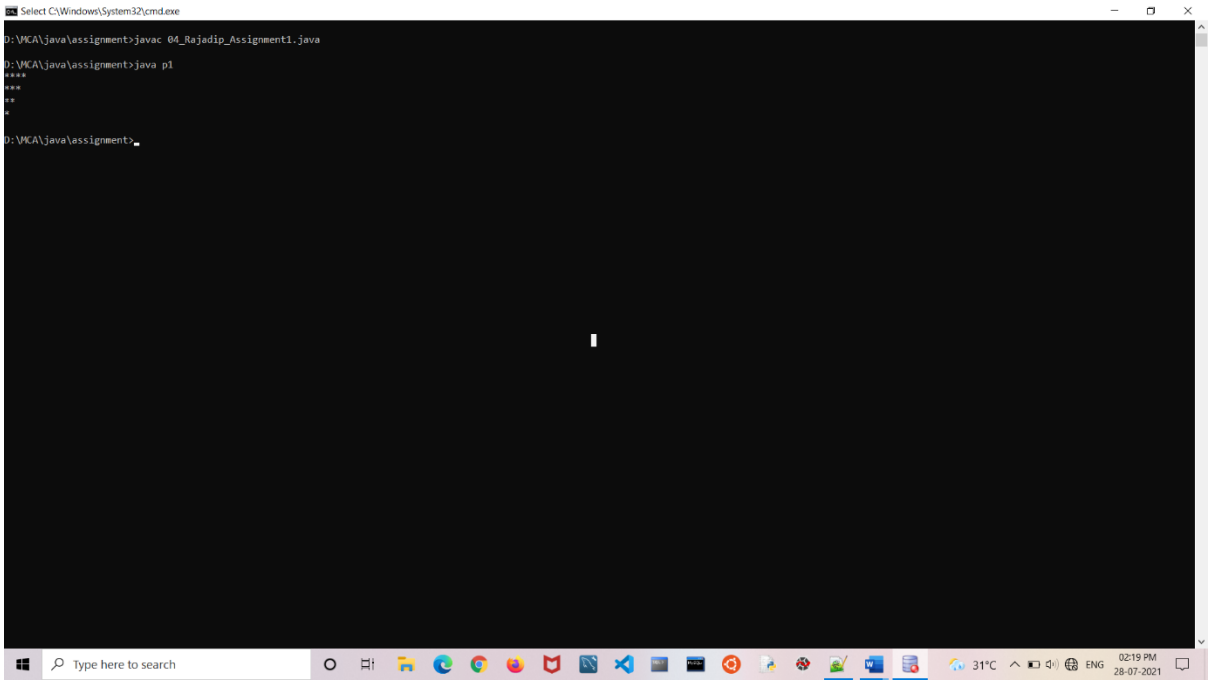
MCA sem-2

04_Rajadip_Chavda

JAVA

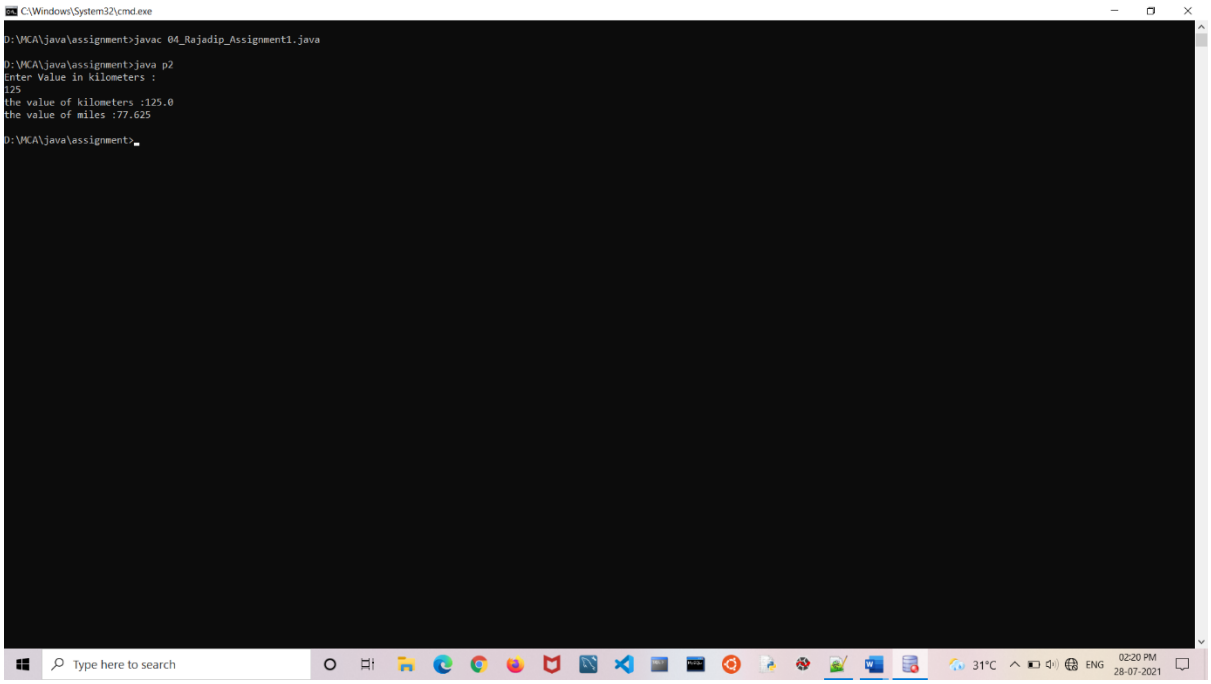
Practical Assignment -1

```
import java.util.*;
import java.lang.*;
class p1
{
	/*Write a program to print the following pattern :
	****
	***
	**
	*


	*/
	public static void main(String args[])
	{
		System.out.println("****\n***\n**\n*");
	}
}
```



```
class p2
{
	//Write a program that reads a distance in kilometers from the keyboard and output the distance as miles.
	public static void main(String args[])
	{
		Scanner p2=new Scanner(System.in);
		System.out.println("Enter Value in kilometers :");
		double value=p2.nextDouble();
		double miles=0.621;
		System.out.println("the value of kilometers :"+value);
		System.out.println("the value of miles :"+value*miles);
	}
```

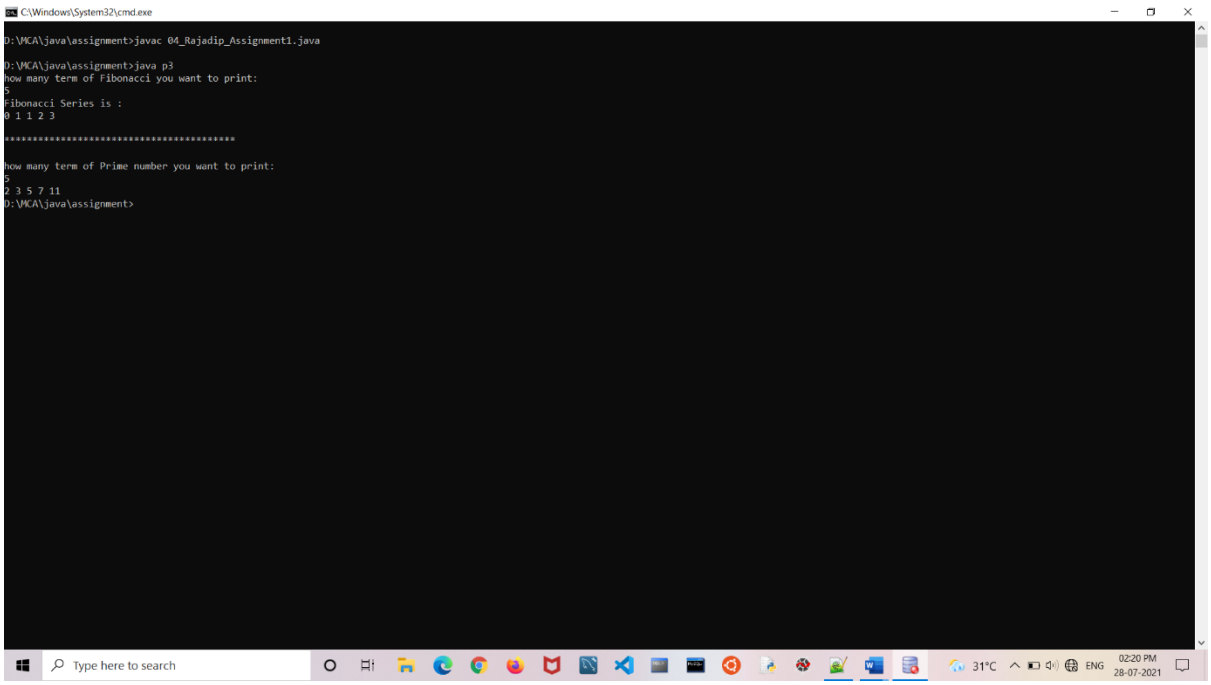
}



```
class p3
{
        //Write an application that generates the first 15 numbers in the Fibonacci series and first 15 prime numbers.
        public static void main(String args[])
        {
                Scanner p3=new Scanner(System.in);
                System.out.println("how many term of Fibonacci you want to print: ");
                int value=p3.nextInt();
                int a,b;
                int nextvalue;
```

```
        a=0;b=1;
        nextvalue=a+b;
        System.out.println("Fibonacci Series is :");
        System.out.print("0 1");
        for(int i=0;i<=value-3;i++)
        {
            System.out.print(" "+nextvalue);
            a=b;
            b=nextvalue;
            nextvalue=a+b;
        }


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


        System.out.println("how many term of Prime number you
want to print: ");
        int prime=p3.nextInt();
        int count=0,number=0,i=1,j=1;
        while(number<prime)
        {
            j=1;
            count=0;
            while(j<=i)
```

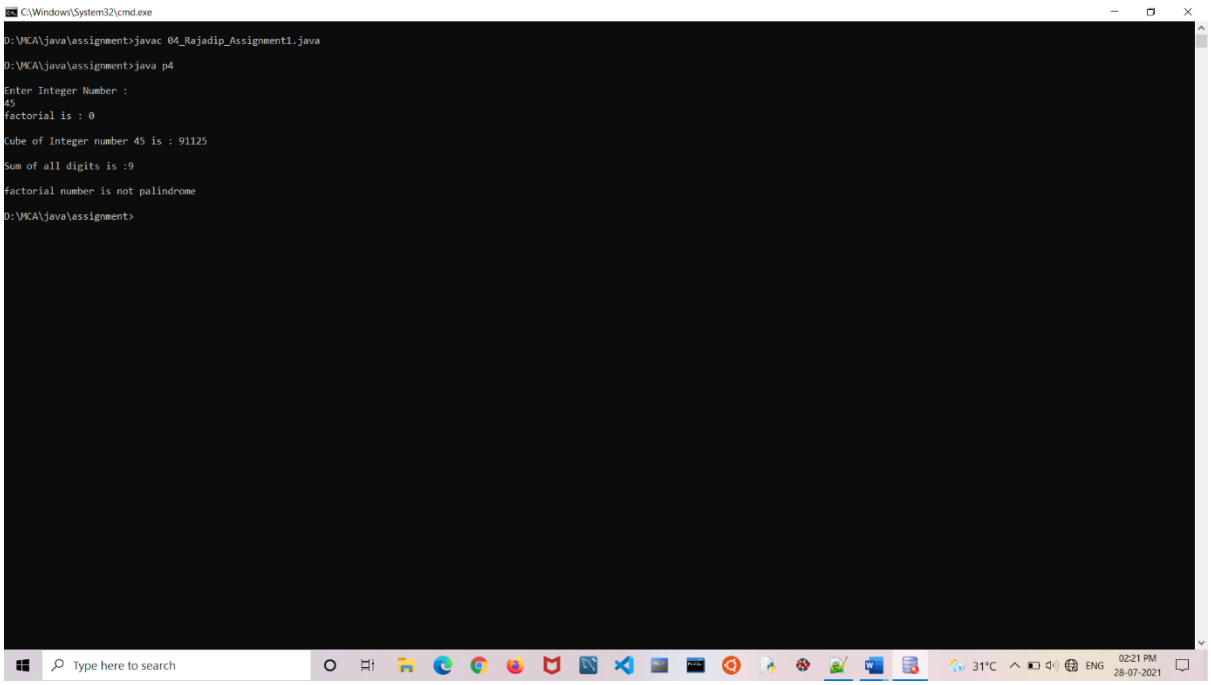
```
                {
                        if(i%j==0)
                                count++;
                        j++;
                }
                if(count==2)
                {
                        System.out.printf("%d ",i);
                        number++;
                }
                i++;
        }
    }
}
```



```
class p4
{
    //Write and run a Java program to calculate factorial and the cube of the given integer number. Also find the sum and multiplication of its digits. Also
    //check whether the no entered is palindrome or not
    static int getsum(int n)
    {
        int sum=0;
        while(n!=0)
        {
            sum=sum+n%10;
            n=n/10;
        }
        return sum;
    }
    public static void main(String args[])
    {
        Scanner p4=new Scanner(System.in);
        System.out.println("\nEnter Integer Number :");
        int number=p4.nextInt();
        int fact=1;
        for(int i=1;i<=number;i++)
```

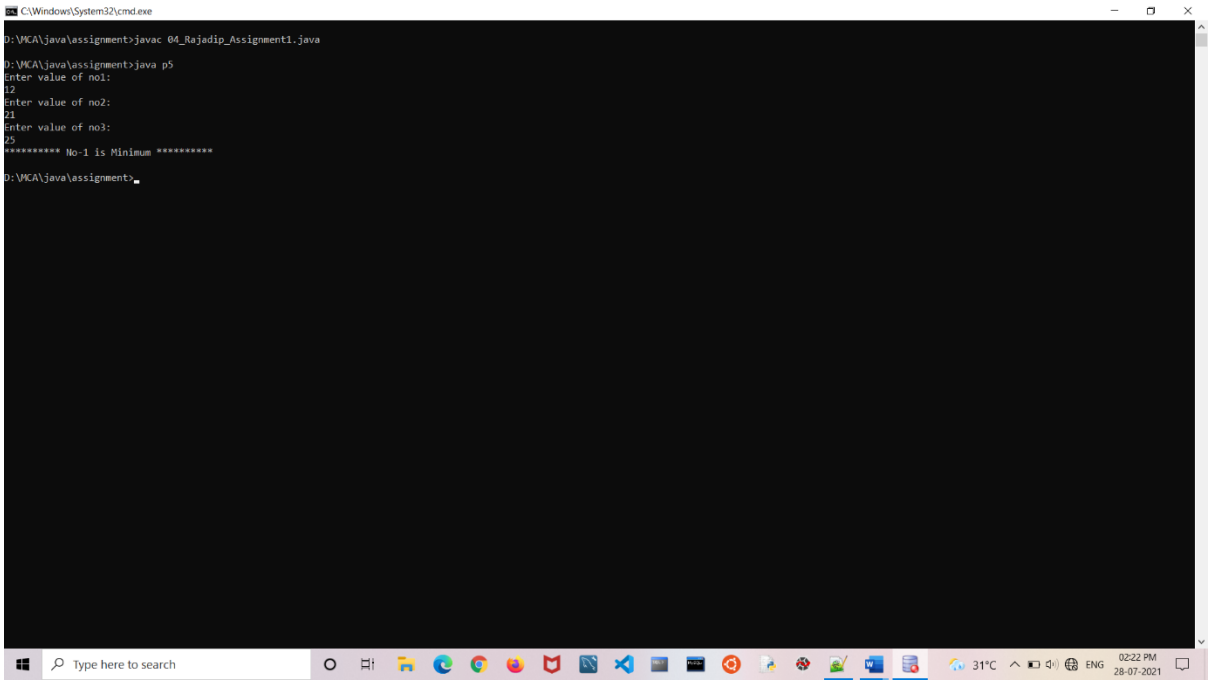
```
        {
            fact=fact*i;
        }
        System.out.println("factorial is : "+fact);

        int cube;
        cube=number*number*number;
        System.out.println("\nCube of Integer number
"+number+" is : "+cube);


        System.out.println("\nSum of all digits is
:"+getsum(number));

        int r,sum=0,temp;    //It is the number variable to be
checked for palindrome
        temp=number;
        while(number>0)
        {
            r=number%10;  //getting remainder
            sum=(sum*10)+r;
            number=number/10;
        }
        if(temp==sum)
```

```
                System.out.println("\nfactorial number is palindrome number ");
            else
                System.out.println("\nfactorial number is not palindrome");



        }
}
```



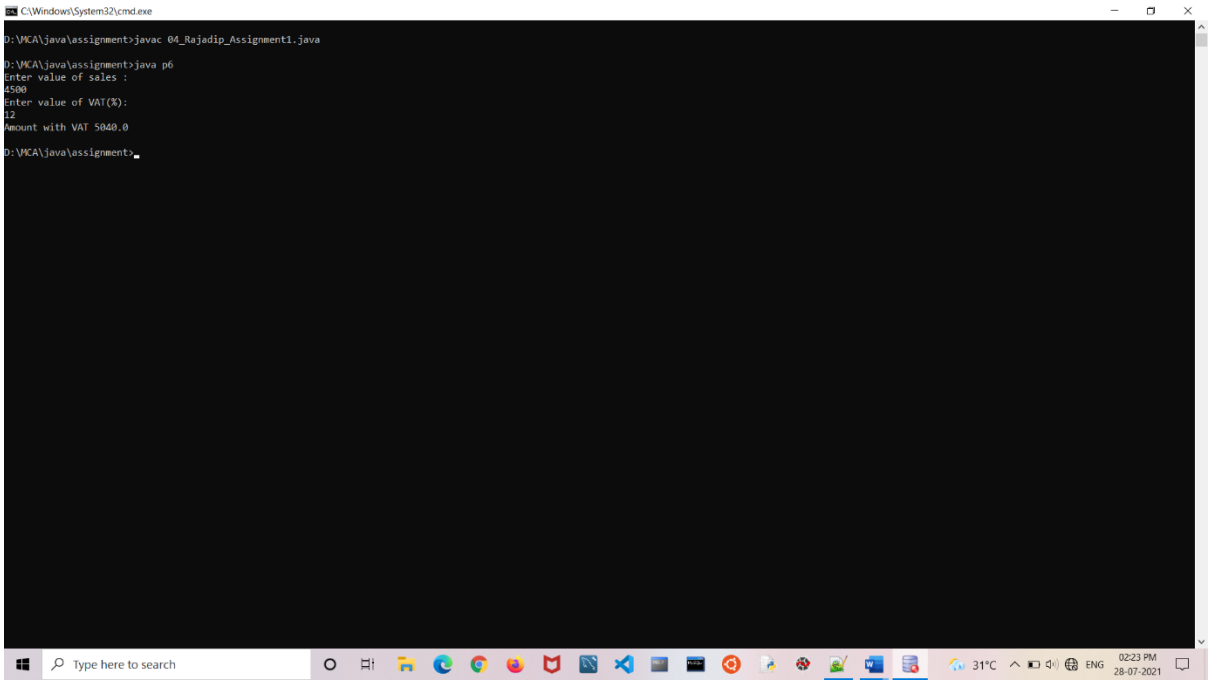
```
class p5
{
    //write a Java program to calculate the minimum of three integer numbers
    public static void main(String args[])
```

```
	{
		Scanner p5=new Scanner(System.in);
		System.out.println("Enter value of no1:");
		int no1=p5.nextInt();
		System.out.println("Enter value of no2:");
		int no2=p5.nextInt();
		System.out.println("Enter value of no3:");
		int no3=p5.nextInt();
		if(no1<no2 & no1<no3)
		{
			System.out.println("********** No-1 is Minimum
**********");
		}
		else if(no2<no3)
		{
			System.out.println("********** No-2 is Minimum
**********");
		}
		else
		{
			System.out.print("********** No-3 is Minimum
**********");
		}
	}
}
```



