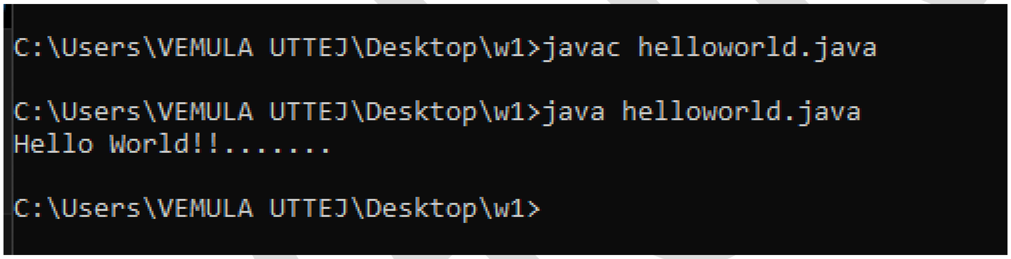
WEEK-1

1. Write a Java program print “Hello World”

PROGRAM:

```
import java.io.*;
public class Helloworld  //class Helloworld
{
   public static void main(String args[])
   {
           System.out.println("Hello World!!......................");
   }
}
```

Output:



```
C:\Users\VEMULA UTTEJ\Desktop\w1>javac helloworld.java

C:\Users\VEMULA UTTEJ\Desktop\w1>java helloworld.java
Hello World!!.......

C:\Users\VEMULA UTTEJ\Desktop\w1>
```

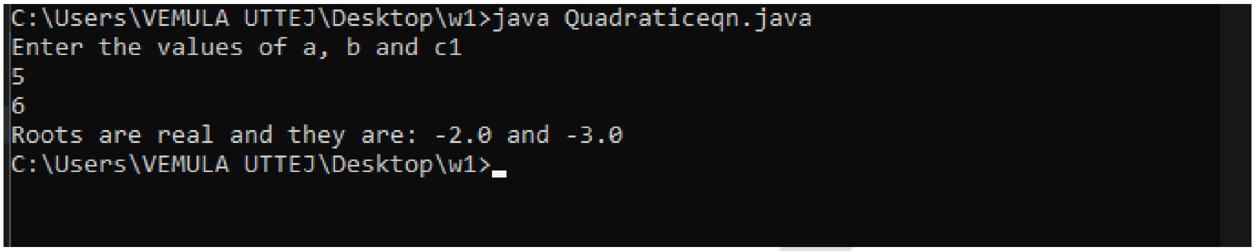
2. Write a Java program that prints all real and imaginary solutions to the quadratic equation ax2 + bx +c=0. Read in a, b, c and use the quadratic formula

Program :

```
import java.io.*;

import java.util.Scanner;

class Quadraticeqn

{

        public static void main(String args[])

        {

                Scanner in=new Scanner(System.in);int

                a,b,c,d;

                double r1,r2;

                System.out.print("Enter the values of a, b and c");

                a=in.nextInt();

                b=in.nextInt();

                c=in.nextInt();

                 d=((b*b)-4*a*c);

                if(d>0)

                {
```

```
    r1=((-b+Math.sqrt(d))/(2*a));

    r2=((-b-Math.sqrt(d))/(2*a));

    System.out.print("Roots are real and they are: "+r1+"

        and "+r2);


    }

    else if(d<0)

    {

        r1=((-b)/(2*a));  //real part

        r2=((Math.sqrt(-d))/(2*a)); //imaginary part

         System.out.print("Roots are imaginary and they

        are:"+r1+"+i"+r2+" and "+r1+"-i"+r2);

        }

    else

    {

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

        System.out.print("Roots are real & distinct and they are: "+r1+" and
        "+r1);

}
```

OUTPUT:



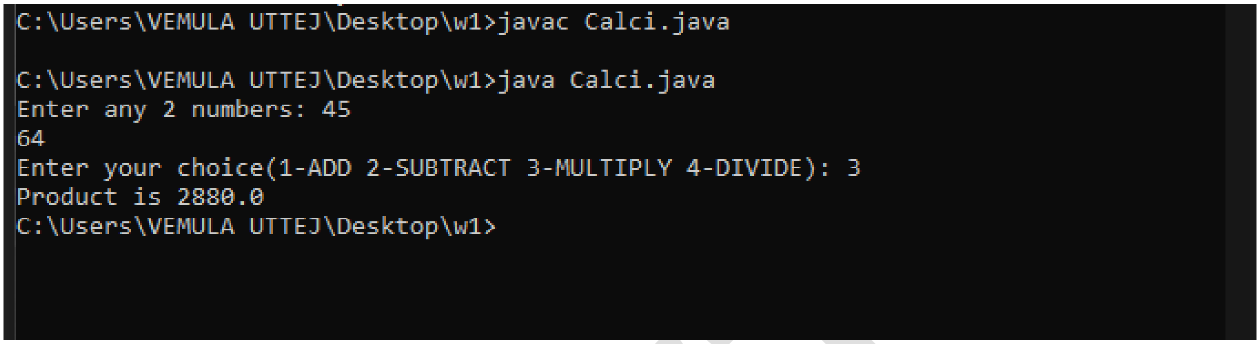
```
C:\Users\VEMULA UTTEJ\Desktop\w1>java Quadraticeqn.java
Enter the values of a, b and c1
5
6
Roots are real and they are: -2.0 and -3.0
C:\Users\VEMULA UTTEJ\Desktop\w1>
```

3. Write a Java program to implement calculatoroperations

PROGRAM :

```
import java.io.*;
import java.util.Scanner;
class Calci
{
    public static void main(String args[])
    {
            float a, b;
            int c;
            Scanner in=new Scanner(System.in);
            System.out.print("Enter any 2 numbers: ");
            a=in.nextInt();
            b=in.nextInt();
            System.out.print("Enter your choice(1-ADD 2-SUBTRACT 3-MULTIPLY
4-DIVIDE): ");
            c=in.nextInt();
            switch(c)
            {
                    case 1:
                    {
                            System.out.print("Sum is "+(a+b));
                            break;
                    }
                    case 2:
                    {
                            System.out.print("Difference is "+(a-b));
                            break;
                    }
                    case 3:
                    {
                            System.out.print("Product is "+(a*b));
                            break;
                    }
                    case 4:
                    {
                            System.out.print("Quotient is "+(a/b));
                            break;
                    }
                    default:
                            System.out.print("OOPS! invalid choice");
            }
    }
}
```

OUTPUT :



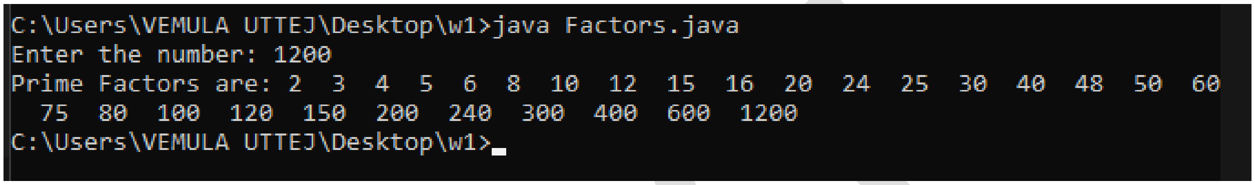
```
C:\Users\VEMULA UTTEJ\Desktop\w1>javac Calci.java

C:\Users\VEMULA UTTEJ\Desktop\w1>java Calci.java
Enter any 2 numbers: 45
64
Enter your choice(1-ADD 2-SUBTRACT 3-MULTIPLY 4-DIVIDE): 3
Product is 2880.0
C:\Users\VEMULA UTTEJ\Desktop\w1>
```

4. Write a java program to find prime factors of givennumber

PROGRAM :

```
import java.io.*;
import java.util.Scanner;
class Factors
{
    public static void main(String args[])
    {
            int a;
            System.out.print("Enter the number: ");
            Scanner in=new Scanner(System.in);
             a=in.nextInt();
            System.out.print("Prime Factors are: ");
            for(int i=2;i<=a;i++)
            {
                    if((a%i)==0)
                            System.out.print(i+"  ");
            }
    }
}
```

OUTPUT :



```
C:\Users\VEMULA UTTEJ\Desktop\w1>java Factors.java
Enter the number: 1200
Prime Factors are: 2  3  4  5  6  8  10  12  15  16  20  24  25  30  40  48  50  60
  75  80  100  120  150  200  240  300  400  600  1200
C:\Users\VEMULA UTTEJ\Desktop\w1>
```

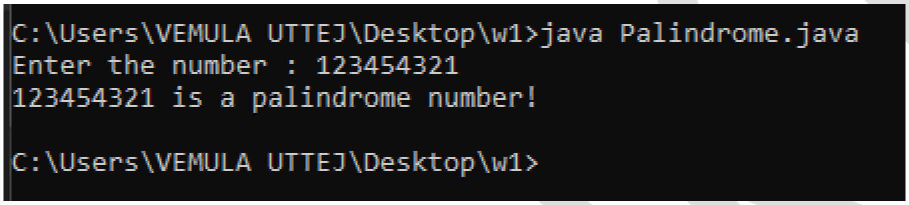
5. Write a java program to find whether given number isPalindrome or not

PROGRAM :

```
import java.io.*;
import java.util.*;
class Palindrome
{
    public static void main(String args[])
    {
            int num, temp, rem=0, sum=0;
            System.out.print("Enter the number : ");
            Scanner vn=new Scanner(System.in);
            num=vn.nextInt();
            temp=num;
            while(num>0)
            {
                    rem=num%10;
                    num=num/10;
```

```
                sum=rem+(sum*10);
            }
            if(temp==sum)
                System.out.println(temp+" is a palindrome number!");
             else
                System.out.println(temp+" is not a palindrome number!");
    }
}
```

OUTPUT :



```
C:\Users\VEMULA UTTEJ\Desktop\w1>java Palindrome.java
Enter the number : 123454321
123454321 is a palindrome number!

C:\Users\VEMULA UTTEJ\Desktop\w1>
```

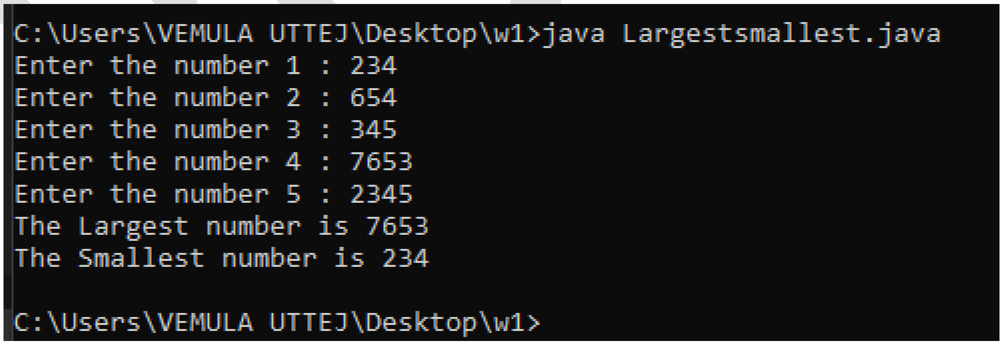
6. Write an application that declares 5 integers, determines and prints the largest and smallest in the group.

PROGRAM:

```
import java.io.*;
 import java.util.*;
 class Largestsmallest
{
        public static void main(String args[])
        {
                int max, min;
                int a[]=new int[5];
                 for(int i=0; i<5; i++)
                {
```

```
                Scanner vn=new Scanner(System.in);
                System.out.print("Enter the number "+(i+1)+" : ");
                a[i]=vn.nextInt();
            }
            max=a[0];
            min=a[0];
            for(int i=0;i<5;i++)
            {
                if(a[i]>max)
                    max=a[i];
                if(a[i]<min)
                    min=a[i];
            }


            System.out.println("The Largest number is "+max);
            System.out.println("The Smallest number is "+min);
        }
    }
```

<u>OUTPUT</u> :



```
C:\Users\VEMULA UTTEJ\Desktop\w1>java Largestsmallest.java
Enter the number 1 : 234
Enter the number 2 : 654
Enter the number 3 : 345
Enter the number 4 : 7653
Enter the number 5 : 2345
The Largest number is 7653
The Smallest number is 234

C:\Users\VEMULA UTTEJ\Desktop\w1>
```

WEEK-2

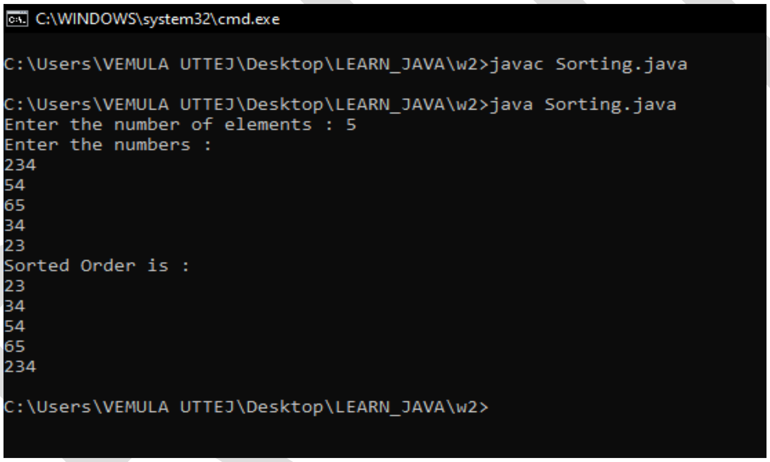
1.Write a Java program to sort given list of numbers.

Program :

```
import java.io.*;
import java.util.*;
class sort
{
   public static void main(String args[])
   {
            int n, i, j, t;
            Scanner vn1=new Scanner(System.in);
            System.out.print("Enter the number of elements : ");
            n=vn1.nextInt();
            int a[]=new int[n];
            System.out.println("Enter the numbers : ");
            for(i=0; i<n; i++)
            {
                      Scanner vn=new Scanner(System.in);
                      a[i]=vn.nextInt();
            }
            for(i=0;i<n;i++)
            {
                      for(j=0;j<n-1;j++)
                      {
                                if(a[i]<a[j])
                                          {
```

```
                    t=a[i];
                    a[i]=a[j];
                    a[j]=t;
                  }
              }
        }

        System.out.print("Sorted Order is :\n");
        for (i=0;i<n;i++)
        System.out.println(""+a[i]);
    }
}
```

<u>Output</u>:



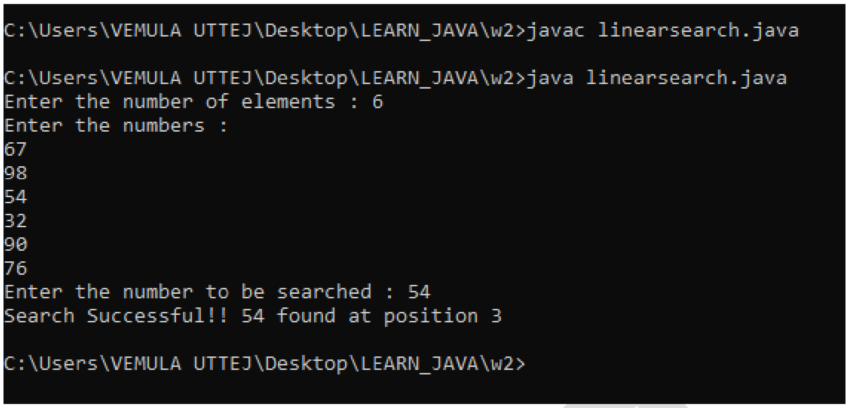
```
C:\Users\VEMULA UTTEJ\Desktop\LEARN_JAVA\w2>javac Sorting.java

C:\Users\VEMULA UTTEJ\Desktop\LEARN_JAVA\w2>java Sorting.java
Enter the number of elements : 5
Enter the numbers :
234
54
65
34
23
Sorted Order is :
23
34
54
65
234

C:\Users\VEMULA UTTEJ\Desktop\LEARN_JAVA\w2>
```

2. Write a Java program to implement linear search.

<u>Program</u> :

```
import java.io.*;
import java.util.*;
class lsearch
{
    public static void main(String arg[])
    {
        int n, find, i, search=0;
        Scanner vn1=new Scanner(System.in);
        System.out.print("Enter the number of elements : ");
        n=vn1.nextInt();
        int v[]=new int[n];
        System.out.println("Enter the numbers : ");
        for(i=0; i<n; i++)
        {
            Scanner vn=new Scanner(System.in);
            v[i]=vn.nextInt();
        }
        System.out.print("Enter the number to be searched : ");
        find=vn1.nextInt();
        for(i=0; i<n; i++)
        {
            if(find==v[i])
            {
                search=1;
                break;
            }
        }
        if(search==1)
            System.out.println("Search Successful!!
    "+find+" found at position "+(i+1));
        else
            System.out.println("Search Failed!! Match not
    found");
    }
    }
```

Output :



```
C:\Users\VEMULA UTTEJ\Desktop\LEARN_JAVA\w2>javac linearsearch.java

C:\Users\VEMULA UTTEJ\Desktop\LEARN_JAVA\w2>java linearsearch.java
Enter the number of elements : 6
Enter the numbers :
67
98
54
32
90
76
Enter the number to be searched : 54
Search Successful!! 54 found at position 3

C:\Users\VEMULA UTTEJ\Desktop\LEARN_JAVA\w2>
```

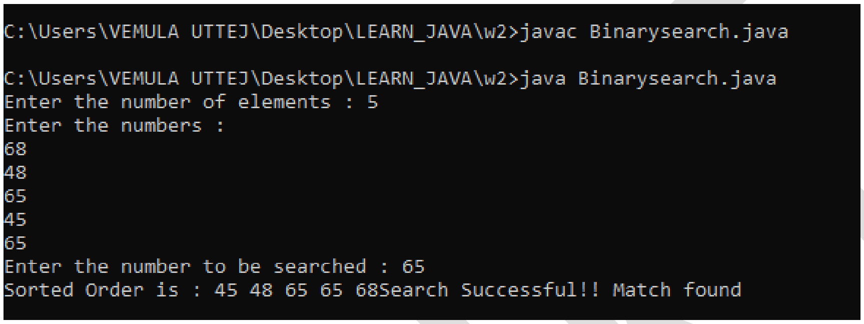
3. Write a Java program to implement binary search.

Programs :

```
import java.io.*;
import java.util.*;
class binsearch
{
   public static void main(String args[])
   {
           int n, i, j, t, find, search=0, first, mid=0, last;
           Scanner vn1=new Scanner(System.in);
           System.out.print("Enter the number of elements : ");
           n=vn1.nextInt();
           int v[]=new int[n];
           System.out.println("Enter the numbers : ");
           for(i=0; i<n; i++)
           {
                   Scanner vn=new Scanner(System.in);
                   v[i]=vn.nextInt();
           }
           System.out.print("Enter the number to be searched : ");
           find=vn1.nextInt();
           //sorting

for(i=0;i<n;i++)
           {
                   for(j=0;j<n-1;j++)
```

```
            {
                if(v[i]<v[j])
                    {
                      t=v[i];
                      v[i]=v[j];
                      v[j]=t;
                    }
                }
        }
        System.out.print("Sorted Order is :");
        for (i=0;i<n;i++)
        System.out.print(" "+v[i]);
        first=0;
        last=n-1;
        for(i=0; i<n; i++)
        {
            mid=(first+last)/2;
            if(find==v[mid])
            {
                search=1;
                break;
            }
            else if(find<v[mid])
                last=mid-1;
            else
                first=mid+1;
        }
        if(search==1)
            System.out.println("Search Successful!! Match found");
        else
            System.out.println("Search Failed!! Match not found");
    }
  }
```

Output:



```
C:\Users\VEMULA UTTEJ\Desktop\LEARN_JAVA\w2>javac Binarysearch.java

C:\Users\VEMULA UTTEJ\Desktop\LEARN_JAVA\w2>java Binarysearch.java
Enter the number of elements : 5
Enter the numbers :
68
48
65
45
65
Enter the number to be searched : 65
Sorted Order is : 45 48 65 65 68Search Successful!! Match found
```

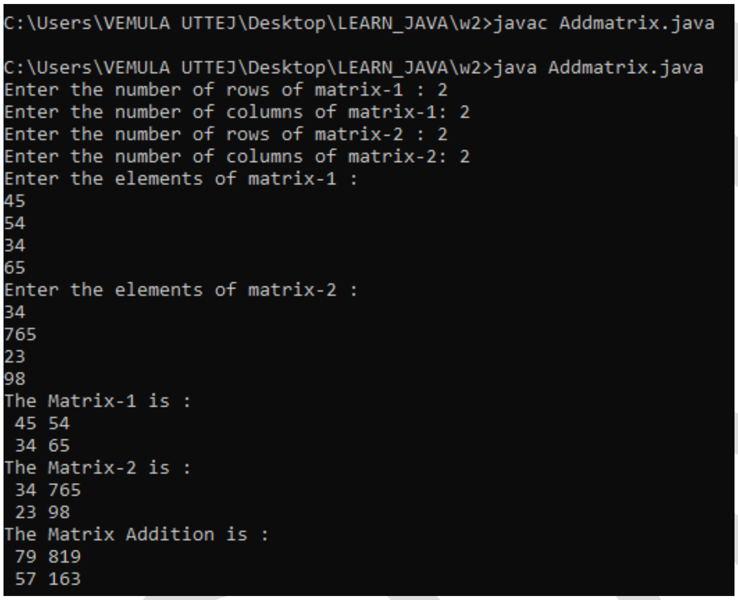
4. Write a java program to add two given matrices.

