sc.nextLine();
                                        address=sc.nextLine();
                                        contact=sc.nextLine();
                                        fee=sc.nextInt();
                                        NonHostler nh=new NonHostler(fName,lName,address,contact,fee);
                                        nh.Show();
                                        break;
                                default:
                                        if(k!=3)
                                                System.out.println("invalid choice:");
                                }
                        }while(k!=3);
                }
        }
```

Output:



Week-8

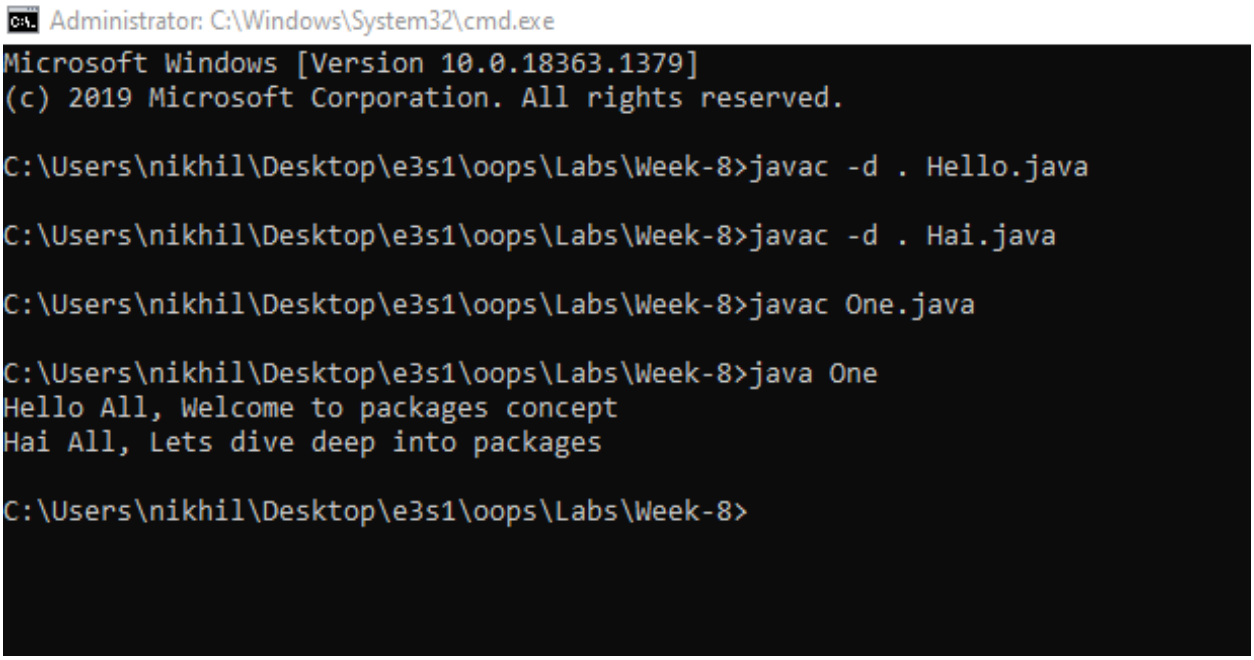
1. Write a Program to create your own package. Package should have more than two classes. write a Program that uses the classes from the package.

```
package OOPS.One;
public class Hai
{
	public void show()
	{
		System.out.println("Hai All, Lets dive deep into packages");
	}
}
```

```
package OOPS.One;
public class Hello
{
	public void show()
	{
		System.out.println("Hello All, Welcome to packages concept");
	}
}
```

```
/*
Write a Program to create your own package. Package should have more than two
classes. write a Program that uses the classes from the package.
*/
import OOPS.One.Hello;
import OOPS.One.Hai;
class One
{
	public static void main(String args[])
	{
		Hello h1=new Hello();
		h1.show();
		Hai h2=new Hai();
		h2.show();
	}
}
```

Output:



2. Create a package named org.shapes. Create some classes in the package representing some common geometric shapes like Square, Triangle, Circle and so on. write a Program that uses the classes from the package.

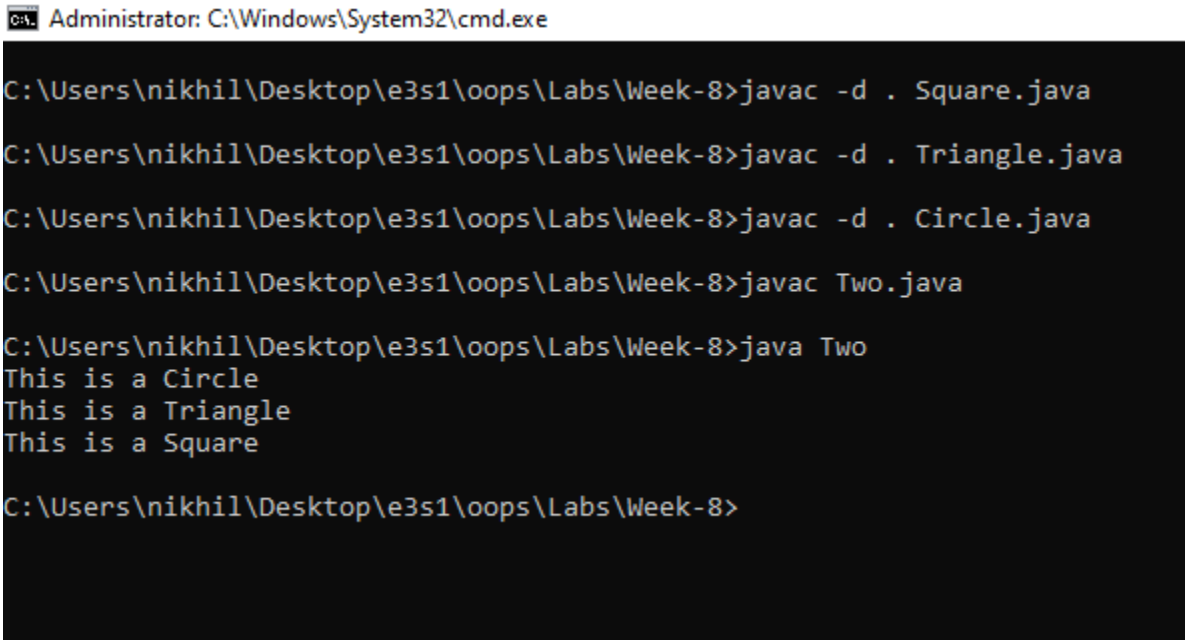
```
package OOPS.Two.org.shapes;
public class Square
{
	public Square()
	{
		System.out.println("This is a Square");
	}
}
```

```
package OOPS.Two.org.shapes;
public class Triangle
{
	public Triangle()
	{
		System.out.println("This is a Triangle");
	}
}
```

```
package OOPS.Two.org.shapes;
public class Circle
{
	public Circle()
	{
		System.out.println("This is a Circle");
	}
}
```

```
/*
Create a package named org.shapes. Create some classes in the package representing
some common geometric shapes like Square, Triangle, Circle and so on. write a Program
that uses the classes from the package.
*/
import OOPS.Two.org.shapes.Circle;
import OOPS.Two.org.shapes.Triangle;
import OOPS.Two.org.shapes.Square;
public class Two
{
        public static void main(String args[])
        {
                Circle c=new Circle();
                Triangle t=new Triangle();
                Square s=new Square();
        }
}
```

Output:



3. Write a Java program to create package called dept. Create four classes as CSE, ECE, ME and CE add methods in each class which can display subject names of your respect year. access this package classes from main class

```
package OOPS.Three.dept;
public class CSE
{
        public CSE()
        {
                System.out.println("Subjects:C,OOPS using java,AI,DS,DAA");
        }
}
```

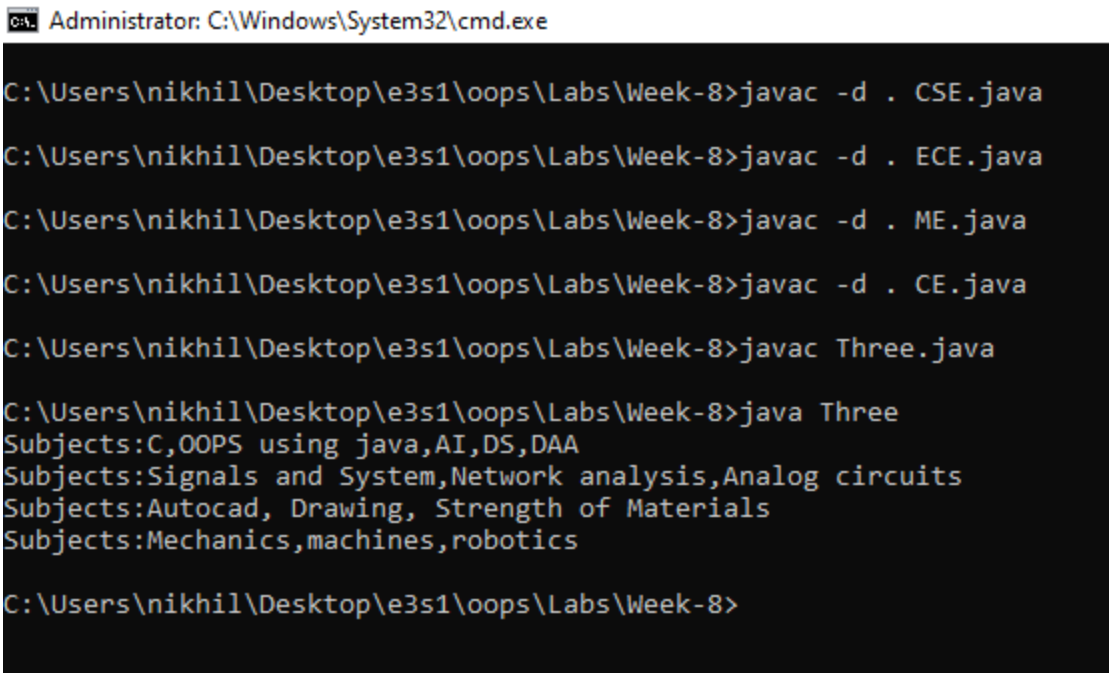
```
package OOPS.Three.dept;
public class CE
```

```
	{
		public CE()
		{
		System.out.println("Subjects:Autocad, Drawing, Strength of Materials");
		}
}
```

```
	package OOPS.Three.dept;
	public class ECE
	{
		public ECE()
		{
			System.out.println("Subjects:Signals and System,Network analysis,Analog
	circuits");
		}
}
```

```
	package OOPS.Three.dept;
	public class ME
	{
		public ME()
		{
			System.out.println("Subjects:Mechanics,machines,robotics");
		}
}
```

```
	/*
	 Write a Java program to create package called dept. Create four classes as CSE, ECE,
	ME and CE add methods in each class which can display subject names of your respect
	year. access this package classes from main class
	*/
	import OOPS.Three.dept.CSE;
	import OOPS.Three.dept.ECE;
	import OOPS.Three.dept.CE;
	import OOPS.Three.dept.ME;
	public class Three
	{
		public static void main(String args[])
		{
			CSE cse=new CSE();
			ECE ece=new ECE();
			CE ce=new CE();
			ME me=new ME();
		}
}
```



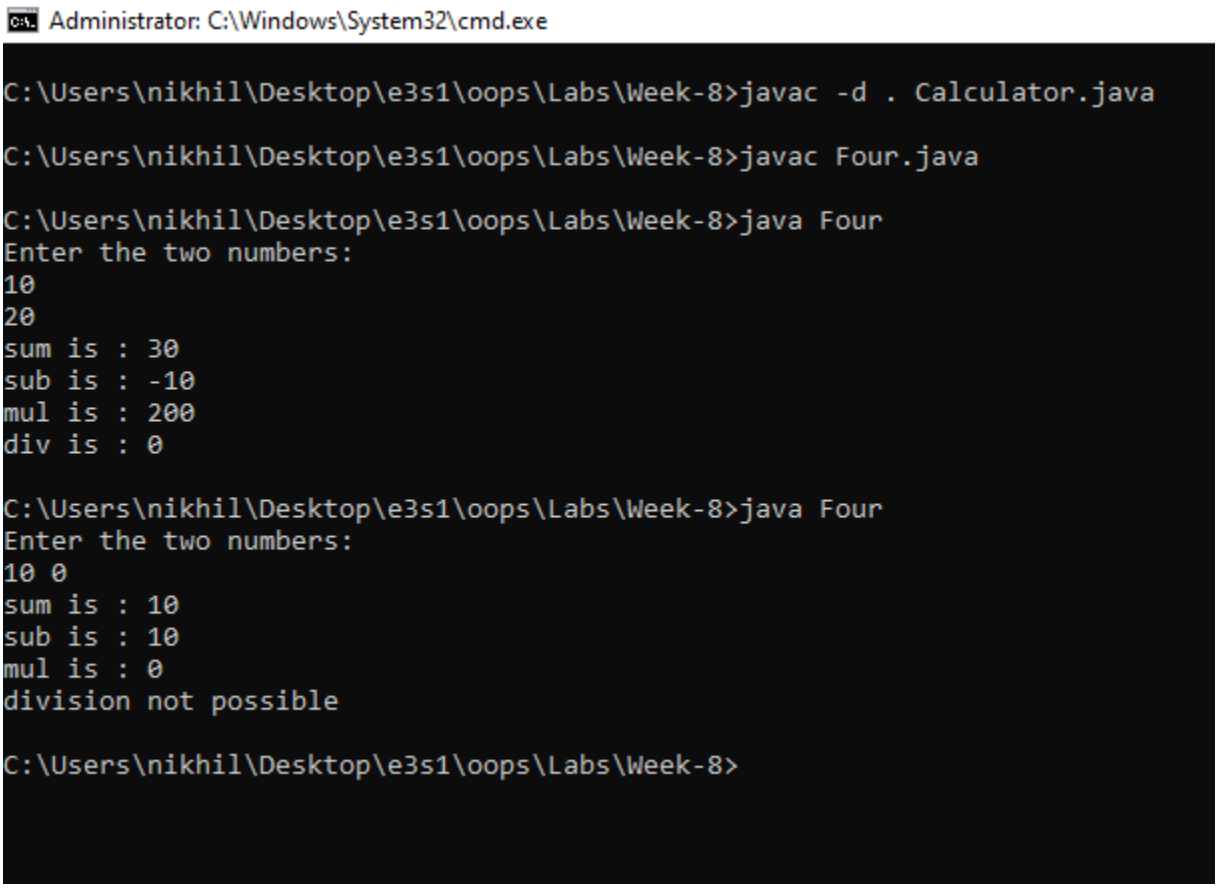
4. Write a Calculator program : Include all calculator operations in as classes in a Package “Calculator” and import in to main class.