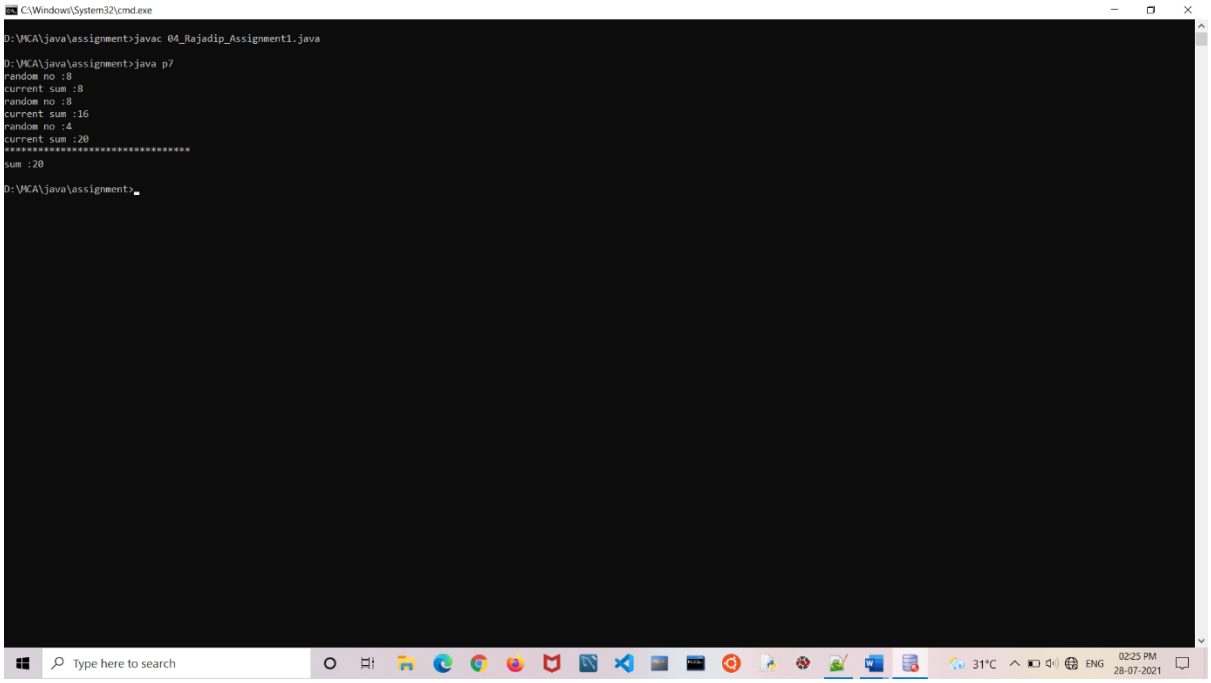
```
class p6
{
	/*WAP in JAVA that calculates the VAT on an amount of sale.
	The program takes amount of sales and VAT rate as input and outputs the total amount inclusive of VAT.*/
	public static void main(String args[])
	{
		Scanner p6=new Scanner(System.in);
		System.out.println("Enter value of sales :");
		double sales=p6.nextDouble();
		System.out.println("Enter value of VAT(%):");
		double vat=p6.nextDouble();
```

```
            double rate;
            rate=(sales *vat)/100;
            rate=rate+sales;
            System.out.println("Amount with VAT "+rate);
        }
}
```

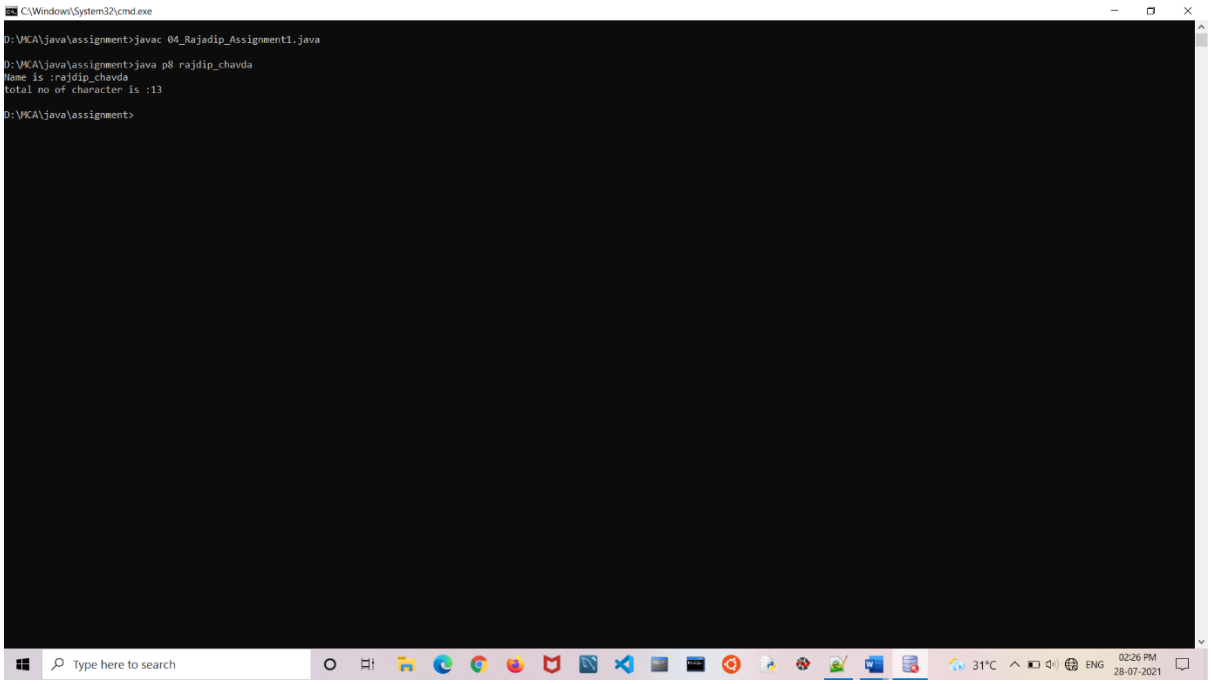


```
class p7
{
        //Use While loop to generate random numbers and maintain a running sum of these values. Terminate when the sum exceeds 20.
        //(Note: use Math.random() method to obtain numbers.)
        public static void main(String args[])
        {
```

```
        int number = 0;
        int sum=0;
        while(sum<20)
        {
            number = 1+(int)(10*Math.random());
            System.out.println("random no :"+number);
            sum+=number;
            System.out.println("current sum :"+sum);
        }

    System.out.println("*********************************");
        System.out.println("sum :"+sum);
    }
}
```

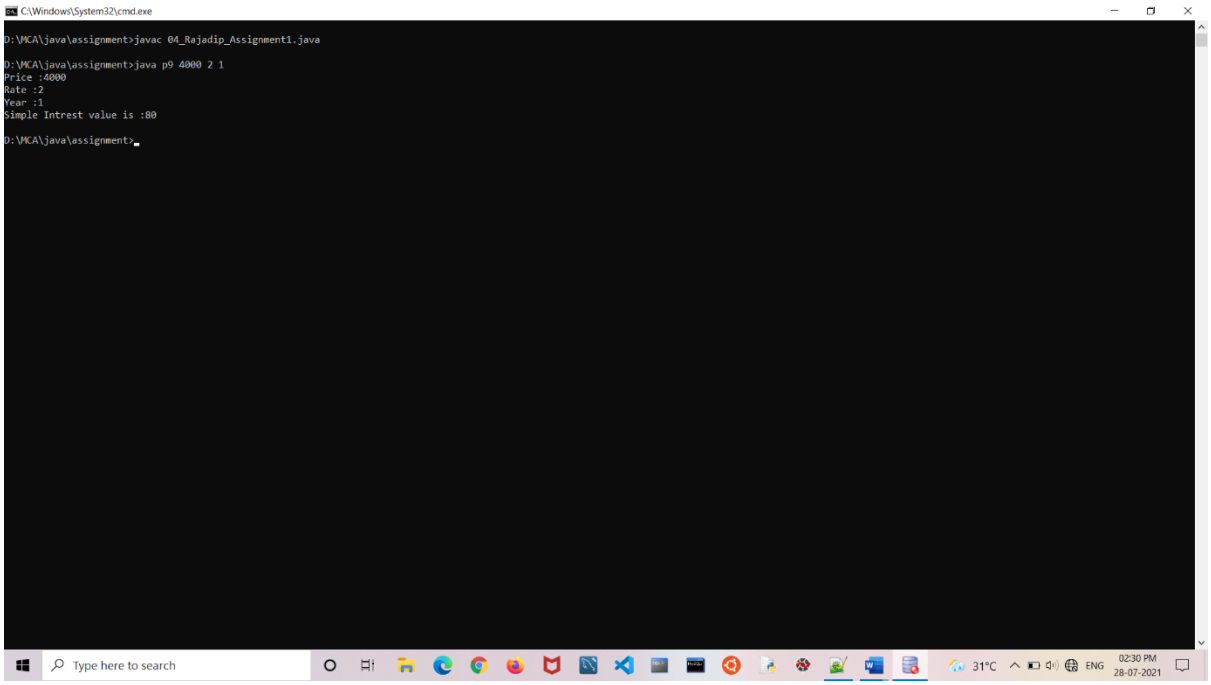


```
class p8
{
	//Write an application that counts the total number of characters in all of its command-line argument
	public static void main(String args[])
	{
		System.out.println("Name is :"+args[0]);
		int count=0;
		String s=args[0];
		for(int i=0;i<s.length();i++)
		{
			if(s.charAt(i)!=' ')
			{
				count++;
			}
		}
		System.out.println("total no of character is :"+count);
	}
}
```



```
class p9
{
	//Write a java program to calculate Simple Interest using Command Line Arguments. (Hint use Wrapper classes)
	public static void main(String args[])
	{
		int price=Integer.parseInt(args[0]);
		int rate=Integer.parseInt(args[1]);
		int year=Integer.parseInt(args[2]);
		int intrest;
		System.out.println("Price :"+price);
		System.out.println("Rate :"+rate);
		System.out.println("Year :"+year);
```

```
            intrest=(price*rate*year)/100;
            System.out.println("Simple Intrest value is :"+intrest);
        }
}
```



Practical Assignment -2

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

```
import java.util.StringTokenizer;
import java.lang.StringBuffer;

/*
1) Write application that creates an array of double, to provide following functionality.
- display the length of the array and its elements.
- Display an array. (Use for each version of loop for display).
- compute the sume of the squares of these numbers.
- Determine Mean and Median of an array.
- Sort an array – Ascending and Descending. Use any two sorting algorithm. User can also select the sorting method.
- Search an element from the array, i.e. returns the location of the element of an array that matches an indicated value.
- Copy of an array.
- Reverse of an array.
*/

class p1
{
   static void sumOfSquares(double[] arr,int n)
   {
      double avg=0,sum=0;
      for(double i : arr)
      {
```

```
    avg +=i;
}
avg = avg /n;
//System.out.println(avg);
for(double j : arr)
{
                    //System.out.println(Math.pow((j-avg),2));
    sum +=Math.pow((j-avg),2);
}
System.out.println("Sum of Squares = "+sum);
System.out.println("Mean of Array : "+avg);

bubbleSort(arr,n);
//Median of an Array
double m=0;
if(n%2==1)
{
    m=arr[(n+1)/2-1];
}
else
{
    m=(arr[n/2-1]+arr[n/2])/2;
}
```

```
    System.out.println("Median :"+m);

}

static void bubbleSort(double[] arr,int n)
{
    double temp;
    for (int i = 0; i < n-1; i++)
    {   for (int j = 0; j < n-i-1; j++)
        {
            if (arr[j] > arr[j+1])
            {
                temp = arr[j];
                arr[j] = arr[j+1];
                arr[j+1] = temp;
            }
        }
    }

}
     static void selectionSort(double[] arr,int n)
{
    int min ;
    for(int i=0;i< n-1;i++)
```

```
    {
        min = i;
        for(int j=i+1;j<n;j++)
        {
            if(arr[j] < arr[min])
                min = j;
        }

        double temp= arr[min];
        arr[min] = arr[i];
        arr[i] =temp;
    }
    printArray(arr);
}

static void printArray(double[] arr)
{
    System.out.println("Array Elements are ::  ");

    for(double i : arr)
    {
        System.out.println(i);
    }
}
```

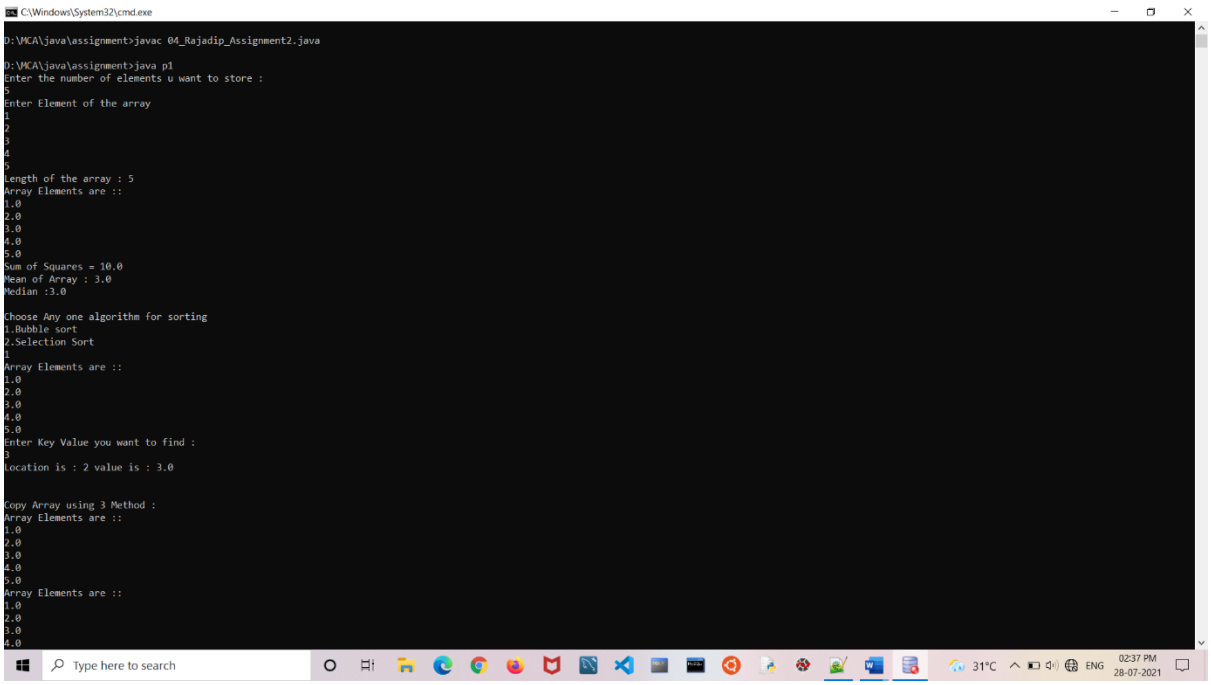
```
static int searchElement(double[] arr,int n,double key)
{
   for(int i=0;i<n;i++)
   {
      if(arr[i] == key)
      {
         return i;
      }
   }
   return 0;
}
static void copyArray(double[] arr,int n)
{
   System.out.println("Copy Array using 3 Method : ");
   double[] arrB = new double[n];
   double[] arrC = new double[n];
   //copy location of array to another array
   arrB = arr;
   printArray(arrB);

   for (int i = 0; i < n; i++)
      arrC[i] = arr[i];
```

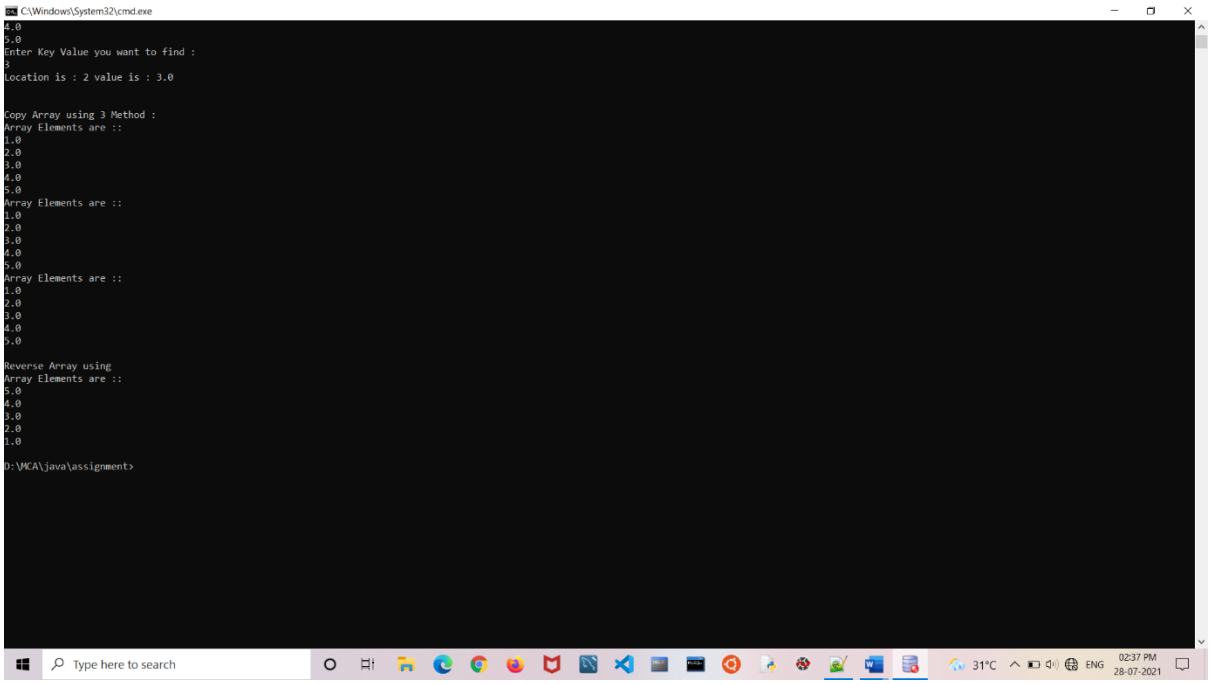
```
    printArray(arrC);

    //array copy using clone
    double[] arrD = arr.clone();
    printArray(arrD);
}

static void reverseArray(double[] arr,int n)
{
    System.out.println("Reverse Array using ");
    double temp;
    for(int i=0;i<n/2;i++)
    {
        temp = arr[i];
        arr[i] = arr[n-i-1];
        arr[n-i-1] = temp;
    }
    printArray(arr);
}
public static void main(String[] args)
{
    int n;
    Scanner sc = new Scanner(System.in);
```

```
System.out.println("Enter the number of elements u want to store : ");
n = sc.nextInt();

double arr[] = new double[n];

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

for(int i=0;i<n;i++)
{
   arr[i]=sc.nextDouble();
}

System.out.println("Length of the array : "+ n);

printArray(arr);

sumOfSquares(arr,n);
System.out.println();

int choice;
System.out.println("Choose Any one algorithm for sorting");
System.out.println("1.Bubble sort");
System.out.println("2.Selection Sort");
```

```
        choice = sc.nextInt();

        if(choice == 1)
        {
           bubbleSort(arr,n);
           printArray(arr);
        }
        else if(choice == 2)
        {
           selectionSort(arr,n);
        }
        else{
           System.out.println("select between 1 | 2 ");
        }



        //Searching Elemetn
        int loc;
        double key=0;
        System.out.println("Enter Key Value you want to find : ");
        key = sc.nextDouble();
        loc = searchElement(arr,n,key);
        System.out.println("Location is : "+ loc + " value is : " +
arr[loc]+"\n\n");
```

```
        //Copy elements of array from one array  to another array
        copyArray(arr,n);

        System.out.println();

        reverseArray(arr,n);
    }
}
```