Output :

```
import java.io.*;
import java.util.*;
class addm
{
   public static void main(String args[])
   {
            int m1, n1, m2, n2, i, j;
            Scanner vn1=new Scanner(System.in);
            System.out.print("Enter the number of rows of matrix-1 : ");
            m1=vn1.nextInt();
            System.out.print("Enter the number of columns of matrix-1: ");
            n1=vn1.nextInt();
            System.out.print("Enter the number of rows of matrix-2 : ");
            m2=vn1.nextInt();
            System.out.print("Enter the number of columns of
matrix-2: ");
        n2=vn1.nextInt();
        if(m1==m2&&n1==n2)
        {
            int add[][]=new int[m1][n1];
            int v1[][]=new int[m1][n1];
            System.out.println("Enter the elements of
matrix-1 : ");
            for(i=0; i<m1; i++)
            {
                for(j=0; j<n1; j++)
                {
                    Scanner vn=new
Scanner(System.in);
                    v1[i][j]=vn.nextInt();
                }
            }
            int v2[][]=new int[m2][n2];
            System.out.println("Enter the elements of
matrix-2 : ");
            for(i=0; i<m2; i++)
            {
                for(j=0; j<n2; j++)
                {
                    Scanner vn=new
Scanner(System.in);
                    v2[i][j]=vn.nextInt();
                }
            }
            System.out.println("The Matrix-1 is : ");
            for(i=0; i<m1; i++)
            {
                for(j=0; j<n1; j++)
                {
                    System.out.print(" "+v1[i][j]);
                }
                System.out.println(" ");
            }
            System.out.println("The Matrix-2 is : ");
            for(i=0; i<m2; i++)
            {
                for(j=0; j<n2; j++)
                {
                    System.out.print(" "+v2[i][j]);
                }
                System.out.println(" ");
            }
            //matrix addition

            for(i=0; i<m1; i++)
            {
                for(j=0; j<n1; j++)
                {
                    add[i][j]=v1[i][j]+v2[i][j];
                }
            }

            System.out.println("The Matrix Addition is : ");
            for(i=0; i<m1; i++)
            {

for(j=0; j<n1; j++)
                {
                    System.out.print(" "+add[i][j]);
                }
                System.out.println(" ");
            }
        }
        else
            System.out.print("Rows and Columns not
matched. Matrix addition not possible!");
```

```
    }
}
```

Output :



```
C:\Users\VEMULA UTTEJ\Desktop\LEARN_JAVA\w2>javac Addmatrix.java

C:\Users\VEMULA UTTEJ\Desktop\LEARN_JAVA\w2>java Addmatrix.java
Enter the number of rows of matrix-1 : 2
Enter the number of columns of matrix-1: 2
Enter the number of rows of matrix-2 : 2
Enter the number of columns of matrix-2: 2
Enter the elements of matrix-1 :
45
54
34
65
Enter the elements of matrix-2 :
34
765
23
98
The Matrix-1 is :
 45 54
 34 65
The Matrix-2 is :
 34 765
 23 98
The Matrix Addition is :
 79 819
 57 163
```

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

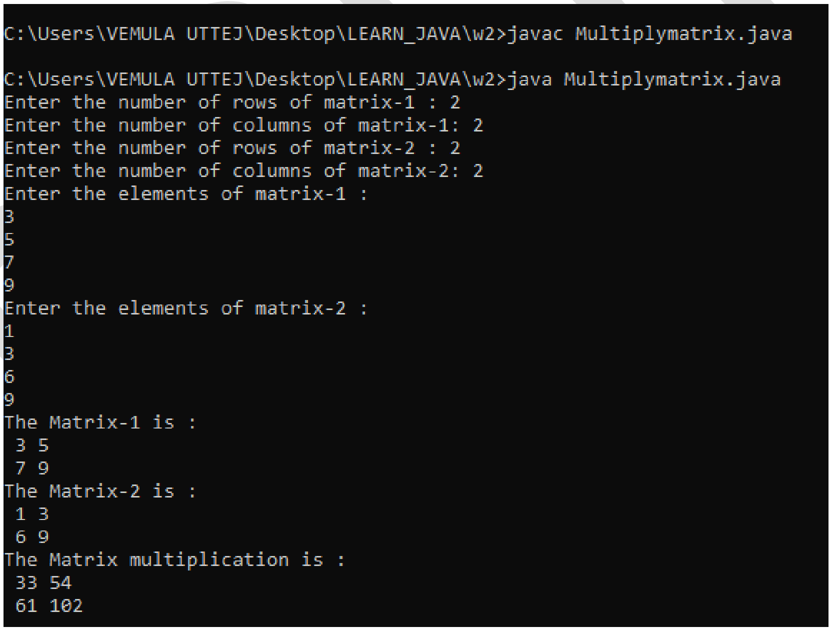
Program :

```
import java.io.*;
import java.util.*;
class mulm
{
    public static void main(String args[])
    {
            int m1, n1, m2, n2, i, j, k, sum=0, product=1;
            Scanner vn1=new Scanner(System.in);

System.out.print("Enter the number of rows of matrix-1 : ");
            m1=vn1.nextInt();
            System.out.print("Enter the number of columns of
matrix-1: ");
```

```
        n1=vn1.nextInt();
        System.out.print("Enter the number of rows of matrix-
2 : ");
        m2=vn1.nextInt();
        System.out.print("Enter the number of columns of
matrix-2: ");
        n2=vn1.nextInt();
        if(n1==m2)
        {
            int mul[][]=new int[m1][m2];
            int v1[][]=new int[m1][n1];
            System.out.println("Enter the elements of
matrix-1 : ");
            for(i=0; i<m1; i++)
            {
                for(j=0; j<n1; j++)
                {
                    Scanner vn=new
Scanner(System.in);
                    v1[i][j]=vn.nextInt();
                }
            }
            int v2[][]=new int[m2][n2];
            System.out.println("Enter the elements of
matrix-2 : ");
            for(i=0; i<m2; i++)
            {
                for(j=0; j<n2; j++)
                {
                    Scanner vn=new
Scanner(System.in);
                    v2[i][j]=vn.nextInt();
                }
            }
            System.out.println("The Matrix-1 is : ");
            for(i=0; i<m1; i++)
        {
            for(j=0; j<n1; j++)
            {
                System.out.print(" "+v1[i][j]);
            }
            System.out.println(" ");
        }
        System.out.println("The Matrix-2 is : ");

for(i=0; i<m2; i++)
        {
            for(j=0; j<n2; j++)
            {
                System.out.print(" "+v2[i][j]);
            }
            System.out.println(" ");
        }
        //matrix multiplication

        for(i=0; i<m1; i++)
        {
            for(j=0; j<n2; j++)
            {
                for(k=0;k<n1;k++)
                {

 mul[i][j]+=v1[i][k]*v2[k][j];
                }
            }

        }

        System.out.println("The Matrix multiplication is
: ");
        for(i=0; i<m1; i++)
```

```
                {
                        for(j=0; j<n2; j++)
                        {
                                System.out.print(" "+mul[i][j]);
                        }
                        System.out.println(" ");
                }
        }
        else
                System.out.print("Rows and Columns not matched. Matrix Multiplication not possible!");



    }
}
```

Output :



```
C:\Users\VEMULA UTTEJ\Desktop\LEARN_JAVA\w2>javac Multiplymatrix.java

C:\Users\VEMULA UTTEJ\Desktop\LEARN_JAVA\w2>java Multiplymatrix.java
Enter the number of rows of matrix-1 : 2
Enter the number of columns of matrix-1: 2
Enter the number of rows of matrix-2 : 2
Enter the number of columns of matrix-2: 2
Enter the elements of matrix-1 :
3
5
7
9
Enter the elements of matrix-2 :
1
3
6
9
The Matrix-1 is :
 3 5
 7 9
The Matrix-2 is :
 1 3
 6 9
The Matrix multiplication is :
 33 54
 61 102
```

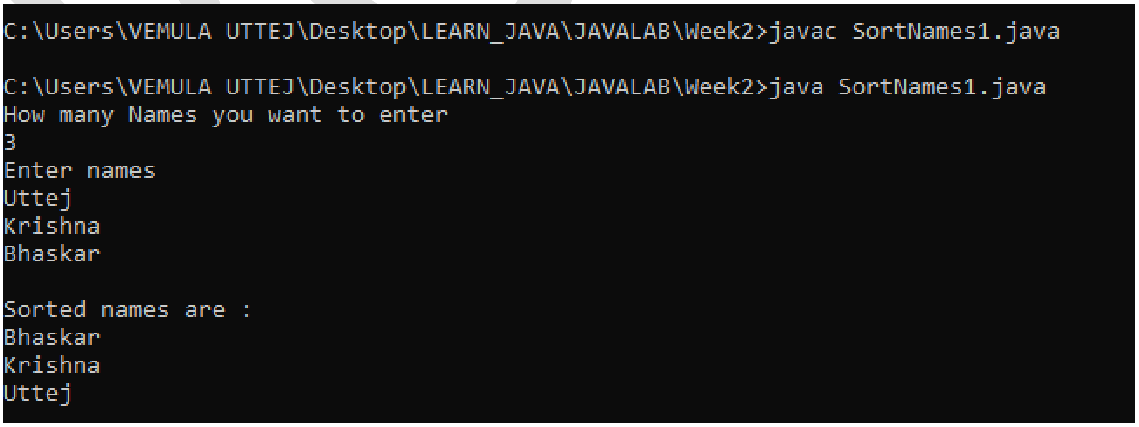
6. Write a java program for sorting a given list of names.

Program :

```
import java.util.Scanner;
import java.util.Arrays;
```

```
class SortNames1
{
        public static void main(String[] args)
        {
                int n;
                System.out.println("How many Names you want to enter");
                Scanner in=new Scanner(System.in);
                n=in.nextInt();
                System.out.println("Enter names");
                String name[]=new String[n];
                for(int i=0;i<n;i++)
                        name[i]=in.next();

                Arrays.sort(name);

                System.out.println("\nSorted names are :");
                for(int i=0;i<n;i++)
                        System.out.println(name[i]);
        }
}
```

Output :



```
C:\Users\VEMULA UTTEJ\Desktop\LEARN_JAVA\JAVALAB\Week2>javac SortNames1.java

C:\Users\VEMULA UTTEJ\Desktop\LEARN_JAVA\JAVALAB\Week2>java SortNames1.java
How many Names you want to enter
3
Enter names
Uttej
Krishna
Bhaskar

Sorted names are :
Bhaskar
Krishna
Uttej
```

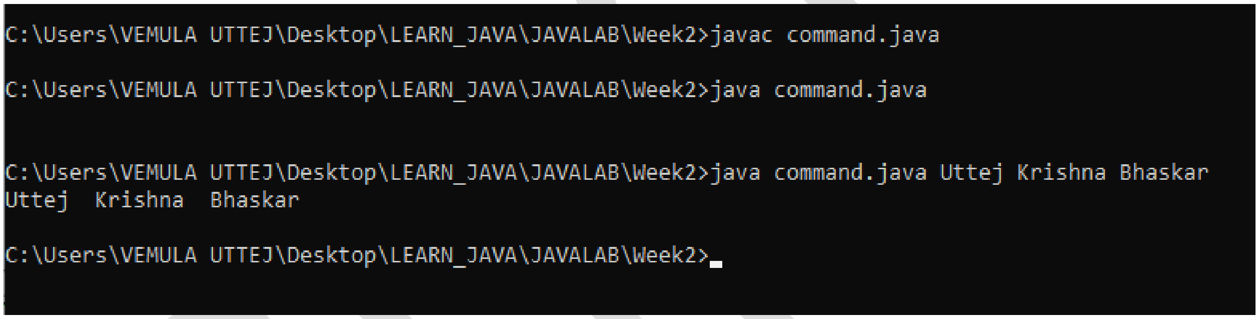
7. Write a Java program to give an example for command line arguments.

Program :

```
class command
```

```
{
   public static void main(String arg[])
   {
      for(int i=0;i<arg.length;i++)
      System.out.print(arg[i]+"  ");
      System.out.println();
   }
}
```

Output :



```
C:\Users\VEMULA UTTEJ\Desktop\LEARN_JAVA\JAVALAB\Week2>javac command.java

C:\Users\VEMULA UTTEJ\Desktop\LEARN_JAVA\JAVALAB\Week2>java command.java


C:\Users\VEMULA UTTEJ\Desktop\LEARN_JAVA\JAVALAB\Week2>java command.java Uttej Krishna Bhaskar
Uttej  Krishna  Bhaskar

C:\Users\VEMULA UTTEJ\Desktop\LEARN_JAVA\JAVALAB\Week2>_
```

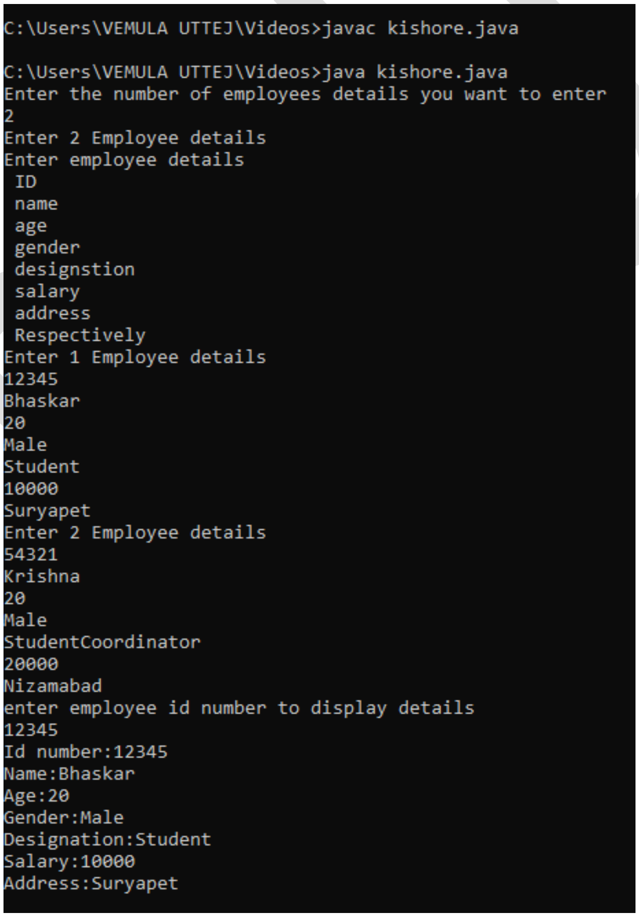
WEEK -3:

1. **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.**

Program :

```
import java.util.*;
class kishore{
        public static void main(String args[]){
                Scanner ip=new Scanner(System.in);
System.out.println("Enter the number of employees details you want to enter");
                int n=ip.nextInt();
                System.out.println("Enter " +n+ " Employee details");
                int id[]=new int[n];
                String name[]=new String[n];
                int age[]=new int[n];
                String gender[]=new String[n];
                String des[]=new String[n];
                int salary[]=new int[n];
                String add[]=new String[n];
System.out.println("Enter employee details \n ID \n name \n age \n gender                    \n
designstion \n salary \n address \n Respectively");

for(int i=0;i<n;i++){
                        System.out.println("Enter "+(i+1)+" Employee details");
                        id[i]=ip.nextInt();
                        name[i]=ip.next();
                        age[i]=ip.nextInt();
                        gender[i]=ip.next();
                        des[i]=ip.next();
                        salary[i]=ip.nextInt();
                        add[i]=ip.next();
                }
        System.out.println("enter employee id number to display details");
                int ID=ip.nextInt();
                for(int i=0;i<n;i++){
                        if(ID==id[i]){
                            System.out.println("Id number:"+id[i]);
                            System.out.println("Name:"+name[i]);
                            System.out.println("Age:"+age[i]);
                            System.out.println("Gender:"+gender[i]);
                            System.out.println("Designation:"+des[i]);
                            System.out.println("Salary:"+salary[i]);
                            System.out.println("Address:"+add[i]);
                        }
                }
        }
}
```

OUTPUT:



```
C:\Users\VEMULA UTTEJ\Videos>javac kishore.java

C:\Users\VEMULA UTTEJ\Videos>java kishore.java
Enter the number of employees details you want to enter
2
Enter 2 Employee details
Enter employee details
 ID
 name
 age
 gender
 designstion
 salary
 address
 Respectively
Enter 1 Employee details
12345
Bhaskar
20
Male
Student
10000
Suryapet
Enter 2 Employee details
54321
Krishna
20
Male
StudentCoordinator
20000
Nizamabad
enter employee id number to display details
12345
Id number:12345
Name:Bhaskar
Age:20
Gender:Male
Designation:Student
Salary:10000
Address:Suryapet
```

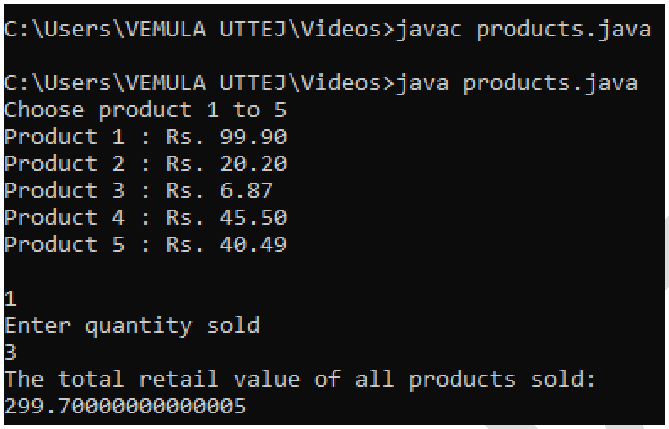
2. A mail-order house sells five products whose retail prices are as follows : Product 1 : Rs. 99.90 , Product 2 : Rs. 20.20 , Product 3 : Rs. 6.87 , Product 4 : Rs. 45.50 and Product 5 : Rs. 40.49 . Each product has Prdouct_Id, Product_Name, Product_Quantity, Product_Price. Write an application that 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.

Program :

```
import java.util.Scanner;
public class products
{
    public static void main(String[] args)
    {
    Scanner input = new Scanner (System.in);
    double product1;
    double product2;
    double product3;
    double product4;
    double product5;
    int quantity;
    double totalSales = 0;
    System.out.println("Choose product 1 to 5 ");
    System.out.println("Product 1 : Rs. 99.90\nProduct 2 : Rs. 20.20\nProduct 3 : Rs. 6.87
                    \nProduct 4 : Rs. 45.50\nProduct 5 : Rs. 40.49");
    int productNo=input.nextInt();
    System.out.println("Enter quantity sold ");
    quantity = input.nextInt();
    switch (productNo)
    {
    case 1: product1 = 99.90;
            totalSales+=(99.90*quantity);
            break;
    case 2: product2 = 20.20;
            totalSales+=(20.20*quantity);
            break;
    case 3: product3 = 6.87;
            totalSales+=(6.87*quantity);
            break;
    case 4: product4 = 45.50;
            totalSales+=(45.50*quantity);
            break;
    case 5: product5 = 40.49;
            totalSales+=(40.49*quantity);
            break;
    }

    System.out.println("The total retail value of all products sold:\n"+totalSales);
    }
```

}

OUTPUT:



```
C:\Users\VEMULA UTTEJ\Videos>javac products.java

C:\Users\VEMULA UTTEJ\Videos>java products.java
Choose product 1 to 5
Product 1 : Rs. 99.90
Product 2 : Rs. 20.20
Product 3 : Rs. 6.87
Product 4 : Rs. 45.50
Product 5 : Rs. 40.49

1
Enter quantity sold
3
The total retail value of all products sold:
299.70000000000005
```

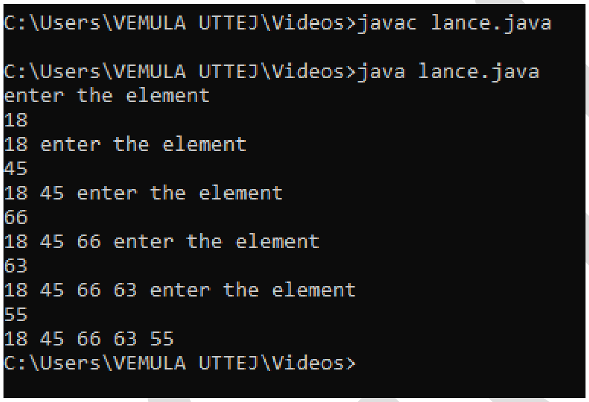
3. **Write java program that inputs 5 numbers, each between 10 and 100 inclusive. As each number is read display it only if it’s not a duplicate of any number already read display the complete set of unique values input after the user enters each new value.**

<u>Program</u> :

```
import java.util.*;
class lance
{
    public static void main(String args[])
    {
            Scanner s=new Scanner(System.in);
            int arr[]=new int[5];
            for(int i=0;i<5;i++)
            {
            int count=0;
            System.out.println("enter the element");
            int e=s.nextInt();
            for(int j=0;j<arr.length;j++)
            {
            if(e==arr[j])
            {
            count=1;
            System.out.println("number already exists,enter another number");
            i=i-1;
            break;
            }
            else if(e<10||e>100)
            {
            count=1;
            System.out.println("number is not in range (10&100)enter a valid number");
            i=i-1;
            break;
            }
```

```
                }
                if(count!=1)
                {
                arr[i]=e;

for (int k=0;k<=i;k++)
                {
                        System.out.print(arr[k]+" ");
                }
    }
    }
}
}
```

Output :



```
C:\Users\VEMULA UTTEJ\Videos>javac lance.java

C:\Users\VEMULA UTTEJ\Videos>java lance.java
enter the element
18
18 enter the element
45
18 45 enter the element
66
18 45 66 enter the element
63
18 45 66 63 enter the element
55
18 45 66 63 55
C:\Users\VEMULA UTTEJ\Videos>
```