```
package OOPS.Four.Calci;
public class Calculator
{
        public void Sum(int a,int b)
        {       System.out.println("sum is : "+(a+b));    }
        public void Sub(int a,int b)
        {       System.out.println("sub is : "+(a-b));    }
        public void Mul(int a,int b)
        {       System.out.println("mul is : "+(a*b));    }
        public void Div(int a,int b)
        {
                if(b!=0)
                        System.out.println("div is : "+(a/b));
                else
                        System.out.println("division not possible");
        }
}
```

```
import java.util.*;
import OOPS.Four.Calci.Calculator;
class Four
{
        public static void main(String args[])
        {
                int a,b;
                System.out.println("Enter the two numbers:");
                Scanner sc=new Scanner(System.in);
                a=sc.nextInt();
```

```
                        b=sc.nextInt();
                        Calculator c=new Calculator();
                        c.Sum(a,b);
                        c.Sub(a,b);
                        c.Mul(a,b);
                        c.Div(a,b);
                }
}
```

Output:



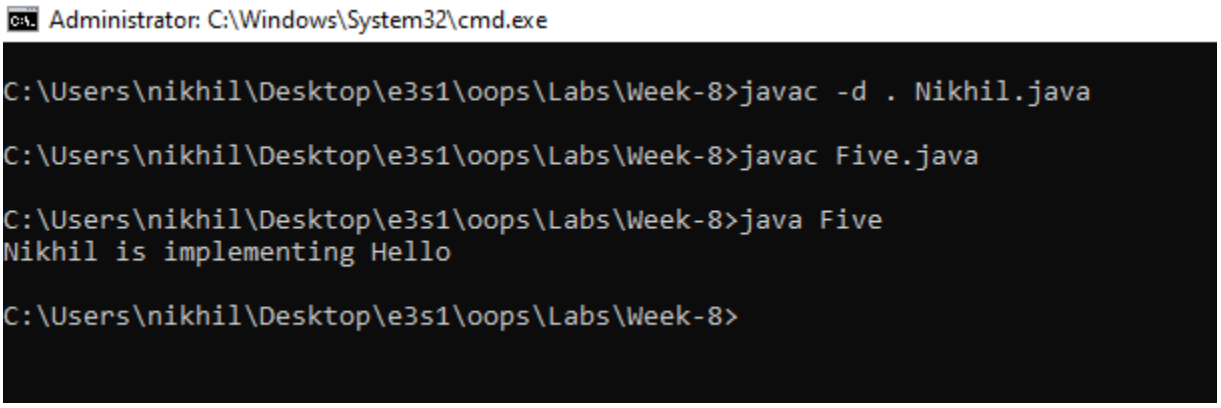
5. Write a program for the following a. Example to use interfaces in Packages. b. Example to create sub package in a package

```
        package OOPS.Five;
        interface Hello                 //(b)
        {
                public void display();
        }
        public class Nikhil implements Hello
        {
                public void display()
                {
```

```
                System.out.println("Nikhil is implementing Hello");
        }
}
```

```
/*
Write a program for the following
a. Example to use interfaces in Packages. b. Example to create sub package in a
package.
*/
import OOPS.Five.Nikhil;  // this package is a nested package(a)
class Five
{
        public static void main(String args[])
        {
                Nikhil nik=new Nikhil();
                nik.display();
        }
}
```

Output:



Week-9

1. Program for demonstrating the use of throw, throws & finally - Create a class with a main( ) that throws an object of class Exception inside a try block. Give the constructor for Exception a String argument. Catch the exception inside a catch clause and print the String argument. Add a finally clause and print a message to prove you were there.

Source Code:

```
import java.util.*;

class One
{
	int show(int a,int b) throws ArithmeticException
	{
		return a/b;
	}
	public static void main(String args[])
	{
		int a,b;
		System.out.println("Enter two numbers:");
		Scanner sc=new Scanner(System.in);
		a=sc.nextInt();
		b=sc.nextInt();
		try
		{
			One A=new One();
			System.out.println(A.show(a,b));
			throw new Exception("hello world");
		}
		catch(ArithmeticException E)
		{
			System.out.println("Error:"+E.getMessage());
		}
		catch(Exception e)
		{
			System.out.println("Error:"+e.getMessage());
		}
		finally
		{
			System.out.println("Any how i will get executed");
		}
	}
}
```

Output:

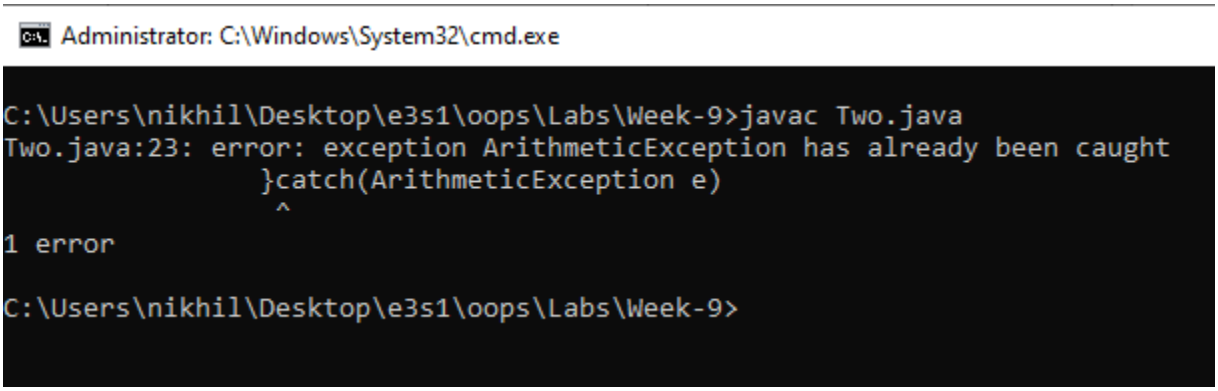
```
Microsoft Windows [Version 10.0.18363.1379]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\nikhil\Desktop\e3s1\oops\Labs\Week-9>javac One.java

C:\Users\nikhil\Desktop\e3s1\oops\Labs\Week-9>java One
Enter two numbers:
10 0
Error:/ by zero
Any how i will get executed

C:\Users\nikhil\Desktop\e3s1\oops\Labs\Week-9>java One
Enter two numbers:
10 10
1
Error:hello world
Any how i will get executed

C:\Users\nikhil\Desktop\e3s1\oops\Labs\Week-9>
```

2. Write a program that shows that the order of the catch blocks is important. If you try to catch a superclass exception type before a subclass type, the compiler should generate errors.

Source Code:

```
class Two
{
	int division(int a,int b) throws ArithmeticException
	{
		return a/b;
	}
	public static void main(String args[])
	{
		Two t=new Two();
		try
		{
			System.out.println(t.division(100,0));
		}
		catch(Exception E)
		{
			System.out.println(E.getMessage());
		}catch(ArithmeticException e)
		{
			System.out.println(e.getMessage());
		}
	}
}
```

Output:



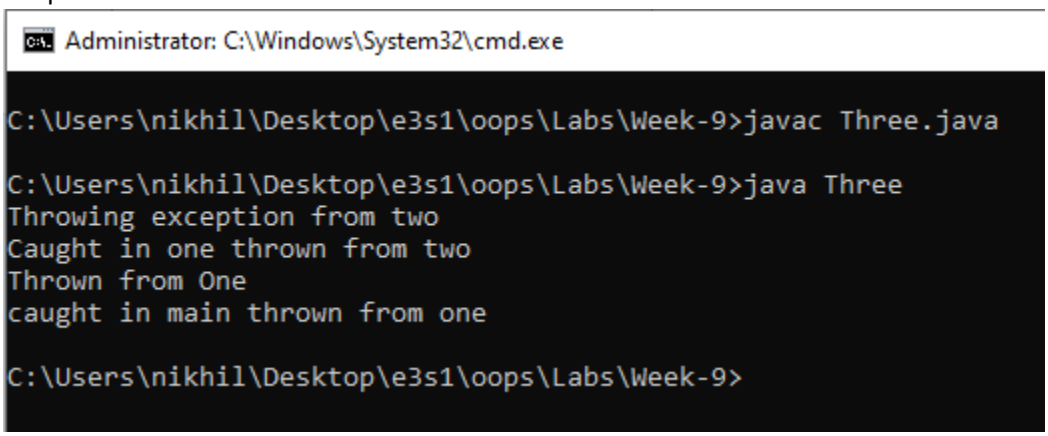
3. Write a program to rethrow an exception – Define methods one() & two(). Method two() should initially throw an exception. Method one() should call two(), catch the exception and rethrow it Call one() from main() and catch the rethrown

Source Code

```
class Three
{
	void two() throws Exception
	{
		System.out.println("Throwing exception from two");
		throw new Exception();
	}
	void one() throws Exception
	{
		try{
			this.two();
		}
		catch(Exception e)
		{
			System.out.println("Caught in one thrown from two");
			System.out.println("Thrown from One");
			throw e;
		}
	}
	public static void main(String args[])
```

```
	{
		Three t=new Three();
		try
		{
			t.one();
		}
		catch(Exception e)
		{
			System.out.println("caught in main thrown from one");
		}
	}
}
```

Output:

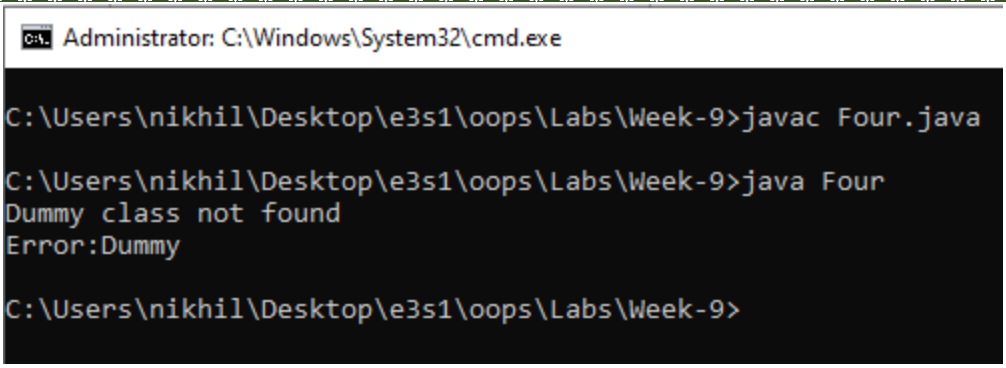


4. Exception Handling program for ClassNotFoundException--thrown if a program can not find a class it depends at runtime (i.e., the class's ".class" file cannot be found or was removed from the CLASSPATH).

Source Code:

```
class Four
{
	public static void main(String args[])
	{
		try
		{
			Class.forName("Dummy");
		}
		catch(ClassNotFoundException e)
		{
			System.out.println("Dummy class not found");
			System.out.println("Error:"+e.getMessage());
		}
		catch(Exception E)
		{
			System.out.println("Error:"+E.getMessage());
		}
	}
}
```

Output:

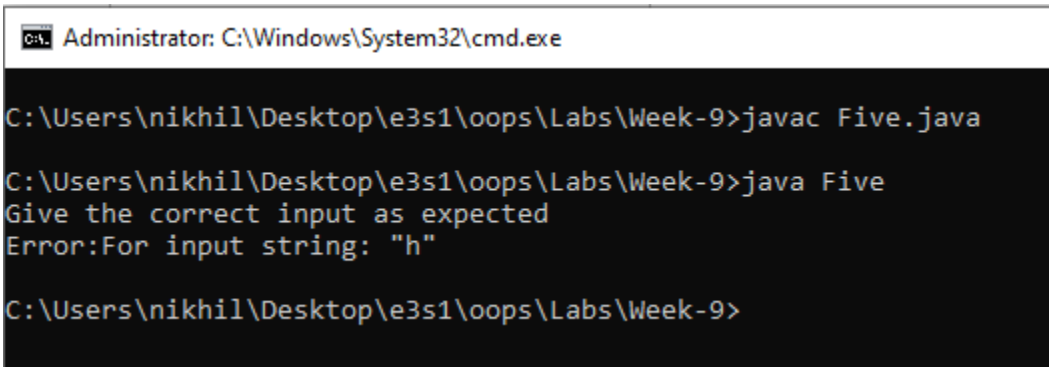


5. Exception Handling program for NumberFormatException--thrown if a program is attempting to convert a string to a numerical datatype, and the string contains inappropriate characters (i.e. 'z' or 'Q').

Source Code:

```
class Five
{
	public static void main(String args[])
	{
		try
		{
			int a=Integer.parseInt("h");
		}
		catch(NumberFormatException e)
		{
			System.out.println("Give the correct input as expected");
			System.out.println("Error:"+e.getMessage());
		}
		catch(Exception E)
		{
			E.printStackTrace();
		}
}
```

Output:



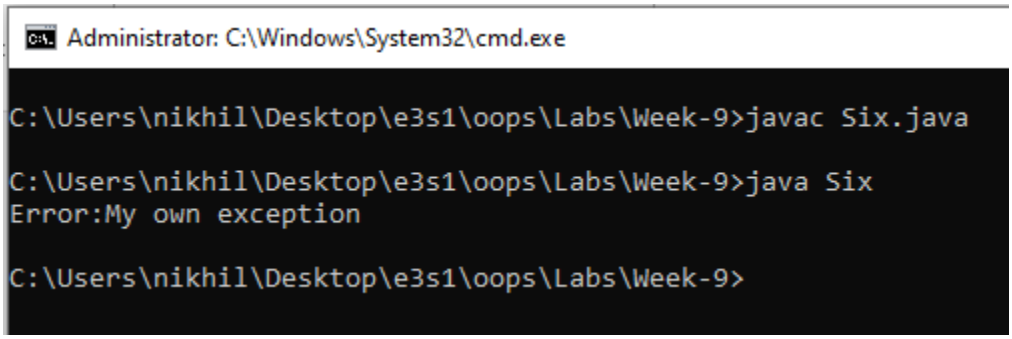
6. Create your own exception class using the extends keyword. Write a constructor for this class that takes a String argument and stores it inside the object with a String reference. Write a method that prints out the stored String. Create a try- catch clause to exercise your new exception.

Source Code:

```
class MyException extends Exception
{
	MyException(String s)
	{
		super(s);
	}
```

```
}
class Six
{
	public static void main(String args[])
	{
		try
		{
			throw new MyException("My own exception");
		}
		catch(MyException e)
		{
			System.out.println("Error:"+e.getMessage());
		}
	}
}
```

Output:

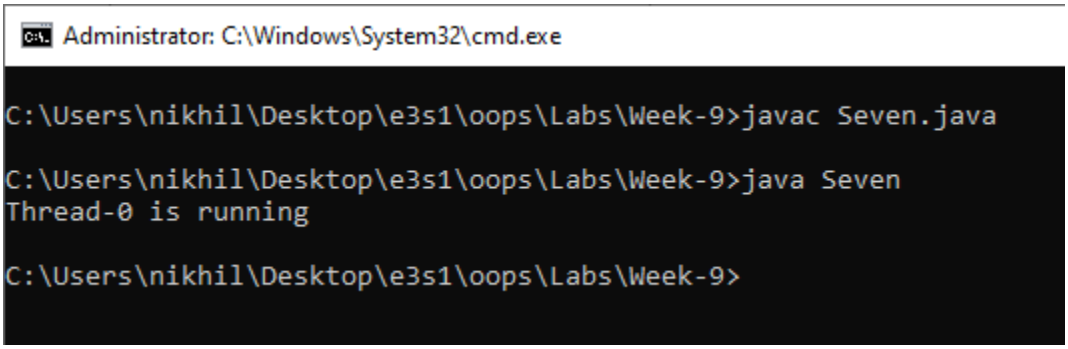


7. Write a program to create MyThread class with run() method and then attach a thread to this MyThread class object

Source Code

```
class MyThread extends Thread
{
	public void run()
	{
		System.out.println(Thread.currentThread().getName()+" is running");
	}
}
class Seven
{
	public static void main(String args[])
	{
		MyThread mt=new MyThread();
		mt.start();
	}
}
```

Output

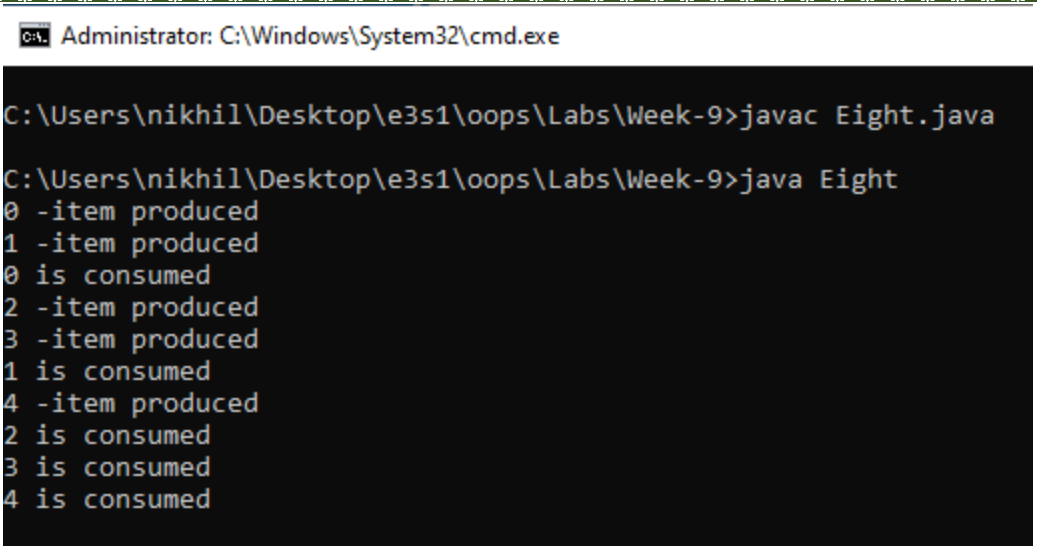


8. Write a program where the consumer thread checks the data production status [is over or not] for every 10 ms.

Source Code:

```
import java.util.*;

class Producer extends Thread
{
	Vector<Integer> v;
	public Producer()
	{
		v=new Vector<Integer>();
	}
	public void run()
	{
		for(int i=0;i<5;i++)
		{
			while(v.size()>2)
			{
				try{Thread.sleep(1);}
				catch(InterruptedException e){e.printStackTrace();}
			}
			System.out.println(i+" -item produced");
			v.add((Integer)i);
			try{Thread.sleep(1);}
			catch(InterruptedException e){e.printStackTrace();}
		}
	}
}
class Consumer extends Thread
{
	Producer p;
	public Consumer(Producer p)
	{
		this.p=p;
	}
	public void run()
	{
		for(int i=0;i<5;i++)
		{
			while(this.p.v.size()<1)
			{
				try{Thread.sleep(1);}
				catch(InterruptedException e){e.printStackTrace();}
			}
			System.out.println(this.p.v.remove(0)+" is consumed");
			try{Thread.sleep(10);}
			catch(InterruptedException e){e.printStackTrace();}
		}
	}
}
class Eight
{
	public static void main(String args[])
	{
		Producer p=new Producer();
		Consumer c=new Consumer(p);
		Thread t1=new Thread(p);
		Thread t2=new Thread(c);
		t1.start();
		t2.start();
	}
}
```

Output:



9. Write a Program using Threads to simulate a traffic light. The Signal lights should glow after each 10 second, one by one. For example: Firstly Red, then after 10 seconds, red will be put to off and yellow will start glowing and then accordingly green.

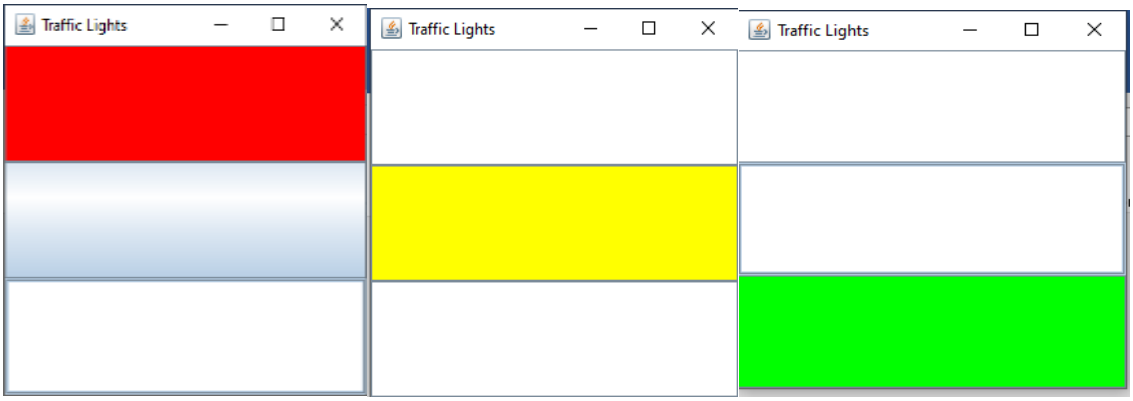
Source Code:

```
import java.util.*;
import java.awt.*;
import javax.swing.*;

class Nine extends JFrame implements Runnable
{
	JButton red,green,yellow;
	class Red extends Thread
	{
		public void run()
		{
			System.out.println("Red");
			green.setBackground(Color.white);
			red.setBackground(Color.red);
			try
			{
				Thread.sleep(10000);
			}
			catch(InterruptedException e)
			{
				e.printStackTrace();
			}
		}
	}
	class Yellow extends Thread
	{
		public void run()
		{
			red.setBackground(Color.white);
			yellow.setBackground(Color.yellow);
			System.out.println("Yellow");
			try
			{
				Thread.sleep(10000);
			}
			catch(InterruptedException e)
			{
				e.printStackTrace();
			}
		}
	}
	class Green extends Thread
	{
```

```
		public void run()
		{
			yellow.setBackground(Color.white);
			green.setBackground(Color.green);
			System.out.println("Green");
			try
			{
				Thread.sleep(10000);
			}
			catch(InterruptedException e)
			{
				e.printStackTrace();
			}
		}
	}
	public void run()
	{
		Thread redThread=new Thread(new Red());
		redThread.start();
		synchronized(redThread){
			try
			{
				redThread.wait();
			}
			catch(InterruptedException e)
			{
				e.printStackTrace();
			}
		}
		Thread yellowThread=new Thread(new Yellow());
		yellowThread.start();
		synchronized(yellowThread){
			try
			{
				yellowThread.wait();
			}
			catch(InterruptedException e)
			{
				e.printStackTrace();
			}
		}
		Thread greenThread=new Thread(new Green());
		greenThread.start();
		synchronized(greenThread){
			try
			{
				greenThread.wait();
			}
			catch(InterruptedException e)
			{
				e.printStackTrace();
			}
		}
		System.exit(0);
	}
	public Nine()
	{
		this.setVisible(true);
		this.setTitle("Traffic Lights");
		this.setBackground(Color.BLACK);
		this.setSize(300,300);
		this.setLayout(new GridLayout(3,1));
		red=new JButton();
		yellow=new JButton();
		green=new JButton();
		this.add(red);
		this.add(yellow);
		this.add(green);
		new Thread(this).start();
	}
	public static void main(String args[])
	{
		new Nine();
```