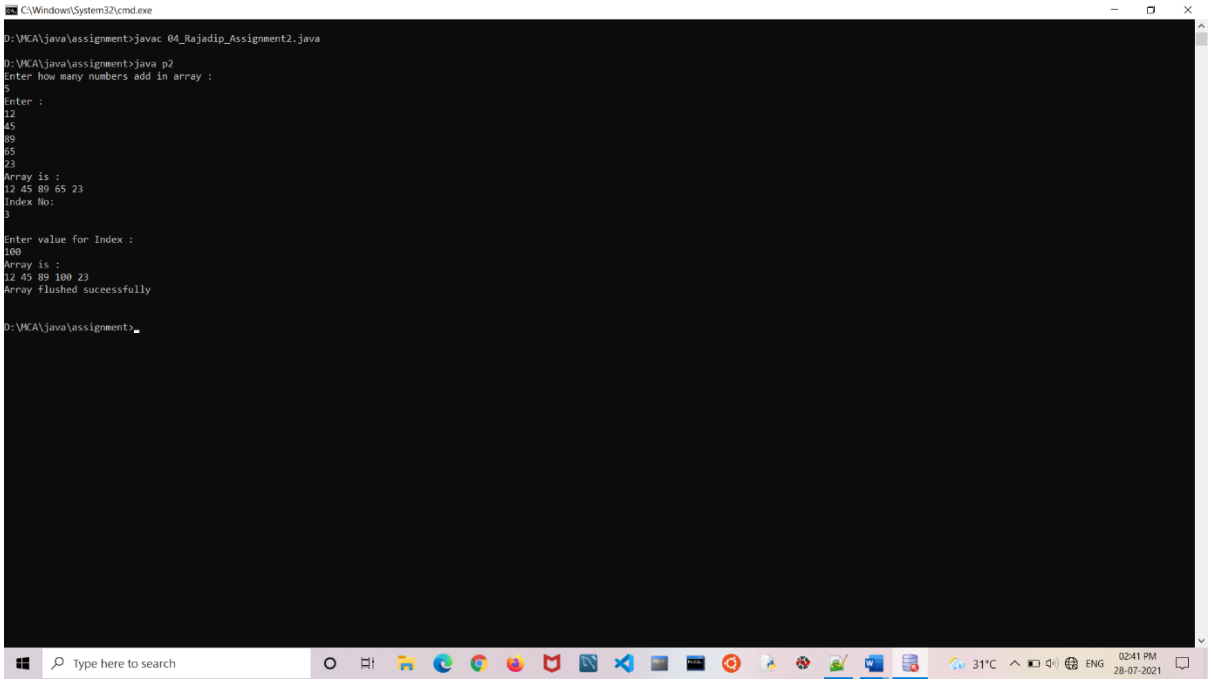




```
/*
2) Write a program in Java to create variable size array. Data can be numeric.
- Flush an array
- add a number at a specified location in an anay.
- display the array
*/

class p2
{
   static void printArray(int[] arr,int n)
   {
```

```
    System.out.println("Array is : ");
    for(int i=0;i<n;i++)
    {
        System.out.print(arr[i]+ " ");
    }
}
public static void main(String[] args)
{
    Scanner sc = new Scanner(System.in);
    int n;
    System.out.println("Enter how many numbers add in array : ");
    n = sc.nextInt();
    System.out.println("Enter :");
    int[] arr = new int[n];
    for(int i=0;i<n;i++)
    {
        arr[i] = sc.nextInt();
    }
    printArray(arr,n);

    int key,val;
    System.out.println("\nIndex No: ");
    key = sc.nextInt();
    System.out.println("\nEnter value for Index : ");
```

```
        val = sc.nextInt();
        if(key < n)
        {
            arr[key] = val;
        }
        else{
            System.out.println("Index is greater than range");
        }
        printArray(arr,n);
        Arrays.fill(arr,0);
        System.out.println("\nArray flushed suceessfully \n" );
    }
}
```



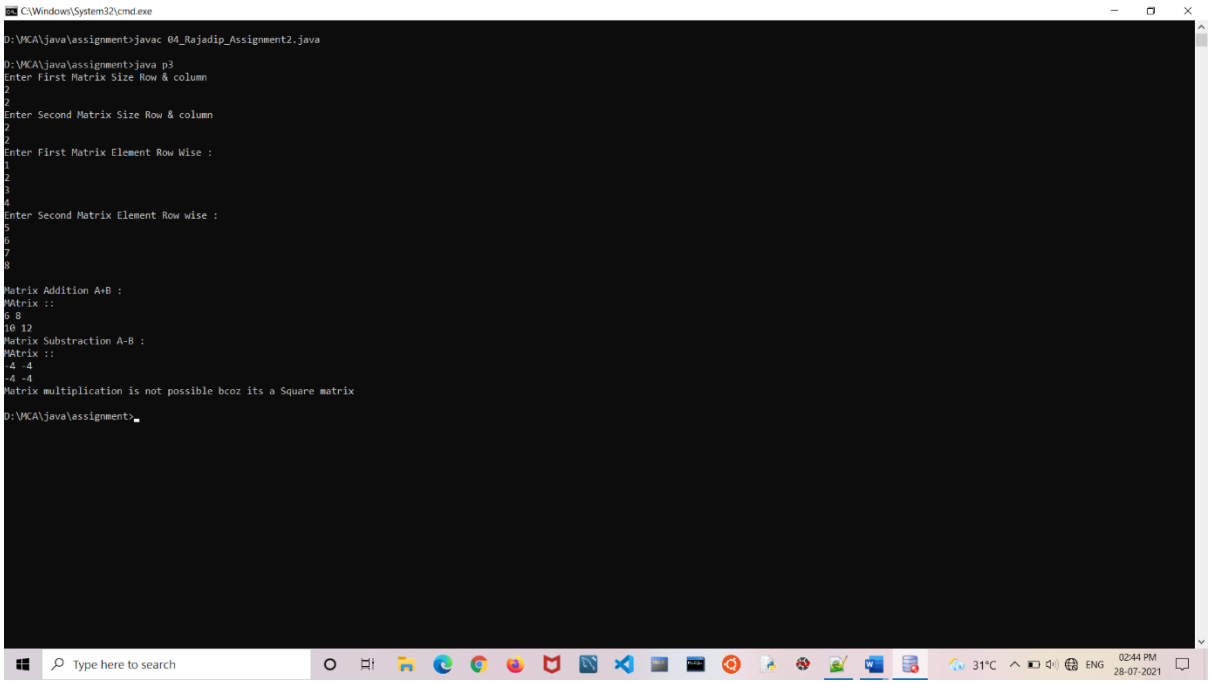
```
/*
3) Program in Java to find A+B, A-B, A*B and transpose of A, where A is a matrix of 3*3 and B is a matrix of 3*4.
Take the values in matrixes A and B from the user.
*/

class p3
{
   static void matrixMulti(int[][] A,int[][] B,int row1,int row2,int col2)
   {
      int c[][] = new int[row1][col2];
      System.out.println("MAtrix Multiplication : ");
       for(int i=0;i<row1;i++){
         for(int j=0;j<col2;j++)
         {
            c[i][j]=0;
            for(int k=0;k<row2;k++)
            {
               c[i][j]+=A[i][k]*B[k][j];
            }
            System.out.print(c[i][j]+" ");
         }
         System.out.println();
```

```
    }
}
static void summationMatrix(int[][] A,int[][] B ,int row1,int col1)
{
    int c[][] = new int[row1][col1];
     for(int i=0;i<row1;i++)
     {
        for(int j=0;j<col1;j++)
        {
            c[i][j] +=A[i][j] + B[i][j];
        }
     }
     System.out.println("Matrix Addition A+B :");
     printMatrix(c,row1,col1);
}
static void substractionMatrix(int[][] A,int[][] B ,int row1,int col1)
{
    int c[][] = new int[row1][col1];
     for(int i=0;i<row1;i++)
     {
        for(int j=0;j<col1;j++)
        {
            c[i][j] +=A[i][j] - B[i][j];
        }
```

```
    }
    System.out.println("Matrix Substraction A-B :");
    printMatrix(c,row1,col1);
}
static void printMatrix(int[][] arr,int row1,int col2)
{
   System.out.println("MAtrix ::  ");
   for(int i=0;i<row1;i++)
   {
      for(int j=0;j<col2;j++)
      {
         System.out.print(arr[i][j] + " ");
      }
      System.out.println();
   }
}
public static void main(String[] args)
{
   /* A+B A-B A*B*/
   Scanner sc = new Scanner(System.in);
   int row1,col1,row2, col2;
   System.out.println("Enter First Matrix Size Row & column");
   row1 = sc.nextInt();
   col1 = sc.nextInt();
```

```
System.out.println("Enter Second Matrix Size Row & column");
row2 = sc.nextInt();
col2 = sc.nextInt();

int[][] A = new int[row1][col1];
int[][] B = new int[row2][col2];

System.out.println("Enter First Matrix Element Row Wise : ");
for(int i=0;i<row1;i++)
{
   for(int j=0;j<col1;j++)
   {
      A[i][j] = sc.nextInt();
   }
}
System.out.println("Enter Second Matrix Element Row wise : ");
for(int i=0;i<row2;i++)
{
   for(int j=0;j<col2;j++)
   {
      B[i][j] = sc.nextInt();
   }
}
//printMatrix(A,row1,col1);
```

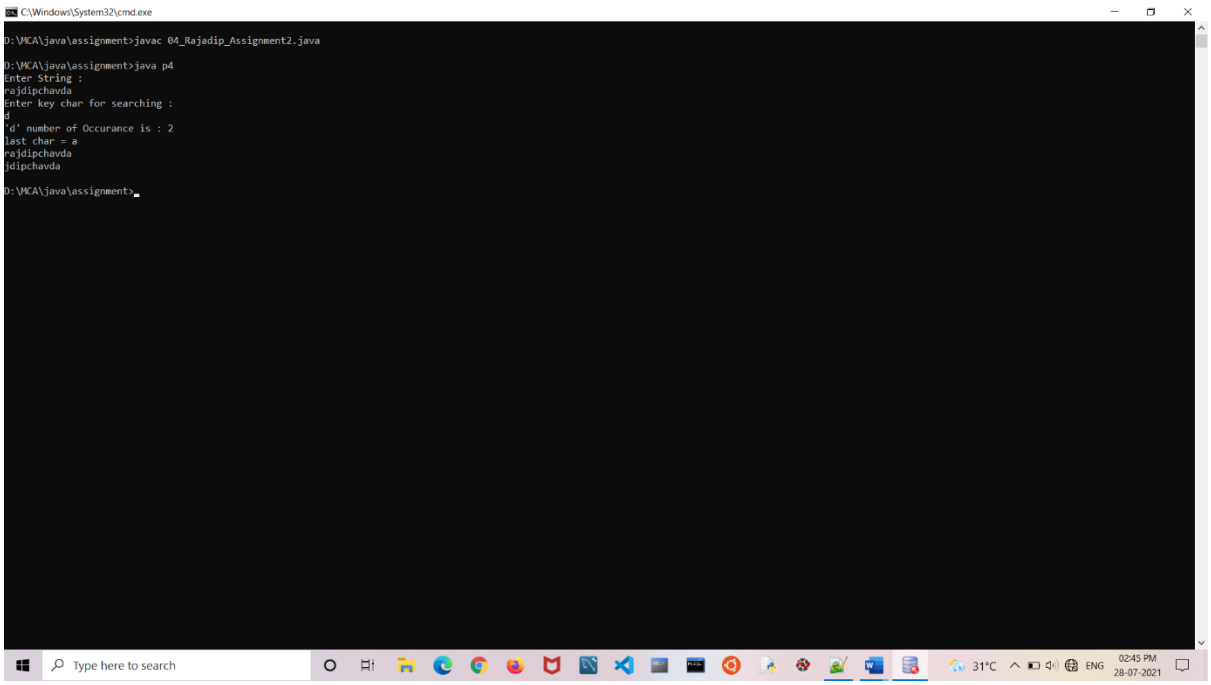
```
        //printMatrix(B,row2,col2);


        System.out.println();
        if(row1 == col1 && row2 == col2)
        {
            summationMatrix(A,B,row1,col1);
            substractionMatrix(A,B,row1,col1);
            System.out.println("Matrix multiplication is not possible bcoz
its a Square matrix");
        }
        else
        {
            System.out.println("Addtion matrix and substraction of matrix
Requires Square Matrix ");
            if(row2 == col1)
                matrixMulti(A,B,row1,row2,col2);
            else
                System.out.println("Matrix multiplication is not possible");
        }
    }
}
```



```
/*
4) Write and run a JAVA program that reads a string from the user and perform the following.
- counts number of occurance of a given character (for example, " a") in a string.
- searches the last occurance of a character in a string.
- removes the unneccessary spaces from a string : leading and trailing spaces.
- displays the substring formed by the last ten characters of a string
*/

class p4
{
    public static void main(String[] args)
```

```
  {
     Scanner sc = new Scanner(System.in);
     System.out.println("Enter String : ");
     String str;
     str = sc.nextLine();
     char key;
      System.out.println("Enter key char for searching : ");
     key = sc.next().charAt(0);
     //System.out.println(str);
     char[] ch = str.toCharArray();
     int count=0;
     for(char i: ch)
     {
        if(i == key)
        {
           count++;
        }
     }
     System.out.println("'"+key +"' number of Occurance is : " +
count);

     //searches last occurance of char in string
     System.out.println("last char = " + str.charAt(str.length()- 1));
```

```
        //remove extra space from string
        System.out.println(str.replaceAll("\\s+"," ").trim());

        //substring
        String last10Digit = " ";
        last10Digit =  str.substring(str.length() - 10);

        System.out.println(last10Digit);
    }

}
```



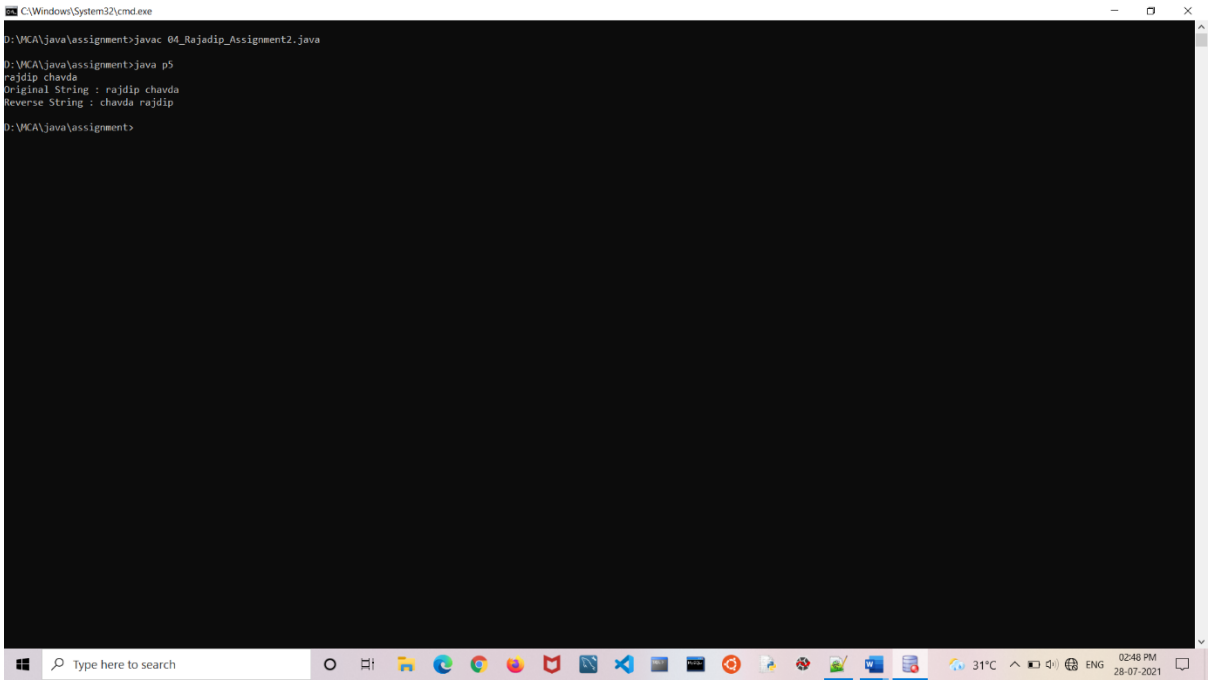
```
/*
```

5) WAP that inputs a line of text, tokenizes the line with StringTokenizer and outputs the tokens in reverse order.

*/

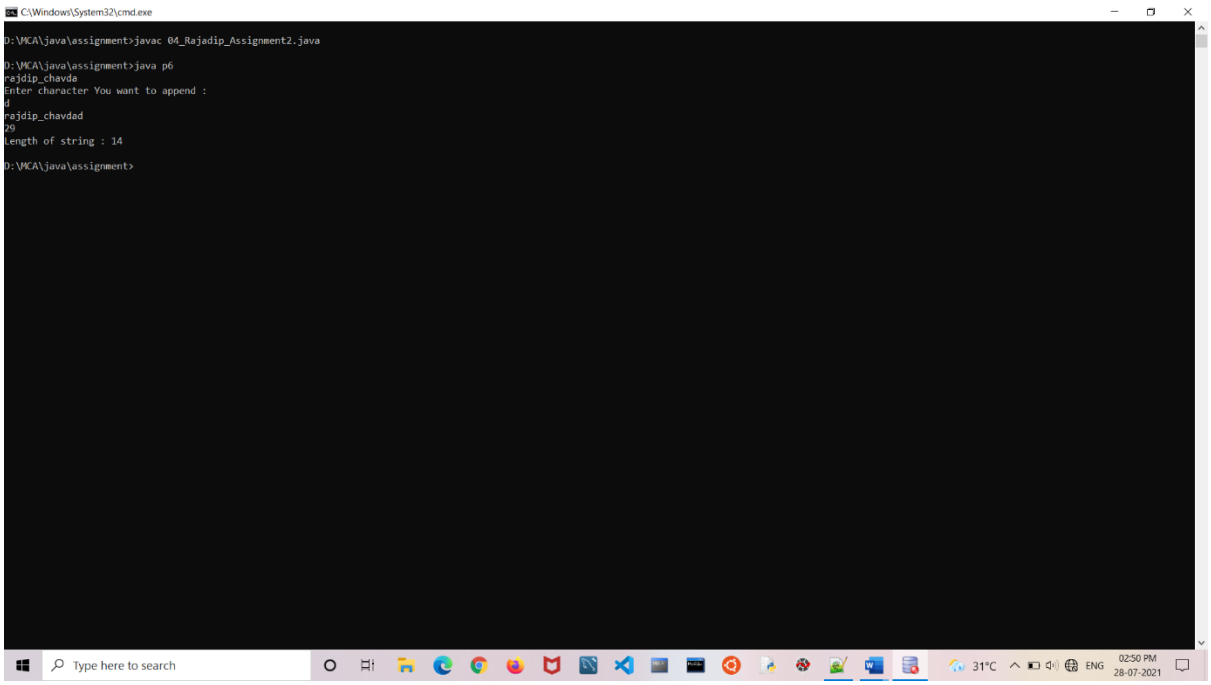
```
class p5
{
   public static void main(String[] args)
   {
      Scanner sc =new Scanner(System.in);

      String str;
      str = sc.nextLine();

      System.out.println("Original String : "+ str);

      StringTokenizer st = new StringTokenizer(str);
      String strRev= " ";
      while(st.hasMoreTokens())
      {
         strRev = st.nextToken() + " " + strRev;
      }

      System.out.println("Reverse String : " + strRev);
```

```
    }
}
```



```
/*
6) Create a StringBuffer and illustrate how to append character.
Display capacity, length of the StringBuffer.
*/


class p6
{
    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        String str = " ";
```

```
        str = sc.nextLine();
        StringBuffer sb = new StringBuffer(str);
        System.out.println("Enter character You want to append : ");
        char newStr = sc.next().charAt(0);
        sb.append(newStr);
        System.out.println(sb);
        System.out.println(sb.capacity());
        System.out.println("Length of string : "+sb.length());
    }
}
```



```
/*
7) Write an application that reads and processes strings from the
console. Perform the following functions based on the menu choice
selected by the user..
- Reverse the sequence of strings and then display it.
- Reverse the sequence of characters in each string and then display
it.
- rearrange the strings according to the length of the string.
- Sorting
- Concatenation
- Change them to uppercase, lowercase depending on user’s choice.
*/

class p7
{
   static Scanner sc =new Scanner(System.in);

   static void reverseSeqOfString(String str)
   {
      String s[] = str.split(" ");
      String seqStr = " ";

      for(int i=s.length-1;i>=0;i--)
         seqStr += s[i] + " ";
```