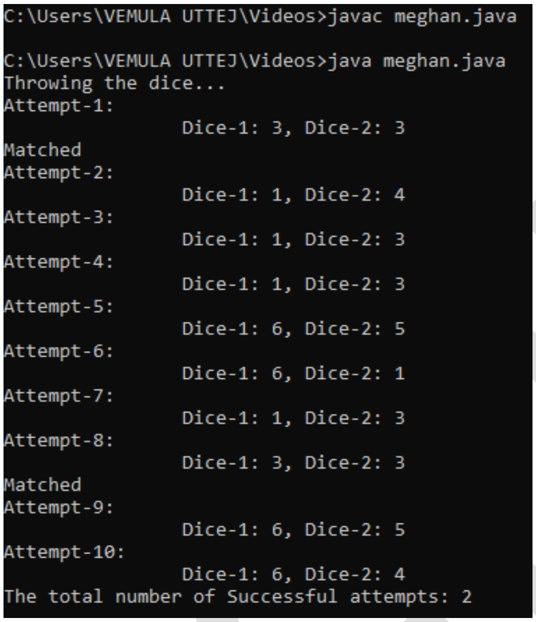
4. **Write a java program : rolling a pair of dices 10 times [each attempt should be delayed by 10000 ms] and count number Successful attempts. successful attempt : If the pair of Dice results in same values.**

Program :

```
import java.util.*;
class meghan {
  static int dice_number(){
  return (int)(Math.random()*6)+1;
  }
  public static void main(String[] args) throws InterruptedException {
        int count=0;
        System.out.println("Throwing the dice...");
        for (int a=1;a<=10;a++){
        int dice1=dice_number();
        int dice2=dice_number();
        System.out.println("Attempt-"+a+": \n\t\tDice-1: "+dice1+", Dice-2: "+dice2);
        Thread.sleep(1000);
        if(dice1==dice2){
        count++;
        System.out.println("Matched");
        }
        }
```

```
                System.out.println("The total number of Successful attempts: "+count);
        }
    }
```

Output :



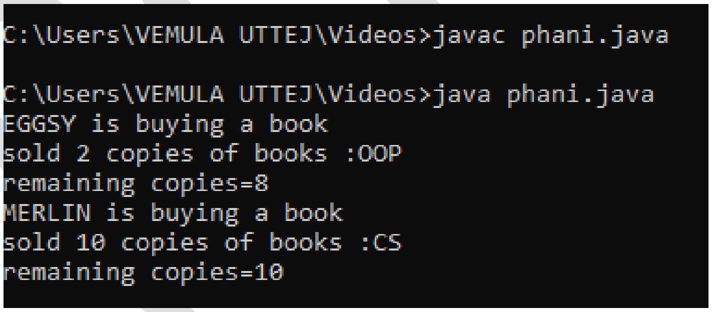
```
C:\Users\VEMULA UTTEJ\Videos>javac meghan.java

C:\Users\VEMULA UTTEJ\Videos>java meghan.java
Throwing the dice...
Attempt-1:
                Dice-1: 3, Dice-2: 3
Matched
Attempt-2:
                Dice-1: 1, Dice-2: 4
Attempt-3:
                Dice-1: 1, Dice-2: 3
Attempt-4:
                Dice-1: 1, Dice-2: 3
Attempt-5:
                Dice-1: 6, Dice-2: 5
Attempt-6:
                Dice-1: 6, Dice-2: 1
Attempt-7:
                Dice-1: 1, Dice-2: 3
Attempt-8:
                Dice-1: 3, Dice-2: 3
Matched
Attempt-9:
                Dice-1: 6, Dice-2: 5
Attempt-10:
                Dice-1: 6, Dice-2: 4
The total number of Successful attempts: 2
```

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.**

Program :

```
import java.util.*;
class paul {
        public static void main(String args[]) {
                Book b1=new Book("OOP","eggsy",10);
                customer c1=new customer("EGGSY","rgukt_basar",10);
                c1.buy(b1,2);
                Book b2=new Book("CS","merlin",20);
                customer c2=new customer("MERLIN","rgukt_basar",20);
                c2.buy(b2,10);
        }
}
class Book {
        String name,author;
        int count;
        public Book(String name,String author,int count)
        {
        this.name=name;
        this.author=author;
        this.count=count;
}
public void sell(int n) {
        if(n<=count) {
        count=count-n;
        System.out.println("sold "+n+" copies of books :"+name);
System.out.println("remaining copies="+count);
}
else {
        System.out.println("stock not available");
}
}
}
class customer {
        String name,address;
        int cid;
        public customer(String name,String address,int cid) {
                this.name=name;
                this.address=address;
                this.cid=cid;
        }
        public void buy(Book b,int n) {
                System.out.println(name+" is buying a book");
                b.sell(n);
}
}
```

OUTPUT:



```
C:\Users\VEMULA UTTEJ\Videos>javac phani.java

C:\Users\VEMULA UTTEJ\Videos>java phani.java
EGGSY is buying a book
sold 2 copies of books :OOP
remaining copies=8
MERLIN is buying a book
sold 10 copies of books :CS
remaining copies=10
```

Week-IV

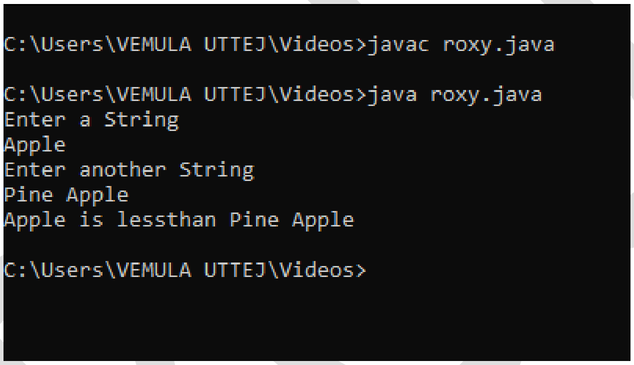
1. **Write an application that uses String method compareTo to compare two strings defined by the user.**

Program :

```
import java.util.Scanner;
class roxy
```

```
{
    public static void main(String args[])
    {
            Scanner sc = new Scanner(System.in);
            System.out.println("Enter a String");
            String s=sc.nextLine();
            System.out.println("Enter another String");
            String s1=sc.nextLine();
            if(s.compareTo(s1)>0)
            {
                    System.out.println(s+" is greater than "+s1);
            }
            else if(s.compareTo(s1)<0)
            {
                    System.out.println(s+" is lessthan "+s1);
            }
            else
            {
                    System.out.println("Given two Strings are equal");
            }
    }
}
```

OUTPUT:



```
C:\Users\VEMULA UTTEJ\Videos>javac roxy.java

C:\Users\VEMULA UTTEJ\Videos>java roxy.java
Enter a String
Apple
Enter another String
Pine Apple
Apple is lessthan Pine Apple

C:\Users\VEMULA UTTEJ\Videos>
```

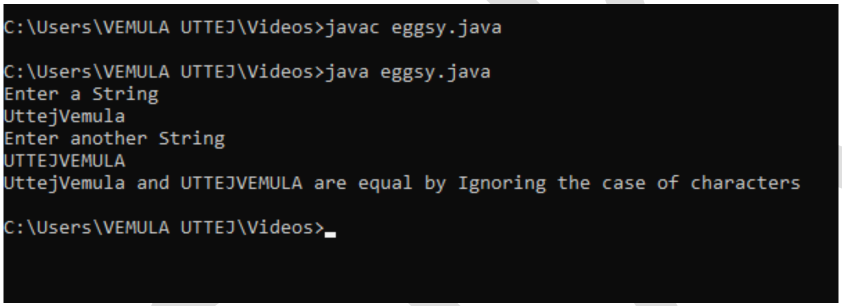
2. **Write an application that uses String method equals and equalsIgnoreCase to tests any two string objects for equality.**

<u>Program :</u>

```
import java.util.Scanner;
class eggsy
{
    public static void main(String args[])
    {
            Scanner sc = new Scanner(System.in);
            System.out.println("Enter a String");
            String s=sc.nextLine();
            System.out.println("Enter another String");
            String s1=sc.nextLine();
    if(s.equals(s1))
    {
```

```
                System.out.println(s+" and "+s1+" are equal");
        }
        else if(s.equalsIgnoreCase(s1))
        {
                System.out.println(s+" and "+s1+" are equal by Ignoring the case of characters");
        }
        else
        {
                System.out.println(s+" and "+s1+" are not equal");
        }
        }
}
```

OUTPUT:



```
C:\Users\VEMULA UTTEJ\Videos>javac eggsy.java

C:\Users\VEMULA UTTEJ\Videos>java eggsy.java
Enter a String
UttejVemula
Enter another String
UTTEJVEMULA
UttejVemula and UTTEJVEMULA are equal by Ignoring the case of characters

C:\Users\VEMULA UTTEJ\Videos>_
```

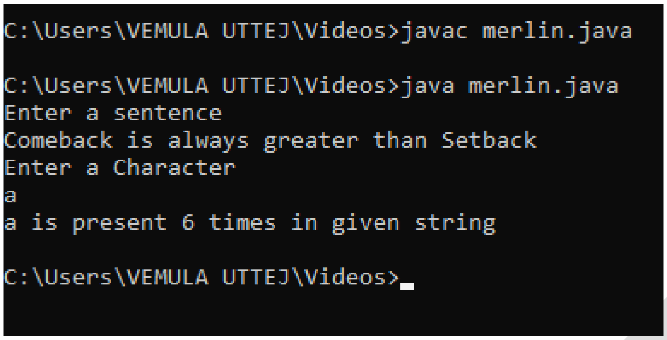
3. **Write an application that uses String method indexOf to determine the total number of occurrences of any given alphabet in a defined text.**

Program :

```
import java.util.Scanner;
class merlin
{
        public static void main(String args[])
        {
                Scanner sc = new Scanner(System.in);
                System.out.println("Enter a sentence");
                String sen = sc.nextLine();
                System.out.println("Enter a Character");
                char ch = sc.next().charAt(0);
                int count=0;
        for(int i=0;i<sen.length();i++)
        {
                if(sen.charAt(sen.indexOf(ch))==sen.charAt(i))
                {
                count++;
                }
        }
        System.out.println(ch+" is present "+count+ " times in given string");
        }
```

}

OUTPUT:



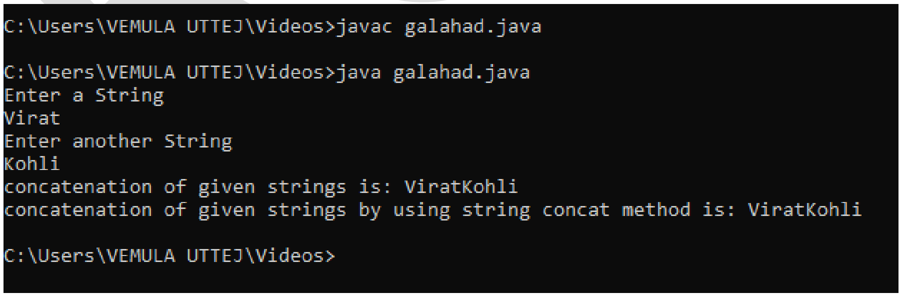
```
C:\Users\VEMULA UTTEJ\Videos>javac merlin.java

C:\Users\VEMULA UTTEJ\Videos>java merlin.java
Enter a sentence
Comeback is always greater than Setback
Enter a Character
a
a is present 6 times in given string

C:\Users\VEMULA UTTEJ\Videos>_
```

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

<u>Program</u> :

```
import java.util.Scanner;
class galahad
{
    public static void main(String args[])
    {
            Scanner sc = new Scanner(System.in);
            System.out.println("Enter a String");
            String s=sc.nextLine();
            System.out.println("Enter another String");
            String s1=sc.nextLine();
            System.out.println("concatenation of given strings is: "+s+s1);
            String s2=s.concat(s1);
            System.out.println("concatenation of given strings by using string concat method is: 
"+s2);
    }
}
```

OUTPUT:



```
C:\Users\VEMULA UTTEJ\Videos>javac galahad.java

C:\Users\VEMULA UTTEJ\Videos>java galahad.java
Enter a String
Virat
Enter another String
Kohli
concatenation of given strings is: ViratKohli
concatenation of given strings by using string concat method is: ViratKohli

C:\Users\VEMULA UTTEJ\Videos>
```

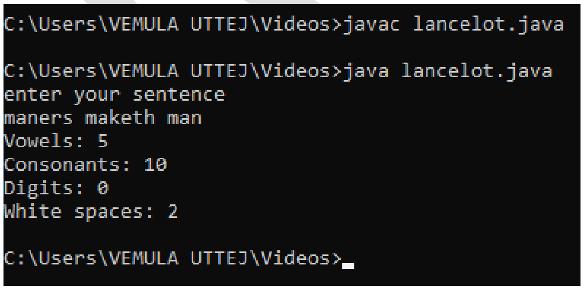
5. Write a Java program to print all vowels in given string and count number of vowels and consonants present in given string.

<u>Program</u> :

```
import java.util.Scanner;
class lancelot
{
```

```
public static void main(String args[])
{
String sen;
Scanner in = new Scanner(System.in);
System.out.println("enter your sentence");
sen = in.nextLine();
int vow=0, con=0, digits=0, spaces=0;
char ch;
sen=sen.toLowerCase();
for(int i = 0; i < sen.length(); ++i)
{
ch = sen.charAt(i);
if(ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u'||ch=='A'|| ch=='E'|| ch=='I'||
    ch=='O'||ch=='U')
{
++vow;
}
else if((ch >= 'a' && ch <= 'z'))
{
++con;
}
else if(ch >= '0' && ch <= '9')
{
++digits;
}
else if(ch == ' ')
{
++spaces;
}
}
System.out.println("Vowels: " +vow);
System.out.println("Consonants: " +con);
System.out.println("Digits: " +digits);
System.out.println("White spaces: " +spaces);
}
}
```

OUTPUT:



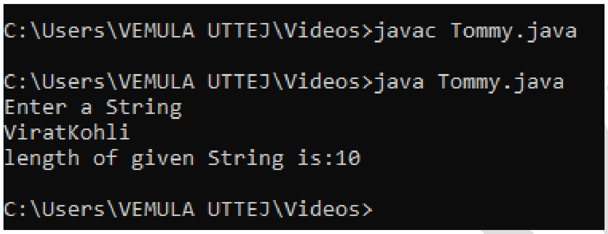
```
C:\Users\VEMULA UTTEJ\Videos>javac lancelot.java

C:\Users\VEMULA UTTEJ\Videos>java lancelot.java
enter your sentence
maners maketh man
Vowels: 5
Consonants: 10
Digits: 0
White spaces: 2

C:\Users\VEMULA UTTEJ\Videos>_
```

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

Program :

```
import java.util.Scanner;
class paul
{
    public static void main(String args[])
    {
            Scanner sc = new Scanner(System.in);
            System.out.println("Enter a String");
            String s=sc.nextLine();
            int a=s.length();
            System.out.println("length of given String is:"+a);
    }
}
```

OUTPUT:



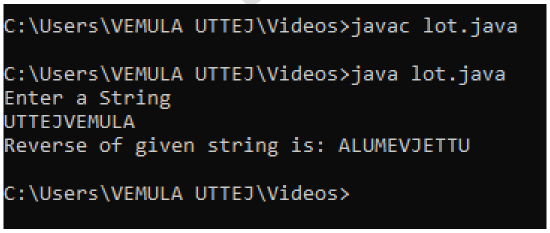
```
C:\Users\VEMULA UTTEJ\Videos>javac Tommy.java

C:\Users\VEMULA UTTEJ\Videos>java Tommy.java
Enter a String
ViratKohli
length of given String is:10

C:\Users\VEMULA UTTEJ\Videos>
```

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

<u>Program</u> :

```
import java.util.Scanner;


class lot
{
    public static void main(String args[])
    {
    Scanner sc = new Scanner(System.in);
    System.out.println("Enter a String");
    String s=sc.nextLine();
    int n=s.length();
    System.out.print("Reverse of given string is: ");
    for(int i=n-1;i>=0;i-- )
    {
            char rev=s.charAt(i);
            System.out.print(rev);
    }
    System.out.println("");
    }
}
```

OUTPUT:



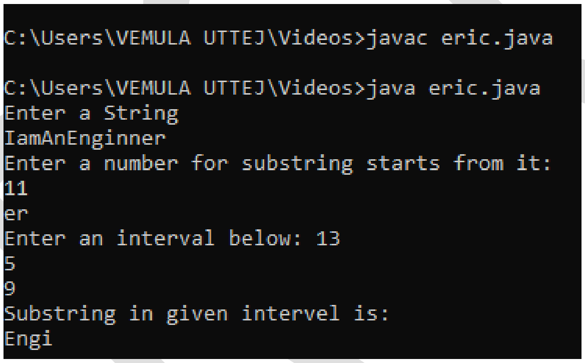
```
C:\Users\VEMULA UTTEJ\Videos>javac lot.java

C:\Users\VEMULA UTTEJ\Videos>java lot.java
Enter a String
UTTEJVEMULA
Reverse of given string is: ALUMEVJETTU

C:\Users\VEMULA UTTEJ\Videos>
```

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

Program :

```
import java.util.Scanner;
class eric
{
    public static void main(String args[])
    {
            Scanner sc = new Scanner(System.in);
            System.out.println("Enter a String");
            String s=sc.nextLine();
            System.out.println("Enter a number for substring starts from it:
                    "+(s.length()));
            int n=sc.nextInt();
            String s1=s.substring(n);
            System.out.println(s1);
            System.out.println("Enter an interval below: "+(s.length()));
            int a=sc.nextInt();
            int b=sc.nextInt();
            String s2=s.substring(a,b);
            System.out.println("Substring in given intervel is:");
            System.out.println(s2);
    }
}
```

OUTPUT:



```
C:\Users\VEMULA UTTEJ\Videos>javac eric.java

C:\Users\VEMULA UTTEJ\Videos>java eric.java
Enter a String
IamAnEnginner
Enter a number for substring starts from it:
11
er
Enter an interval below: 13
5
9
Substring in given intervel is:
Engi
```

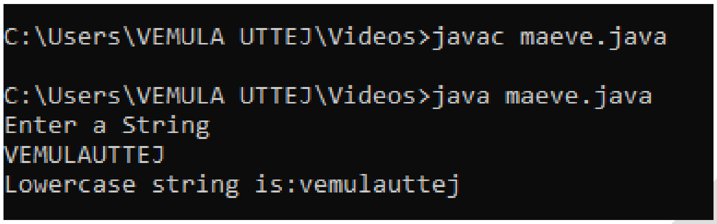
9. **Write an application that changes any given string with uppercase letters, displays it, changes it back to lowercase letters and displays it.**

Program :

```
import java.util.Scanner;
class maeve
{
    public static void main(String args[])
    {
            Scanner sc = new Scanner(System.in);
            System.out.println("Enter a String");
            String s=sc.nextLine();
            String s1=s.toLowerCase();
            System.out.print("Lowercase string is:"+s1);
```

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

OUTPUT:



```
C:\Users\VEMULA UTTEJ\Videos>javac maeve.java

C:\Users\VEMULA UTTEJ\Videos>java maeve.java
Enter a String
VEMULAUTTEJ
Lowercase string is:vemulauttej
```

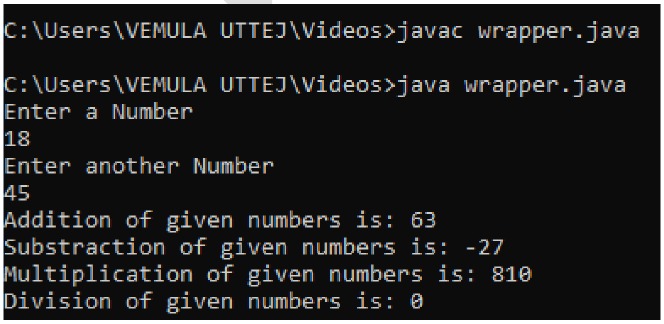
WEEK-5

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

Program :

```
import java.util.*;
class wrapper
{
  public static void main(String args[])
  {
       int a,b;
       Scanner s=new Scanner(System.in);
       System.out.println("Enter a Number");


       a=s.nextInt();
       Integer c=Integer.valueOf(a);
       System.out.println("Enter another Number");
       b=s.nextInt();
       Integer d=Integer.valueOf(b);
       System.out.println("Addition of given numbers is: "+(c+d));
       System.out.println("Substraction of given numbers is: "+(c-d));
       System.out.println("Multiplication of given numbers is: "+(c*d));
       System.out.println("Division of given numbers is: "+(c/d));
  }
}
```

OUTPUT:



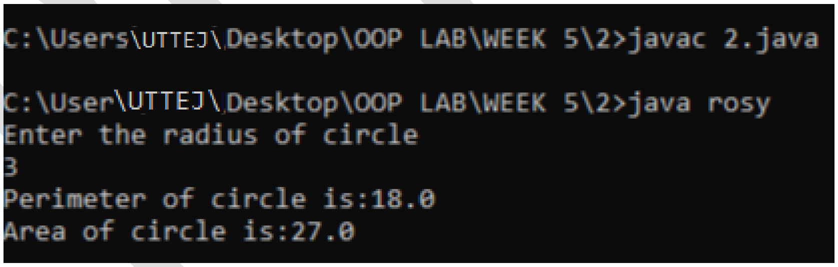
```
C:\Users\VEMULA UTTEJ\Videos>javac wrapper.java

C:\Users\VEMULA UTTEJ\Videos>java wrapper.java
Enter a Number
18
Enter another Number
45
Addition of given numbers is: 63
Substraction of given numbers is: -27
Multiplication of given numbers is: 810
Division of given numbers is: 0
```

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.

```
import java.util.*;
class eggsy
{
        float r;
        float Area()
        {
                return (22/7)*r*r;
        }
        float Perimeter()
        {
                return 2*(22/7)*r;
        }
}
class rosy
{
        public static void main(String args[])
        {
                Scanner s=new Scanner(System.in);
                eggsy c=new eggsy();
                System.out.println("Enter the radius of circle");
                float R=s.nextFloat();
                c.r=R;
                System.out.println("Perimeter of circle is:"+c.Perimeter());
                System.out.println("Area of circle is:"+c.Area());
        }
}
```

OUTPUT:



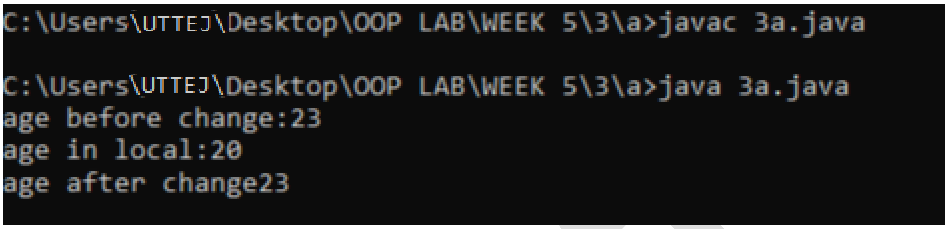
```
C:\Users\UTTEJ\Desktop\OOP LAB\WEEK 5\2>javac 2.java

C:\User\UTTEJ\Desktop\OOP LAB\WEEK 5\2>java rosy
Enter the radius of circle
3
Perimeter of circle is:18.0
Area of circle is:27.0
```

3. Write a JAVA program for the following:

a. Call by value

```
class paul
{
        int age=23;
        void change_age(int age)
        {
                age=age+10;// cahnges will be done in the local variable only
                System.out.println("age in local:"+age);
        }
        public static void main(String args [])
        {
                paul c1=new paul();
                System.out.println("age before change:"+c1.age);
                c1.change_age(10);
                System.out.println("age after change"+c1.age);
        }
}
```

OUTPUT:

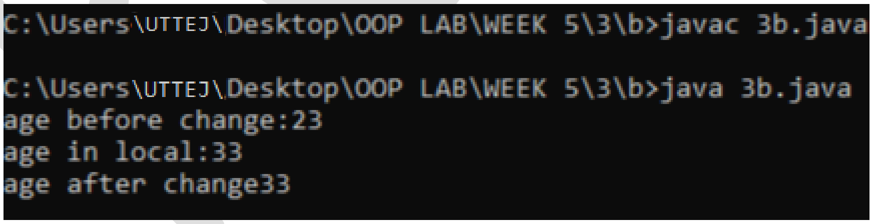


```
C:\Users\UTTEJ\Desktop\OOP LAB\WEEK 5\3\a>javac 3a.java

C:\Users\UTTEJ\Desktop\OOP LAB\WEEK 5\3\a>java 3a.java
age before change:23
age in local:20
age after change23
```

b. Call by object

```
class adam
{
        int age=23;
        void change_age(adam c1)
        {
        age=age+10;// cahnges will be done in the local variable only
        System.out.println("age in local:"+age);
        }
        public static void main(String args [])
        {
                adam c1=new adam();
                System.out.println("age before change:"+c1.age);
                c1.change_age(c1);// passinf object through called method
                System.out.println("age after change"+c1.age);
}
}
```

OUTPUT:

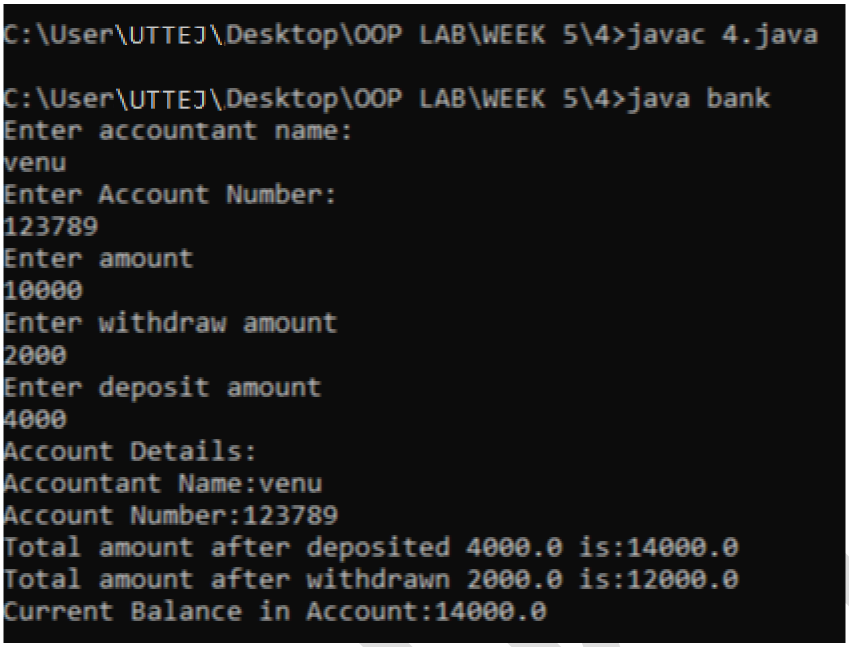


```
C:\Users\UTTEJ\Desktop\OOP LAB\WEEK 5\3\b>javac 3b.java

C:\Users\UTTEJ\Desktop\OOP LAB\WEEK 5\3\b>java 3b.java
age before change:23
age in local:33
age after change33
```

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.

```
import java.util.Scanner;
class Account
{
        String Accountant_name;
        int Account_number;
        double amount,withdraw,deposit;
        Account(String s,double a,int i,double w,double d)
        {
                Accountant_name=s;
        amount=a;
        Account_number=i;
        withdraw=w;
        deposit=d;
    }
    void debit(double w)
    {
    if(amount>w)
    {
        amount=(amount-w);
        System.out.println("Total amount after withdrawn "+withdraw+" is:"+amount);
        double M=(amount-withdraw)+deposit;
        System.out.println("Current Balance in Account:"+M);
    }
    else
    {
        System.out.println("Insufficent accunt balance");
        System.out.println("Enter amount below:"+amount);
    }
}
    void credit(double d)
    {
        amount=(amount+d);
        System.out.println("Accountant Name:"+Accountant_name);
        System.out.println("Account Number:"+Account_number);
        System.out.println("Total amount after deposited "+deposit+" is:"+amount);
    }
}
class bank
    {
    public static void main(String args[])
    {
        Scanner s=new Scanner(System.in);
        System.out.println("Enter accountant name:");
        String name=s.nextLine();
        System.out.println("Enter Account Number:");
        int i=s.nextInt();
        System.out.println("Enter amount");
        double a=s.nextDouble();
        System.out.println("Enter withdraw amount");
        double w=s.nextDouble();
        System.out.println("Enter deposit amount");
        double d=s.nextDouble();
        System.out.println("Account Details:");
        Account Ac=new Account(name,a,i,w,d);
        Ac.credit(d);
        Ac.debit(w);
    }
}
```

OUTPUT:

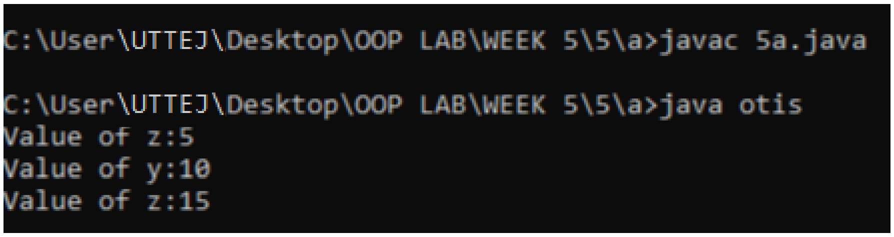


```
C:\User\UTTEJ\Desktop\OOP LAB\WEEK 5\4>javac 4.java

C:\User\UTTEJ\Desktop\OOP LAB\WEEK 5\4>java bank
Enter accountant name:
venu
Enter Account Number:
123789
Enter amount
10000
Enter withdraw amount
2000
Enter deposit amount
4000
Account Details:
Accountant Name:venu
Account Number:123789
Total amount after deposited 4000.0 is:14000.0
Total amount after withdrawn 2000.0 is:12000.0
Current Balance in Account:14000.0
```

5. Write Java program for the following
a. Example for this operator and the use of this keyword.

```
class maeve
{
        maeve(int x,int y,int z)
        {
                this(x,y);
                System.out.println("Value of z:"+z);
        }
        maeve(int x,int y)
        {
                this(x);
                System.out.println("Value of y:"+y);
        }
        maeve(int x)
        {
                System.out.println("Value of z:"+x);
        }
}
class otis
        {
        public static void main(String args[])
        {
                maeve e=new maeve(5,10,15);
        }
}
```

OUTPUT:

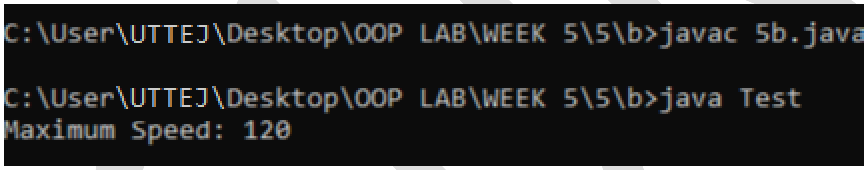


```
C:\User\UTTEJ\Desktop\OOP LAB\WEEK 5\5\a>javac 5a.java

C:\User\UTTEJ\Desktop\OOP LAB\WEEK 5\5\a>java otis
Value of z:5
Value of y:10
Value of z:15
```

b. Example for super keyword.

```
class Vehicle
{
        int maxSpeed = 120;
}
class Car extends Vehicle
{
        int maxSpeed = 180;
        void display()
        {
        System.out.println("Maximum Speed: " + super.maxSpeed);
        }
}
class Test
{
        public static void main(String[] args)
        {
                Car small = new Car();
                small.display();
        }
}
```

OUTPUT:

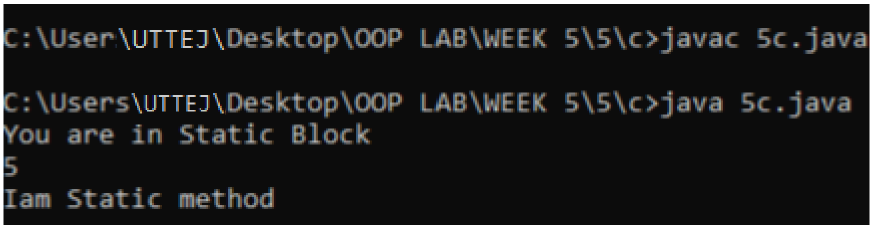


```
C:\User\UTTEJ\Desktop\OOP LAB\WEEK 5\5\b>javac 5b.java

C:\User\UTTEJ\Desktop\OOP LAB\WEEK 5\5\b>java Test
Maximum Speed: 120
```

c. Example for static variables and methods.

```
class eric
{
        static
        {
                System.out.println("You are in Static Block");
        }
        static void display()
        {
                System.out.println("Iam Static method");
        }
        static int a=5;
        public static void main(String args[])
        {
                System.out.println(eric.a);
                eric.display();
        }
}
```

OUTPUT:



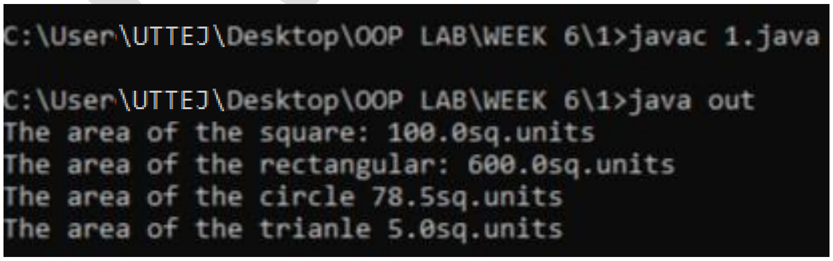
```
C:\User\UTTEJ\Desktop\OOP LAB\WEEK 5\5\c>javac 5c.java

C:\Users\UTTEJ\Desktop\OOP LAB\WEEK 5\5\c>java 5c.java
You are in Static Block
5
Iam Static method
```

WEEK- 6

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

```
class out
{
        public static void main(String a[])
        {
                shape s=new shape();
                s.area(10);
                s.area(20,30);
                s.Area(5);
                s.Area(2,5);
        }
}
class shape
{
        void area(float x)
        {
                System.out.println("The area of the square: "+x*x+"sq.units");
        }
        void area(float x,float y)
        {
                System.out.println("The area of the rectangular: "+x*y+"sq.units");
        }
        void Area(double x)
        {
                double z=3.14*x*x;
                System.out.println("The area of the circle "+z+"sq.units");
        }
        void Area(int x,int y)
        {
                float r=(x*y)/2;
                System.out.println("The area of the trianle "+r+"sq.units");
        }
}
```

OUTPUT:



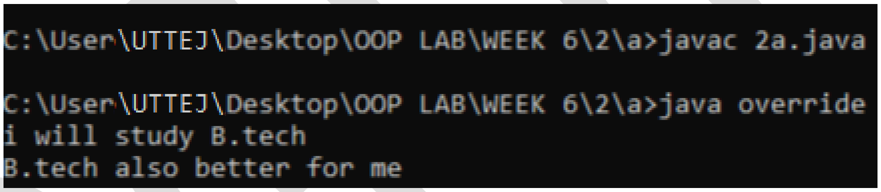
```
C:\User\UTTEJ\Desktop\OOP LAB\WEEK 6\1>javac 1.java

C:\User\UTTEJ\Desktop\OOP LAB\WEEK 6\1>java out
The area of the square: 100.0sq.units
The area of the rectangular: 600.0sq.units
The area of the circle 78.5sq.units
The area of the trianle 5.0sq.units
```

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

```
class parent
{
        void m1()
        {
```

```
		System.out.println("you must study MBA");
		}
		void m2()
		{
			System.out.println("MBA is better to you");
		}
}
class override extends parent
{
		void m1()
		{
			System.out.println("i will study B.tech");
		}
		void m2()
		{
		System.out.println("B.tech also better for me");
		}
		public static void main(String args[])
		{
			override c1= new override();
			c1.m1();
			c1.m2();
		}
}
```

OUTPUT:



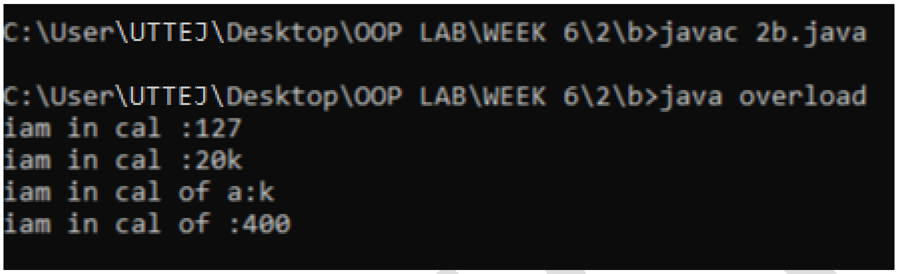
```
C:\User\UTTEJ\Desktop\OOP LAB\WEEK 6\2\a>javac 2a.java

C:\User\UTTEJ\Desktop\OOP LAB\WEEK 6\2\a>java override
i will study B.tech
B.tech also better for me
```

Overloading:

```
import java.util.Scanner;
class over
		{
		void cal(int a, char c)
		{
			System.out.println("iam in cal :"+(a+c));
			System.out.println("iam in cal :"+a+c);
		}
		void cal(char c)
		{
			System.out.println("iam in cal of a:"+c);
		}
		void cal(int a)
		{
			System.out.println("iam in cal of :"+a*a);
		}
}

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

```
        {
                over c1 = new over ();
                c1.cal(20,'k');
                c1.cal('k');
                c1.cal(20);
        }
}
```

OUTPUT:



```
C:\User\UTTEJ\Desktop\OOP LAB\WEEK 6\2\b>javac 2b.java

C:\User\UTTEJ\Desktop\OOP LAB\WEEK 6\2\b>java overload
iam in cal :127
iam in cal :20k
iam in cal of a:k
iam in cal of :400
```

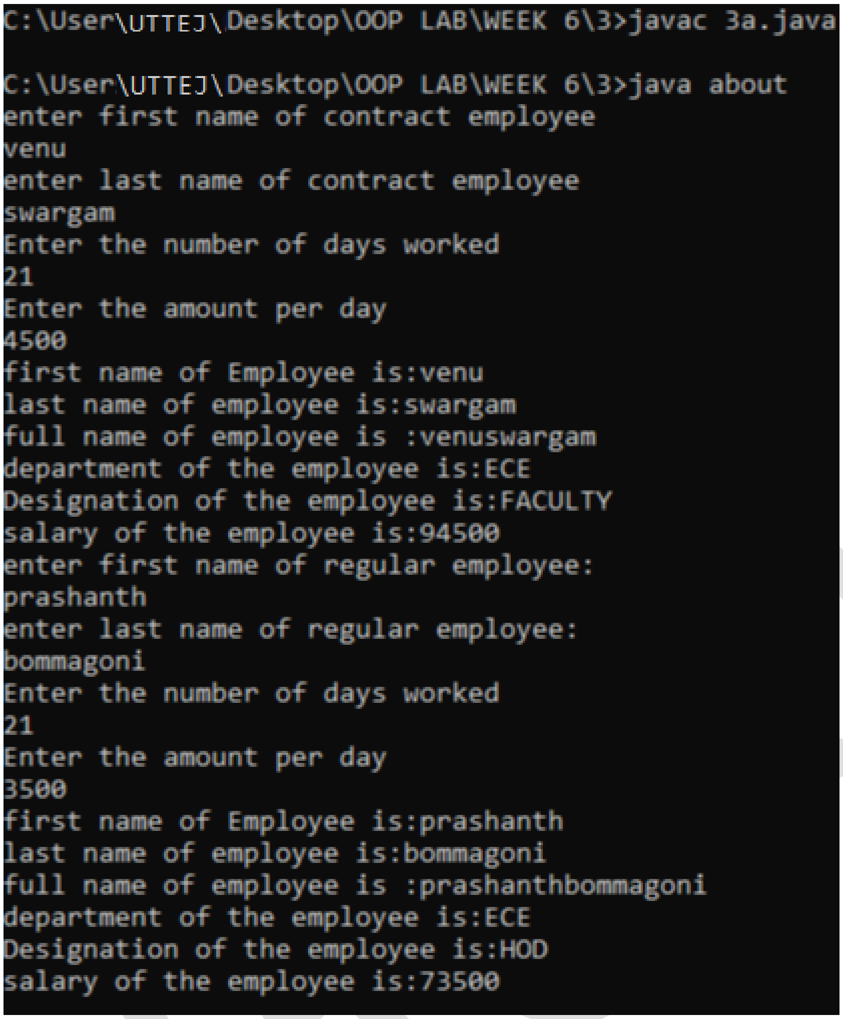
3. Write an application to create a super class Employee with information first name & last name and methods getFirstName(), getLastName() derive the sub-classes ContractEmployee and RegularEmployee with the information about department, designation & method displayFullName() , getDepartment(), getDesig()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

```
import java.util.*;
class Employee
{
        String firstName;
        String lastName;
        void getFirstName(String a){
                firstName=a;
                System.out.println("first name of Employee is:"+firstName);
        }
        void getLastName(String b){
                lastName=b;
                System.out.println("last name of employee is:"+lastName);
        }
}
class ContractEmployee extends Employee
{
        int salary;
        int d;
        int pd;
        String department;
        String designation;
        void displayFullName()
        {
                System.out.println("full name of employee is :"+firstName+lastName);
        }
        void getDepartment(String c)
        {
                department=c;
                System.out.println("department of the employee is:"+department);
        }
```

```
void getDesig(String d)
        {
                designation=d;
                System.out.println("Designation of the employee is:"+designation);
        }
        void getsalary(int j,int k){
                d=j;
                pd=k;
                salary=d*pd;
                System.out.println("salary of the employee is:"+salary);
        }
}
class RegularEmployee extends Employee
{
        int salary;
        int d;
        int pd;
        String department;
        String designation;
        void displayFullName()
        {
                System.out.println("full name of employee is :"+firstName+lastName);
        }
        void getDepartment(String c)
        {
                department=c;
                System.out.println("department of the employee is:"+department);
        }
        void getDesig(String d)
        {
                designation=d;
                System.out.println("Designation of the employee is:"+designation);
        }
        void getsalary(int l,int z){
                d=l;
                pd=z;
                salary=d*pd;
                System.out.println("salary of the employee is:"+salary);
        }
}
class about
{
        public static void main (String args[])
        {
                Scanner in=new Scanner(System.in);
                System.out.println("enter first name of contract employee");
                String m=in.next();
                System.out.println("enter last name of contract employee");
                String n=in.next();
                System.out.println("Enter the number of days worked");
                int j=in.nextInt();
                System.out.println("Enter the amount per day");
                int k=in.nextInt();
```

```
                    ContractEmployee s1=new ContractEmployee();
                    s1.getFirstName(m);

                    s1.getLastName(n);
                    s1.displayFullName();
                    s1.getDepartment("ECE");
                    s1.getDesig("FACULTY");
                    s1.getsalary(j,k);
                    RegularEmployee s2=new RegularEmployee();
                    System.out.println("enter first name of regular employee:");
                    String h=in.next();
                    System.out.println("enter last name of regular employee:");
                    String i=in.next();
                    System.out.println("Enter the number of days worked");
                    int l=in.nextInt();
                    System.out.println("Enter the amount per day");
                    int z=in.nextInt();
                    s2.getFirstName(h);
                    s2.getLastName(i);
                    s2.displayFullName();
                    s2.getDepartment("ECE");
                    s2.getDesig("HOD");
                    s2.getsalary(l,z);
        }
}
```

OUTPUT:

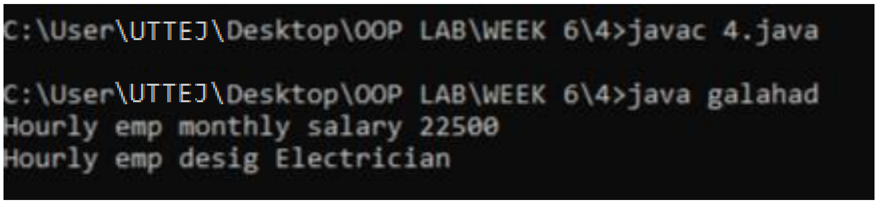


```
C:\User\UTTEJ\Desktop\OOP LAB\WEEK 6\3>javac 3a.java

C:\User\UTTEJ\Desktop\OOP LAB\WEEK 6\3>java about
enter first name of contract employee
venu
enter last name of contract employee
swargam
Enter the number of days worked
21
Enter the amount per day
4500
first name of Employee is:venu
last name of employee is:swargam
full name of employee is :venuswargam
department of the employee is:ECE
Designation of the employee is:FACULTY
salary of the employee is:94500
enter first name of regular employee:
prashanth
enter last name of regular employee:
bommagoni
Enter the number of days worked
21
Enter the amount per day
3500
first name of Employee is:prashanth
last name of employee is:bommagoni
full name of employee is :prashanthbommagoni
department of the employee is:ECE
Designation of the employee is:HOD
salary of the employee is:73500
```

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.