```
        }
    }
```

Output:

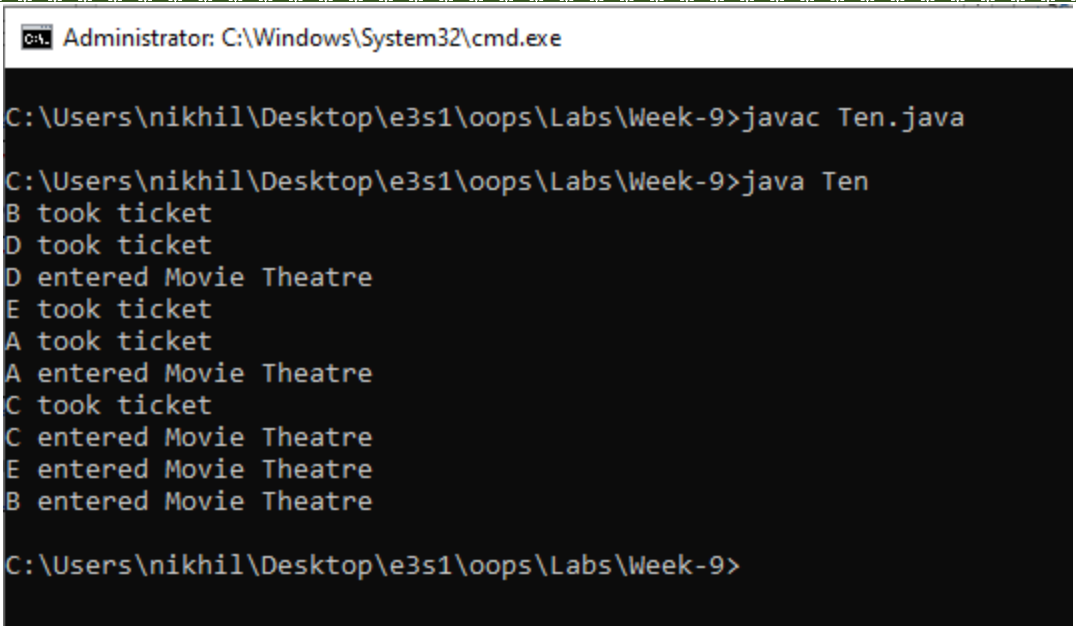


10. Write a Program using Threads for the following case study: Movie Theatre To watch a movie the following process is to be followed, at first get the ticket then show the ticket. Assume that N persons are trying to enter the Theatre hall all at once, display their sequence of entry into theater. Note: The person should enter only after getting a ticket and showing it to the boy.

Source Code:

```
class Customer extends Thread
{
        String name;
        Theatre theatre;
        Customer(String name,Theatre theatre){
                this.name=name;
                this.theatre=theatre;
        }
        public void run(){
                theatre.getTicket(this);
                theatre.enterTheatre(this);
        }
}
class Theatre
{
        public synchronized  void getTicket(Customer c)
        {       System.out.println(c.name+" took ticket");
        }
        public synchronized void enterTheatre(Customer c)
        {       System.out.println(c.name+" entered Movie Theatre");
        }
}
class Ten
{
        public static void main(String args[])
        {
                Theatre t=new Theatre();
                new Thread(new Customer("A",t)).start();
                new Thread(new Customer("B",t)).start();
                new Thread(new Customer("C",t)).start();
                new Thread(new Customer("D",t)).start();
                new Thread(new Customer("E",t)).start();
        }
}
```

Output:

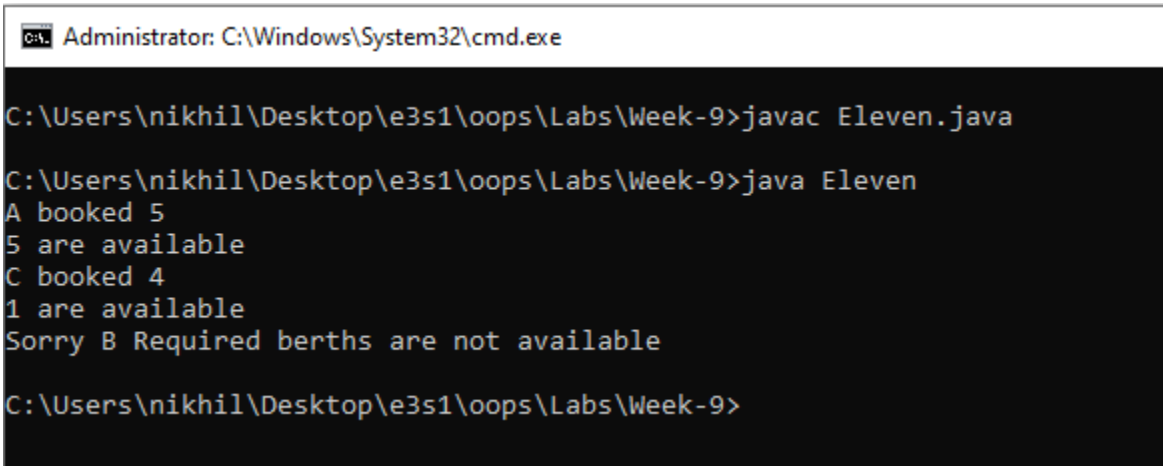


11. Write a Program using Threads for the following case study: Train Reservation system
To reserve a berth the following process need to be followed, at first check the number of available berths with the requested berths, if the number of requested berths are less than or equal to available berths then allot berth and print ticket or else display no berths are available. Assume that N persons are trying to reserve the berth, display their sequence of reservation status along with the number of available berths. Note: The person can print ticket only if berth is confirmed.

Source Code:

```
class Train
{
	int berths;
	Train(int berths)
	{	this.berths=berths;	}
	public synchronized void reserve(Customer c)
	{
		if(c.no_of_berths<=berths)
		{
			berths-=c.no_of_berths;
			System.out.println(c.name+" booked "+c.no_of_berths);
			System.out.println(berths +" are available");
		}
		else
		{
			System.out.println("Sorry "+c.name+" Required berths are not available");;
		}
	}
}
class Customer extends Thread
{
	String name;
	Train t;
	int no_of_berths;
	Customer(String name,Train t,int no_of_berths)
	{
		this.name=name;
		this.t=t;
		this.no_of_berths=no_of_berths;
	}
	public  void run()
	{
		this.t.reserve(this);
	}
```

```
	}
	class Eleven
	{
		public static void main(String args[])
		{
			Train t=new Train(10);
			new Customer("A",t,5).start();
			new Customer("B",t,3).start();
			new Customer("C",t,4).start();
		}
}
```



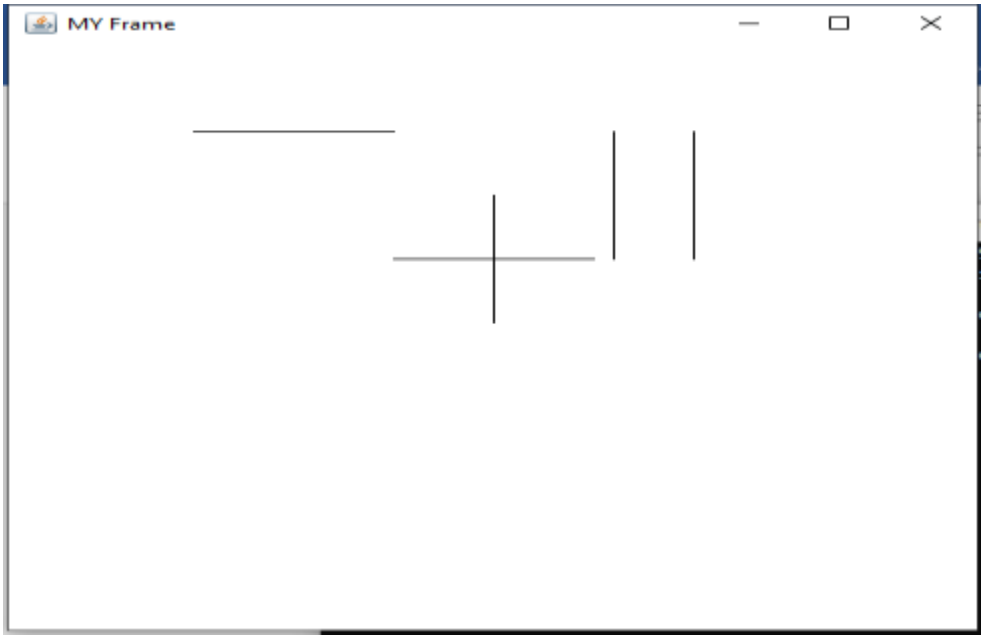
Week-10

1. Write a program for the following a. display a frame with title MyFrame b. draw a horizontal line. c. Draw one line perpendicular to other. One line parallel to other.

Source Code:

```
import java.util.*;
import java.awt.*;

class One extends Frame
{
        One()
        {
                this.setTitle("MY Frame");
                this.setSize(500,500);
                this.setVisible(true);
        }
        public void paint(Graphics g)
        {
                g.drawLine(100,100,200,100);    //horizontal line
                g.drawLine(200,200,300,200);
                g.drawLine(250,250,250,150);

                g.drawLine(310,100,310,200);
                g.drawLine(350,100,350,200);
        }
        public static void main(String args[])
        {
                One o=new One();
        }
}
```

Output:

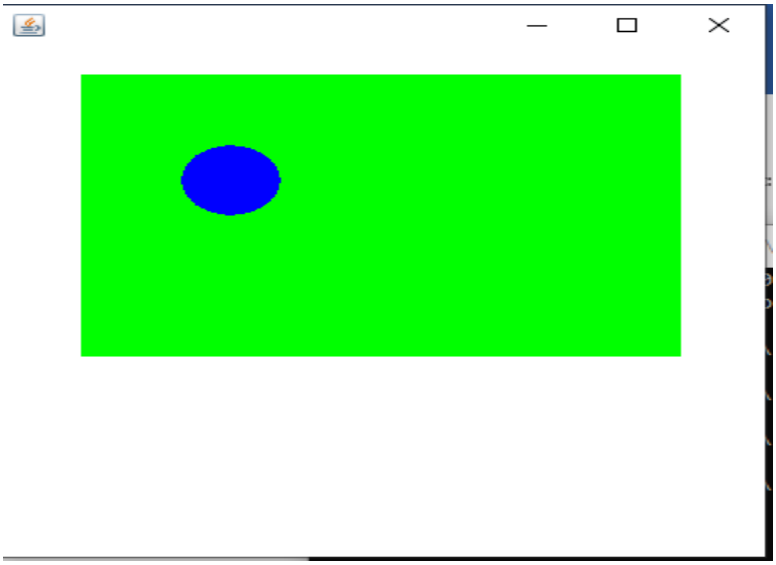


2. Create an application to display a circle within rectangle and fill different colors in the circle & rectangle

Source Code:

```
import java.awt.*;
class Two extends Frame
{
        Two()
        {
                this.setSize(400,400);
                this.setVisible(true);
        }
        public void paint(Graphics g)
        {
                //g.fillRect(x,y,width,height);
                g.setColor(Color.green);
                g.fillRect(50,50,300,200);
                g.setColor(Color.blue);
                g.fillOval(100,100,50,50);
        }
        public static void main(String args[])
        {
                Two t=new Two();
        }
}
```

Output:



3. Write an application that displays any string. Choose color from combo box to change the color of this displayed string and choose its size & type respectively from another two combo boxes.

Source Code

```
import java.awt.*;
import java.awt.event.ItemEvent;
import java.awt.event.ItemListener;

class Three extends Frame
{
	Choice c1,c2,c3;
	Color color=Color.black;
	int type=Font.BOLD;
	int size=50;
	String n1,n2,n3;
	Three()
	{
		this.setVisible(true);
		this.setSize(600,600);
		c1=new Choice();
		c1.setBounds(50,200,100,50);
		c1.add("red"); c1.add("green"); c1.add("blue");
		c2=new Choice();
		c2.setBounds(200,200,100,50);
		c2.add("bold"); c2.add("plain"); c2.add("italic");
		c3=new Choice();
		c3.setBounds(50,100,100,50);
		c3.add("10"); c3.add("50"); c3.add("100");
		c1.setFont(new Font("arial",Font.BOLD,30));
		c2.setFont(new Font("arial",Font.BOLD,30));
		c3.setFont(new Font("arial",Font.BOLD,30));
		c1.addItemListener(new ItemListener() {
			public void itemStateChanged(ItemEvent e) {
				n1=e.getItem().toString();
				if(n1.equals("red"))
					color=Color.red;
				else if(n1.equals("green"))
					color=Color.green;
				else if(n1.equals("blue"))
					color=Color.blue;
				repaint();
			}
		});
		c2.addItemListener(new ItemListener() {
			public void itemStateChanged(ItemEvent e){
				n2=e.getItem().toString();
				if(n2.equals("bold"))
					type=Font.BOLD;
				else if(n2.equals("plain"))
					type=Font.PLAIN;
				else if(n2.equals("italic"))
					type=Font.ITALIC;
				repaint();
			}
		});
		c3.addItemListener(new ItemListener() {
			public void itemStateChanged(ItemEvent e){
```