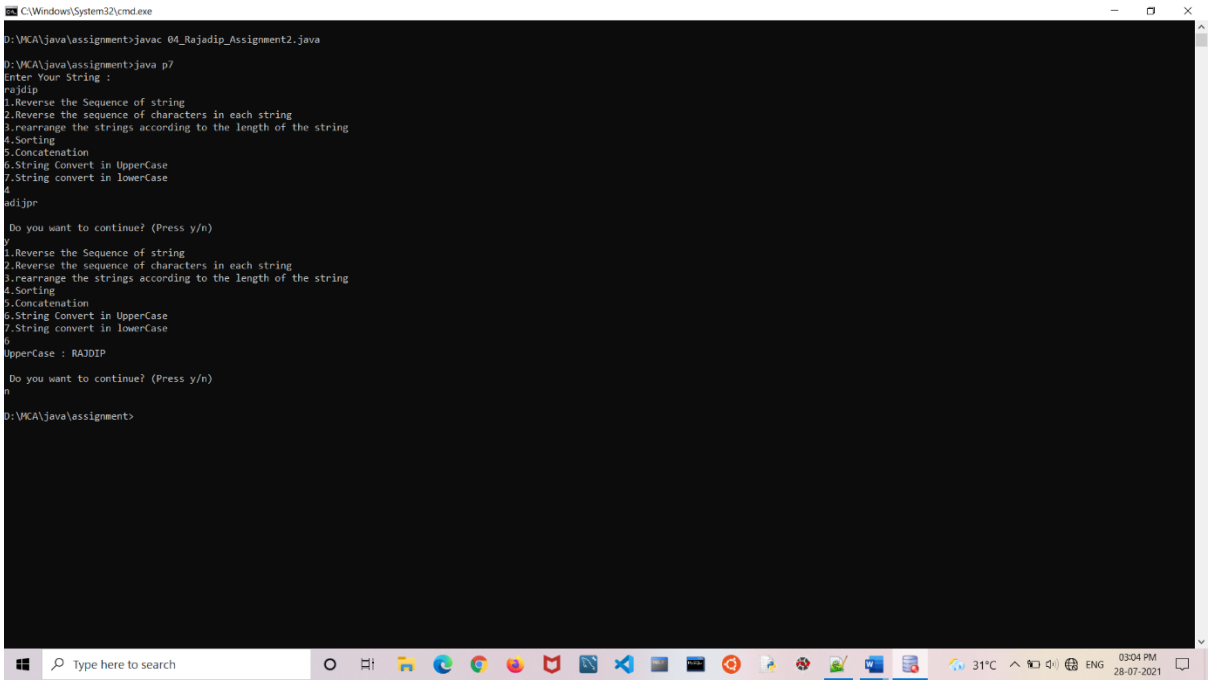
```
        System.out.println("Reverse the sequence of strings\n" + seqStr
+"\n");
    }
    static void reverseSeqOfCharEachString(String str)
    {

        String s[] = str.split(" ");
        String seqStr = " ";
        String tmp;
        for(String w:s)
        {
            StringBuilder sb=new StringBuilder(w);
            sb.reverse();
            seqStr += sb.toString() + " ";
        }
        System.out.println("Reverse the sequence of character of each
strings\n" + seqStr+"\n");
     }
    static void rearrangeString(String str)
    {
        String[] ch = str.split(" ");
       /* for(int i=1;i < ch.length;i++)
        {
```

```
        String temp = ch[i];
        int j = i-1;
        while( j >= 0 && temp.length() < ch[j].length() )
        {
           ch[j+1] = ch[j];
           j--;
        }
        ch[j+1] = temp;
    }*/
    System.out.println(ch.toString());
    System.out.println("Rearrange String : "+str);
}
static void sortingString(String str)
{
    char sortStr[] = str.toCharArray();
    Arrays.sort(sortStr);
    System.out.println(sortStr);
}
static void ConcatenationString(String str)
{
    String newStr =" " ;
    System.out.println("Enter string u want to append");
    newStr = sc.nextLine();
    String concateStr = str.concat(newStr);
```

```
        System.out.println(concateStr);
    }
    static void upperCaseString(String str)
    {
        System.out.println("UpperCase : "+str.toUpperCase());
    }
     static void lowerCaseString(String str)
    {
        System.out.println("LowerCase "+str.toLowerCase());
    }

    public static void main(String[] args)
    {
        System.out.println("Enter Your String : ");
        String str=sc.nextLine();
        char ch = ' ';
        int choice ;

        do
        {
        System.out.println("1.Reverse the Sequence of string ");
        System.out.println("2.Reverse the sequence of characters in
each string ");
```

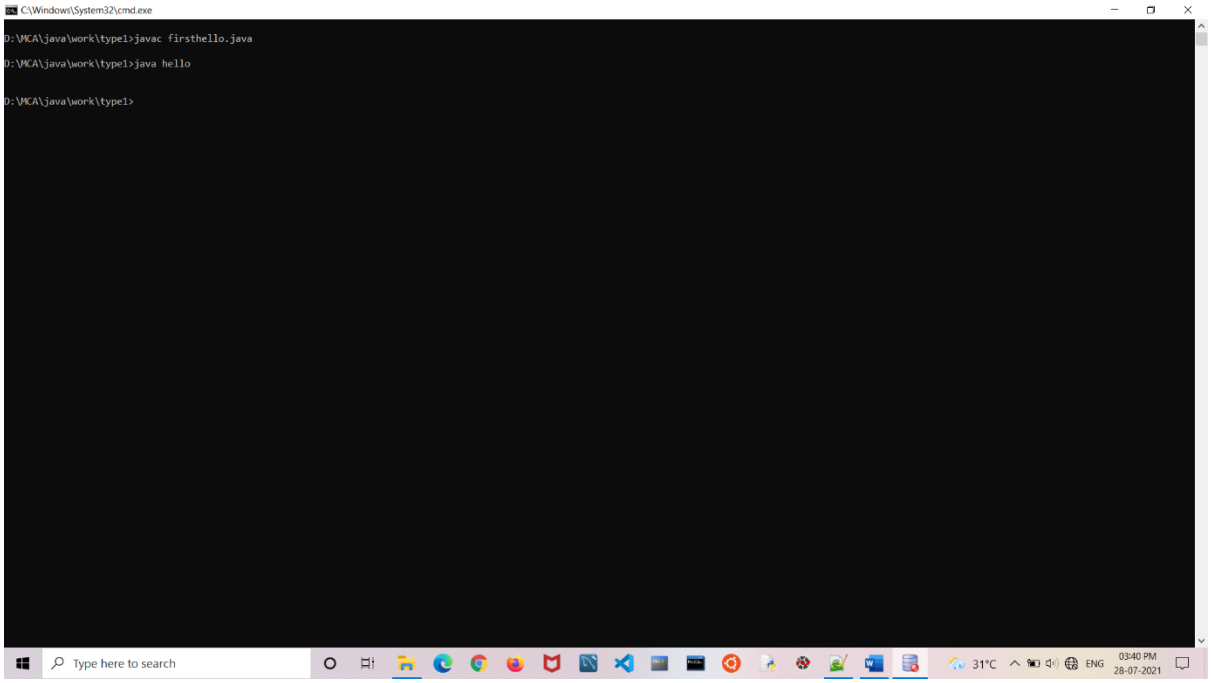
```
System.out.println("3.rearrange the strings according to the length of the string");
System.out.println("4.Sorting");
System.out.println("5.Concatenation");
System.out.println("6.String Convert in UpperCase");
System.out.println("7.String convert in lowerCase");

choice = sc.nextInt();
switch(choice)
{
case 1:
   reverseSeqOfString(str);
   break;
case 2:
   reverseSeqOfCharEachString(str);
   break;
case 3:
   rearrangeString(str);
   break;
case 4:
   sortingString(str);
   break;
case 5:
   ConcatenationString(str);
```

```
            break;
        case 6:
            upperCaseString(str);
            break;
        case 7:
            lowerCaseString(str);;
            break;
        case 8:
            System.exit(0);
        default:
            System.out.println("Enter Valid choice");
        }
      System.out.println("\n Do you want to continue? (Press y/n)");
      ch = sc.next().charAt(0);
      }while(ch == 'y');
   }
}
```


C:\Windows\System32\cmd.exe
D:\MCA\java\assignment>javac 04_Rajadip_Assignment2.java
D:\MCA\java\assignment>java p7
Enter Your String :
rajdip
1.Reverse the Sequence of string
2.Reverse the sequence of characters in each string
3.rearrange the strings according to the length of the string
4.Sorting
5.Concatenation
6.String Convert in UpperCase
7.String convert in lowerCase
4
adijpr
Do you want to continue? (Press y/n)
y
1.Reverse the Sequence of string
2.Reverse the sequence of characters in each string
3.rearrange the strings according to the length of the string
4.Sorting
5.Concatenation
6.String Convert in UpperCase
7.String convert in lowerCase
6
UpperCase : RAJDIP
Do you want to continue? (Press y/n)
n
D:\MCA\java\assignment>
Type here to search
31°C
ENG
03:04 PM
28-07-2021

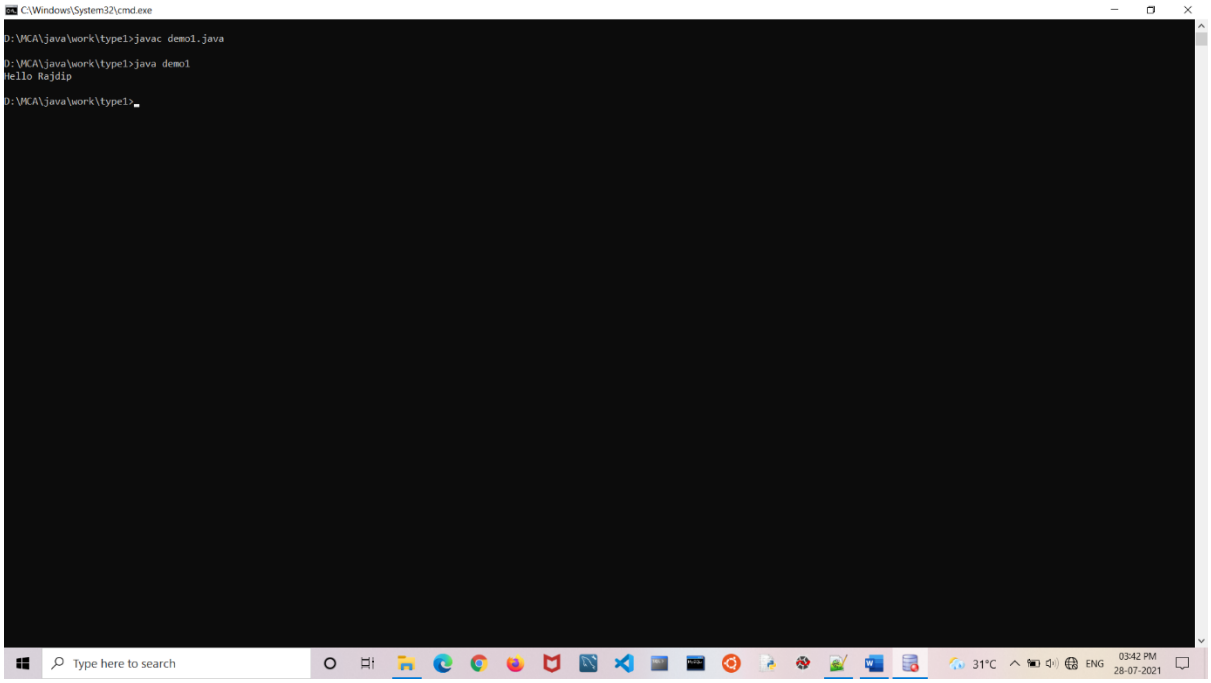
Class Work – Simple Program's

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



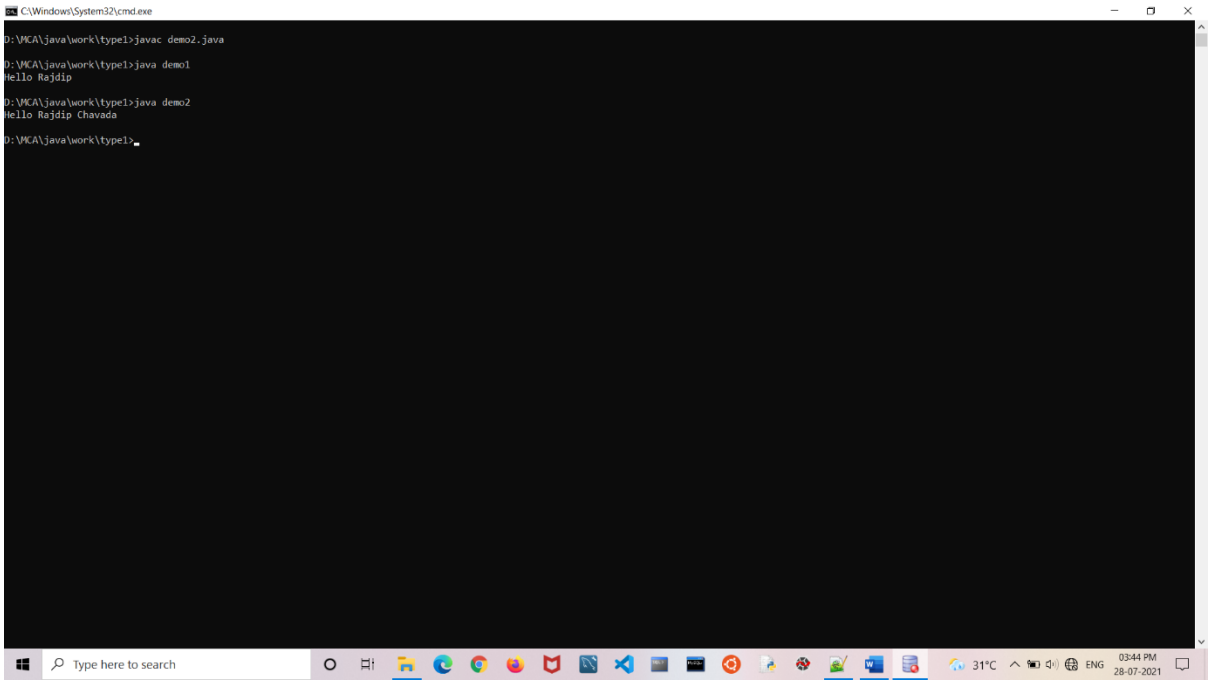
```
class demo1
{
	public static void main(String args[])
```

```
	{
		System.out.println("Hello Rajdip"); // simple hello <name> program
	}
}
```



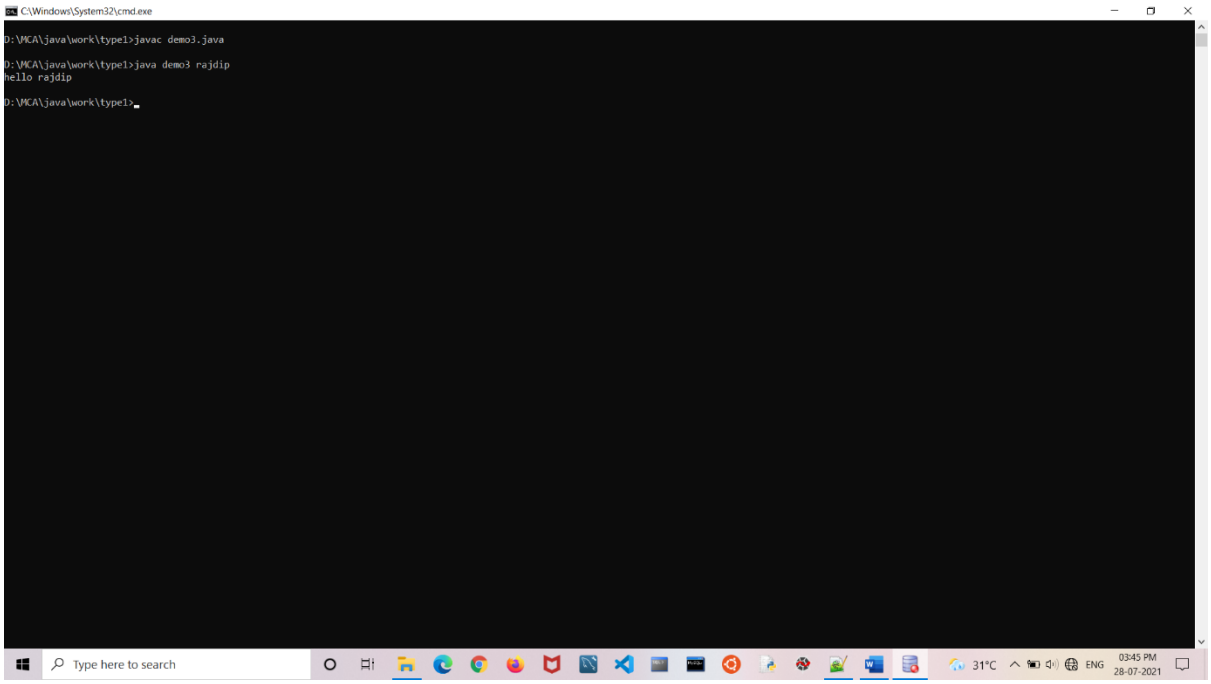
```
class demo1
{
	public static void main(String args[])
	{
		System.out.println("Hello Rajdip");
	}
}
class demo2  //public class always named as file name
```

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

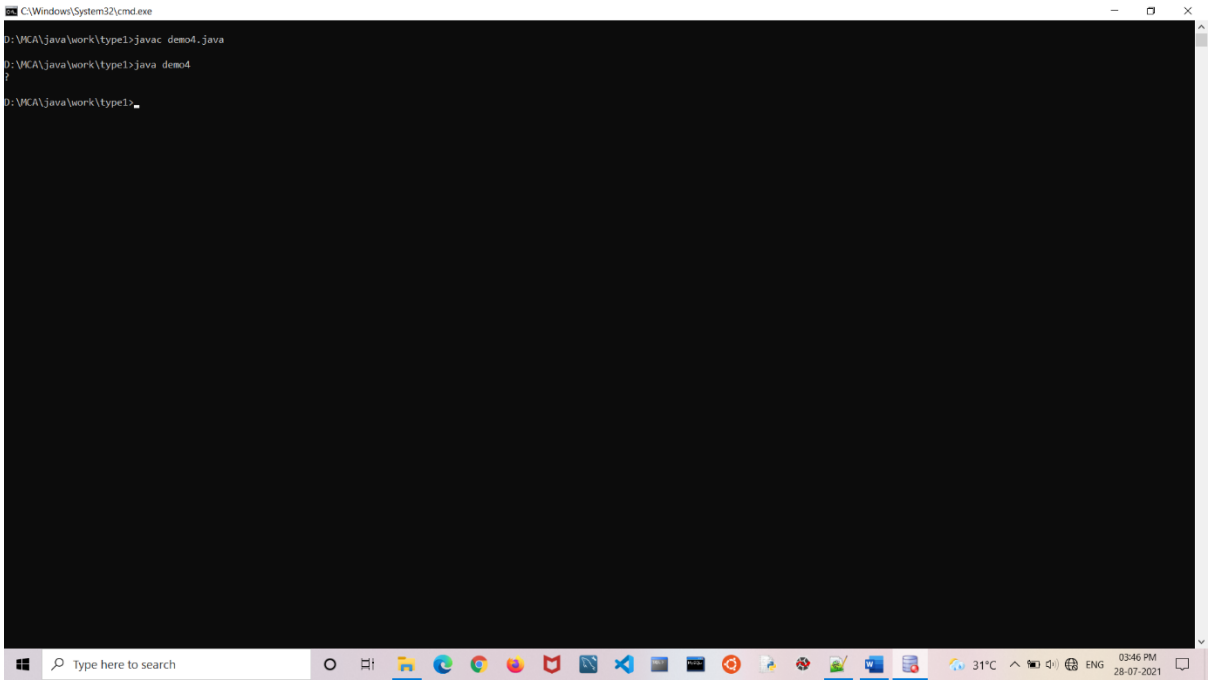


```
class demo3
{
	public static void main(String args[])
	{
		for(String name : args) //concept of jagged array
		{
			System.out.println("hello "+name);
```

```
        }
    }
}
```