```
class contractemployee
{
        String desig;
        double salary;
        public void setDesignation(String desig)
        {
                this.desig=desig;
        }
        public String getdesignation()
        {
                return desig;
        }
}
class hourlyemp extends contractemployee
{
        int hours,wageph;
        hourlyemp(String design,int hours,int wageph)
        {
                super.setDesignation(design);
                this.hours=hours;
                this.wageph=wageph;
        }
        int calwages()
        {
                return hours*wageph*30;
        }
        public String getDesignation()
        {
                return super.desig;
        }
}
class Weeklyemp extends contractemployee
{
        int weeks,wagespwk;
        Weeklyemp(String design,int weeks,int wagepwk)
        {
                super.setDesignation(design);
                this.weeks=weeks;
                this.wagespwk=wagespwk;
        }
        int calwages()
        {
                return weeks*wagespwk*30;
        }
        public String getDesignation()
        {
                return super.desig;
        }
}
class galahad


{
        public static void main(String a[])
        {
                hourlyemp he=new hourlyemp("Electrician",10,75);
                System.out.println("Hourly emp monthly salary "+he.calwages());
                System.out.println("Hourly emp desig "+he.getdesignation());
        }
}
```

OUTPUT:



```
C:\User\UTTEJ\Desktop\OOP LAB\WEEK 6\4>javac 4.java

C:\User\UTTEJ\Desktop\OOP LAB\WEEK 6\4>java galahad
Hourly emp monthly salary 22500
Hourly emp desig Electrician
```

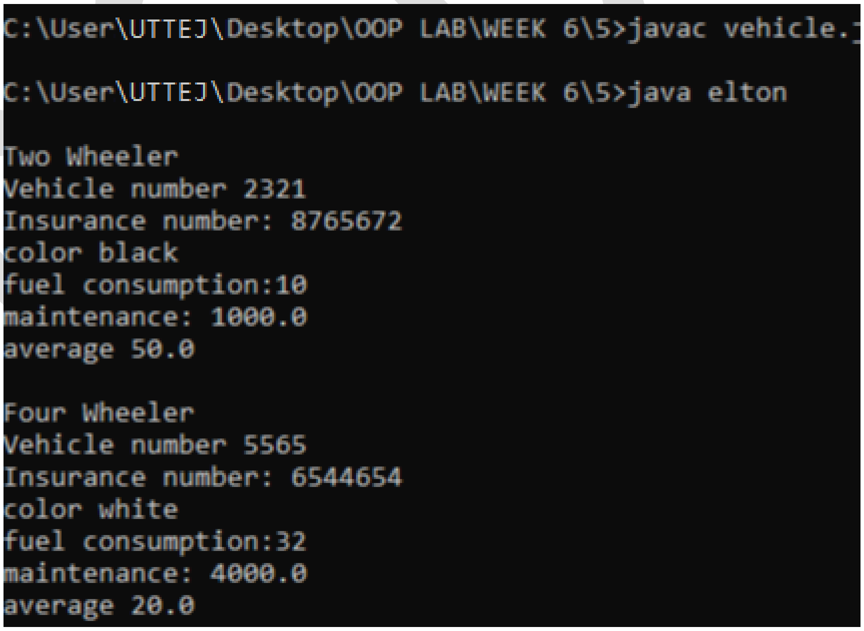
5. Write an application to create a super class Vehicle with information vehicle number,insurance number,color and methods getConsumption() displayConsumption().

Derive the sub-classes TwoWheeler and FourWheeler with method maintenance() and average() to print the maintenance And average of vehicle.

```
import java.util.*;
class vehicle{
        int v;
        int i;
        String c;
        void info(int v,int i,String c){
                this.v=v;
                this.i=i;
                this.c=c;
        }
                void getconsumption(int f){
                System.out.println();
                }
                void displayconsumption(int f) {
                        System.out.println("fuel consumption:"+f);
                }
                void displayinfo(){
                        System.out.println("Vehicle number "+v);
                        System.out.println("Insurance number: "+i);
                        System.out.println("color "+c);
        }
}
class twowheeler extends vehicle{
        double a;
        double m;
        void setg(double a,double m){
                this.a=a;
                this.m=m;
        }
        double getaverage(){
                return a;
        }
        double getmaintenance(){
                return m;
        }
}

class fourwheeler extends vehicle{
        double a;
        double m;
        void seth(double a,double m){
                this.a=a;
                this.m=m;
        }
        double getaverage(){
                return a;
        }
        double getmaintenance(){
                return m;
        }
}
```

```
class elton{
        public static void main(String args[]){
                twowheeler s1=new twowheeler();
                s1.info(2321,8765672,"black");
                s1.getconsumption(12);
                s1.setg(50,1000);
                double m=s1.getmaintenance();
                double a=s1.getaverage();
                System.out.println("Two Wheeler");
                s1.displayinfo();
                s1.displayconsumption(10);
                System.out.println("maintenance: "+m);
                System.out.println("average "+a);
                fourwheeler s2=new fourwheeler();
                s2.info(5565,6544654,"white");
                s2.getconsumption(32);
                s2.seth(20,4000);
                double m1=s2.getmaintenance();
                double a1=s2.getaverage();
                System.out.println("Four Wheeler");
                s2.displayinfo();
                s2.displayconsumption(32);
                System.out.println("maintenance: "+m1);
                System.out.println("average "+a1);
        }
}
```

OUTPUT:



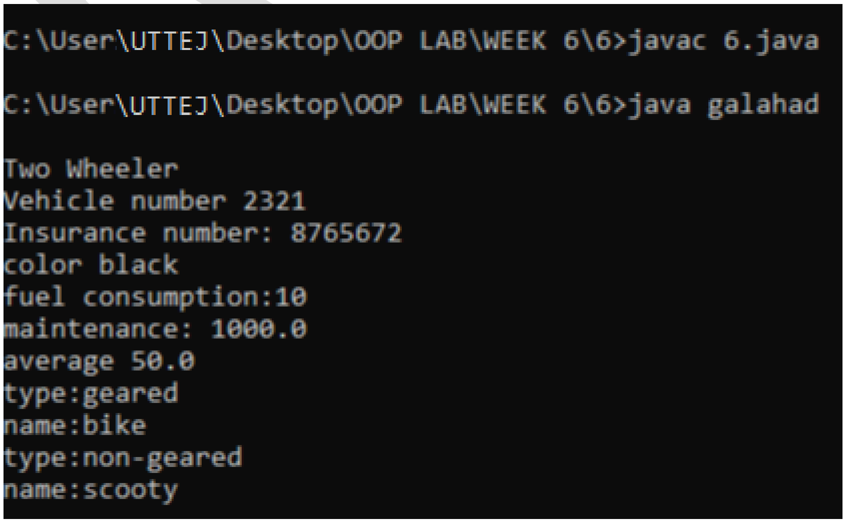
```
C:\User\UTTEJ\Desktop\OOP LAB\WEEK 6\5>javac vehicle.

C:\User\UTTEJ\Desktop\OOP LAB\WEEK 6\5>java elton

Two Wheeler
Vehicle number 2321
Insurance number: 8765672
color black
fuel consumption:10
maintenance: 1000.0
average 50.0

Four Wheeler
Vehicle number 5565
Insurance number: 6544654
color white
fuel consumption:32
maintenance: 4000.0
average 20.0
```

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 non-geared two wheeler.

```
import java.util.*;
class vehicle{
        int v;
```

```
    int i;
    String c;
    void info(int v,int i,String c){
        this.v=v;
        this.i=i;
        this.c=c;
    }
        void getconsumption(int f){
        System.out.println();
        }
        void displayconsumption(int f) {
            System.out.println("fuel consumption:"+f);
        }
        void displayinfo(){
            System.out.println("Vehicle number "+v);
            System.out.println("Insurance number: "+i);
            System.out.println("color "+c);
    }
}
class twowheeler extends vehicle{
    double a;
    double m;
    void setg(double a,double m){
        this.a=a;
        this.m=m;
    }
    double getaverage(){
        return a;
    }

double getmaintenance(){
        return m;
    }
}
class geared extends twowheeler{
    String type;
    String name;
    geared(String type,String name){
        this.type=type;
        this.name=name;
    }
    String getType(){
        return type;
    }
    String getName(){
        return name;
    }
}
class nongeared extends twowheeler{
    String type;
    String name;
    nongeared(String type,String name){
        this.type=type;
        this.name=name;
    }
    String getType(){
        return type;
    }
    String getName(){
        return name;
    }
}
class galahad{
    public static void main(String args[]){
        twowheeler s1=new twowheeler();
        s1.info(2321,8765672,"black");
        s1.getconsumption(12);
        s1.setg(50,1000);
        double m=s1.getmaintenance();
        double a=s1.getaverage();
        System.out.println("Two Wheeler");
        s1.displayinfo();
        s1.displayconsumption(10);
        System.out.println("maintenance: "+m);
        System.out.println("average "+a);
        geared s2=new geared("geared","bike");
        String t=s2.getType();
        String n=s2.getName();
        System.out.println("type:"+t);
        System.out.println("name:"+n);
        nongeared s3=new nongeared("non-geared","scooty");
        String u=s3.getType();
        String v=s3.getName();
        System.out.println("type:"+u);
        System.out.println("name:"+v);


}

}
```

OUTPUT:



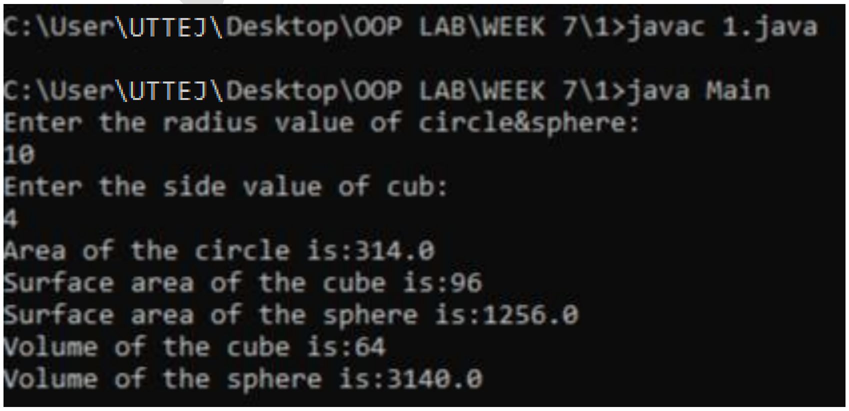
```
C:\User\UTTEJ\Desktop\OOP LAB\WEEK 6\6>javac 6.java

C:\User\UTTEJ\Desktop\OOP LAB\WEEK 6\6>java galahad

Two Wheeler
Vehicle number 2321
Insurance number: 8765672
color black
fuel consumption:10
maintenance: 1000.0
average 50.0
type:geared
name:bike
type:non-geared
name:scooty
```

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.**

```
import java.util.Scanner;
abstract class Shape
{
abstract void getArea(int r,int a);
abstract void getVolume(int r,int a);
}
class Calculations extends Shape
{
        public void getArea(int r,int a)
        {
                System.out.println("Area of the circle is:" +(3.14*r*r));
                System.out.println("Surface area of the cube is:"+(6*a*a));
                System.out.println("Surface area of the sphere is:"+(4*3.14*r*r));
        }
        public void getVolume(int r,int a)
        {
                System.out.println("Volume of the cube is:" +(a*a*a));
                System.out.println("Volume of the sphere is:"+((4/3)*3.14*r*r*r));
        }
}
class Main
        {
        public static void main(String args[])
        {
                Scanner sc=new Scanner(System.in);
                System.out.println("Enter the radius value of circle&sphere:");
                int R=sc.nextInt();
                System.out.println("Enter the side value of cub:");
                int A=sc.nextInt();

                Shape sh=new Calculations();
                sh.getArea(R,A);
                sh.getVolume(R,A);
        }
}
```

OUTPUT:

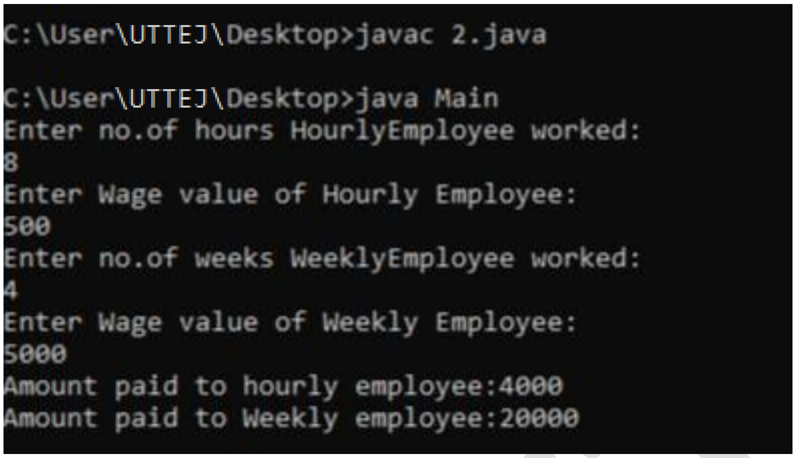


```
C:\User\UTTEJ\Desktop\OOP LAB\WEEK 7\1>javac 1.java

C:\User\UTTEJ\Desktop\OOP LAB\WEEK 7\1>java Main
Enter the radius value of circle&sphere:
10
Enter the side value of cub:
4
Area of the circle is:314.0
Surface area of the cube is:96
Surface area of the sphere is:1256.0
Volume of the cube is:64
Volume of the sphere is:3140.0
```

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.

```
import java.util.Scanner;
abstract class Employee
{
        abstract void getAmount(int m,int n);
}
class HE extends Employee
{
        public void getAmount(int m,int wage)
        {
                System.out.println("Amount paid to hourly employee:"+(m*wage));
        }
}
class WE extends Employee
{
        public void getAmount(int p,int Wage)
        {
                System.out.println("Amount paid to Weekly employee:"+(p*Wage));
        }
}
class Main
{
        public static void main(String args[])
        {
                Scanner sc=new Scanner(System.in);
                System.out.println("Enter no.of hours HourlyEmployee worked: ");
                int a=sc.nextInt();
                System.out.println("Enter Wage value of Hourly Employee:");
                int b=sc.nextInt();
                System.out.println("Enter no.of weeks WeeklyEmployee worked:");
                int c=sc.nextInt();
                System.out.println("Enter Wage value of Weekly Employee:");
                int d=sc.nextInt();
                Employee sc1= new HE();

                Employee sc2= new WE();
                sc1.getAmount(a,b);
                sc2.getAmount(c,d);
        }
}
```

OUTPUT:

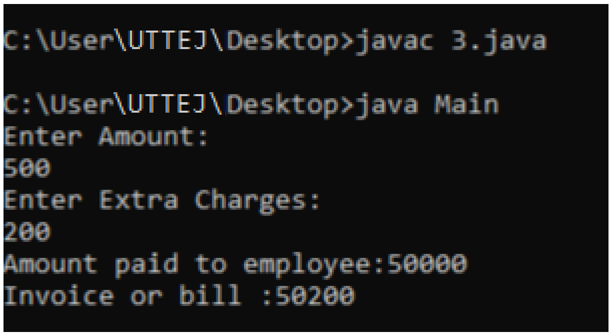


```
C:\User\UTTEJ\Desktop>javac 2.java

C:\User\UTTEJ\Desktop>java Main
Enter no.of hours HourlyEmployee worked:
8
Enter Wage value of Hourly Employee:
500
Enter no.of weeks WeeklyEmployee worked:
4
Enter Wage value of Weekly Employee:
5000
Amount paid to hourly employee:4000
Amount paid to Weekly employee:20000
```

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

```
import java.util.Scanner;
interface payable
{
        void getAmount(int m,int n);
}
class cal implements payable
{
        public void getAmount(int m,int n)
        {
        System.out.println("Amount paid to employee:"+(m*100));
        System.out.println("Invoice or bill :"+((m*100)+n));
        }
}
class Main
{
        public static void main(String args[])
        {
                Scanner sc=new Scanner(System.in);
                System.out.println("Enter Amount: ");
                int a=sc.nextInt();
                System.out.println("Enter Extra Charges: ");
                int b=sc.nextInt();
                payable pb=new cal();
                pb.getAmount(a,b);
        }
}
```

OUTPUT:



```
C:\User\UTTEJ\Desktop>javac 3.java

C:\User\UTTEJ\Desktop>java Main
Enter Amount:
500
Enter Extra Charges:
200
Amount paid to employee:50000
Invoice or bill :50200
```

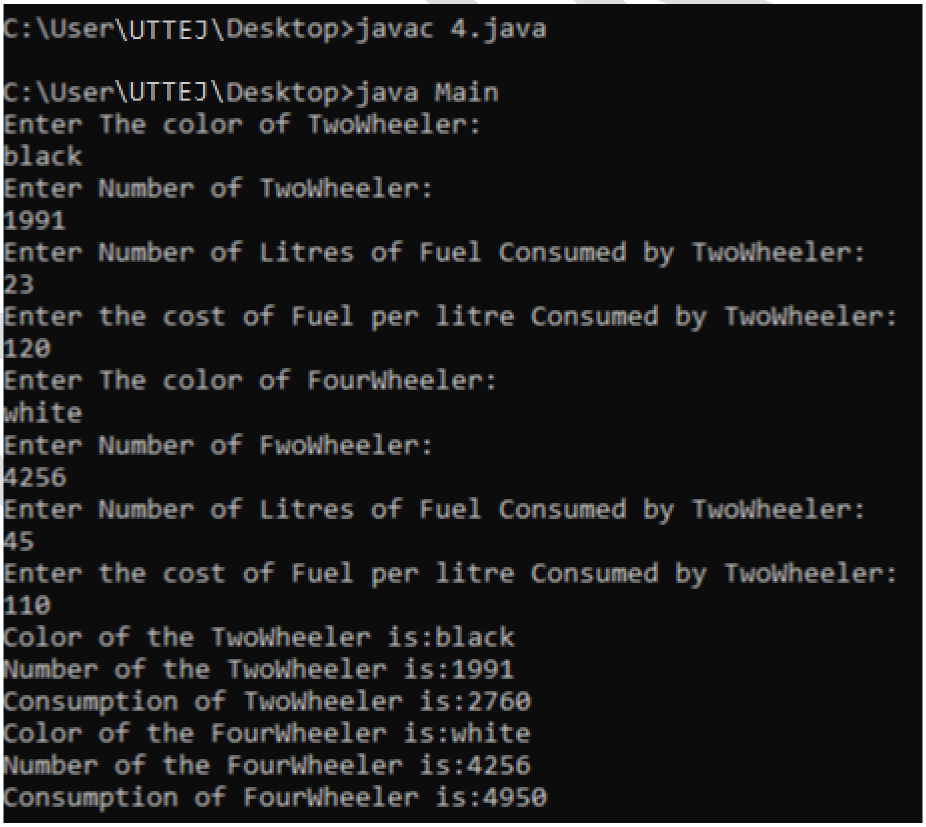
4. Create an Interface Vehicle with method getColor(),getNumber(), getConsumption()

calculate the fuel consumed, name and color for TwoWheeler and Four Wheeler By implementing interface Vehicle.