```
                                        n3=e.getItem().toString();
                                        if(n3.equals("10"))
                                                size=10;
                                        else if(n3.equals("50"))
                                                size=50;
                                        else if(n3.equals("100"))
                                                size=100;
                                        repaint();
                                }
                        });
                        this.add(c1); this.add(c2); this.add(c3);
                }
                public void paint(Graphics g)
                {
                        g.setFont(new Font("arial",type,size));
                        g.setColor(color);
                        g.drawString("Hello Java",400,400);
                }
                public static void main(String args[])
                {
                        Three t=new Three();
                }
}
```

Output:

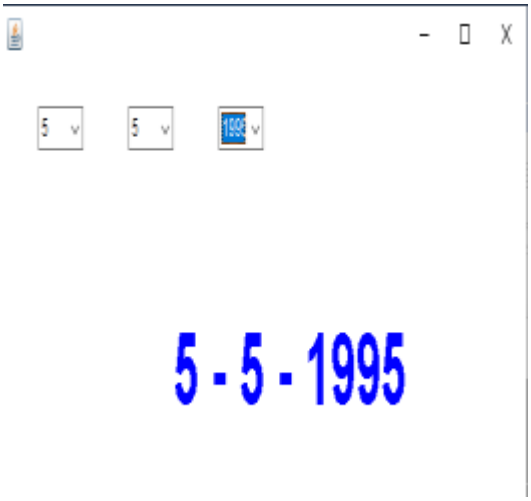
4. Write a small application with a default date 01/01/2000 and three combo boxes displaying valid days, months & year (1990 – 2050). Change the displayed date with the one chosen by user from these combo boxes.

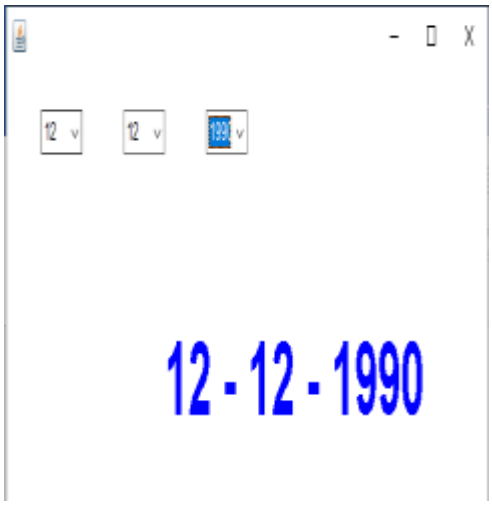
Source Code:

```
import java.awt.*;
import java.awt.event.ItemListener;
import java.awt.event.ItemEvent;

class Four extends Frame
{
        Choice c1,c2,c3,c4;
        String day="1",month="1",year="2020";
        //label n1,n2,n3;
        Four()
```

```
	{
		this.setVisible(true);
		this.setSize(600,600);
		c1=new Choice();
		c2=new Choice();
		c3=new Choice();
		c4=new Choice();
		for(int i=1;i<=31;i++)
			c1.add(String.valueOf(i));
		for(int i=1;i<=12;i++)
			c2.add(String.valueOf(i));
		for(int i=1990;i<=2050;i++)
			c3.add(String.valueOf(i));
		c4.add("10");
		c1.setBounds(50,50,50,50);
		c2.setBounds(150,50,50,50);
		c3.setBounds(250,50,50,50);
		c4.setBounds(350,50,50,50);
		c4.setVisible(false);
		this.add(c1); this.add(c2); this.add(c3); this.add(c4);


		c1.addItemListener(new ItemListener() {
			public void itemStateChanged(ItemEvent e){
				day=e.getItem().toString();
			repaint();
			}
		});
		c2.addItemListener(new ItemListener() {
			public void itemStateChanged(ItemEvent e){
				month=e.getItem().toString();
			repaint();
			}
		});
		c3.addItemListener(new ItemListener() {
			public void itemStateChanged(ItemEvent e){
				year=e.getItem().toString();
			repaint();
			}
		});

	}
	public void paint(Graphics g)
	{
		g.setFont(new Font("arial",Font.BOLD,50));
		g.setColor(Color.blue);
		g.drawString(day+" - " +month+" - "+year,200,200);
	}
	public static void main(String args[])
	{
		Four f=new Four();
	}
}
```

Output:





5. Create a GUI with title STUDENT which has labels roll no., name, course, gender, class, address with textboxes for taking input from the user(without any functionality) and checkboxes for selecting the course, radio buttons for selecting gender with appropriate background color

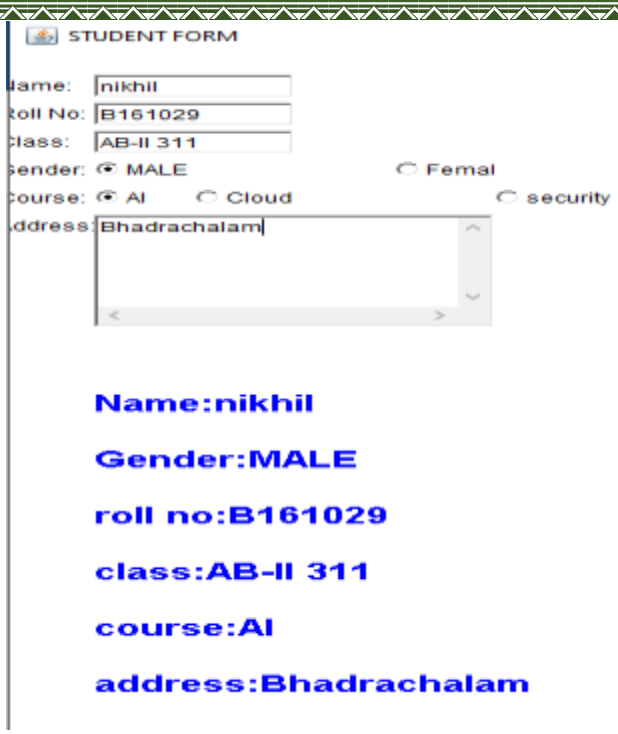
Source Code:

```
import java.awt.*;
import java.awt.event.*;
class SS extends Frame implements ItemListener, ActionListener
{
        Label name,rno,course,gender,cls,address,dummy1;
        String sname="",srno="",scourse="",sgender="",scls="",saddress="";
        TextField tname,trno,tcls;
        Checkbox c1,c2,c3,cm,cf;
        CheckboxGroup cg,cc;
        TextArea taddress;
        SS()
        {
                this.setSize(1000,1000);
                this.setVisible(true);
                this.setTitle("STUDENT FORM");
                this.setBackground(Color.white);

                name=new Label("Name:");
                rno=new Label("Roll No:");
                course=new Label("Course:");
                gender=new Label("Gender:");
                cls=new Label("Class:");
                address=new Label("Address:");
                dummy1=new Label("dummy:");
                dummy1.setVisible(false);

                tname=new TextField(20);
                trno=new TextField(20);
                tcls=new TextField(20);
```

```
		cg=new CheckboxGroup();
		cm=new Checkbox("MALE",cg,false);
		cf=new Checkbox("Female",cg,false);

		cc=new CheckboxGroup();
		c1=new Checkbox("AI",cc,false);
		c2=new Checkbox("Cloud",cc,false);
		c3=new Checkbox("security",cc,false);

		taddress=new TextArea(3,20);

		name.setBounds(0,50,50,20);
		tname.setBounds(50,50,100,20);
		rno.setBounds(0,75,50,20);
		trno.setBounds(50,75,100,20);
		cls.setBounds(0,100,50,20);
		tcls.setBounds(50,100,100,20);
		gender.setBounds(0,125,50,20);
		cm.setBounds(50,125,100,20);
		cf.setBounds(200,125,50,20);
		course.setBounds(0,150,50,20);
		c1.setBounds(50,150,50,20);
		c2.setBounds(100,150,100,20);
		c3.setBounds(250,150,100,20);
		address.setBounds(0,175,50,20);
		taddress.setBounds(50,175,200,100);

		tname.addActionListener(this);
		trno.addActionListener(this);
		tcls.addActionListener(this);
		c1.addItemListener(this);
		c2.addItemListener(this);
		c3.addItemListener(this);
		cm.addItemListener(this);
		cf.addItemListener(this);
		taddress.addTextListener(new TextListener(){
			public void textValueChanged(TextEvent e){
				saddress=taddress.getText();
				repaint();
			}
		});

		this.add(name); this.add(tname);
		this.add(rno);    this.add(trno);
		this.add(cls); this.add(tcls);
		this.add(gender); this.add(cm); this.add(cf);
		this.add(course); this.add(c1); this.add(c2); this.add(c3);
		this.add(address); this.add(taddress);
		this.add(dummy1);
	}
	public void paint(Graphics g)
	{
		Font f=new Font("arial",Font.BOLD,20);
```

```
                g.setFont(f);
                g.setColor(Color.blue);
                g.drawString("Name:"+sname,50,350);
                g.drawString("Gender:"+sgender,50,400);
                g.drawString("roll no:"+srno,50,450);
                g.drawString("class:"+scls,50,500);
                g.drawString("course:"+scourse,50,550);
                g.drawString("address:"+saddress,50,600);
        }
        public void actionPerformed(ActionEvent e)
        {

                sname=tname.getText();
                srno=trno.getText();
                scls=tcls.getText();
                repaint();
        }
        public void itemStateChanged(ItemEvent e){
                if(c1.getState()==true)
                        scourse=c1.getLabel();
                else if(c2.getState()==true)
                        scourse=c2.getLabel();
                else if(c3.getState()==true)
                        scourse=c3.getLabel();
                if(cm.getState()==true)
                        sgender=cm.getLabel();
                else if(cf.getState()==true)
                        sgender=cf.getLabel();
                repaint();
        }
}
class Five
{
        public static void main(String args[])
        {
                SS s= new SS();
        }
}
```

Output:



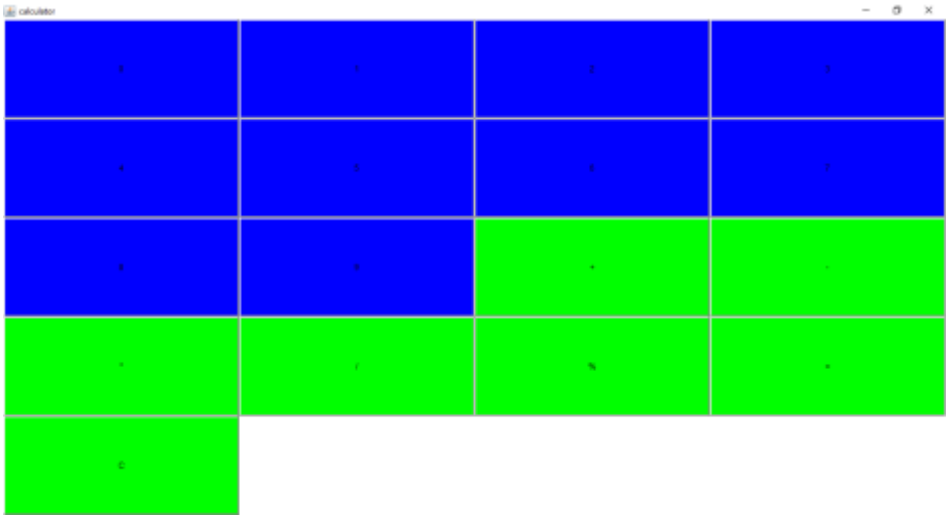
6. Create a GUI application to display a calculator using grid Layout (You do not have to provide functionality).

Source Code:

```
import java.awt.*;

class Six extends Frame
{
	Button b[]=new Button[10];
	Button o[]=new Button[7];
	String op[]=new String[]{"+","-","*","/","%","=","C"};
	Six()
	{
		this.setVisible(true);
		this.setSize(600,600);
		this.setLayout(new GridLayout(5,4));
		this.setTitle("calculator");
		for(int i=0;i<10;i++)
		{
			b[i]=new Button(""+i);
			b[i].setBackground(Color.blue);
			this.add(b[i]);
		}
		for(int i=0;i<=6;i++)
		{
			o[i]=new Button(op[i]);
			o[i].setBackground(Color.green);
			this.add(o[i]);
		}

	}
	public static void main(String args[])
	{
```

```
                    Six s=new Six();
                }
}
```

Output:



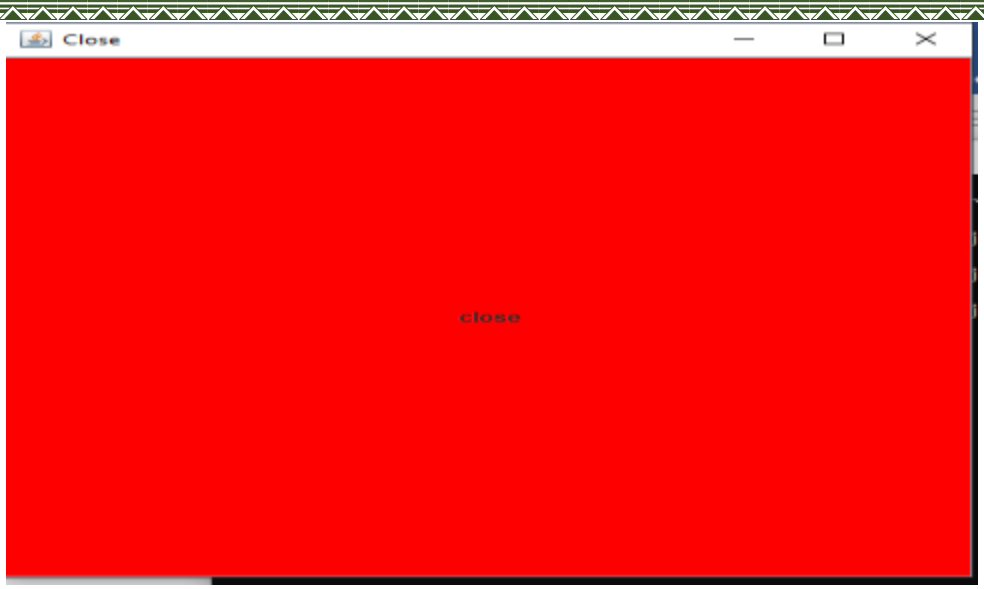
Week-11

1. Write a program to create a frame by creating an object to Jframe class and include close button to terminate the application of the frame.

Source Code:

```
import java.awt.*;
import javax.swing.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

class First extends JFrame implements ActionListener
{
	JButton b;
	public First()
	{
		this.setVisible(true);
		this.setSize(500,500);
		this.setTitle("Close ");
		b=new JButton("close");
		b.setBackground(Color.red);
		b.setBounds(200,200,100,100);
		b.addActionListener(this);
		this.add(b);
	}
	public void actionPerformed(ActionEvent e)
	{
		String button=e.getActionCommand();
		if(button.equals("close"))
			System.exit(0);
	}
}
class One
{
	public static void main(String args[])
	{
		First f=new First();
	}
}
```

Output:

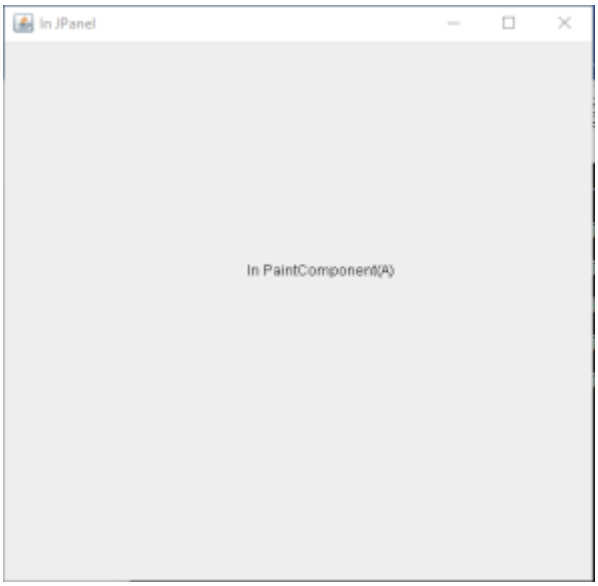


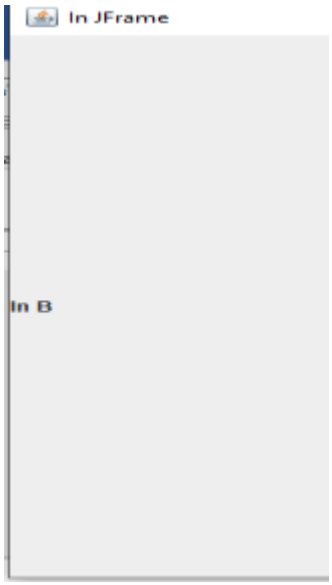
2. Write program for the following.
   a. Display text in the frame by overriding PaintComponent() method of Jpanel class.
   b. Display some text in the frame with the help of a Label.

Source Code:

```
import java.awt.*;
import javax.swing.*;
class A extends JPanel
{
	A()
	{
		JFrame f=new JFrame();
		f.setVisible(true);
		f.setSize(500,500);
		f.setTitle("In JPanel");
		f.add(this);
	}
	protected void paintComponent(Graphics g)
	{
		g.drawString("In PaintComponent(A)",200,200);
	}
}
class B extends JFrame
{
	B()
	{
		this.setVisible(true);
		this.setTitle("In JFrame");
		this.setSize(500,500);
		JLabel label=new JLabel();
		label.setBounds(200,200,200,100);
		label.setText("In B");
		this.add(label);
	}
}
class Two
```

```
	{
		public static void main(String args[])
		{
			A a=new A();
			B b=new B();
		}
}
```





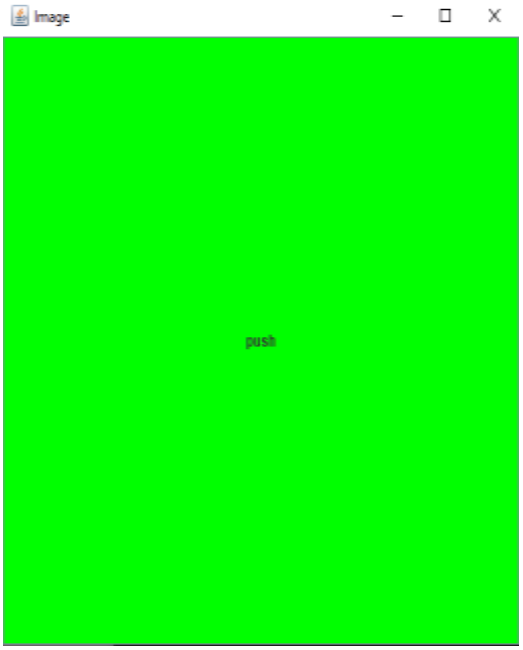
3. Write a program to create a push button , when the button is clicked an image is displayed in the frame

Source code:

```
	/*
	Write a program to create a push button , when the button is clicked an image is
	displayed in the frame
	*/

	import java.awt.*;
	import javax.swing.*;
	import java.awt.event.ActionEvent;
	import java.awt.event.ActionListener;

	class Third extends JFrame implements ActionListener
	{
		JButton jb;
		ImageIcon ii;
		JLabel jl;
		Third()
		{
			this.setVisible(true);
			this.setTitle("Image");
			this.setSize(500,500);
			jb=new JButton("push");
			this.setBackground(Color.red);
			jb.setBackground(Color.green);
```

```
                        jb.setBounds(100,100,100,100);
                        jb.addActionListener(this);
                        this.add(jb);
                }
                public void actionPerformed(ActionEvent e)
                {
                        String s=e.getActionCommand();
                        if(s.equals("push"))
                        {
                                jb.setVisible(false);
                                ii=new ImageIcon("/Users/nikhil/Desktop/e3s1/oops/Labs/Week-
        11/img.jpeg");
                                jl=new JLabel(ii);
                                this.setSize(ii.getIconWidth()+100,ii.getIconHeight()+100);
                                jl.setBounds(0,0,ii.getIconWidth(),ii.getIconHeight());
                                this.add(jl);
                        }
                }
        }
        class Three
        {
                public static void main(String args[])
                {
                        new Third();
                }
}
```

Output:



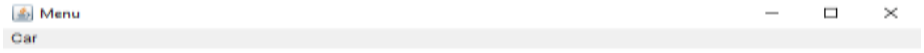
4. Write a program to create a menu with several menu items

Source Code:

```
/*
 Write a program to create a menu with several menu items
*/

import java.awt.*;
import javax.swing.*;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;

class Fourth extends Frame implements ActionListener
{
	String item="";
	Fourth()
	{
		this.setVisible(true);
		this.setSize(500,500);
		this.setTitle("Menu");

		MenuBar mb=new MenuBar();
		this.setMenuBar(mb);

		Menu m=new Menu("Car");
		mb.add(m);

		MenuItem m1=new MenuItem("Tata");
		MenuItem m2=new MenuItem("Toyota");
		MenuItem m3=new MenuItem("Hundai");
		MenuItem m4=new MenuItem("Ford");

		m1.addActionListener(this);
		m2.addActionListener(this);
		m3.addActionListener(this);
		m4.addActionListener(this);

		m.add(m1); m.add(m2); m.add(m3); m.add(m4);
	}
	public void actionPerformed(ActionEvent e)
	{
		item=e.getActionCommand();
		repaint();
	}
	public void paint(Graphics g)
	{
		Font f=new Font("arial",Font.BOLD,50);
		g.setFont(f);
		g.drawString("Selected Car is :"+ item,100,400);
	}
}
class Four
{
	public static void main(String args[])
	{
		Fourth f=new Fourth();
```

```
                }
}
```

Output:



Selected Car is :Ford

5. Create an application Form for University Enroll ment with the following Fields. a. Check box b. Text area c. List box d. Display text e. Push buttons f. Combo box. g. Radio buttons. h. Back ground color

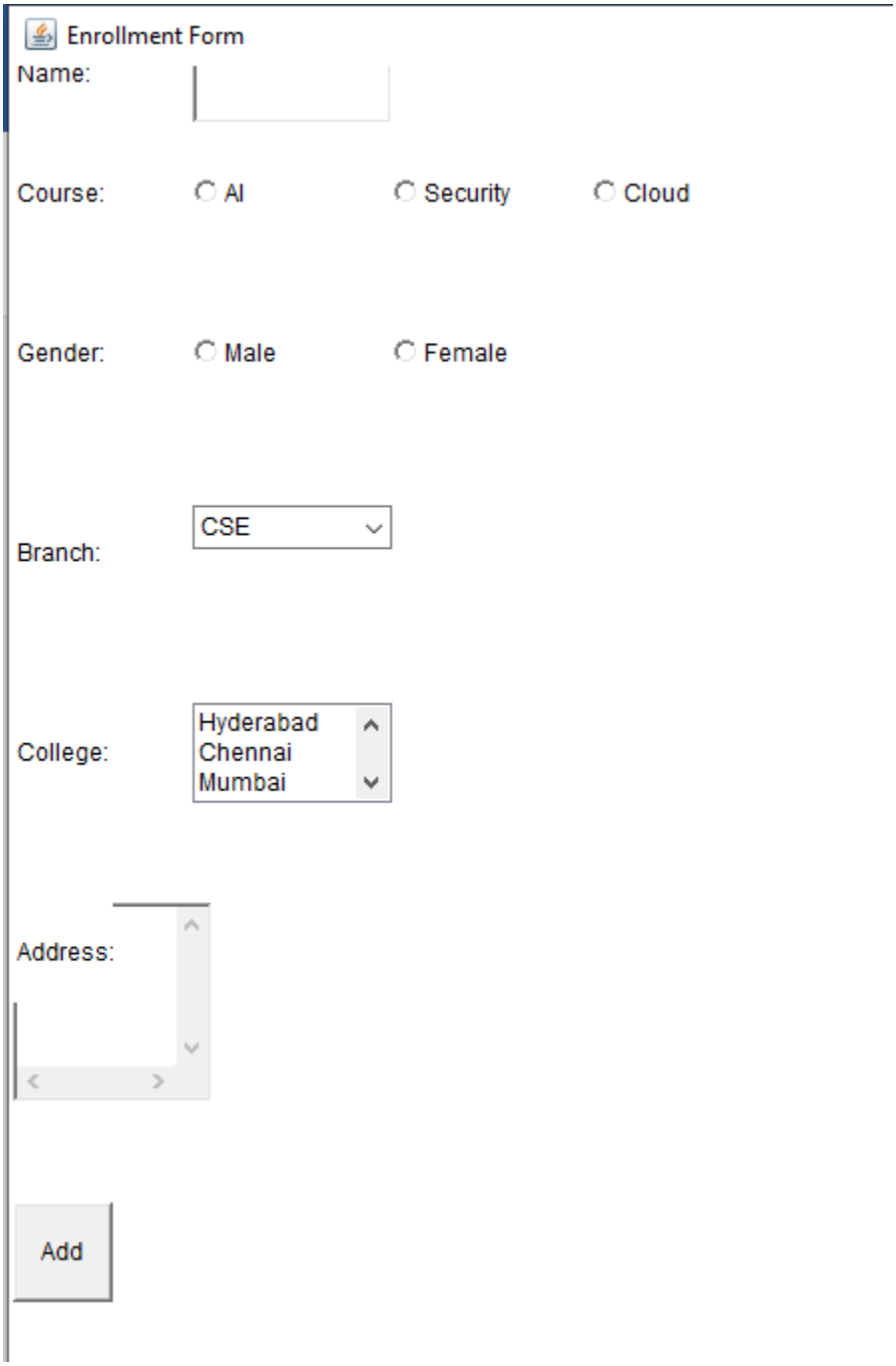
Source Code:

```
/*
Create an application Form for University Enroll ment with the following Fields.
a. Check box b. Text area c. List box d. Display text e. Push buttons f. Combo
box. g. Radio buttons. h. Back ground color
*/

import java.awt.*;

class Fifth extends Frame
{
        Label nL,cL,aL,gL,bL,clL;
        TextField tname;
        TextArea taddress;
        Checkbox c1,c2,c3,c4,c5;
        CheckboxGroup cg,cg1;
        Button b1;
        Choice ch1;
        List l1;
        Fifth()
        {
                this.setVisible(true);
                this.setSize(700,700);
                this.setTitle("Enrollment Form");
                nL=new Label("Name:");
                cL=new Label("Course:");
                aL=new Label("Address:");
                gL=new Label("Gender:");
                bL=new Label("Branch:");
                clL=new Label("College:");
```