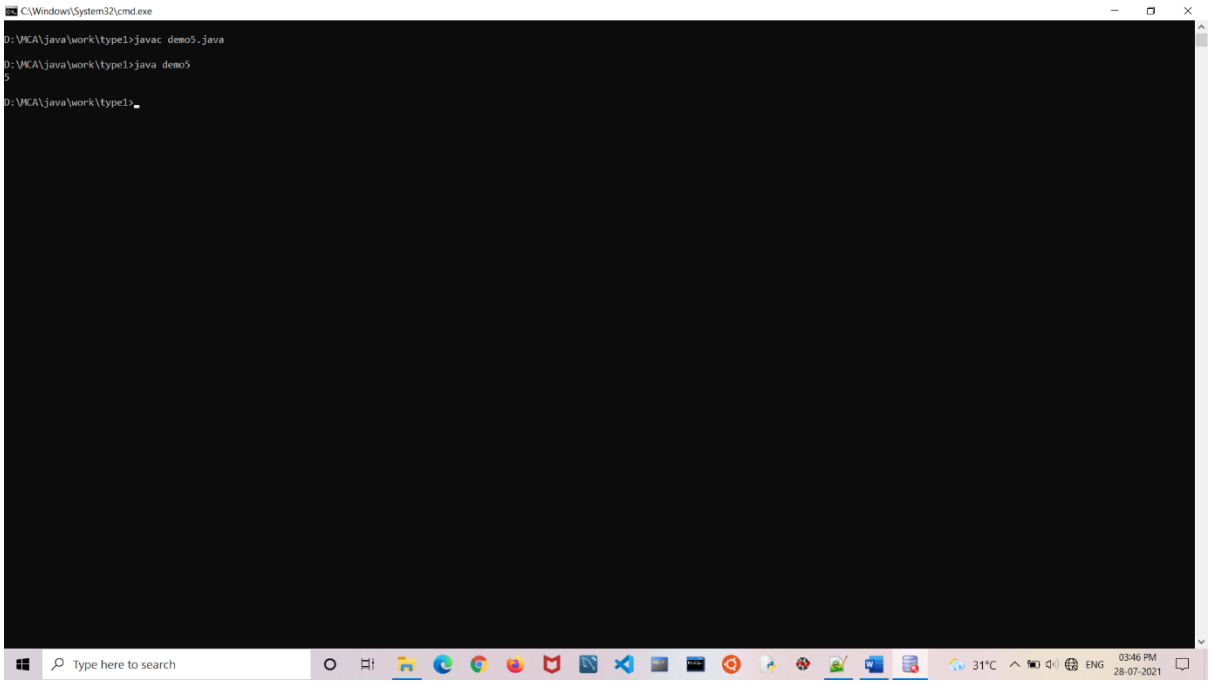


```
class demo4
{
    public static void main(String args[])
    {
        char ch='\u0B85';
        System.out.println(ch);
    }
}
```



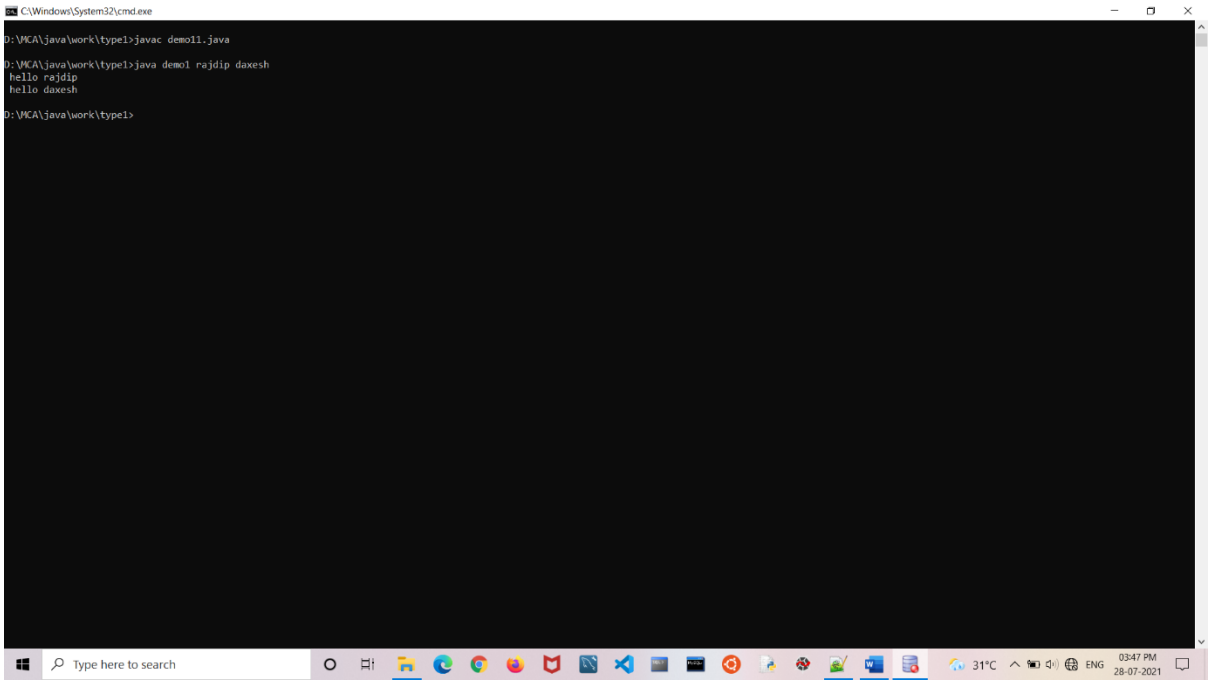
```
class demo5
{
	public static void main(String args[])
	{
		int[][] a=new int[10][5];
		a[2]=a[0];
		for(int i=0;i<a.length;i++)
		{
			for(int j=0;j<a[i].length;j++)
			{
				a[i][j]=i+j;
			}
		}
```

```
            System.out.println(a[1][4]);
        }
}
```



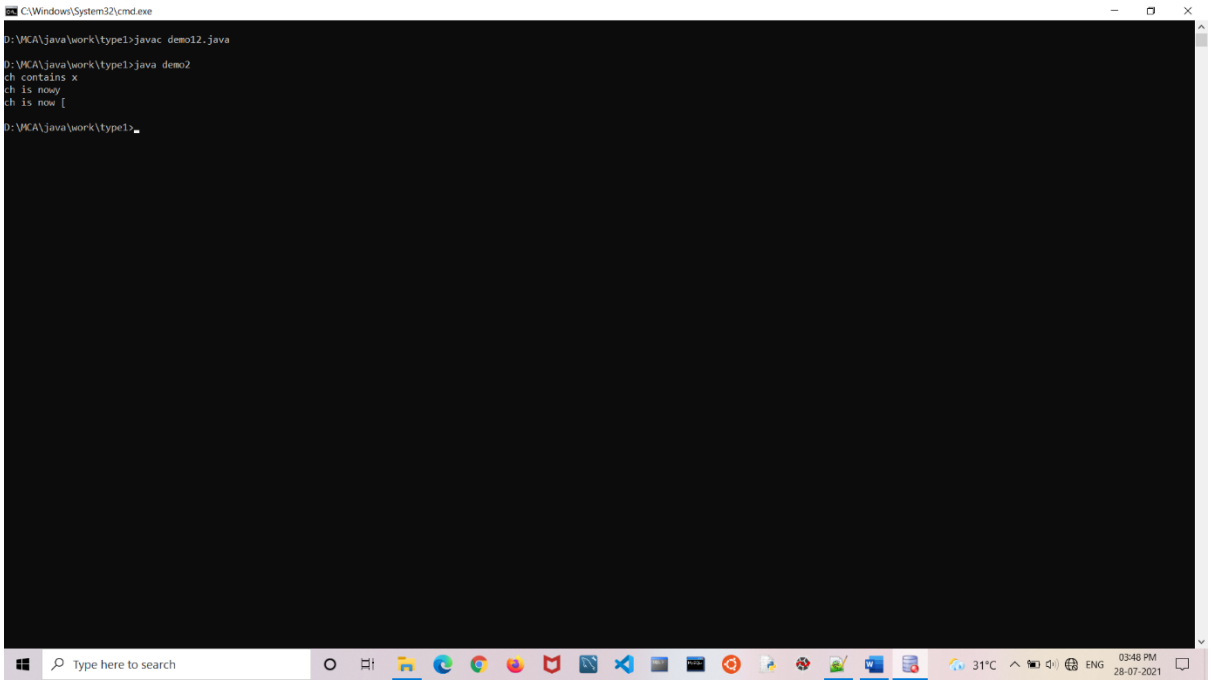
```
class demo1
{
        public static void main(String args[])
        {
            /*System.out.println("there are "+ args.length + " command line arguments");
            System.out.println("there are "+ args[0] + " "+ args[1] );*/
            for(int i=0;i<args.length;i++)
            System.out.println(" hello " + args[i]);
```

```
        }
}
```



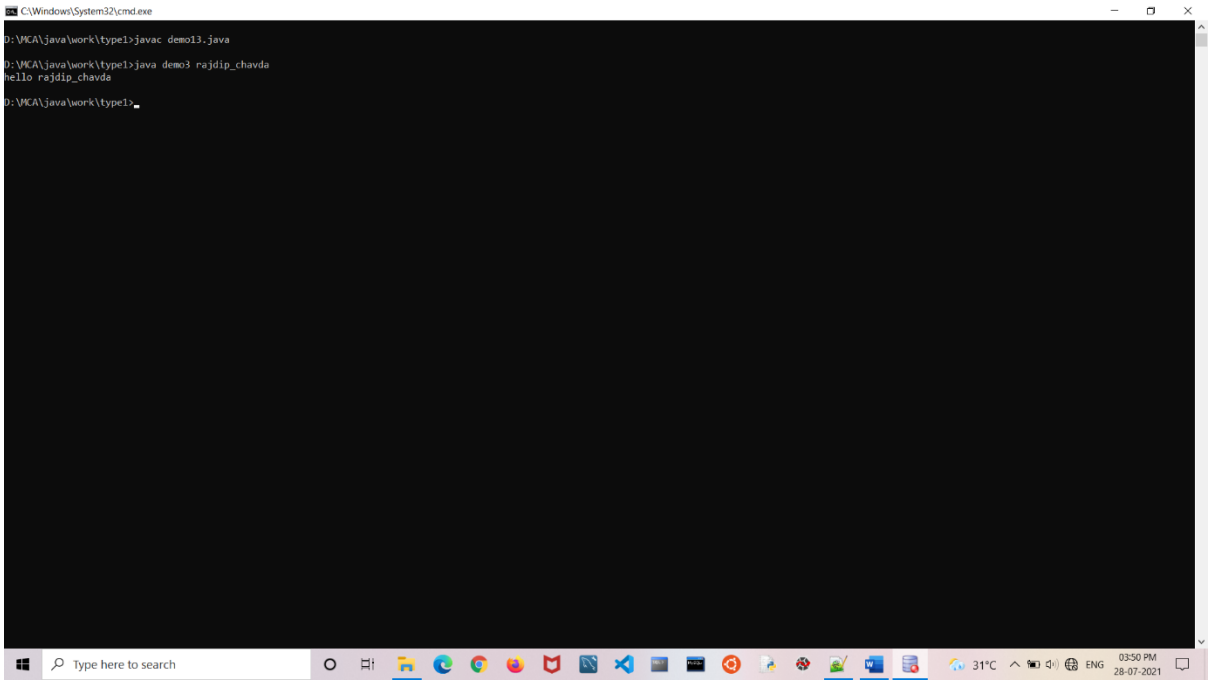
```
class demo2
{
        public static void main(String args[])
        {
                char ch;
                ch='x';
                System.out.println("ch contains " + ch);
                ch++;
                System.out.println("ch is now" + ch);
                ch=90;
                ch++;
```

```
            System.out.println("ch is now " + ch);
        }
}
```



```
package p1;


class c1
{
        public void m1()
        {
            System.out.println("m1 of c1");
        }
        public static void main(String args[])
        {
```

```
        c1 obj=new c1();
        obj.m1();
    }
}
```



```
import java.util.Scanner;
class hello
{
    public static void main(String args[])
    {
        System.out.println("hello");
    }
}
class helloname
```

```
{
	public static void main(String args[])
	{
		System.out.print("hello "+args[0]);
	}
}
class dynamic
{
	public static void main(String args[])
	{
		for(String name:args)
		{
			System.out.println("hello "+name);
			System.out.println("hello "+args);
		}
	}
}
class jagged
{
	public static void main(String args[])
	{
		int p[][]={{1,2,3},{4,5,6},{7,8,9}};
		for(int i[]:p)
		{
```

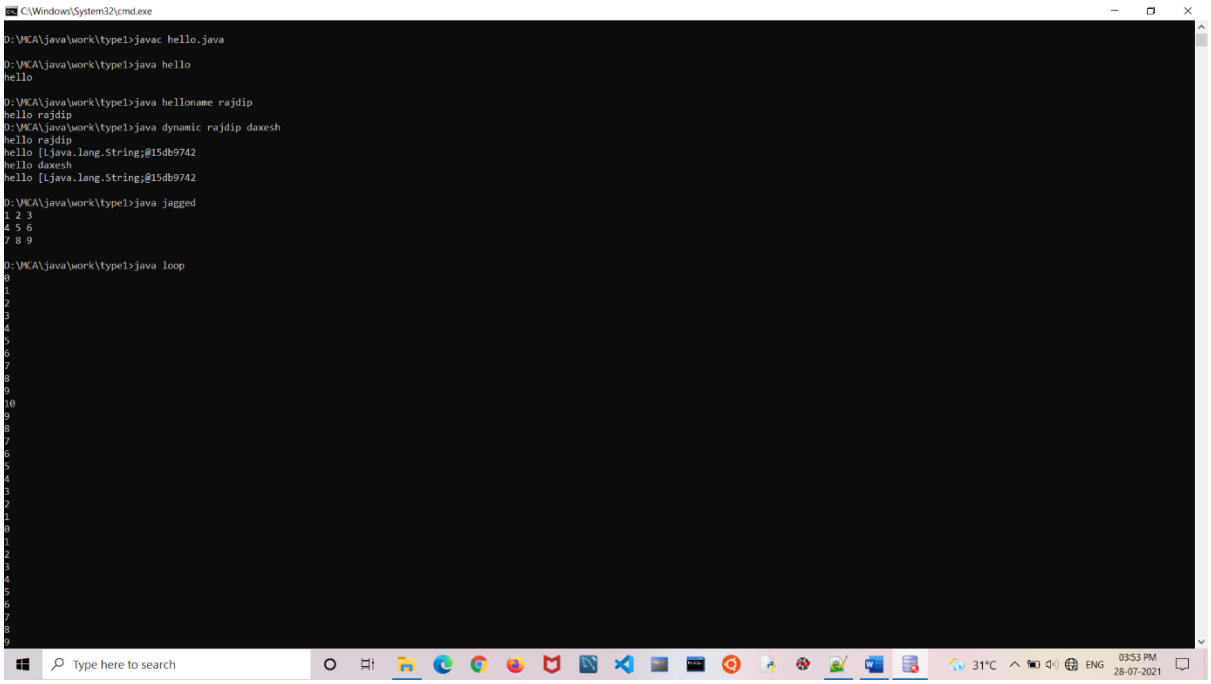
```
                for(int j:i)
                {
                        System.out.print(j+" ");
                }
                System.out.println(" ");
        }
    }
}
class loop
{
    public static void main(String args[])
    {
        int i=0;
        for(i=0;i<10;i++)
        {
            System.out.println(i+" ");
        }
        while(i>0)
        {
            System.out.println(i+" ");
            i--;
        }
        do
        {
```

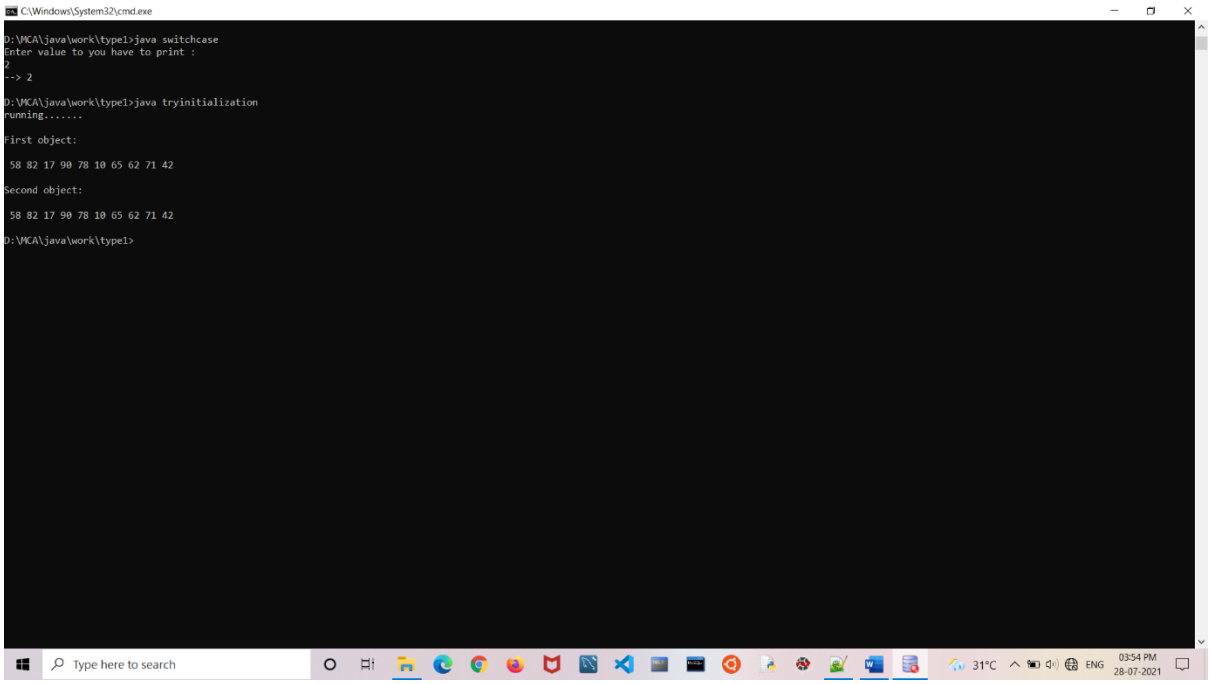
```
                System.out.println(i+" ");
                i++;
            }while(i<=10);
        }
}

class switchcase
{
        public static void main(String args[])
        {
            Scanner n1=new Scanner(System.in);
            System.out.println("Enter value to you have to print :");
            int n=n1.nextInt();

            if(n<=5)
            {
                switch(n)
                {
                    case 1:
                        System.out.println("--> 1");
                        break;
                    case 2:
                        System.out.println("--> 2");
                        break;
```

```
                        case 3:
                                System.out.println("--> 3");
                                break;
                        case 4:
                                System.out.println("--> 4");
                                break;
                        case 5:
                                System.out.println("--> 5");
                                break;
                        }
                }

        }
}
class tryinitialization
{
        static int value[]=new int[10];

        static
        {
                System.out.println("running.....");
                for(int i=0;i<value.length;i++)
                {
                        value[i]=(int)(100.0*Math.random());
```

```
            }

        }
        void list()
        {
            System.out.println();
            for(int nvalue:value)
            {
                System.out.print(" "+nvalue);
            }
            System.out.println();
        }
        public static void main(String[] args)
        {
            tryinitialization example = new tryinitialization();
            System.out.println("\nFirst object:");
            example.list();
            tryinitialization nextexample = new tryinitialization();
            System.out.println("\nSecond object:");
            nextexample.list();

        }
}
class some
```