```
import java.util.Scanner;
interface Vehicle
{
        void getColor(String c);
        void getNumber(int n);
        void getConsumption(int l,int fuelcost);
}
class TwoWheeler implements Vehicle
{
        public void getColor(String c)
        {
                System.out.println("Color of the TwoWheeler is:"+c);
        }
        public void getNumber(int n)
        {
                System.out.println("Number of the TwoWheeler is:"+n);
        }
        public void getConsumption(int l,int fuelcost)
        {
                System.out.println("Consumption of TwoWheeler is:"+(l*fuelcost));
        }
}
class FourWheeler implements Vehicle
{
        public void getColor(String c)
        {
                System.out.println("Color of the FourWheeler is:"+c);
        }
        public void getNumber(int n)
        {
                System.out.println("Number of the FourWheeler is:"+n);
        }
        public void getConsumption(int l,int fuelcost)
        {
                System.out.println("Consumption of FourWheeler is:"+(l*fuelcost));
        }
}
class Main
{
        public static void main(String args[])
        {
                Scanner sc=new Scanner(System.in);

                System.out.println("Enter The color of TwoWheeler:");
                String C=sc.nextLine();
                System.out.println("Enter Number of TwoWheeler:");
                int N=sc.nextInt();
                System.out.println("Enter Number of Litres of Fuel Consumed by
TwoWheeler:");
                int L=sc.nextInt();
                System.out.println("Enter the cost of Fuel per litre Consumed by TwoWheeler:");
```

```
                    int FC=sc.nextInt();
                    System.out.println("Enter The color of FourWheeler:");
                    String C1=sc.next();
                    System.out.println("Enter Number of FwoWheeler:");
                    int N1=sc.nextInt();
                    System.out.println("Enter Number of Litres of Fuel Consumed by
TwoWheeler:");
                    int L1=sc.nextInt();
                    System.out.println("Enter the cost of Fuel per litre Consumed by TwoWheeler:");
                    int FC1=sc.nextInt();
                    Vehicle TW=new TwoWheeler();
                    Vehicle FW=new FourWheeler();
                    TW.getColor(C);
                    TW.getNumber(N);
                    TW.getConsumption(L,FC);
                    FW.getColor(C1);
                    FW.getNumber(N1);
                    FW.getConsumption(L1,FC1);
          }
}
```

OUTPUT:



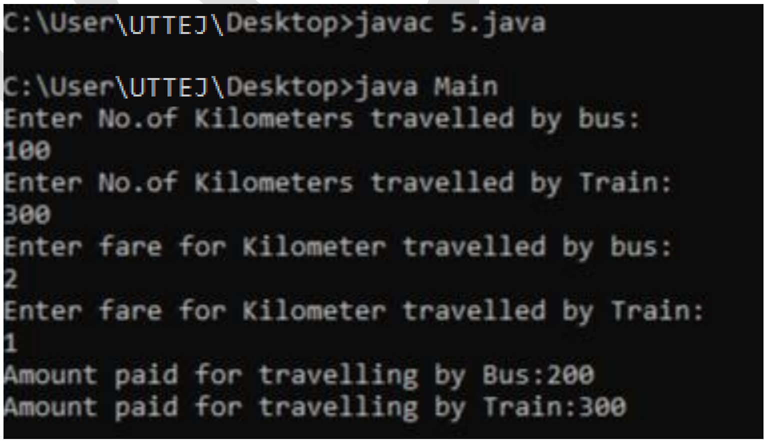
```
C:\User\UTTEJ\Desktop>javac 4.java

C:\User\UTTEJ\Desktop>java Main
Enter The color of TwoWheeler:
black
Enter Number of TwoWheeler:
1991
Enter Number of Litres of Fuel Consumed by TwoWheeler:
23
Enter the cost of Fuel per litre Consumed by TwoWheeler:
120
Enter The color of FourWheeler:
white
Enter Number of FwoWheeler:
4256
Enter Number of Litres of Fuel Consumed by TwoWheeler:
45
Enter the cost of Fuel per litre Consumed by TwoWheeler:
110
Color of the TwoWheeler is:black
Number of the TwoWheeler is:1991
Consumption of TwoWheeler is:2760
Color of the FourWheeler is:white
Number of the FourWheeler is:4256
Consumption of FourWheeler is:4950
```

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.

```
import java.util.Scanner;
interface Fare
{
          void getAmount(int K,int C);
}
class Bus implements Fare
```

```
{
        public void getAmount(int K,int C)
        {
                System.out.println("Amount paid for travelling by Bus:"+(K*C));
        }
}
class Train implements Fare
{
        public void getAmount(int K,int C)
        {
                System.out.println("Amount paid for travelling by Train:"+(K*C));
        }
}
class Main
{
        public static void main(String args[])
        {
                Scanner sc=new Scanner(System.in);
                System.out.println("Enter No.of Kilometers travelled by bus:");
                int k=sc.nextInt();
                System.out.println("Enter No.of Kilometers travelled by Train:");
                int k1=sc.nextInt();
                System.out.println("Enter fare for Kilometer travelled by bus:");
                int f=sc.nextInt();
                System.out.println("Enter fare for Kilometer travelled by Train:");
                int f1=sc.nextInt();
                Fare b=new Bus();
                Fare t=new Train();
                b.getAmount(k,f);
                t.getAmount(k1,f1);
        }
}
```

OUTPUT:



```
C:\User\UTTEJ\Desktop>javac 5.java

C:\User\UTTEJ\Desktop>java Main
Enter No.of Kilometers travelled by bus:
100
Enter No.of Kilometers travelled by Train:
300
Enter fare for Kilometer travelled by bus:
2
Enter fare for Kilometer travelled by Train:
1
Amount paid for travelling by Bus:200
Amount paid for travelling by Train:300
```

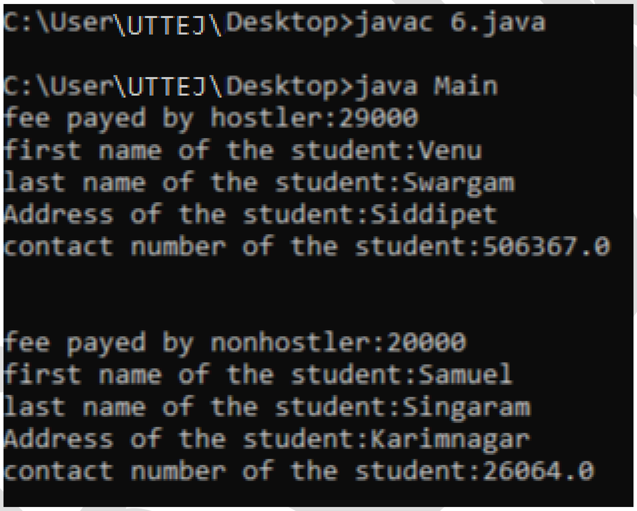
6. Create an Interface StudentFee with method getAmount(),getFirstName(),getLastName(), getAddress(), getContact(). Calculate the amount paid by the Hostler and NonHostler student by implementing interface Student Fee.

```
interface Studentfee
{
        void getAmount(int clgfee);
        void getFirstname(String fname);
```

```
    void getLastname(String lname);
    void getAddress(String address);
    void getContact(double contact);
}
class Hostler implements Studentfee
{
    public void getAmount(int clgfee)
    {
        int hostelfee=5000;
        System.out.println("fee payed by hostler:"+(clgfee+hostelfee));
    }
    public void getFirstname(String fname)
    {
        System.out.println("first name of the student:"+fname);
    }
    public void getLastname(String lname)
    {
        System.out.println("last name of the student:"+lname);
    }
    public void getAddress(String address)
    {
        System.out.println("Address of the student:"+address);
    }
    public void getContact(double contact)
    {
        System.out.println("contact number of the student:"+contact);
    }
}
class NonHostler implements Studentfee
{
    public void getAmount(int clgfee)
    {
        System.out.println("fee payed by nonhostler:"+(clgfee));
    }
    public void getFirstname(String fname)
    {
        System.out.println("first name of the student:"+fname);
    }
    public void getLastname(String lname)
    {
        System.out.println("last name of the student:"+lname);
    }
    public void getAddress(String address)
    {
        System.out.println("Address of the student:"+address);
    }
    public void getContact(double contact)
    {
        System.out.println("contact number of the student:"+contact);
    }
}
```

```
class Main
{
        public static void main(String args[])
        {
                Studentfee h= new Hostler();
                h.getAmount(24000);
                h.getFirstname("Venu");
                h.getLastname("Swargam");
                h.getAddress("Siddipet");
                h.getContact(506367);
                System.out.println("\n");
                Studentfee h1= new NonHostler();
                h1.getAmount(20000);
                h1.getFirstname("Samuel");
                h1.getLastname("Singaram");
                h1.getAddress("Karimnagar");
                h1.getContact(26064);
        }
}
```

OUTPUT:



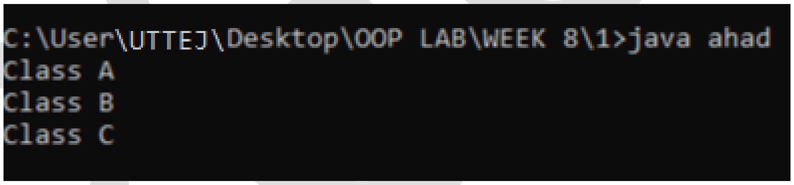
```
C:\User\UTTEJ\Desktop>javac 6.java

C:\User\UTTEJ\Desktop>java Main
fee payed by hostler:29000
first name of the student:Venu
last name of the student:Swargam
Address of the student:Siddipet
contact number of the student:506367.0


fee payed by nonhostler:20000
first name of the student:Samuel
last name of the student:Singaram
Address of the student:Karimnagar
contact number of the student:26064.0
```

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 pack;
public class A
{
public void display1()
        {
        System.out.println("Class A");
        }
}
package pack;
public class B
{
        public void display2()
        {

System.out.println("Class B");
        }
}
package pack;
public class C
{
        public void display3()
        {
                System.out.println("Class C");
        }
}
import pack.A;
import pack.B;
import pack.C;
public class ahad{
        public static void main(String args[])
        {
                A s1=new A();
                B s2=new B();
                C s3=new C();
                s1.display1();
                s2.display2();
                s3.display3();
        }
}
```

OUTPUT:



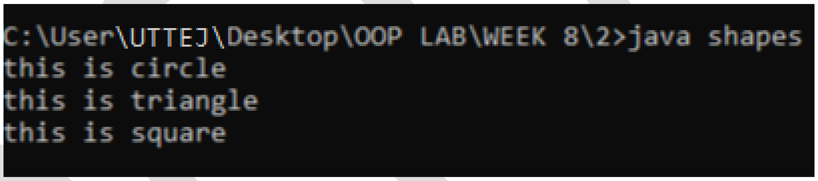
```
C:\User\UTTEJ\Desktop\OOP LAB\WEEK 8\1>java ahad
Class A
Class B
Class C
```

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 org.shapes;
public class circle
{
        public void a()
        {
                System.out.println("this is circle");
        }
}
package org.shapes;
public class square
{
        public void c()
        {
                System.out.println("this is square");
        }
}
```

```
package org.shapes;
public class triangle
{


public void b()
            {
                        System.out.println("this is triangle");
            }
}
import org.shapes.circle;
import org.shapes.triangle;
import org.shapes.square;
public class shapes
{
            public static void main(String args[])
            {
                        circle s1=new circle();
                          s1.a();
                        triangle s2=new triangle();
                          s2.b();
                        square s3=new square();
                          s3.c();
            }
}
```

OUTPUT:

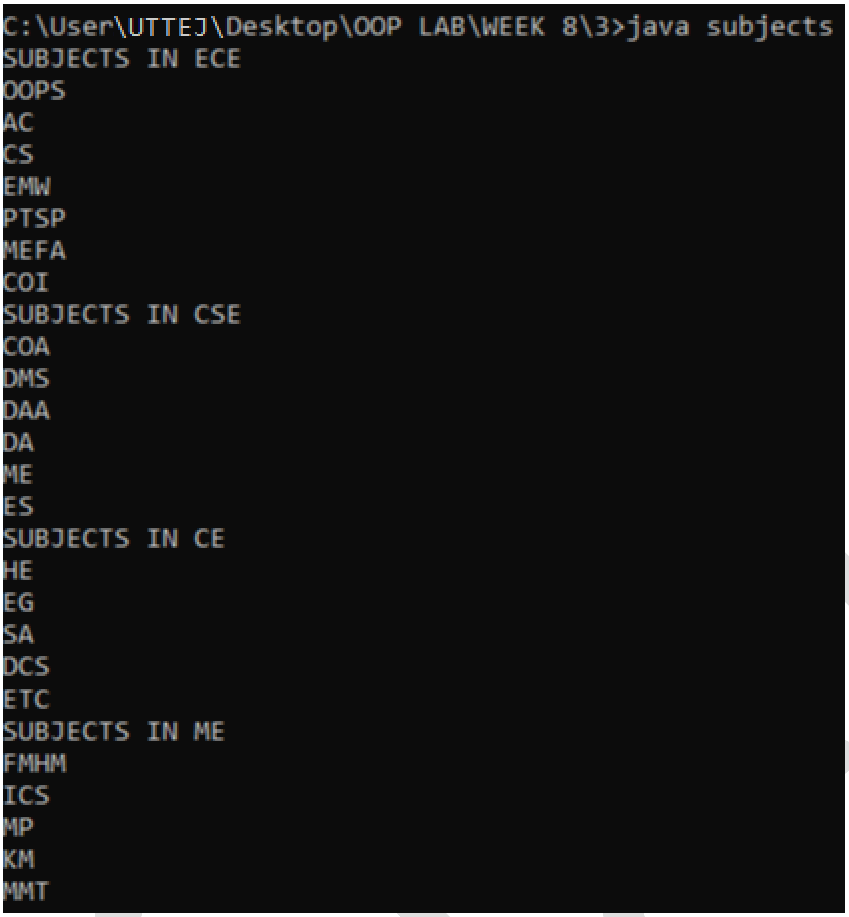


```
C:\User\UTTEJ\Desktop\OOP LAB\WEEK 8\2>java shapes
this is circle
this is triangle
this is square
```

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 dept;
public class ECE{
            public void a()
            {
                        System.out.println("SUBJECTS IN ECE");
                        System.out.println("OOPS\nAC\nCS\nEMW\nPTSP\nMEFA\nCOI");
            }
}
package dept;
public class CSE{
            public void b()
            {
                        System.out.println("SUBJECTS IN CSE");
                        System.out.println("COA\nDMS\nDAA\nDA\nME\nES");
            }
}
package dept;
public class CE{
    public void c()
    {
        System.out.println("SUBJECTS IN CE");
        System.out.println("HE\nEG\nSA\nDCS\nETC");
    }
}


package dept;
public class ME{
    public void d()
    {
        System.out.println("SUBJECTS IN ME");
        System.out.println("FMHM\nICS\nMP\nKM\nMMT");
    }
}
import dept.ECE;
import dept.CSE;
import dept.CE;
import dept.ME;
public class subjects
{
    public static void main(String args[])
    {
        ECE s1=new ECE();
        s1.a();
        CSE s2=new CSE();
        s2.b();
        CE s3=new CE();
        s3.c();
        ME s4=new ME();
        s4.d();
    }
}
```

OUTPUT:

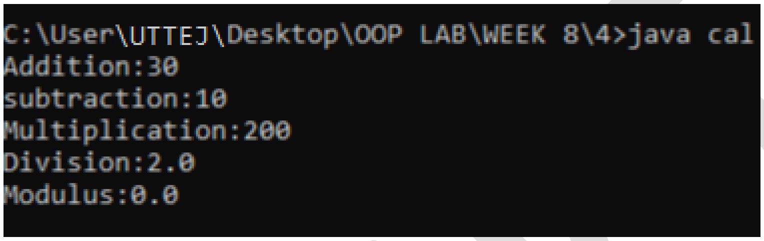


```
C:\User\UTTEJ\Desktop\OOP LAB\WEEK 8\3>java subjects
SUBJECTS IN ECE
OOPS
AC
CS
EMW
PTSP
MEFA
COI
SUBJECTS IN CSE
COA
DMS
DAA
DA
ME
ES
SUBJECTS IN CE
HE
EG
SA
DCS
ETC
SUBJECTS IN ME
FMHM
ICS
MP
KM
MMT
```

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

```
package calculator;
public class num
{
        public int add(int a,int b){
        return a+b;
        }
        public int sub(int a,int b){
        return a-b;
        }
        public int mul(int a,int b){
        return a*b;
        }
        public double div(int a,int b){
        return a/b;
        }
        public double mod(int a,int b){
        return a%b;
        }
}
import calculator.num;
public class cal
{
	public static void main(String args[])
	{
		num n=new num();
		System.out.println("Addition:"+n.add(20,10));
		System.out.println("subtraction:"+n.sub(20,10));
		System.out.println("Multiplication:"+n.mul(20,10));
		System.out.println("Division:"+n.div(20,10));
		System.out.println("Modulus:"+n.mod(20,10));
	}
}
```

OUTPUT:



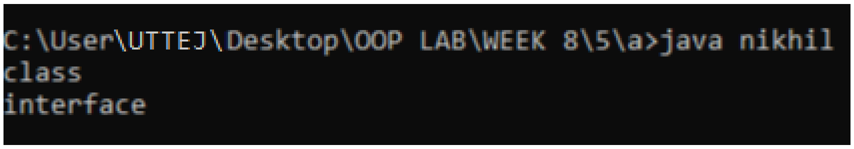
```
C:\User\UTTEJ\Desktop\OOP LAB\WEEK 8\4>java cal
Addition:30
subtraction:10
Multiplication:200
Division:2.0
Modulus:0.0
```

5. Write a program for the following

a. Example to use interfaces in Packages.

```
package nikki;
public interface n{
	public void e();
}
package nikki;
public class nik{
	public void v(){
		System.out.println("class");
	}
}


import nikki.nik;
import nikki.n;
class nikhil implements n{
	public void e(){
	System.out.println("interface");
	}
	public static void main(String args[]){
		nik s1=new nik();
		s1.v();
		nikhil s2=new nikhil();
		s2.e();
	}
}
```

OUTPUT:



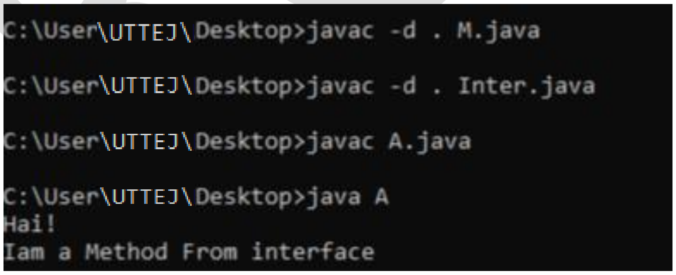
```
C:\User\UTTEJ\Desktop\OOP LAB\WEEK 8\5\a>java nikhil
class
interface
```

b. Example to create sub package in a package.

```
package Oops.java;
public class M
```

```
{
        public void m()
        {
                System.out.println("Hai!");
        }
}
package Oops.java;
public interface Inter
{
public void m1();
}
import Oops.java.M;
import Oops.java.Inter;
class A implements Inter
{
        public void m1()
        {
                System.out.println("Iam a Method From interface");
        }
        public static void main(String args[])
        {
                M c=new M();
                c.m();
                A a=new A();
                a.m1();
        }
}
```

OUTPUT:



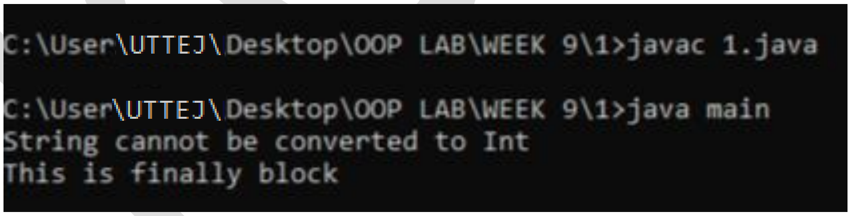
```
C:\User\UTTEJ\Desktop>javac -d . M.java

C:\User\UTTEJ\Desktop>javac -d . Inter.java

C:\User\UTTEJ\Desktop>javac A.java

C:\User\UTTEJ\Desktop>java A
Hai!
Iam a Method From interface
```

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.

```
class ThrowandThrows
{
        ThrowandThrows(String s)throws Exception
        {
                System.out.printf("%d",s);
                throw new Exception();
        }
}
class main
{
        public static void main(String args[])
        {
                try
                {
                        ThrowandThrows s1=new ThrowandThrows("oops");
                }
                catch(Exception e)
                {
                        System.out.println("String cannot be converted to Int");
                }
                finally
                {
                        System.out.println("This is finally block");
                }
        }
}
```

OUTPUT:



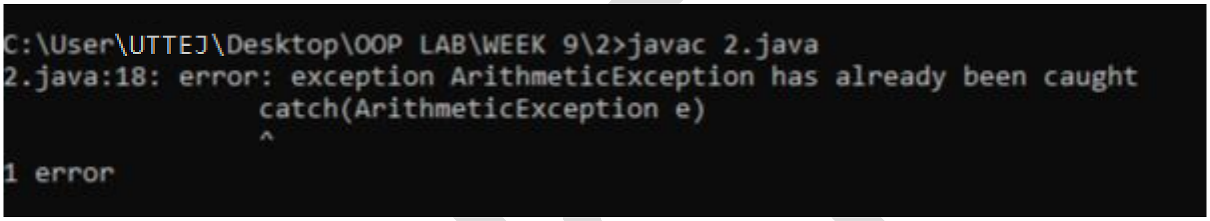
```
C:\User\UTTEJ\Desktop\OOP LAB\WEEK 9\1>javac 1.java

C:\User\UTTEJ\Desktop\OOP LAB\WEEK 9\1>java main
String cannot be converted to Int
This is finally block
```

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.

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

{
                int a=10,b=0,sum;
                try
                {
                        sum=a/b;
                        System.out.println("sum ="+sum);
     }
```

```
                    catch(Exception e)
                    {
                              System.out.println("Not a valid input");
        }
                    catch(ArithmeticException e)
        {
                              System.out.println("You can not divide something by 0");
                    }
          }
}
```

OUTPUT:

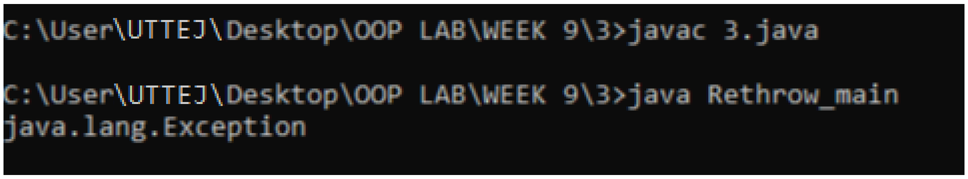


```
C:\User\UTTEJ\Desktop\OOP LAB\WEEK 9\2>javac 2.java
2.java:18: error: exception ArithmeticException has already been caught
                catch(ArithmeticException e)
                ^
1 error
```

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 .