```
                tname=new TextField(20);
                taddress=new TextArea(3,20);

                cg=new CheckboxGroup();
                c1=new Checkbox("AI",cg,false);
                c2=new Checkbox("Security",cg,false);
                c3=new Checkbox("Cloud",cg,false);

                cg1=new CheckboxGroup();
                c4=new Checkbox("Male",cg1,false);
                c5=new Checkbox("Female",cg1,false);

                b1=new Button("Add");

                ch1=new Choice();
                ch1.add("CSE");
                ch1.add("ECE");
                ch1.add("EEE");
                ch1.add("Mech");

                l1=new List(4);
                l1.add("Hyderabad");
                l1.add("Chennai");
                l1.add("Mumbai");
                l1.add("Kharagpur");
                l1.add("Khanpur");
                nL.setBounds(10,10,50,50);
                tname.setBounds(100,10,100,50);
                cL.setBounds(10,70,50,50);
                c1.setBounds(100,70, 80,50);
                c2.setBounds(200, 70, 80,50);
                c3.setBounds(300,70,80,50);
                gL.setBounds(10,150,50,50);
                c4.setBounds(100,150,70,50);
                c5.setBounds(200,150,70,50);
                bL.setBounds(10,250,50,50);
                ch1.setBounds(100,250,100,50);
                clL.setBounds(10,350,50,50);
                l1.setBounds(100,350,100,50);
                aL.setBounds(10,450,50,50);
                taddress.setBounds(10,450,100,100);
                b1.setBounds(10,600,50,50);
                this.add(nL);     this.add(tname);
                this.add(cL);     this.add(c1); this.add(c2); this.add(c3);
                this.add(gL);     this.add(c4); this.add(c5);
                this.add(bL);     this.add(ch1);
                this.add(clL);    this.add(l1);
                this.add(aL);      this.add(taddress);
                this.add(b1);
        }
}
```

```
        class Five
        {
                public static void main(String args[])
                {
                        Fifth f=new Fifth();
                }
}
```

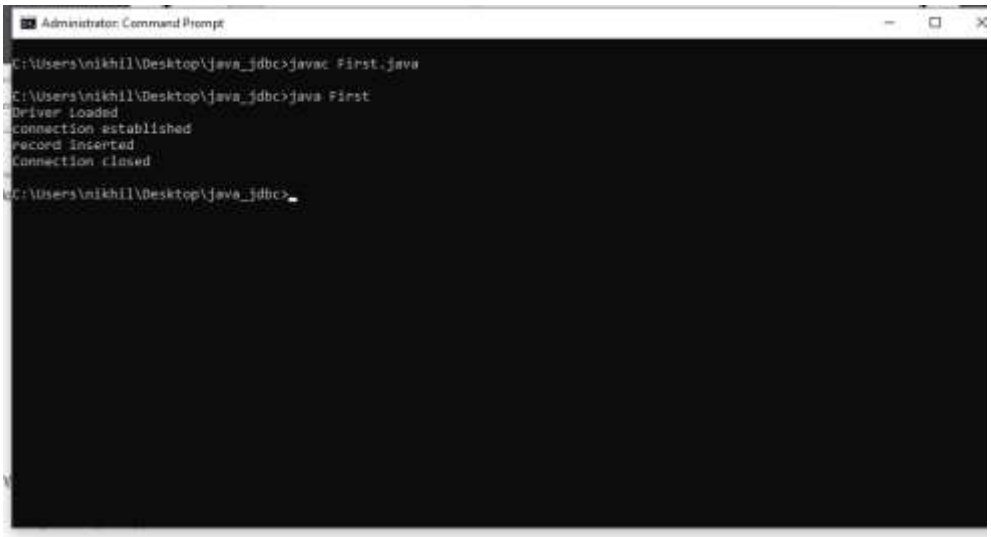
Output:

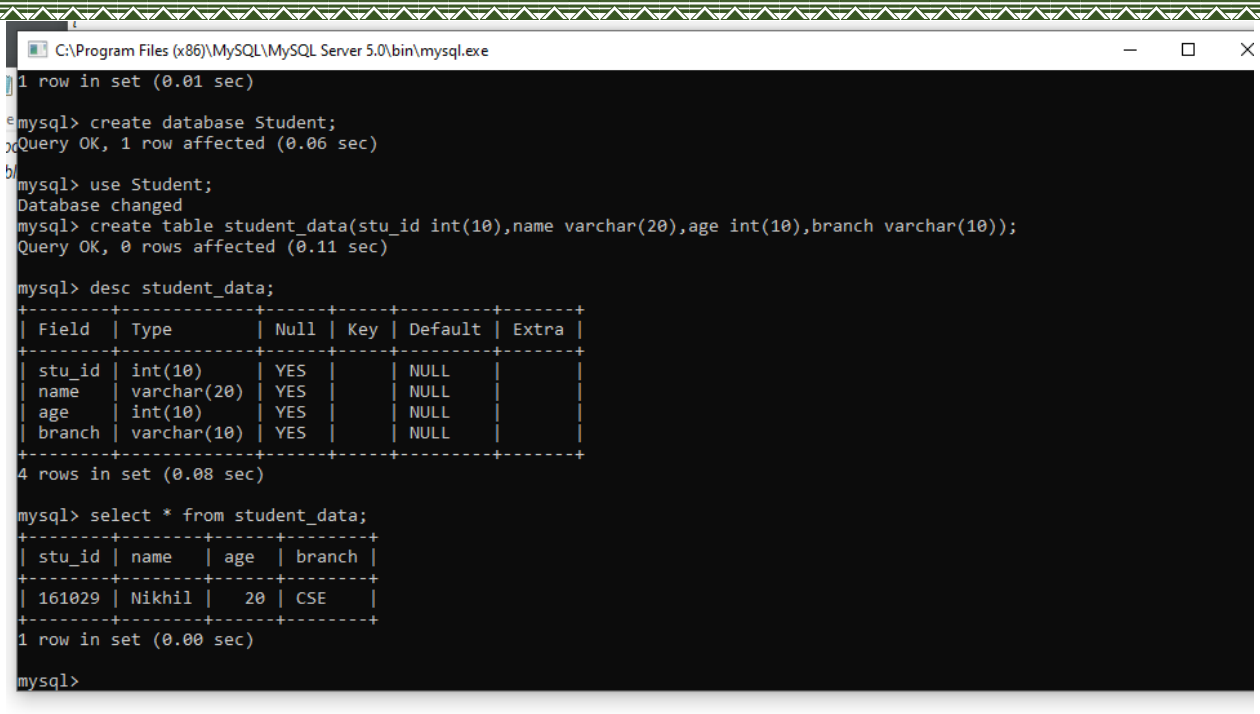


Week-12

1. Write a program to insert data into student table.

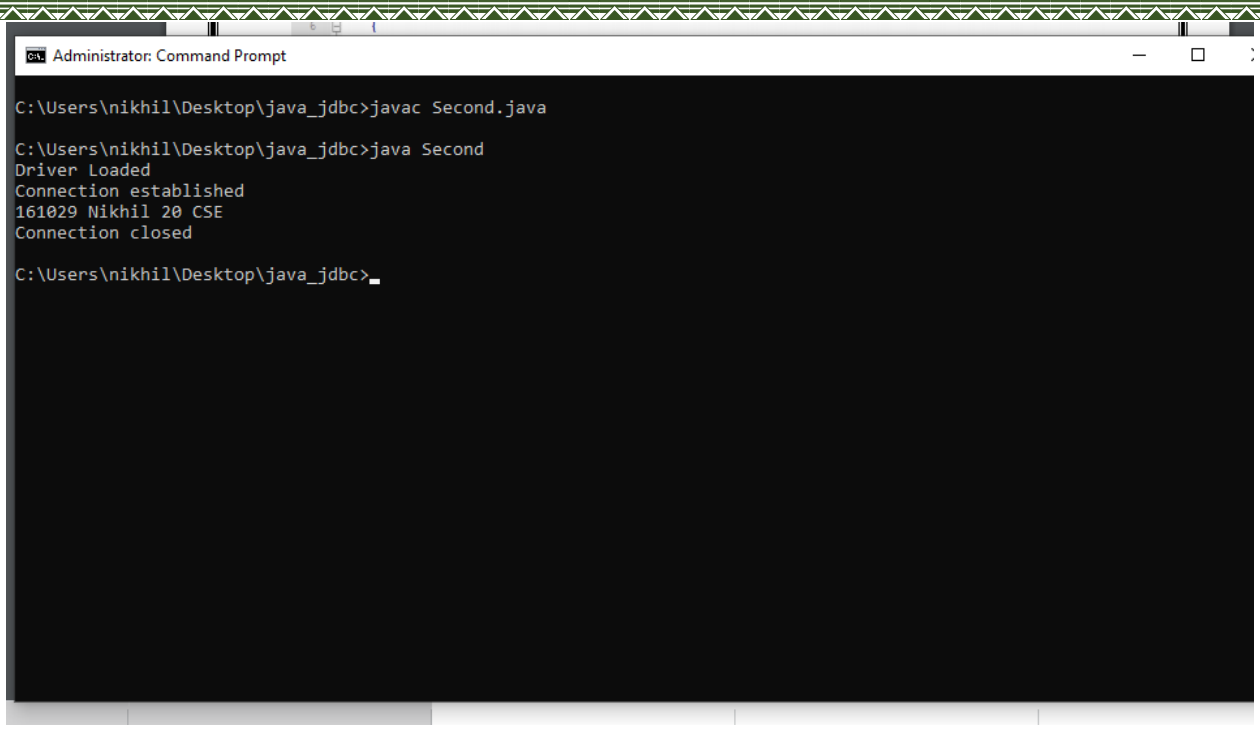
```
2.  import java.sql.*;
3.  public class First
4.  {
5.          public static void main(String args[])
6.          {
7.                  try
8.                  {
9.                          Class.forName("com.mysql.jdbc.Driver");
10.                         System.out.println("Driver Loaded");
11.                         Connection
    con=DriverManager.getConnection("jdbc:mysql://localhost:3306/Student","root","rgukt123");
12.                         System.out.println("connection established");
13.                         Statement st=con.createStatement();
14.                         st.executeUpdate("insert into student_data values(161029,'Nikhil',20,'CSE');");
15.                         System.out.println("record inserted");
16.                         con.close();
17.                         System.out.println("Connection closed");
18.                 }
19.                 catch(Exception e)
20.                 {
21.                         System.out.println(e);
22.                 }
23.         }
24. }
25.
```





2. Write a program to retrieve the data from the table Student.

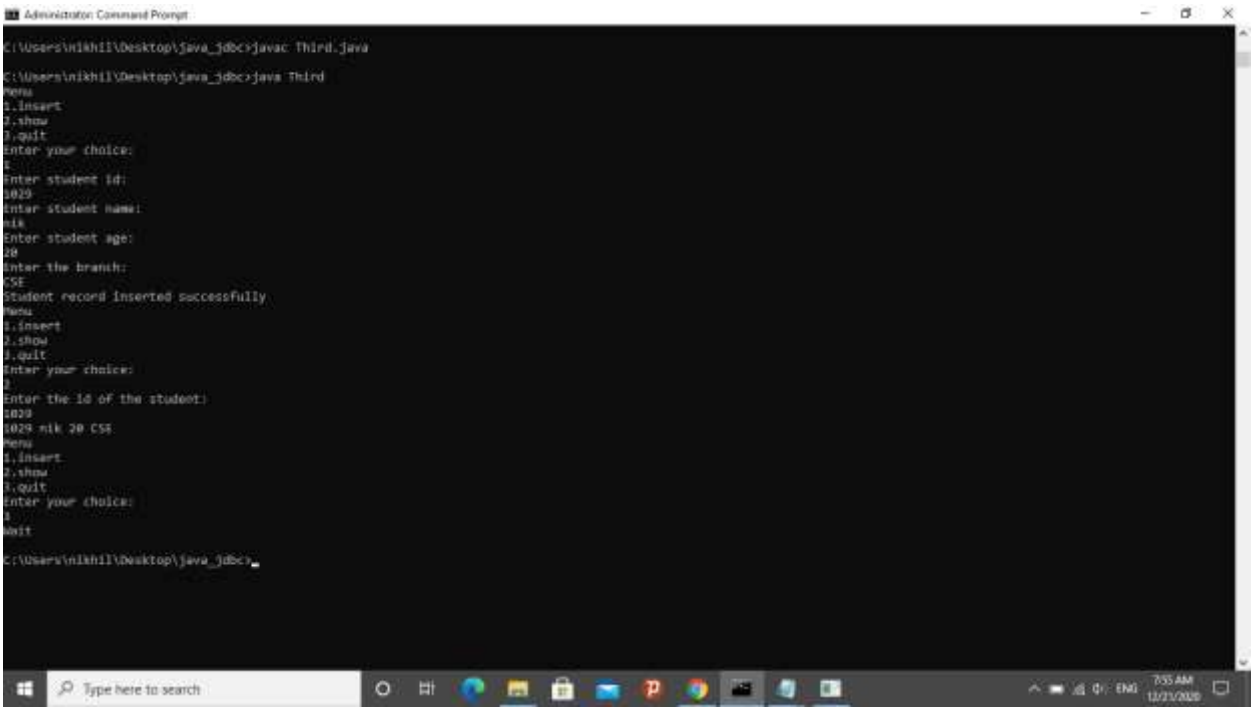
```
import java.sql.*;
public class Second
{
	public static void main(String args[])
	{
		try
		{
			Class.forName("com.mysql.jdbc.Driver");
			System.out.println("Driver Loaded");
			Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/Student","root","rgukt123");
			System.out.println("Connection established");
			Statement st=con.createStatement();
			ResultSet rs=st.executeQuery("select * from student_data;");
			while(rs.next())
			{
				System.out.println(rs.getInt(1)+" "+rs.getString(2)+" "+rs.getInt(3)+" "+rs.getString(4));
			}
			con.close();
			System.out.println("Connection closed");
		}
		catch(Exception e)
		{
			System.out.println(e);
		}
	}
}
```

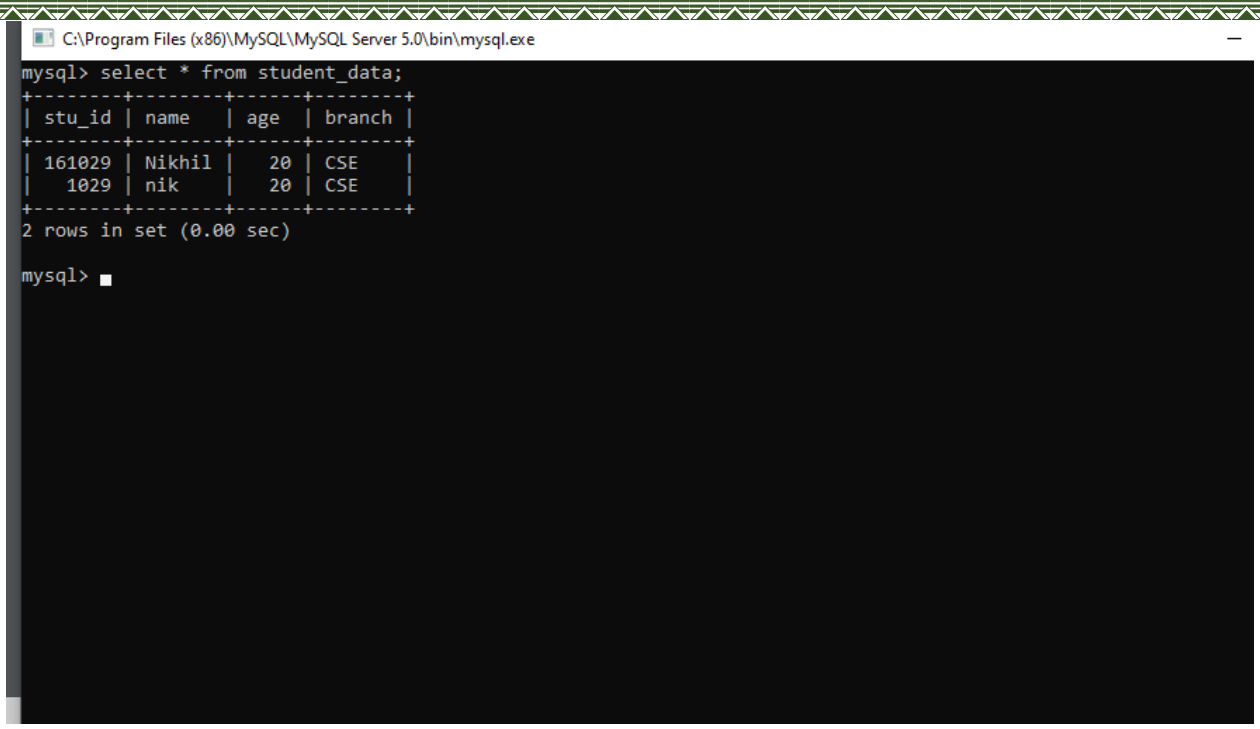


3. Create a Form to insert and retrieve the data from Database as user prefer.