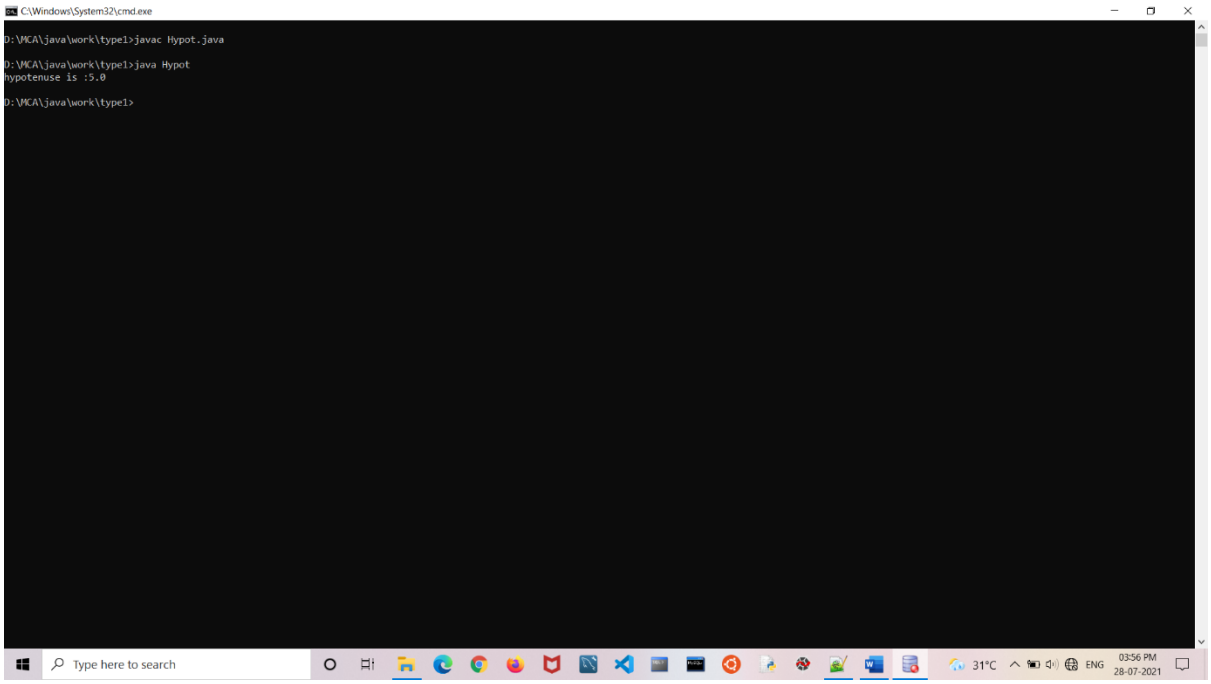
```
{
	public static void main(String args[])
	{
		Rectangle rect;
		rect.width=40;


	}
}
```





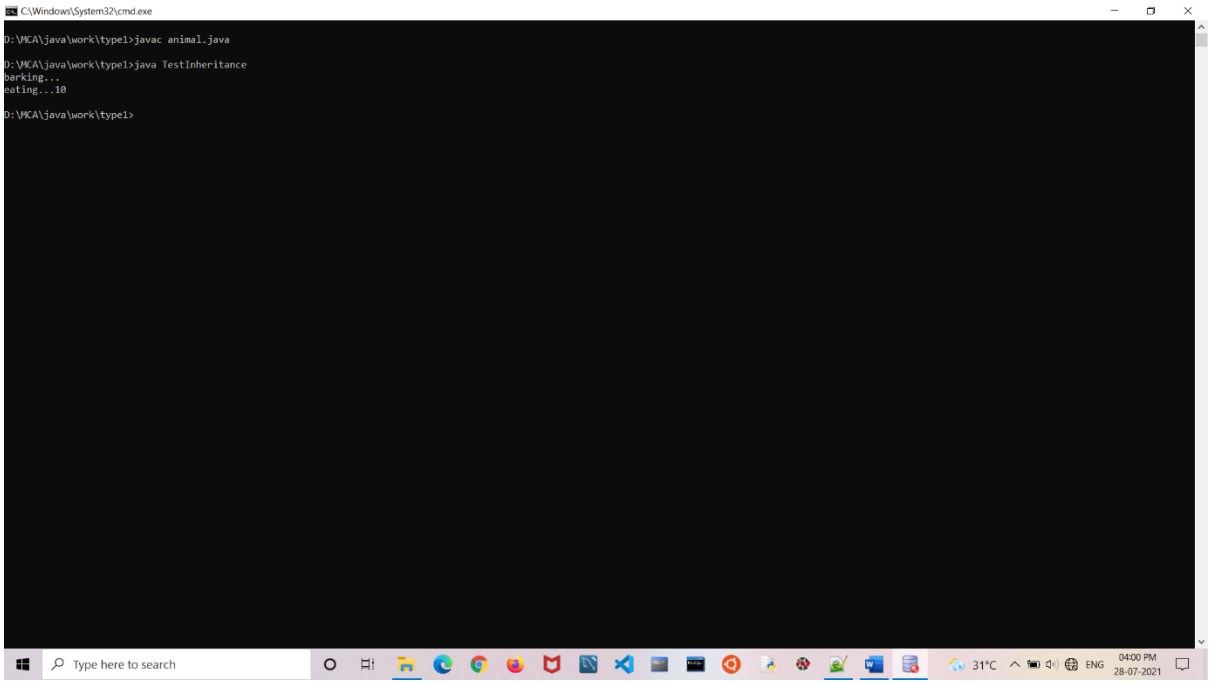
```
/*use the pythagorean theorem to
find the length of the hypotenuse
given the lengths of the two opposing sides.
*/
class Hypot{
    public static void main(String args[])
    {
        double x,y,z;
        x=3;
        y=4;
        z=java.lang.Math.sqrt(x*x + y*y);

        System.out.println("hypotenuse is :" + z );
```

```
    }
}
```



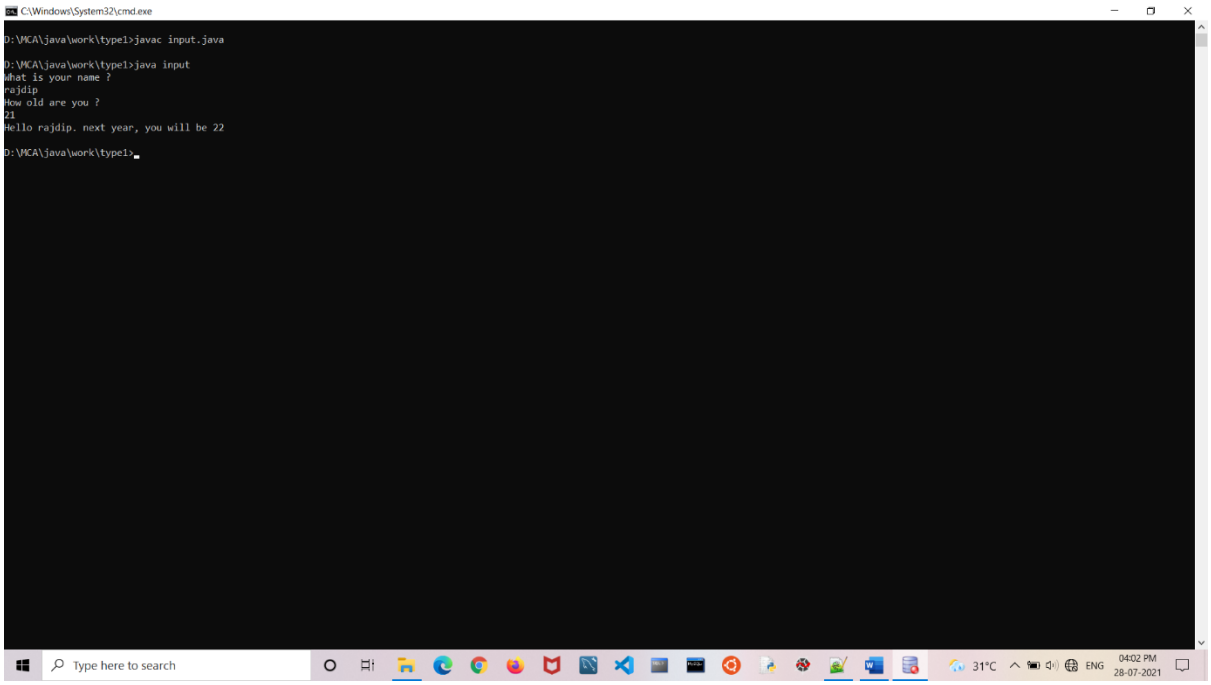
```
class Animal
{
        int a=10;
        void eat()
        {
                System.out.println("eating..."+a);
        }
}
class Dog extends Animal{
void bark(){System.out.println("barking...");}
}
```

```
class TestInheritance{
public static void main(String args[]){
Dog d=new Dog();
d.bark();
d.eat();
}}
```



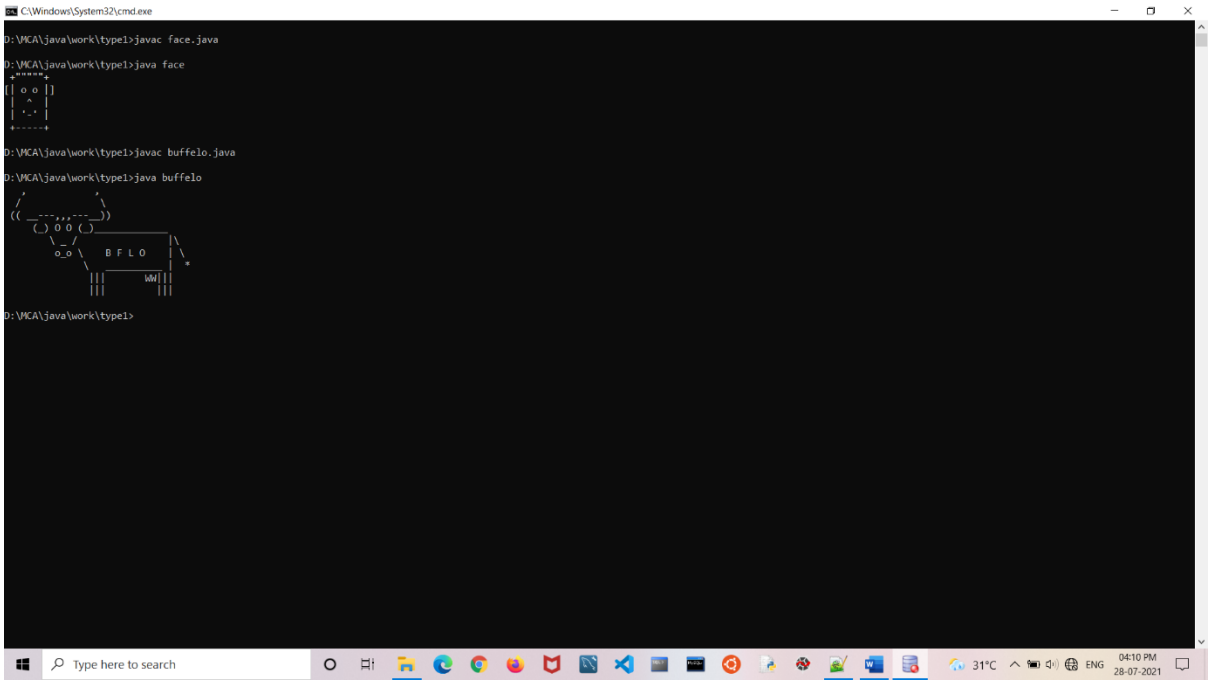
```
import java.util.Scanner;
public class input
{
	public static void main(String args[])
	{
		Scanner in=new Scanner(System.in);
```

```
            System.out.println("What is your name ? ");
            String name=in.nextLine();
            System.out.println("How old are you ? ");
            int age=in.nextInt();

            System.out.println("Hello "+name+". next year, you will be "+(age+1));
        }
}
```



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

```
            System.out.println(" +\"\"\"\"\"+");
            System.out.println("[| o o |]");
            System.out.println(" |  ^  | ");
            System.out.println(" | '-' | ");
            System.out.println(" +-----+ ");
        }
}
class buffelo
{
        public static void main(String args[])
        {
            System.out.println("   ,          , ");
            System.out.println("  /            \\");
            System.out.println(" (( __---,,,---__))");
            System.out.println("    (_) O O (_)_____________");
            System.out.println("       \\ _ /              |\\");
            System.out.println("        o_o \\    B F L O    | \\");
            System.out.println("             \\   __________ |  *");
            System.out.println("              |||       WW|||");
            System.out.println("              |||        |||");
        }
}
```

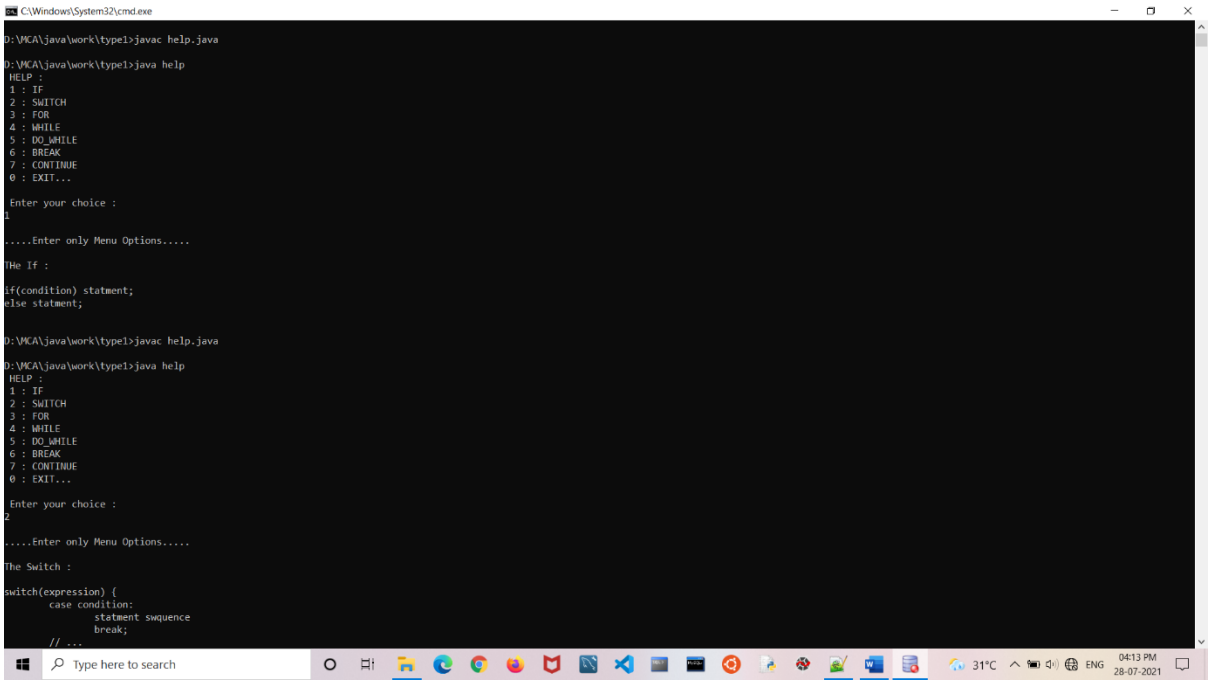


```
class help
{
	public static void main(String args[])
		throws java.io.IOException
		{
			char choice,ignore;
			do
			{
				System.out.println(" HELP : ");
				System.out.println(" 1 : IF ");
				System.out.println(" 2 : SWITCH ");
				System.out.println(" 3 : FOR ");
				System.out.println(" 4 : WHILE ");
```

```
                System.out.println(" 5 : DO_WHILE ");
                System.out.println(" 6 : BREAK ");
                System.out.println(" 7 : CONTINUE ");
                System.out.println(" 0 : EXIT...");

                System.out.println("\n Enter your choice : ");
                choice=(char)System.in.read();
                do
                {
                        ignore=(char)System.in.read();
                }while(ignore!='\n');
                if(choice!='0')
                {
                        System.out.println("\n.....Enter only
Menu Options.....\n");
                }
                else
                {
                        System.out.println("\n ......Exit.....");
                }
        }while(choice!='0' & choice<='0'|choice>='8');

        switch(choice)
        {
```

```
                    case '1':
                        System.out.println("THe If :\n");
                        System.out.println("if(condition)
statment;");
                        System.out.println("else statment;");
                        break;
                    case '2':
                        System.out.println("The Switch :\n");
                        System.out.println("switch(expression)
{");
                        System.out.println("   case condition:");
                        System.out.println("          statment
swquence");
                        System.out.println("          break;");
                        System.out.println("   // ...");
                        System.out.println("}");
                        break;
                    case '3':
                        System.out.println("The For :\n");

        System.out.println("for(init;condition;iteration)");
                        System.out.println("   statment;");
                        break;
                    case '4':
                        System.out.println("The While :\n");
```

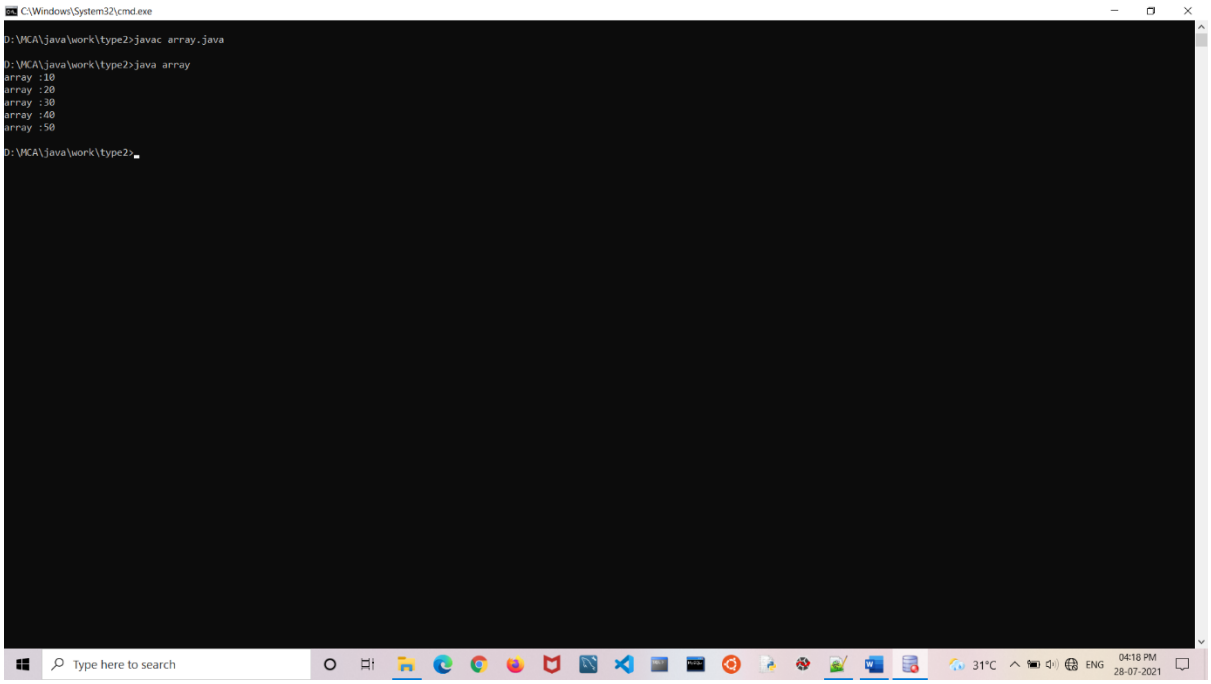
```
                    System.out.println("while(condition)
statment;");
                    break;
                case '5':
                    System.out.println("The do-while :\n");
                    System.out.println("do {");
                    System.out.println(" statment");
                    System.out.println("} while (condition);");
                    break;
                case '6':
                    System.out.println("The break :\n");
                    System.out.println("break; or break label
;");
                    break;
                case '7':
                    System.out.println("The continue :\n");
                    System.out.println("continue; or
continue label;");
                    break;
                case '0':
                    break;
            }
            System.out.println();
        }
}
```