```
class Rethrow
{
          void one() throws Exception
          {
                    try
                    {
                              two();
        }
                    catch(Exception e)
                    {
                              throw e;
        }
          }
   void two() throws Exception
          {
                    throw new Exception();
   }
}
class Rethrow_main
{
          public static void main(String []aa)
          {
                    Rethrow s1=new Rethrow();
                    try
                    {
                              s1.one();
                    }
                    catch(Exception e)
                    {
                              System.out.println(e);
                    }
          }
}
```

OUTPUT:

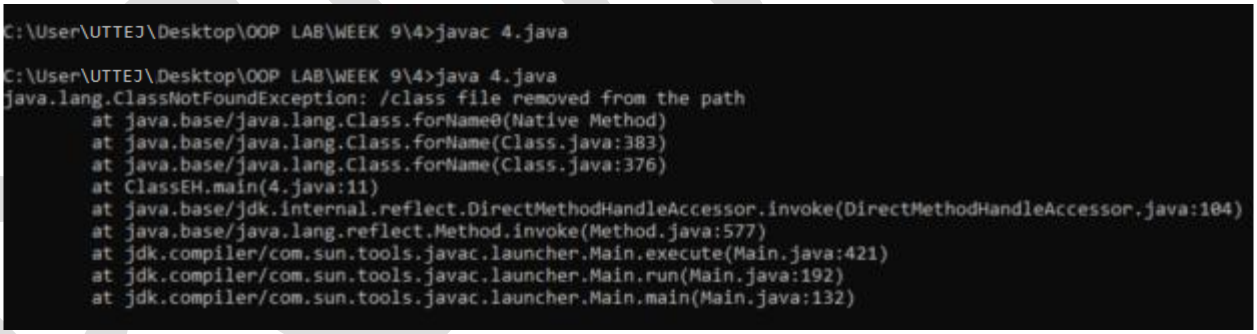


```
C:\User\UTTEJ\Desktop\OOP LAB\WEEK 9\3>javac 3.java

C:\User\UTTEJ\Desktop\OOP LAB\WEEK 9\3>java Rethrow_main
java.lang.Exception
```

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).

```
class ClassEH
{
	public static void main(String []aa)
	{
		try
		{
			Class .forName(".class file removed from the path");
		}
		catch(ClassNotFoundException e)
		{
			e.printStackTrace();
		}
	}
}
```

OUTPUT:

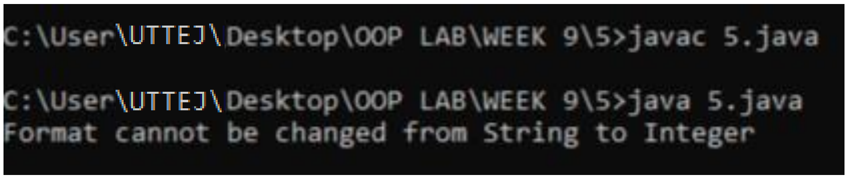


```
C:\User\UTTEJ\Desktop\OOP LAB\WEEK 9\4>javac 4.java

C:\User\UTTEJ\Desktop\OOP LAB\WEEK 9\4>java 4.java
java.lang.ClassNotFoundException: /class file removed from the path
        at java.base/java.lang.Class.forName0(Native Method)
        at java.base/java.lang.Class.forName(Class.java:383)
        at java.base/java.lang.Class.forName(Class.java:376)
        at ClassEH.main(4.java:11)
        at java.base/jdk.internal.reflect.DirectMethodHandleAccessor.invoke(DirectMethodHandleAccessor.java:104)
        at java.base/java.lang.reflect.Method.invoke(Method.java:577)
        at jdk.compiler/com.sun.tools.javac.launcher.Main.execute(Main.java:421)
        at jdk.compiler/com.sun.tools.javac.launcher.Main.run(Main.java:192)
        at jdk.compiler/com.sun.tools.javac.launcher.Main.main(Main.java:132)
```

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').

```
class NFE  // NumberFormatException is a Runtime Exception
{
	public static void main(String []aa)
	{
		String s="rgukt";
		try
		{
			System.out.printf("%d",s);
		}
		catch(Exception e)
		{

		System.out.println("Format cannot be changed from String to Integer");
		}
		}
}
```

OUTPUT:

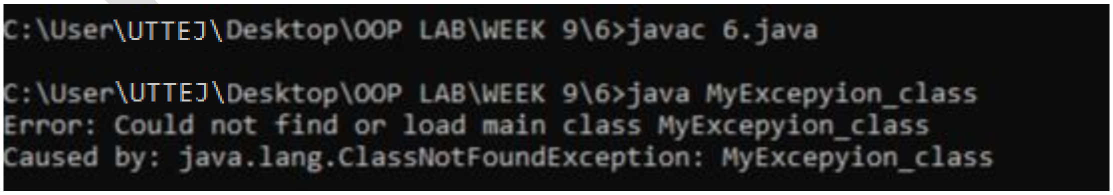


```
C:\User\UTTEJ\Desktop\OOP LAB\WEEK 9\5>javac 5.java

C:\User\UTTEJ\Desktop\OOP LAB\WEEK 9\5>java 5.java
Format cannot be changed from String to Integer
```

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.

```
class MyException extends Exception
{
	String s;
	MyException(String a)
	{
		s=a;
	}
	String Display()
	{
		return s;
	}
}
class MyException_class
{
	public static void main(String []aa)
	{
		try
		{
			throw new MyException("It is My Exception");
		}
		catch(MyException e)
		{
			System.out.println(e.Display());
		}
	}
}
```

OUTPUT:



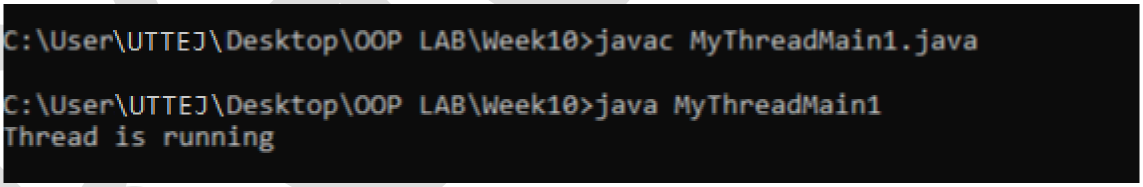
```
C:\User\UTTEJ\Desktop\OOP LAB\WEEK 9\6>javac 6.java

C:\User\UTTEJ\Desktop\OOP LAB\WEEK 9\6>java MyExcepyion_class
Error: Could not find or load main class MyExcepyion_class
Caused by: java.lang.ClassNotFoundException: MyExcepyion_class
```

WEEK -10

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

```
class MyThread implements Runnable
{
        public void run()
        {
                System.out.println("Thread is running");
                try
                {
                        Thread.sleep(5000);
                }
                catch(InterruptedException e)
                {
                        System.out.println("InterruptedException caught");
                }
        }
}
class MyThreadMain1
{
        public static void main(String[] args)
        {
                Thread t=new Thread(new MyThread(),"MyThread");
                t.start();
        }
}
```

OUTPUT:

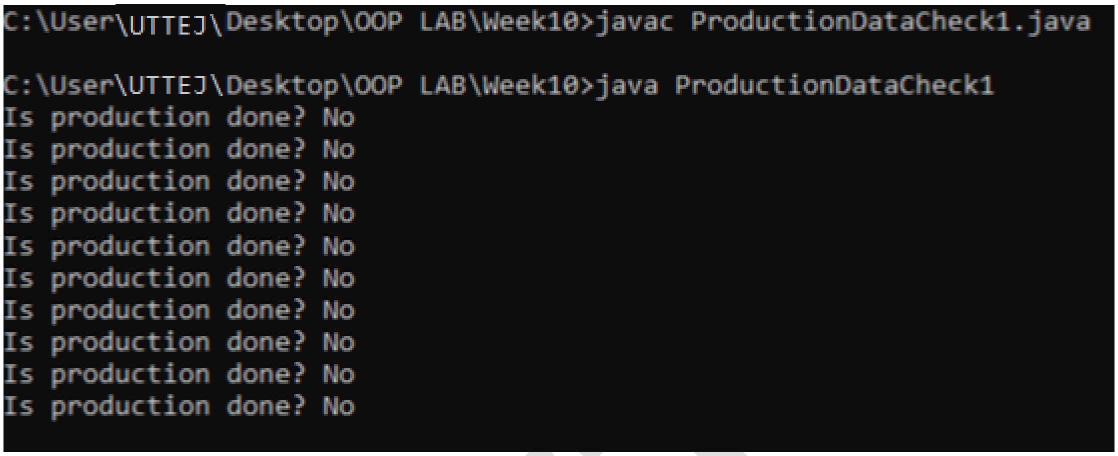


```
C:\User\UTTEJ\Desktop\OOP LAB\Week10>javac MyThreadMain1.java

C:\User\UTTEJ\Desktop\OOP LAB\Week10>java MyThreadMain1
Thread is running
```

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

```
class Producer extends Thread
{
        Producer(){}
        Producer(String n)
        {
                super(n);
        }
        public void run()
        {
                for(int i=0;i<8;i++)
                {
                        try
                        {
                                Thread.sleep(11);
                        }
                        catch(InterruptedException e)
                        {}
                }

        }
}
class Consumer extends Thread
{
        Consumer(){}
        Producer p;
        Consumer(Producer p,String n)
        {
                super(n);
                this.p=p;
        }
        public void run()
        {
                for(int i=0;i<10;i++)
                {
                        System.out.println("Is production done? "+(p.isAlive()?"No":"Yes"));
                        try
                        {
                                Thread.sleep(10);
                        }
                        catch(InterruptedException e)
                        {
                                System.out.println("consumer thread interrupted");
                        }
                }
        }
}
class ProductionDataCheck1
{
        public static void main(String args[])
        {
                Producer p=new Producer("producer");
                Consumer c=new Consumer(p,"consumer");
                p.start();
                c.start();
        }
}
```

OUTPUT:



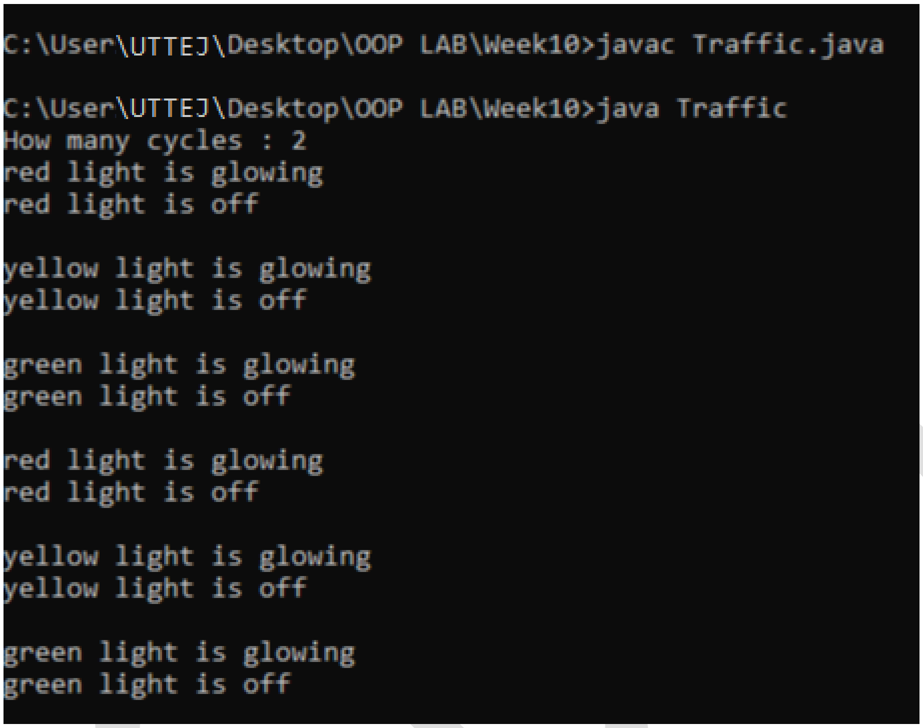
```
C:\User\UTTEJ\Desktop\OOP LAB\Week10>javac ProductionDataCheck1.java

C:\User\UTTEJ\Desktop\OOP LAB\Week10>java ProductionDataCheck1
Is production done? No
Is production done? No
Is production done? No
Is production done? No
Is production done? No
Is production done? No
Is production done? No
Is production done? No
Is production done? No
Is production done? No
```

3. 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.

```
import java.util.Scanner;
class TrafficSignal extends Thread
{
        int n;
        TrafficSignal(int n)
        {
                this.n=n;
        }
        public void run()
        {
                for (int i=0;i<n;i++)
                {
                        try
                        {
                                System.out.println("red light is glowing");
                                Thread.sleep(10000);
                                System.out.println("red light is off\n");
                        System.out.println("yellow light is glowing");
                                Thread.sleep(10000);
                                System.out.println("yellow light is off\n");
                                System.out.println("green light is glowing");
                                Thread.sleep(10000);
                                System.out.println("green light is off\n");
                        }
                        catch(Exception e)
                        {
                                System.out.println(e);
                        }
                }
        }
}
class Traffic
{
        public static void main(String args[])
        {
                System.out.print("How many cycles : ");
```

```
                        int n=new Scanner(System.in).nextInt();//as we are using just one time :)
                        TrafficSignal t=new TrafficSignal(n);
                        t.start();
            }
}
```

OUTPUT:



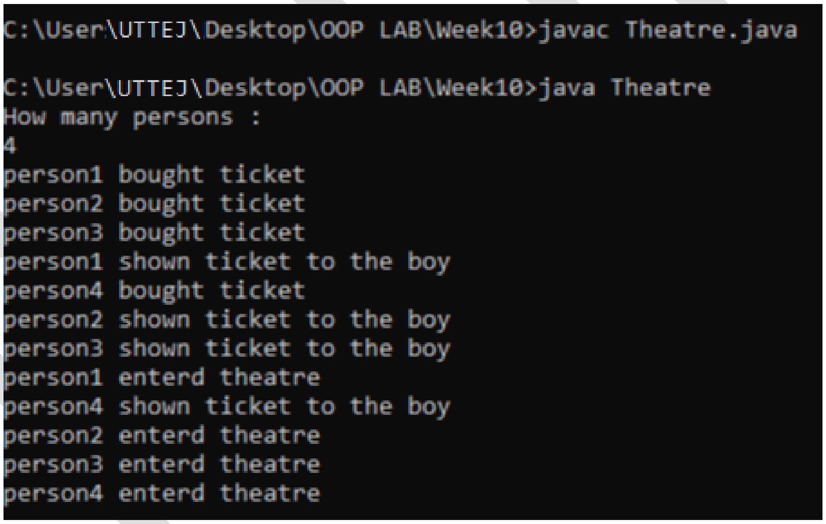
```
C:\User\UTTEJ\Desktop\OOP LAB\Week10>javac Traffic.java

C:\User\UTTEJ\Desktop\OOP LAB\Week10>java Traffic
How many cycles : 2
red light is glowing
red light is off

yellow light is glowing
yellow light is off

green light is glowing
green light is off

red light is glowing
red light is off

yellow light is glowing
yellow light is off

green light is glowing
green light is off
```

4. 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 seuence of entry into theater. Note: The person should enter only after getting a ticket and showing it to the boy.

```
import java.util.Scanner;
class Person extends Thread
{
  int n;
  Person(int n)
  {
    this.n=n;
  }
  public void run()
  {
    try
    {
      System.out.println("person"+n+" bought ticket");
      Thread.sleep(2000);
      System.out.println("person"+n+" shown ticket to the boy");
      Thread.sleep(2000);
      System.out.println("person"+n+" enterd theatre");
    }
    catch(InterruptedException e)
    {}
```

```
   }
}
class Theatre
{
   public static void main(String args[])
   {
      Scanner in=new Scanner(System.in);

     System.out.println("How many persons : ");
      int n=in.nextInt();
      Person p;
      for(int i=1;i<=n;i++)
      {
       p=new Person(i);
       p.start();
        try
        {
        Thread.sleep(900);
        }
        catch(Exception e){}
      }
   }
}
```

OUTPUT:



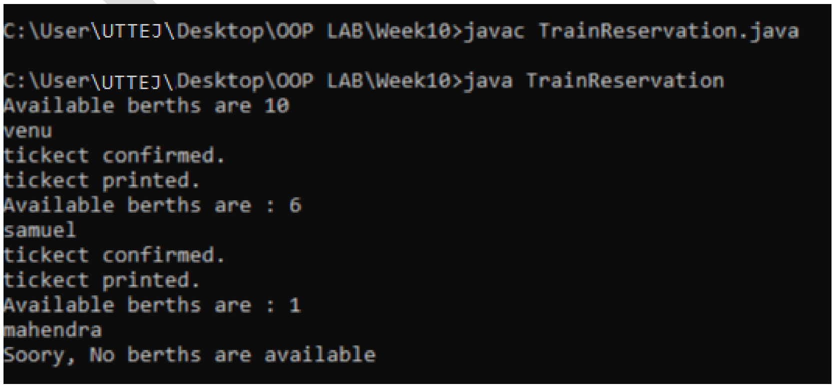
```
C:\User\UTTEJ\Desktop\OOP LAB\Week10>javac Theatre.java

C:\User\UTTEJ\Desktop\OOP LAB\Week10>java Theatre
How many persons :
4
person1 bought ticket
person2 bought ticket
person3 bought ticket
person1 shown ticket to the boy
person4 bought ticket
person2 shown ticket to the boy
person3 shown ticket to the boy
person1 enterd theatre
person4 shown ticket to the boy
person2 enterd theatre
person3 enterd theatre
person4 enterd theatre
```

5. 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 availabe 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.

```
import java.util.Scanner;
class Berths
{
            static int avlBerths=10;
            static{System.out.println("Available berths are "+avlBerths);}
            static synchronized void berthAllotment(int reqBerths)
            {
```

```
        System.out.println(Thread.currentThread().getName());
        if(reqBerths<=avlBerths)
        {
            avlBerths=avlBerths-reqBerths;
            System.out.println("tickect confirmed.");
            System.out.println("tickect printed.");
            System.out.println("Available berths are : "+avlBerths);
        }

else
            System.out.println("Soory, No berths are available");
    }
}
class Person extends Thread
{
    int reqBerths;
    Person(int reqBerths,String personName)
    {
        super(personName);
        this.reqBerths=reqBerths;
    }
    public void run()
    {
        Berths.berthAllotment(reqBerths);
    }
}
class TrainReservation
{
    public static void main(String[] args)
    {
        Person p1=new Person(4,"venu");
        Person p2=new Person(5,"mahendra");
        Person p3=new Person(5,"samuel");
        p1.start();
        p2.start();
        p3.start();
    }
}
```

OUTPUT:



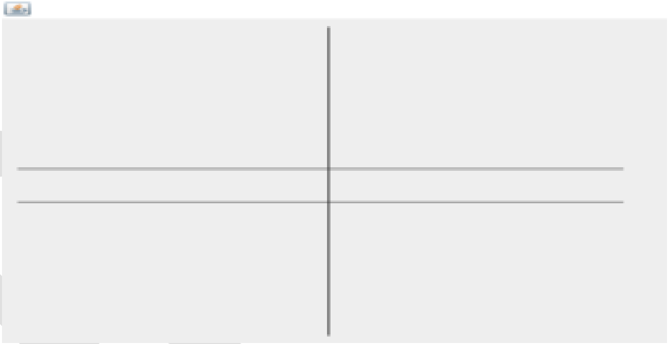
```
C:\User\UTTEJ\Desktop\OOP LAB\Week10>javac TrainReservation.java

C:\User\UTTEJ\Desktop\OOP LAB\Week10>java TrainReservation
Available berths are 10
venu
tickect confirmed.
tickect printed.
Available berths are : 6
samuel
tickect confirmed.
tickect printed.
Available berths are : 1
mahendra
Soory, No berths are available
```

<u>WEEK</u>- 11:

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.

```
import java.awt.*;
import javax.swing.*;
import java.awt.geom.Line2D;
class MyCanvas extends JComponent {
        public void paint(Graphics g)
        {
                g.drawLine(10, 180, 380, 180);
                g.drawLine(10, 220, 380, 220);
                g.drawLine(200, 10, 200, 380);
        }
}
class DrawLines {
        public static void main(String[] a)
        {
                JFrame window = new JFrame();
                window.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
                window.setBounds(400, 400, 400, 400);
                window.getContentPane().add(new MyCanvas());
                window.setVisible(true);
        }
}
```

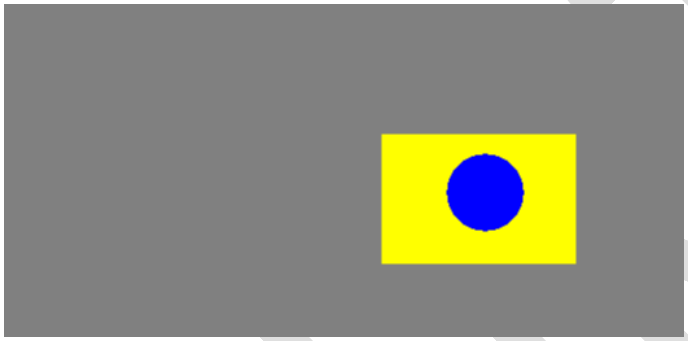
OUTPUT:



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

```
import java.awt.*;
import java.awt.event.*;
class awt6 extends Frame {
    awt6() {
          this.addWindowListener(new WindowAdapter()
          {
               public void windowClosing(WindowEvent e)
                {
                     System.exit(0);
                 }
          });
    }
   public void paint(Graphics g)
    {
            this.setBackground(Color.gray);
            g.setColor(Color.yellow);
            g.fillRect(300,200,150,100);
            g.setColor(Color.blue);
            g.fillOval(350,215,60,60);
              }
public static void main(String args[])
            {
                awt6 h=new awt6();
            h.setVisible(true);
}
}
```

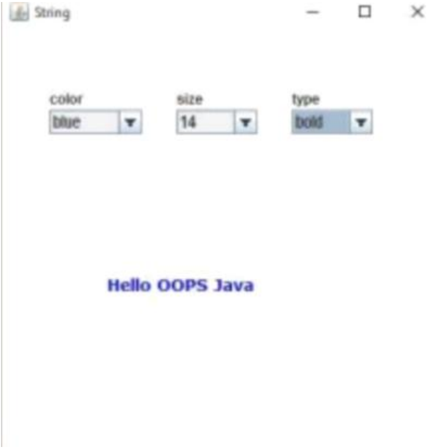
OUTPUT:



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.