```
import java.sql.*;
import java.util.Scanner;
public class Third
{
	public static void main(String args[])
	{
		try
		{
			Scanner sc= new Scanner(System.in);
			Class.forName("com.mysql.jdbc.Driver");
			Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/Student","root","rgukt123");
			Statement st=con.createStatement();
			int choice=0,id,age;
			String name,branch;
			while(choice!=3)
			{
				System.out.println("Menu");
				System.out.println("1.insert");
				System.out.println("2.show");
				System.out.println("3.quit");
				System.out.println("Enter your choice:");
				choice=sc.nextInt();
				switch(choice)
				{
					case 1:	System.out.println("Enter student id:");
						id=sc.nextInt();
						System.out.println("Enter student name:");
						name=sc.next();
						System.out.println("Enter student age:");
						age=sc.nextInt();
						System.out.println("Enter the branch:");
						branch=sc.next();
						String query="insert into student_data (stu_id,name,age,branch)
"+"values(?,?,?,?)";
						PreparedStatement ps=con.prepareStatement(query);
						ps.setInt(1,id);
						ps.setString(2,name);
						ps.setInt(3,age);
						ps.setString(4,branch);
						ps.execute();
						System.out.println("Student record inserted successfully");
						break;
```

```
                                                case 2:     System.out.println("Enter the id of the student:");
                                                            id=sc.nextInt();
                                                            ResultSet rs=st.executeQuery("select * from student_data where
stu_id="+id+";");
                                                            while(rs.next())
                                                            {
                                                                        System.out.println(rs.getInt(1)+" "+rs.getString(2)+"
"+rs.getInt(3)+" "+rs.getString(4));
                                                            }
                                                            break;
                                                default:    System.out.println("Wait");
                                                }
                                    }
                                    con.close();
                        }
                        catch(Exception e)
                        {
                                    System.out.println(e);
                        }
            }}
```





4. Write a program to store an Image and retrieve an image from Database.

```
import javax.swing.*;
import java.awt.*;
import java.io.IOException;
import java.sql.*;
import java.sql.Connection;
public class Image extends Canvas {
        String path;
        public void paint(Graphics g) {
                Toolkit t=Toolkit.getDefaultToolkit();
                java.awt.Image i=t.getImage(path);
                g.drawImage(i, 120,100,this);
        }
        public void display(Image im) {
                Frame f=new JFrame();
                f.add(im);
                f.setSize(400,400);
                f.setVisible(true);
        }
        public static void main(String[] args) throws ClassNotFoundException, SQLException,
IOException {
                Image im=new Image();

                // load the driver
                Class.forName("com.mysql.jdbc.Driver");
                System.out.println("**** Loaded the JDBC driver");
                // Create the connection
```

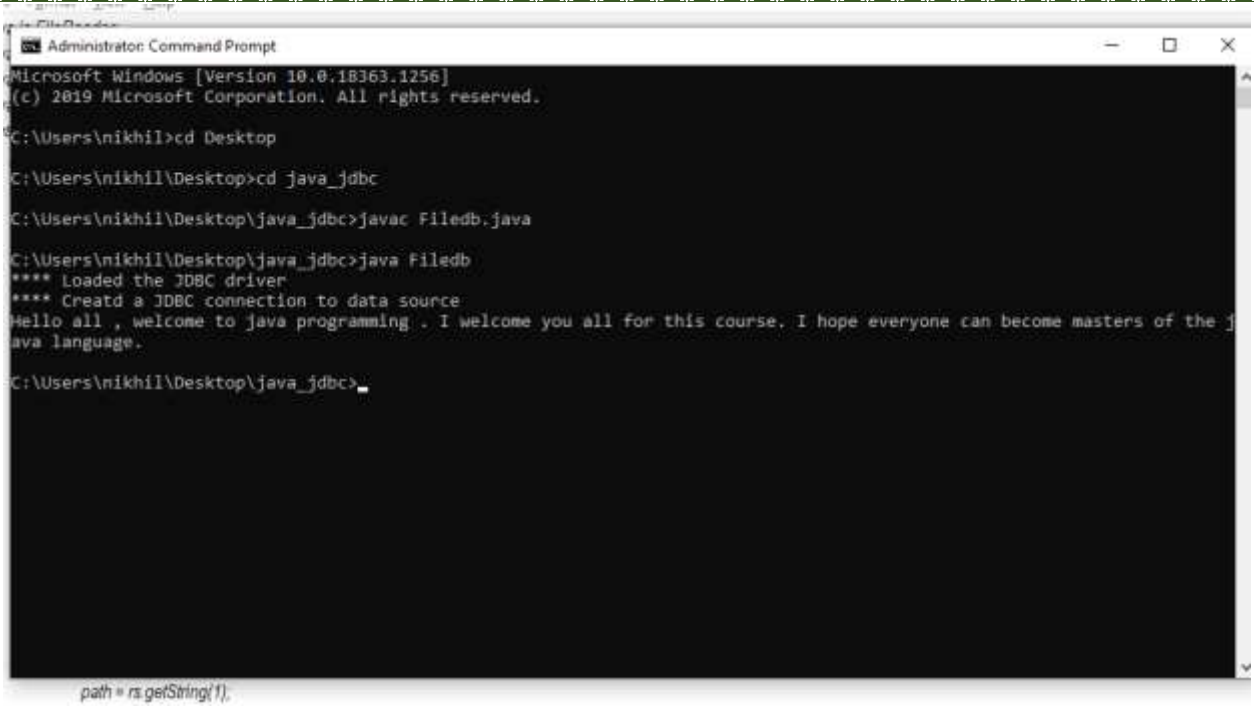
```
                Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/JDBC","root","rgukt123");
                System.out.println("**** Creatd a JDBC connection to data source");
                // create table
                Statement stmt = con.createStatement();
                //stmt.executeUpdate("CREATE TABLE Image " + "(id INTEGER not NULL," +"
image VARBINARY(10000) NOT NULL," +" PRIMARY KEY ( id ))");
                // Insert Image
                PreparedStatement pstmt = con.prepareStatement("INSERT INTO Image(id,
image) VALUES(?,?)");
                pstmt.setInt(1,3); // id
                pstmt.setString(2, "facebook.png");
                // execute
                pstmt.execute();
                //Retrieve image
                ResultSet rs = stmt.executeQuery("SELECT image From Image where id = 1");
                while (rs.next())
                im.path = rs.getString(1).toString();
                // diplay image
                im.display(im);
                // close the connection
                con.close();
                System.out.println("**** Closed JDBC");
        }
}
```

```
        at com.mysql.jdbc.MysqlIO.sendCommand(Mysq
        at com.mysql.jdbc.MysqlIO.sqlQueryDirect(M
        at com.mysql.jdbc.Connection.execSQL(Conne
        at com.mysql.jdbc.PreparedStatement.execut
        at com.mysql.jdbc.PreparedStatement.execut
        at Image.main(Image.java:36)

C:\Users\nikhil\Desktop\java_jdbc>javac Image.java

C:\Users\nikhil\Desktop\java_jdbc>java Image
**** Loaded the JDBC driver
**** Creatd a JDBC connection to data source
**** Closed JDBC
```

5. write a program to store and retrieve file content from the Database

```
import java.io.BufferedReader;
import java.io.FileNotFoundException;
import java.io.FileReader;
import java.io.IOException;
import java.sql.*;
import java.sql.Connection;
```

```
public class Filedb {
	public static void main(String[] args) throws ClassNotFoundException, SQLException,
IOException {
		Filedb f=new Filedb();
		// load the driver
		Class.forName("com.mysql.jdbc.Driver");
		System.out.println("**** Loaded the JDBC driver");
		// Create the connection
		Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/JDBC","root","rgukt123");
		System.out.println("**** Creatd a JDBC connection to data source");
		// create table
		Statement stmt = con.createStatement();
		stmt.executeUpdate("CREATE TABLE File " +"(id INTEGER not NULL," +" file
VARBINARY(10000) NOT NULL," +" PRIMARY KEY ( id ))");

		PreparedStatement pstmt = con.prepareStatement("INSERT INTO File(id, file)
VALUES(?,?)");
		pstmt.setInt(1,1); // id
		pstmt.setString(2, "sample.txt");
		// execute
		pstmt.execute();
		//Retrieve image
		ResultSet rs = stmt.executeQuery("SELECT file From File where id = 1");
		String path = null;
		while (rs.next())
		path = rs.getString(1);
		// display file
		BufferedReader in = new BufferedReader(new FileReader(path));
		String line;
		while((line = in.readLine()) != null)
		{
			System.out.println(line);
		}
		in.close();
		// close the connection
		con.close();
	}
}
```



path = rs.getString(1);