C:\Windows\System32\cmd.exe
D:\MCA\java\work\type1>javac help.java
D:\MCA\java\work\type1>java help
HELP :
1 : IF
2 : SWITCH
3 : FOR
4 : WHILE
5 : DO_WHILE
6 : BREAK
7 : CONTINUE
0 : EXIT...
Enter your choice :
1
.....Enter only Menu Options.....
THe If :
if(condition) statment;
else statment;
D:\MCA\java\work\type1>javac help.java
D:\MCA\java\work\type1>java help
HELP :
1 : IF
2 : SWITCH
3 : FOR
4 : WHILE
5 : DO_WHILE
6 : BREAK
7 : CONTINUE
0 : EXIT...
Enter your choice :
2
.....Enter only Menu Options.....
The Switch :
switch(expression) {
case condition:
statment swquence
break;
// ...
Type here to search
31°C
ENG
04:13 PM
28-07-2021

Class Work – Arrays and sorting and Oprations

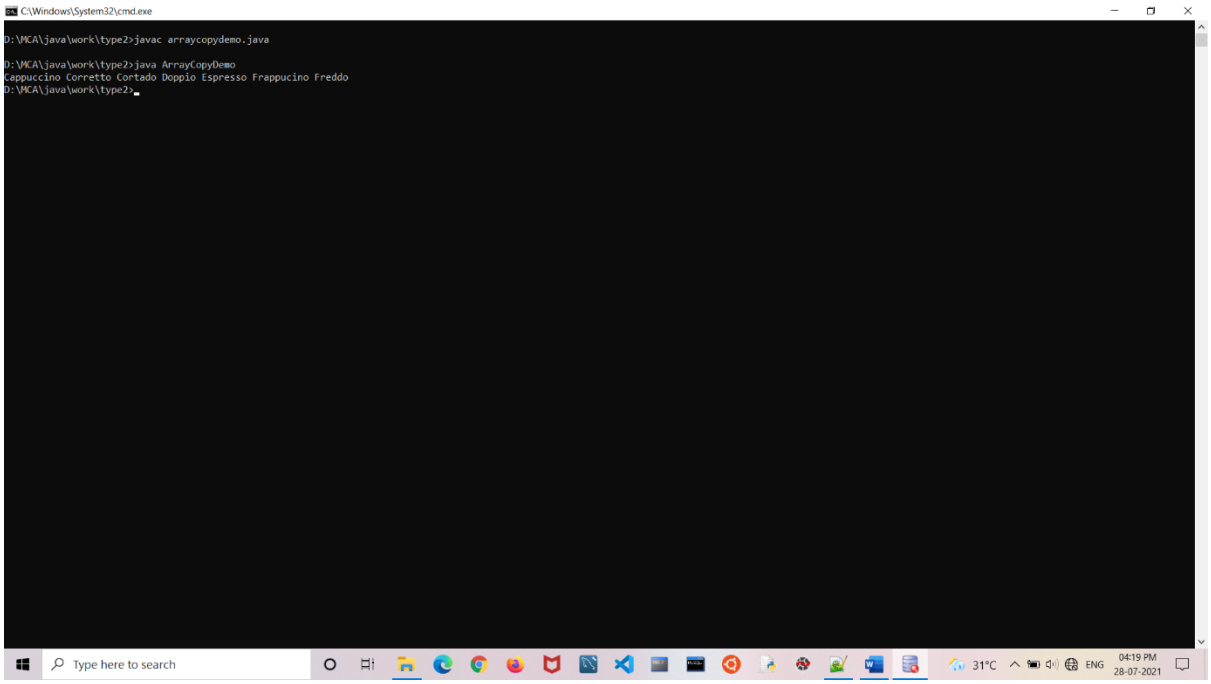
```
class array
{
	public static void main(String args[])
	{
		int a[]={10,20,30,40,50};
		int b;
		for(int i=0;i<a.length;i++)
		{
			for(int j=i+1;j<a.length;j++)
			{
				if(a[j]<a[i])
				{
					b=a[j];
					a[j]=a[i];
					a[i]=b;
				}
			}
		}
		for(int i=0;i<a.length;i++)
		{
```

```
                System.out.println("array :"+a[i]);
            }
        }
}
```



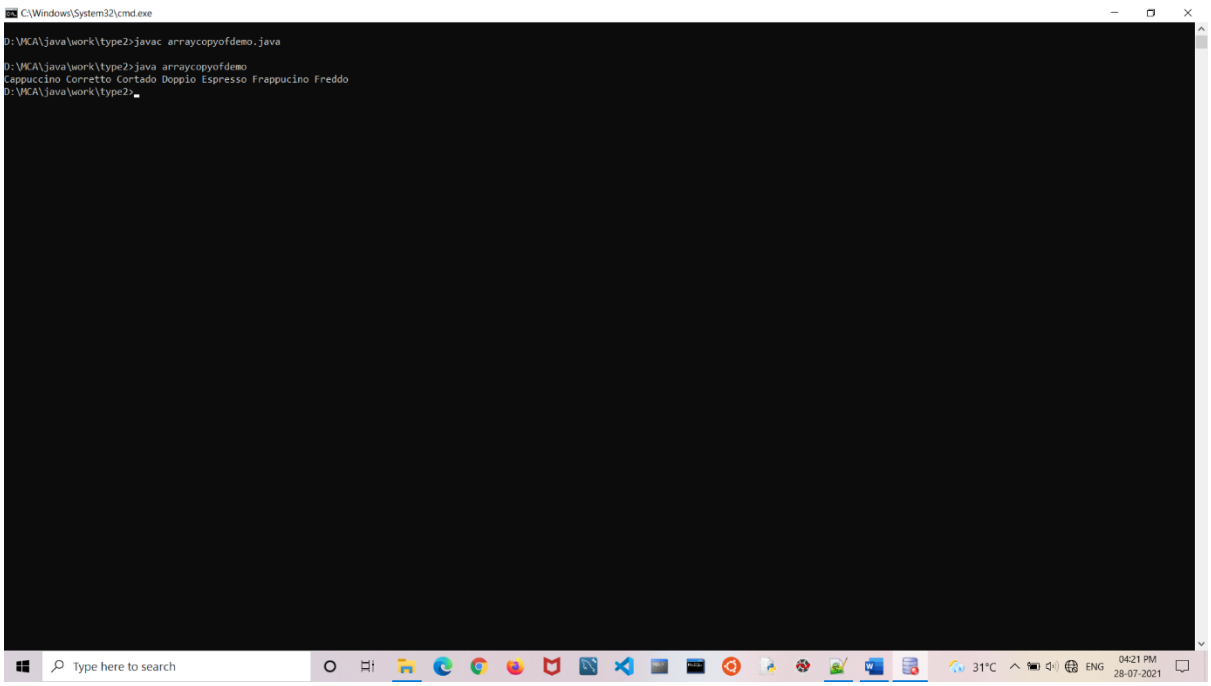
```
class ArrayCopyDemo
{
   public static void main(String[] args)
        {
      String[] copyFrom = {
          "Affogato", "Americano", "Cappuccino", "Corretto",
"Cortado",
          "Doppio", "Espresso", "Frappucino", "Freddo", "Lungo",
"Macchiato",
```

```
        "Marocchino", "Ristretto" };

    String[] copyTo = new String[7];
    System.arraycopy(copyFrom, 2, copyTo, 0, 7);
    for (String coffee : copyTo)
            {
        System.out.print(coffee + " ");
    }
  }
}
```



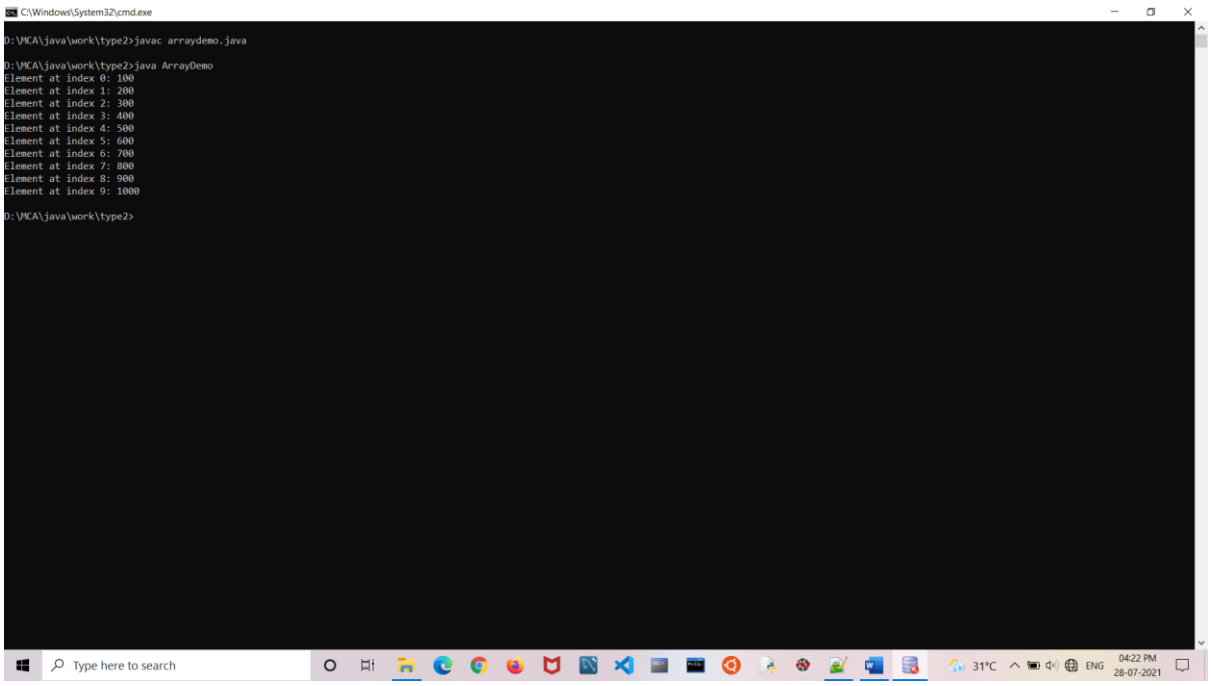
```
class ArrayCopyDemo
{
   public static void main(String[] args)
```

```
        {
        String[] copyFrom = {
            "Affogato", "Americano", "Cappuccino", "Corretto",
"Cortado",
            "Doppio", "Espresso", "Frappucino", "Freddo", "Lungo",
"Macchiato",
            "Marocchino", "Ristretto" };


        String[] copyTo = new String[7];
        System.arraycopy(copyFrom, 2, copyTo, 0, 7);
        for (String coffee : copyTo)
                {
            System.out.print(coffee + " ");
        }
    }
}
```



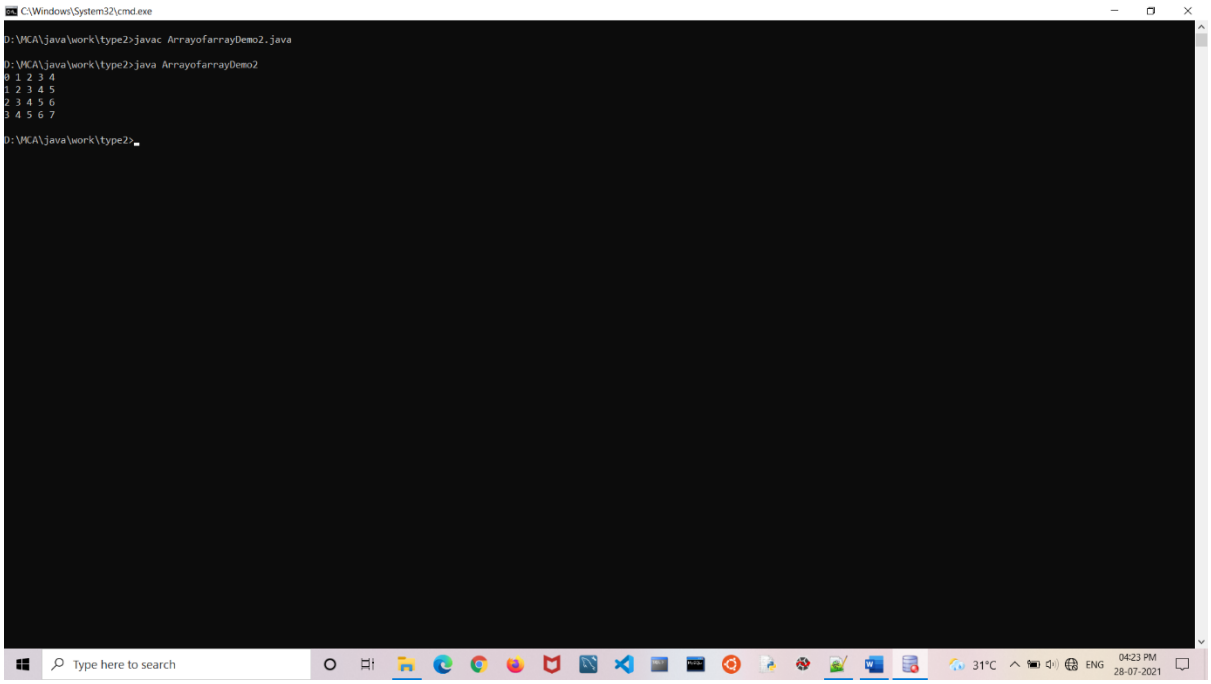
```
class ArrayDemo
{
   public static void main(String[] args)
        {
       // declares an array of integers
       int[] anArray;

       // allocates memory for 10 integers
       anArray = new int[10];

       // initialize first element
       anArray[0] = 100;
       // initialize second element
       anArray[1] = 200;
       // and so forth
       anArray[2] = 300;
       anArray[3] = 400;
       anArray[4] = 500;
       anArray[5] = 600;
       anArray[6] = 700;
       anArray[7] = 800;
       anArray[8] = 900;
       anArray[9] = 1000;
```

```
        System.out.println("Element at index 0: "+ anArray[0]);
        System.out.println("Element at index 1: "+ anArray[1]);
        System.out.println("Element at index 2: "+ anArray[2]);
        System.out.println("Element at index 3: "+ anArray[3]);
        System.out.println("Element at index 4: "+ anArray[4]);
        System.out.println("Element at index 5: "+ anArray[5]);
        System.out.println("Element at index 6: "+ anArray[6]);
        System.out.println("Element at index 7: "+ anArray[7]);
        System.out.println("Element at index 8: "+ anArray[8]);
        System.out.println("Element at index 9: "+ anArray[9]);
    }
}
```

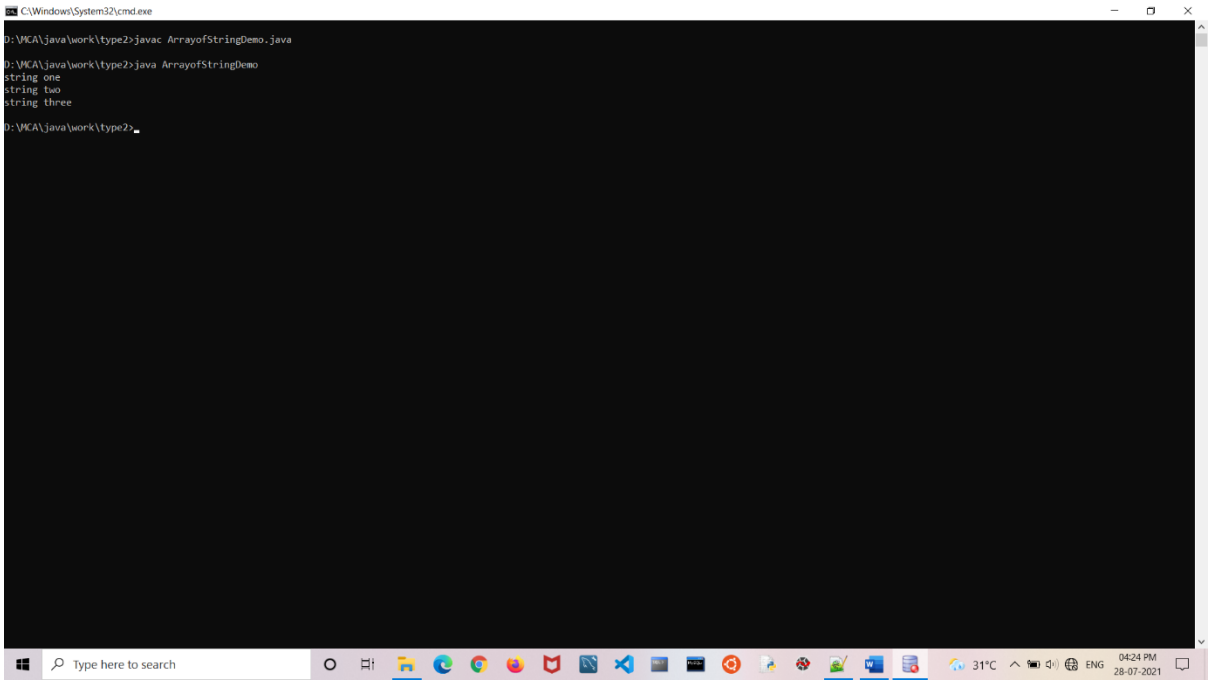


```
---------------------------------------------------------------------------------------
/*Software Even If adviced of the possubility of such Damage*/
public class ArrayofarrayDemo2{
   public  static void main(String args[])
   {
      int[][] amatrix=new int[4][];
      //populate matrix
      for(int i=0;i<amatrix.length;i++)
      {
         amatrix[i] =new int[5];//Create sub-array
         for(int j=0;j<amatrix[i].length;j++)
         {
            amatrix[i][j] =i + j;
         }
      }
      //print matrix
      for(int i=0;i<amatrix.length;i++)
      {
         for(int j=0;j<amatrix[i].length;j++)
         {
            System.out.print(amatrix[i][j] + " ");
         }
         System.out.println();
      }
```

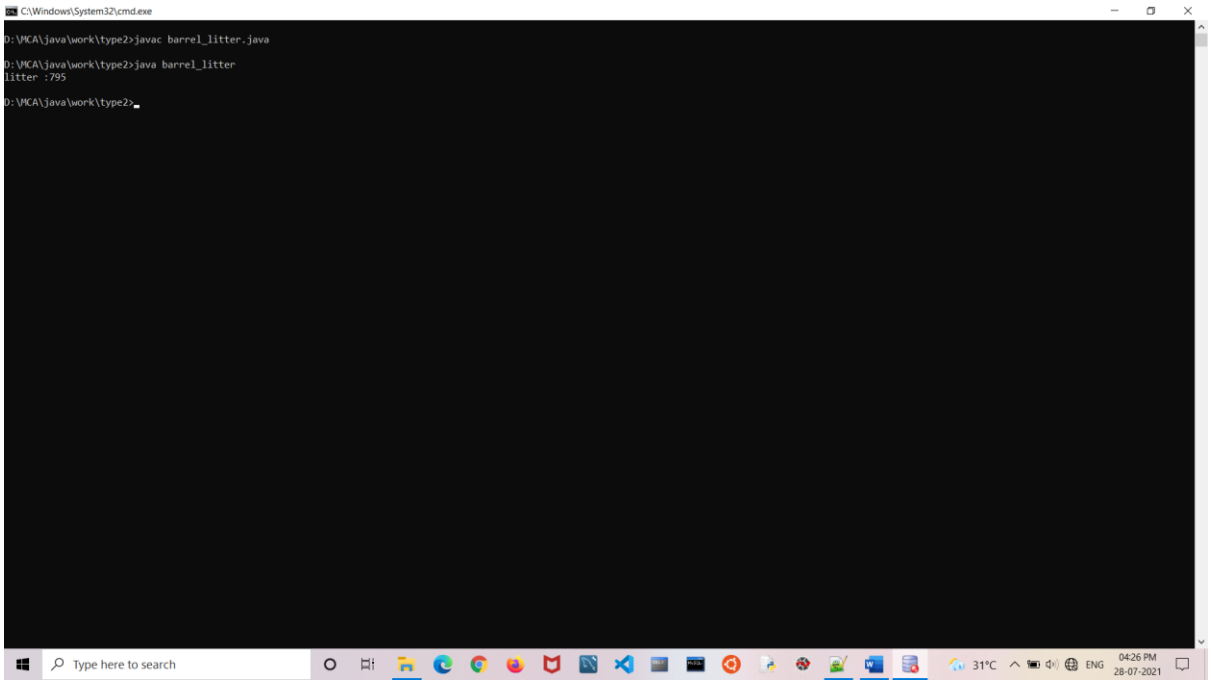
```
    }
}
```



```
public class ArrayofStringDemo{
    public static void main(String args[])
    {
        String[] anArray={"String one","String two","String Three"};
        for(String S:anArray)
        {
            System.out.println(S.toLowerCase());
         }
    }
}
```



```
class barrel_litter
{
	public static void main(String args[])
	{
		int barrel=5;
		int litter=barrel*159;
		System.out.println("litter :"+litter);
	}
}
```