```
import java.applet.Applet;
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
public class Q3 extends Applet
{
public static JLabel l,l1,l2,l3;
public static JComboBox cb,cb1,cb2;
public static void main(String[] args){
Frame f = new Frame("String");
f.setSize(400, 400);
String colors[]={"select","red","blue","green"};
String sizes[]={"select","14","15","16"};
String type[]={"select","bold","italic","normal"};
l=new JLabel("Hello OOPS Java");
l.setBounds(100,200,300,100);
l1=new JLabel("color");
l1.setBounds(50,80,80,20);
l2=new JLabel("size");
l2.setBounds(160,80,80,20);
l3=new JLabel("type");
l3.setBounds(260,80,80,20);
cb=new JComboBox(colors);
cb.setBounds(50,100,80,20);
cb1=new JComboBox(sizes);
cb1.setBounds(160,100,70,20);
cb2=new JComboBox(type);
cb2.setBounds(260,100,70,20);
f.add(cb);
f.add(cb1);
f.add(cb2);
f.add(l);
f.add(l1);
f.add(l2);
f.add(l3);
cb.addActionListener(new ActionListener(){
@Override
public void actionPerformed(ActionEvent e){
if (cb.getSelectedItem()=="red") {
l.setForeground(Color.red);
}else if (cb.getSelectedItem()=="blue") {
l.setForeground(Color.blue);
}else if(cb.getSelectedItem()=="green"){
l.setForeground(Color.green);
} } });
cb2.addActionListener(new ActionListener(){
@Override
public void actionPerformed(ActionEvent e){
if (cb2.getSelectedItem()=="bold") {
l.setFont(new Font("Verdana", Font.BOLD, 14));
}if (cb2.getSelectedItem()=="italic") {
l.setFont(new Font("Verdana", Font.ITALIC, 15));
} if(cb2.getSelectedItem()=="normal"){
l.setFont(new Font("Verdana", Font.PLAIN, 16));
} } });
cb1.addActionListener(new ActionListener(){
@Override
public void actionPerformed(ActionEvent e){
if (cb1.getSelectedItem()=="14") {
l.setFont(new Font("Verdana", Font.PLAIN, 14));
}else if (cb1.getSelectedItem()=="15") {
l.setFont(new Font("Verdana", Font.PLAIN, 15));
}else if(cb1.getSelectedItem()=="16"){
l.setFont(new Font("Verdana", Font.PLAIN, 16));
} } });
f.setLayout(null);
f.setVisible(true);
f.addWindowListener(new WindowAdapter() {
public void windowClosing(WindowEvent e) {
System.exit(0);
}});}}
```

OUTPUT:



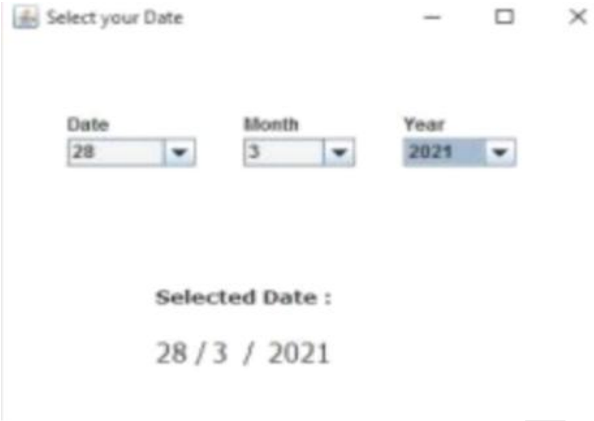
Hello OOPS Java

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.

```
import java.applet.Applet;
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
public class Q4 extends Applet
{
public static void main(String[] args){
Frame f = new Frame("Select your Date");
f.setSize(400, 400);
Integer days[]={01,02,03,04,05,06,07,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31};
Integer months[]={01,02,03,04,05,06,07,8,9,10,11,12};
Integer years[]=new Integer[61];
int k=0;
for (int i=1990; i<=2050;i++ ) {
years[k]=i;
k++;
}
l1=new JLabel("Date");
l1.setBounds(50,80,80,20);
l2=new JLabel("Month");
l2.setBounds(160,80,80,20);
l3=new JLabel("Year");
l3.setBounds(260,80,80,20);
cb=new JComboBox(days);
cb.setBounds(50,100,80,20);
cb1=new JComboBox(months);
cb1.setBounds(160,100,70,20);
cb2=new JComboBox(years);
cb2.setBounds(260,100,70,20);
l=new JLabel("Selected Date :");
l.setBounds(105,160,200,100);
l.setFont(new Font("Verdana", Font.BOLD, 13));
d1=new JLabel("01");
d1.setBounds(105,200,300,100);
d1.setFont(new Font("Verdana", Font.PLAIN, 16));
d4=new JLabel("/");
d4.setBounds(130,200,300,100);
d4.setFont(new Font("Verdana", Font.PLAIN, 16));
d2=new JLabel("01");
d2.setBounds(140,200,300,100);
d2.setFont(new Font("Verdana", Font.PLAIN, 16));
d5=new JLabel("/");
d5.setBounds(160,200,300,100);
d5.setFont(new Font("Verdana", Font.PLAIN, 16));
d3=new JLabel("2000");
d3.setBounds(175,200,300,100);
d3.setFont(new Font("Verdana", Font.PLAIN, 16));
f.add(cb);
f.add(cb1);
f.add(cb2);
f.add(l1);
f.add(l2);
f.add(l3);
f.add(d1);
f.add(d2);
f.add(d3);
f.add(d4);
f.add(d5);
f.add(l);
cb.addActionListener(new ActionListener(){
@Override
public void actionPerformed(ActionEvent e){
int date=(int)cb.getSelectedItem();
d1.setText(String.valueOf(date));
}
});
cb1.addActionListener(new ActionListener(){
@Override
public void actionPerformed(ActionEvent e){
int day=(int)cb1.getSelectedItem();
d2.setText(String.valueOf(day));
}
});
cb2.addActionListener(new ActionListener(){
@Override
public void actionPerformed(ActionEvent e){
int yr=(int)cb2.getSelectedItem();
d3.setText(String.valueOf(yr));
}
});
f.setLayout(null);
f.setVisible(true);

f.addWindowListener(new WindowAdapter() {
public void windowClosing(WindowEvent e) {
System.exit(0);
}
});
}
```

```
}
```

OUTPUT:

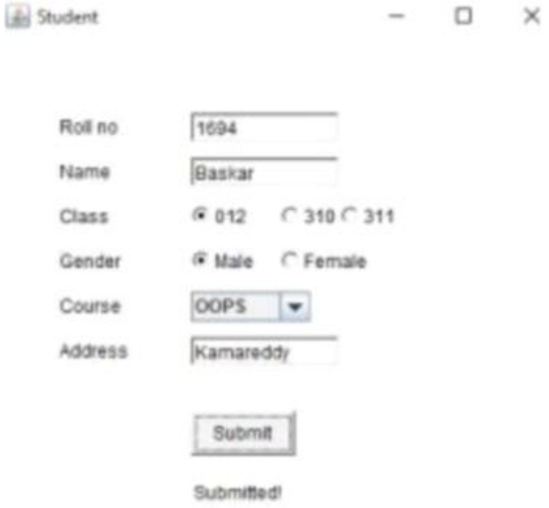


```
Date      Month     Year
28        3         2021

Selected Date :
28 / 3  /  2021
```

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.

```
import java.applet.Applet;
import java.awt.event.*;
import java.awt.*;
import javax.swing.*;
public class Q5 extends Applet{
public static Label l,l1,l2,l3,l4,l5,l6;
public static TextField t1,t2,t3,t4;
public static JComboBox cb;
public static void main(String[] args) {
Frame f=new Frame("Student");
l1=new Label("Roll no");
l1.setBounds(50,80,50,20);
l2=new Label("Name");
l2.setBounds(50,110,50,20);
l3=new Label("Class");
l3.setBounds(50,140,50,20);
l4=new Label("Gender");
l4.setBounds(50,170,50,20);
l5=new Label("Course");
l5.setBounds(50,200,50,20);
l6=new Label("Address");
l6.setBounds(50,230,50,20);
String course[]={"Ds","Daa","OOPS","c"};
cb=new JComboBox(course);

cb.setBounds(140,200,80,20);
t1=new TextField();
t1.setBounds(140,80,100,20);
t2=new TextField();
t2.setBounds(140,110,100,20);
t3=new TextField();
t3.setBounds(140,230,100,20);
CheckboxGroup cbg=new CheckboxGroup();
Checkbox box1=new Checkbox("012",false,cbg);
Checkbox box2=new Checkbox("310",false,cbg);
Checkbox box3=new Checkbox("311",false,cbg);
box1.setBounds(140,140,40,20);
box2.setBounds(200,140,40,20);
box3.setBounds(240,140,40,20);
CheckboxGroup cbg1=new CheckboxGroup();
Checkbox box4=new Checkbox("Male",false,cbg1);
Checkbox box5=new Checkbox("Female",false,cbg1);
box4.setBounds(140,170,60,20);
box5.setBounds(200,170,60,20);
Button b=new Button("Submit");
b.setBounds(140,280,70,30);
l=new Label("Submitted!");
l.setBounds(140,320,60,30);
l.setVisible(false);
b.addActionListener(new ActionListener(){
@Override
public void actionPerformed(ActionEvent e){
l.setVisible(true);
}
});
f.add(l1);
f.add(l2);
f.add(l3);
f.add(l4);
f.add(l6);
f.add(t1);
f.add(t2);
f.add(box1);
f.add(box2);
f.add(box3);
f.add(box4);
f.add(box5);
f.add(l5);
f.add(cb);
f.add(t3);
f.add(b);
f.add(l);
f.setSize(400,400);
f.setLayout(null);
f.setVisible(true);
f.addWindowListener(new WindowAdapter() {
public void windowClosing(WindowEvent e) {
System.exit(0);
}
}); } }
```

OUTPUT:



Roll no 1694

Name Baskar

Class ● 012 ○ 310 ○ 311

Gender ● Male ○ Female

Course OOPS

Address Kamareddy

Submitted!

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

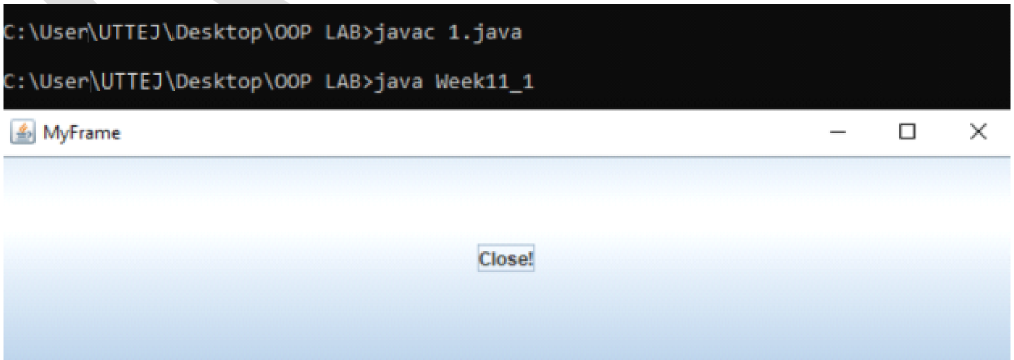
```
import java.applet.Applet;
import java.awt.event.*;
import java.awt.*;
import javax.swing.*;
public class Q6 extends Applet{
Q6(){
o display a calculator using grid Layout (You do not have to
provide
Frame f=new Frame("Calculator");
Button b1=new Button("7");
Button b2=new Button("8");
Button b3=new Button("9");
Button b4=new Button("/");
Button b5=new Button("4");
Button b6=new Button("5");
Button b7=new Button("6");
Button b8=new Button("x");
Button b9=new Button("1");
Button b10=new Button("2");
Button b11=new Button("3");
Button b12=new Button("-");
Button b13=new Button("=");
Button b14=new Button("0");
Button b15=new Button(".");
Button b16=new Button("+");
f.add(b1);
f.add(b2);f.add(b3);f.add(b4);f.add(b5);
f.add(b6);f.add(b7);f.add(b8);f.add(b9);f.add(b10);
f.add(b11);f.add(b12);f.add(b13);f.add(b14);
f.add(b15);f.add(b16);

f.setLayout(new GridLayout(3,3,3,3));
f.setSize(250,250);
f.setVisible(true);
f.addWindowListener(new WindowAdapter() {
public void windowClosing(WindowEvent e) {
System.exit(0);
} }); }
public static void main(String[] args) {
new Calculator();
}}
```

WEEK- 12:

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.

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
class MyFrame extends JFrame {
    MyFrame() {
        JButton button = new JButton("Close!");
        button.setBounds(100, 250, 50, 50);
        button.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent ae) {
                System.exit(0);
            }
        });
        add(button);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setTitle("MyFrame");
        setBounds(50, 200, 200, 200);
        setVisible(true);
    }
}
class Week11_1 {
    public static void main(String[] args) {
        new MyFrame();
    }
}
```

OUTPUT:



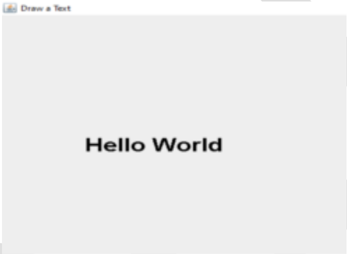
```
C:\User\UTTEJ\Desktop\OOP LAB>javac 1.java

C:\User\UTTEJ\Desktop\OOP LAB>java Week11_1
```

2.Display text in the frame by overriding PaintComponent() method of Jpanel class.

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

```
class Draw_Text extends JPanel
{
        public static void main(String args[])
        {
                JFrame frame=new JFrame("Draw a Text");
                frame.setSize(500,500);
                frame.getContentPane().setBackground(Color.white);
                frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
                Draw_Text obj=new Draw_Text();
                frame.add(obj);
                frame.setVisible(true);
        }
        @Override
        public void paintComponent(Graphics g)
        {
                g.setColor(Color.black);
                Font myFont=new Font("TimesRoman",Font.BOLD,30);
                g.setFont(myFont);
                g.drawString("Hello World",100,225);
        }
}
```

OUTPUT:

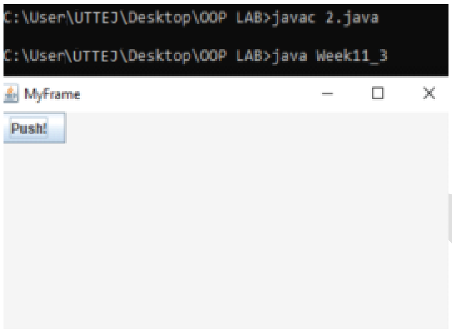


```
Hello World
```

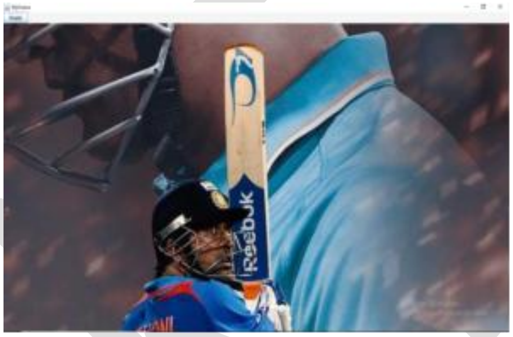
3. Write a program to create a push button , when the button is clicked an image is displayed in the frame.

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
class MyFrame extends JFrame {
    MyFrame() {
        JButton button = new JButton("Push!");
        button.setBounds(100, 250, 50, 50);
        button.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent ae) {
                ImageIcon icon = new ImageIcon("C:/Users/Venu/Desktop/pictures/7041211.jpg");
                JLabel label = new JLabel(icon);
                add(label);
                pack();
            }
        });
        add(button);

setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setTitle("MyFrame");
        setLayout(new BoxLayout(getContentPane(), BoxLayout.Y_AXIS));
        setBounds(50, 200, 430, 330);
        setVisible(true);
    }
}
class Week11_3 {
    public static void main(String[] args) {
        new MyFrame();
    }
}
```

OUTPUT:



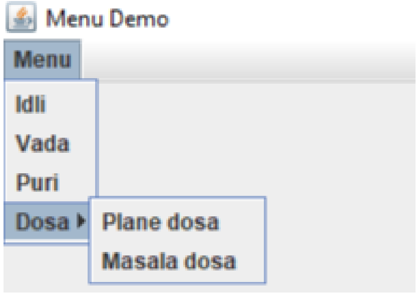
```
C:\User\UTTEJ\Desktop\OOP LAB>javac 2.java

C:\User\UTTEJ\Desktop\OOP LAB>java Week11_3
```



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

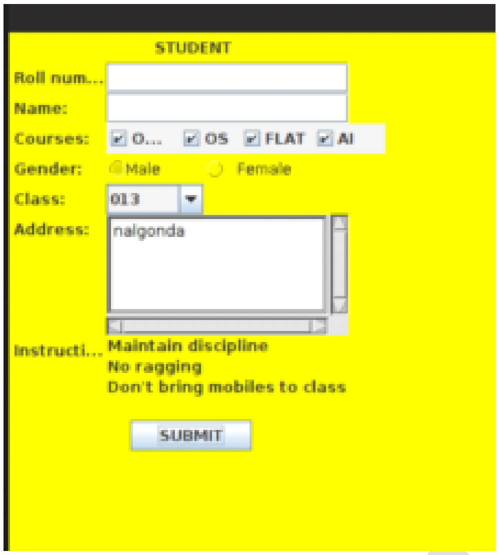
```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
class MyFrame extends JFrame {
    MyFrame() {
        JMenuBar menuBar = new JMenuBar();
        JMenu menu = new JMenu("Menu");
        menu.add(new JMenuItem("Idli"));
        menu.add(new JMenuItem("Vada"));
        menu.add(new JMenuItem("Puri"));
        JMenu subMenu = new JMenu("Dosa");
        subMenu.add(new JMenuItem("Plane dosa"));
        subMenu.add(new JMenuItem("Masala dosa"));
        menu.add(subMenu);
        menuBar.add(menu);
        setJMenuBar(menuBar);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setTitle("Menu Demo");
        setLayout(null);
        setBounds(50, 200, 300, 400);
        setVisible(true);
    }
}
class Week11_4 {
    public static void main(String[] args) {
        new MyFrame();
    }
}
```

OUTPUT:



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

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
class StudentForm extends JFrame {
    StudentForm() {
        JLabel stdLabel = new JLabel("STUDENT");
        stdLabel.setBounds(120, 0, 100, 25);
        add(stdLabel);
        JLabel rnLabel = new JLabel(" Roll number: ");
        rnLabel.setBounds(0, 25, 80, 25);
        add(rnLabel);
        JTextField rnTF = new JTextField(20);
        rnTF.setBounds(80, 25, 200, 25);
        add(rnTF);
        JLabel nameLabel = new JLabel(" Name: ");
        nameLabel.setBounds(0, 50, 80, 25);
        add(nameLabel);
        JTextField nameTF = new JTextField(20);
        nameTF.setBounds(80, 50, 200, 25);
        add(nameTF);
        JLabel courseLabel = new JLabel(" Courses: ");
        courseLabel.setBounds(0, 75, 80, 25);
        add(courseLabel);
        JCheckBox courseCB1 = new JCheckBox("OOPS");
        courseCB1.setBounds(80, 75, 60, 25);
        add(courseCB1);
        JCheckBox courseCB2 = new JCheckBox("OS");
        courseCB2.setBounds(140, 75, 50, 25);
        add(courseCB2);
        JCheckBox courseCB3 = new JCheckBox("FLAT");
        courseCB3.setBounds(190, 75, 60, 25);
        add(courseCB3);
        JCheckBox courseCB4 = new JCheckBox("AI");
        courseCB4.setBounds(250, 75, 60, 25);
        add(courseCB4);
        JLabel genderLabel = new JLabel(" Gender: ");
        genderLabel.setBounds(0, 100, 80, 25);
        add(genderLabel);
        CheckboxGroup cbg = new CheckboxGroup();
        Checkbox genderCB1 = new Checkbox("Male", cbg, true);
        genderCB1.setBounds(80, 100, 60, 25);
        add(genderCB1);
        Checkbox genderCB2 = new Checkbox("Female", cbg, false);
        genderCB2.setBounds(160, 100, 100, 25);
        add(genderCB2);
        JLabel classLabel = new JLabel(" Class: ");
        classLabel.setBounds(0, 125, 80, 25);
        add(classLabel);
        String[] classes = {"013", "309", "311"};
        JComboBox classComboBox = new JComboBox(classes);
        classComboBox.setBounds(80, 125, 80, 25);
        add(classComboBox);
        JLabel addrLabel = new JLabel(" Address: ");
        addrLabel.setBounds(0, 150, 80, 25);
        add(addrLabel);
        TextArea addrTA = new TextArea();
        addrTA.setBounds(80, 150, 200, 100);
        add(addrTA);
        JLabel iLabel = new JLabel(" Instructions: ");
        iLabel.setBounds(0, 250, 80, 25);
        add(iLabel);
        String[] instructions = {"Maintain discipline", "No ragging", "Don't bring mobiles to class"};
        JList iList = new JList(instructions);
        iList.setBounds(80, 250, 200, 60);
        iList.setBackground(Color.YELLOW);
        add(iList);
        JButton submitButton = new JButton("SUBMIT");
        submitButton.setBounds(100, 320, 100, 25);
        add(submitButton);
        setTitle("STUDENT");
        getContentPane().setBackground(Color.YELLOW);
        setLayout(null);
        setSize(300, 400);
        setDefaultCloseOperation(EXIT_ON_CLOSE);
        setVisible(true);
    }
}
class Week11_5 {
    public static void main(String[] args) {
                        new StudentForm();
             }
}
```



STUDENT

Roll num...

Name:

Courses: ☑ O... ☑ OS ☑ FLAT ☑ AI

Gender: ● Male ○ Female

Class: 013

Address: nalgonda

Instructi... Maintain discipline
No ragging
Don't bring mobiles to class