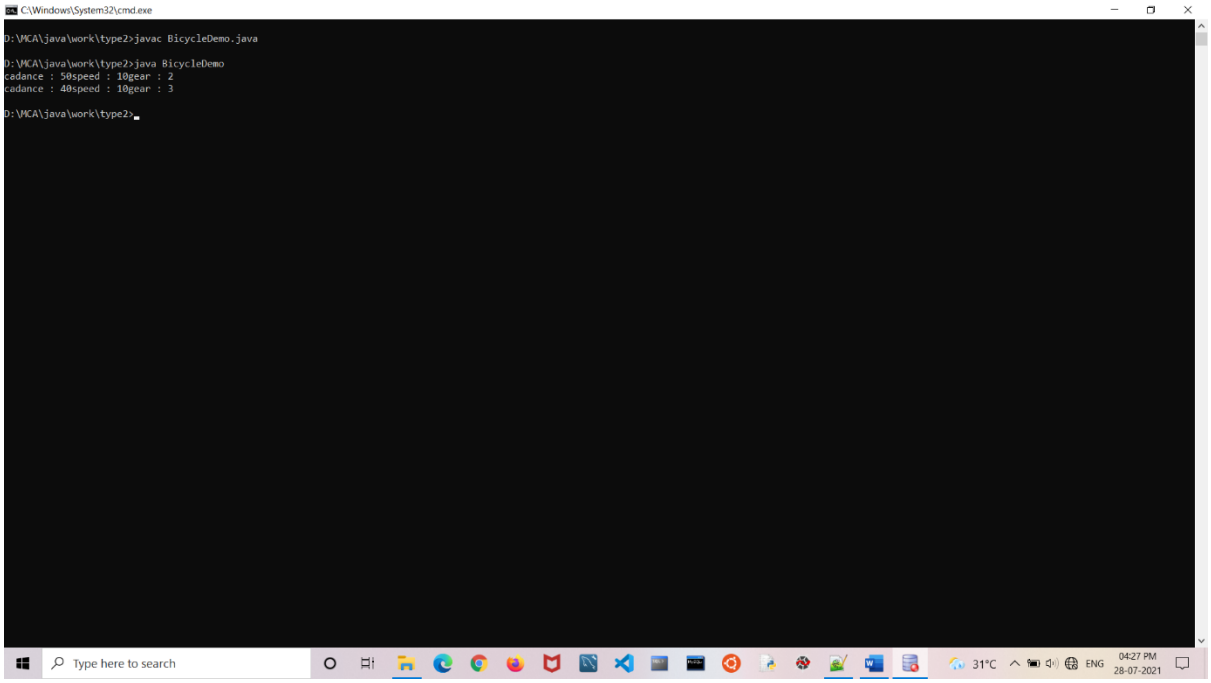


```
class Bicycle
{
        public int cadence=0;
         int speed=0;
         int gear=1;

        void changeCadence(int newValue)
        {
                cadence = newValue;
        }

        void changeGear(int newValue)
        {
```

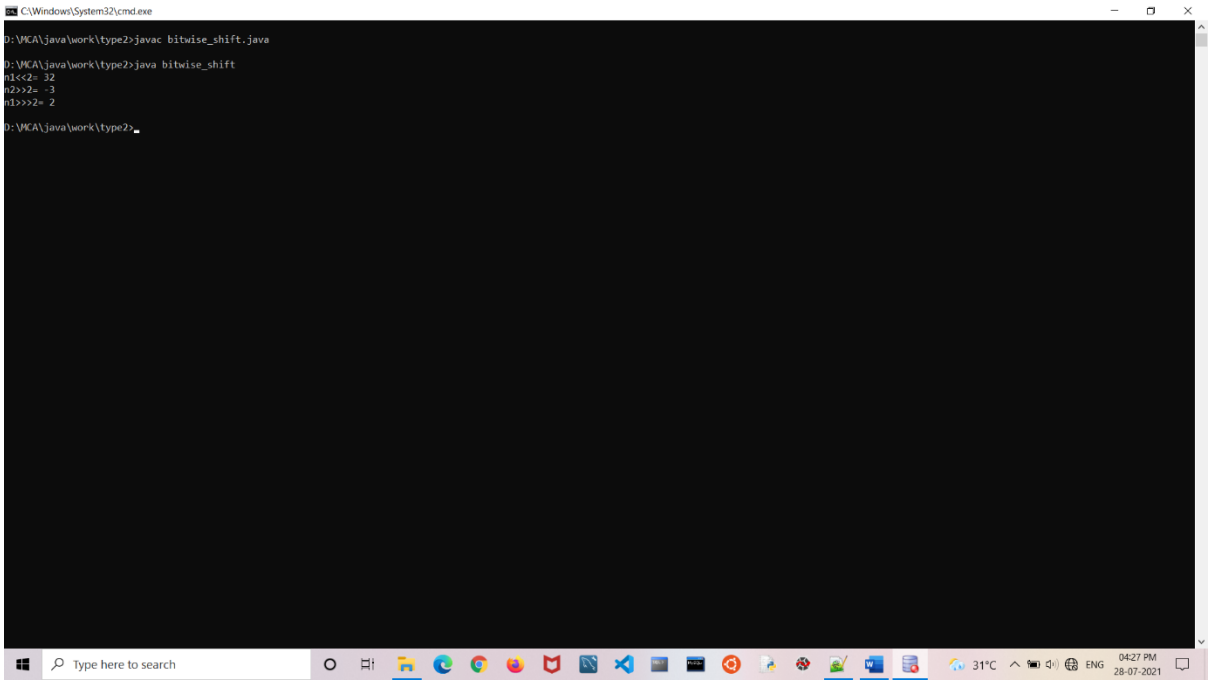
```
        gear = newValue;
    }

    void speedUp(int increment)
    {
        speed = speed+increment;
    }

    void applyBrakes(int decrement)
    {
                speed = speed-decrement;

    }

    void printStates()
    {
        System.out.println("cadance : " + cadence + "speed : " +
speed + "gear : " + gear);
    }
}

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

```
{
        Bicycle bike1= new Bicycle();
        Bicycle bike2 = new Bicycle();

        bike1.changeCadence(50);
        bike1.speedUp(10);
        bike1.changeGear(2);
        bike1.printStates();

        bike2.changeCadence(50);
        bike2.speedUp(10);
        bike2.changeGear(2);
        bike2.changeCadence(40);
        bike2.changeGear(3);
        bike2.printStates();
}
```

}

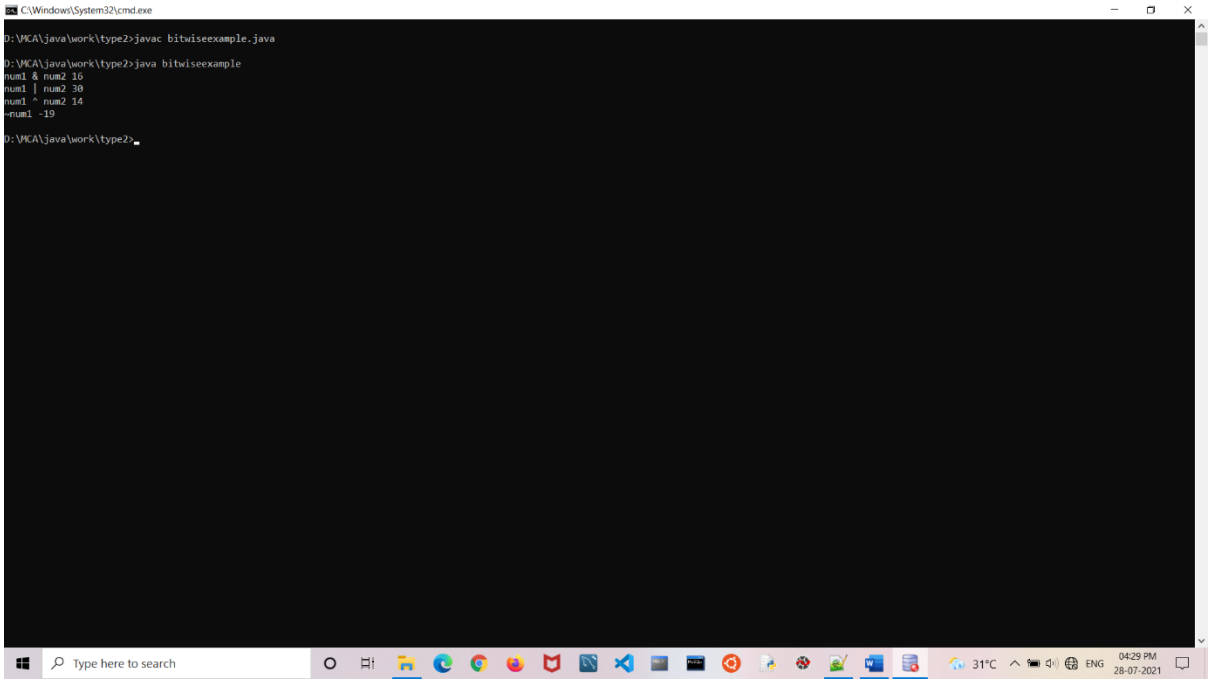


```
public class bitwise_shift
{
	public static void main(String args[])
	{
		int n1=8;
		int n2=-10;
		System.out.println("n1<<2= "+(n1<<2));
		System.out.println("n2>>2= "+(n2>>2));
		System.out.println("n1>>>2= "+(n1>>>2));
	}
}
```



```
public class bitwiseexample
{
	public static void main(String args[])
	{
		int n1=18,n2=28,n3=0;
		System.out.println("num1 & num2 "+(n1&n2));
		System.out.println("num1 | num2 "+(n1|n2));
		System.out.println("num1 ^ num2 "+(n1^n2));
		System.out.println("~num1 "+~n1);
	}
```

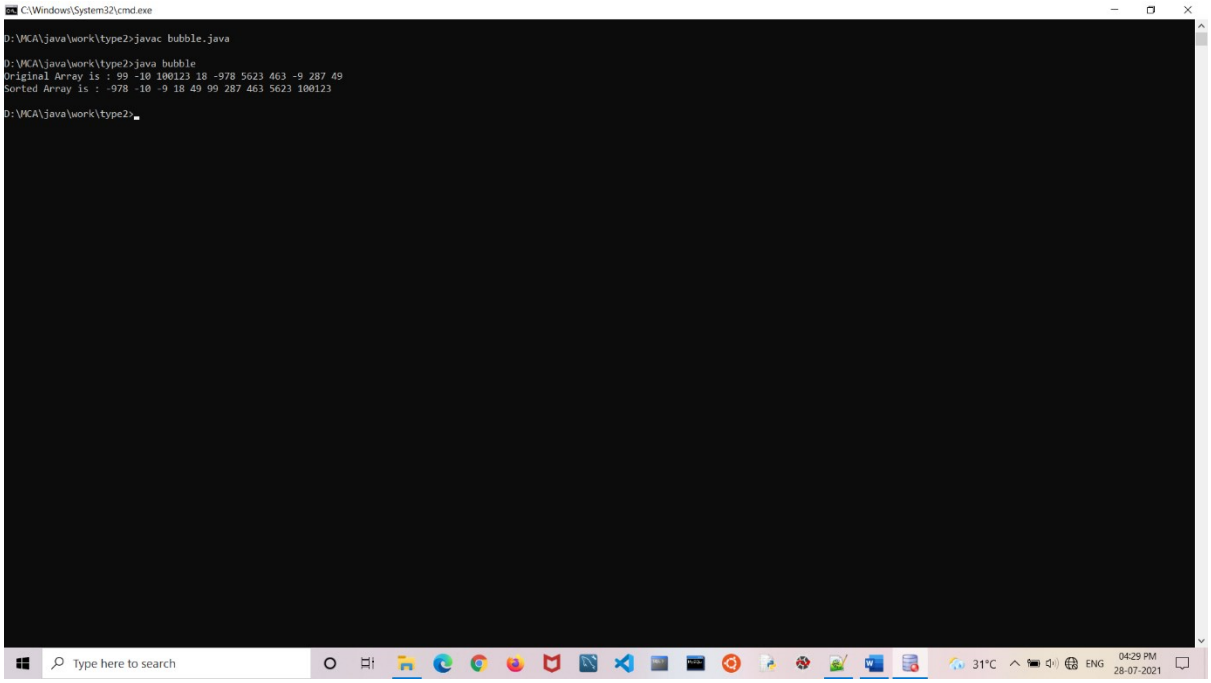
}



```
class bubble
{
	public static void main(String args[])
	{
		int nums[]={99,-10,100123,18,-978,5623,463,-9,287,49};
		int a,b,t;
		int size;
		size=10;
		System.out.print("Original Array is :");
		for(int i=0;i<size;i++)
		{
			System.out.print(" "+nums[i]);
```

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

        for(a=1;a<size;a++)
        {
                for(b=size-1;b>=a;b--)
                {
                        if(nums[b-1]>nums[b])
                        {
                                t=nums[b-1];
                                nums[b-1]=nums[b];
                                nums[b]=t;
                        }
                }
        }
        System.out.print("Sorted Array is :");

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

}

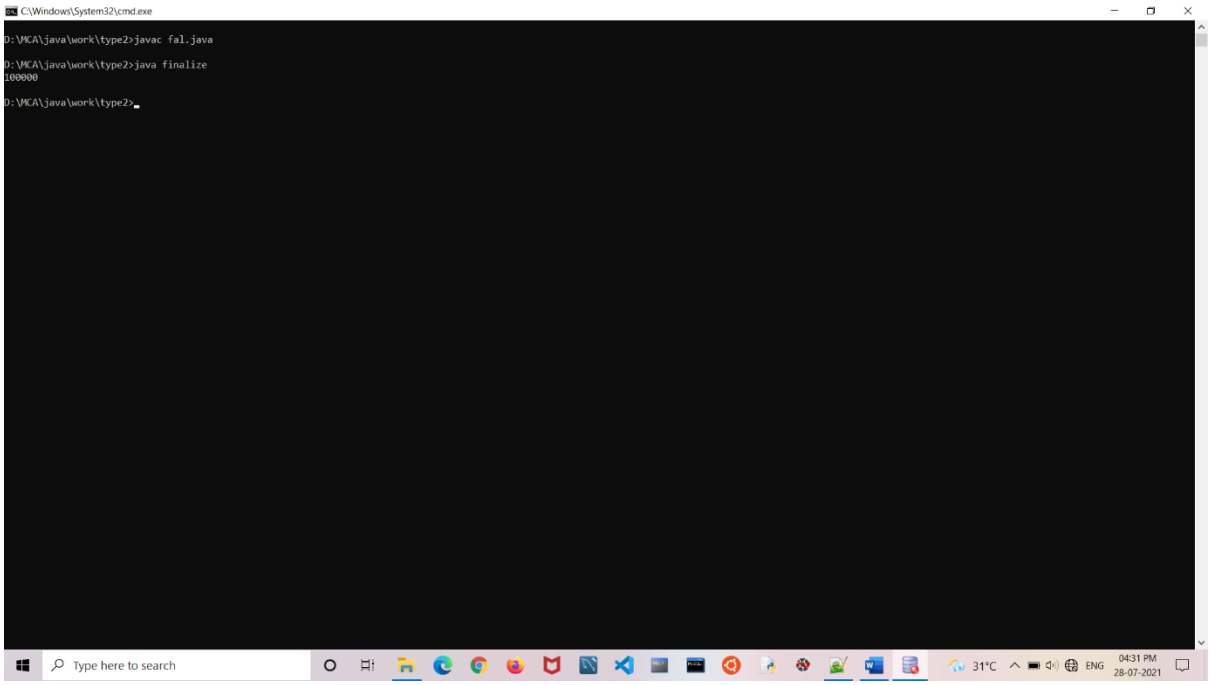


```
class fal
{
	public static void main(String args[])
	{
	double gallons,liters;
	int counter;
	counter=0;

	for(gallons=1;gallons<=100;gallons++)
	{
		liters=gallons*3.7854; //convert to liters
```

```
            System.out.println(gallons+ " gallons is "+ liters+ " liters. "
);
            counter++;
            if(counter == 10)
            {
                  System.out.println();
                  counter=0;
            }
      }
      }
}

class sound
{
      public static void main(String args[])
      {
            double dist;
            dist=7.2*1100;
            System.out.println("the lightning is " +dist+ " feet away.");
      }
}
```

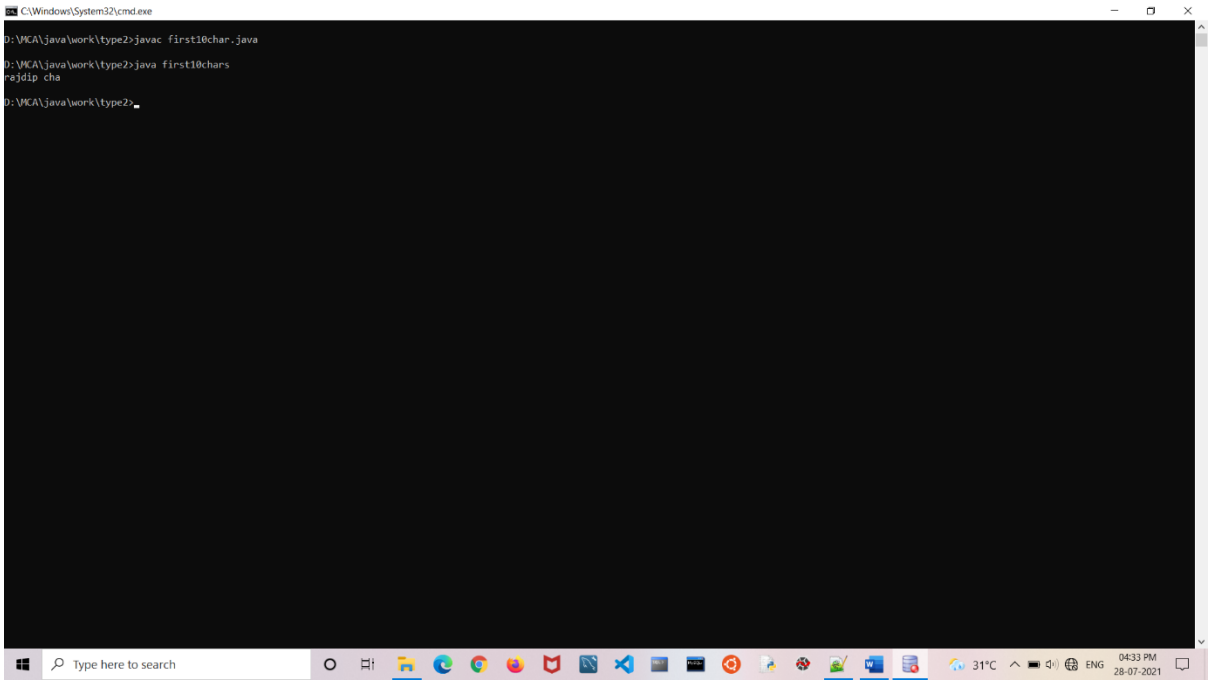
```
class fdemo
{
	int x;
	fdemo(int i)
	{
		x=i;
	}
	protected void finalize()
	{
		System.out.println("finalizing "+ x);
	}
	void generator(int i)
	{
		fdemo o=new fdemo(i);
	}
}
class finalize
{
	public static void main(String args[])
	{
		int count;
		fdemo ob=new fdemo(0);
		for(count=1;count<100000;count++)
		{
```

```
                ob.generator(count);
            }
            System.out.println(count);
        }
}
```



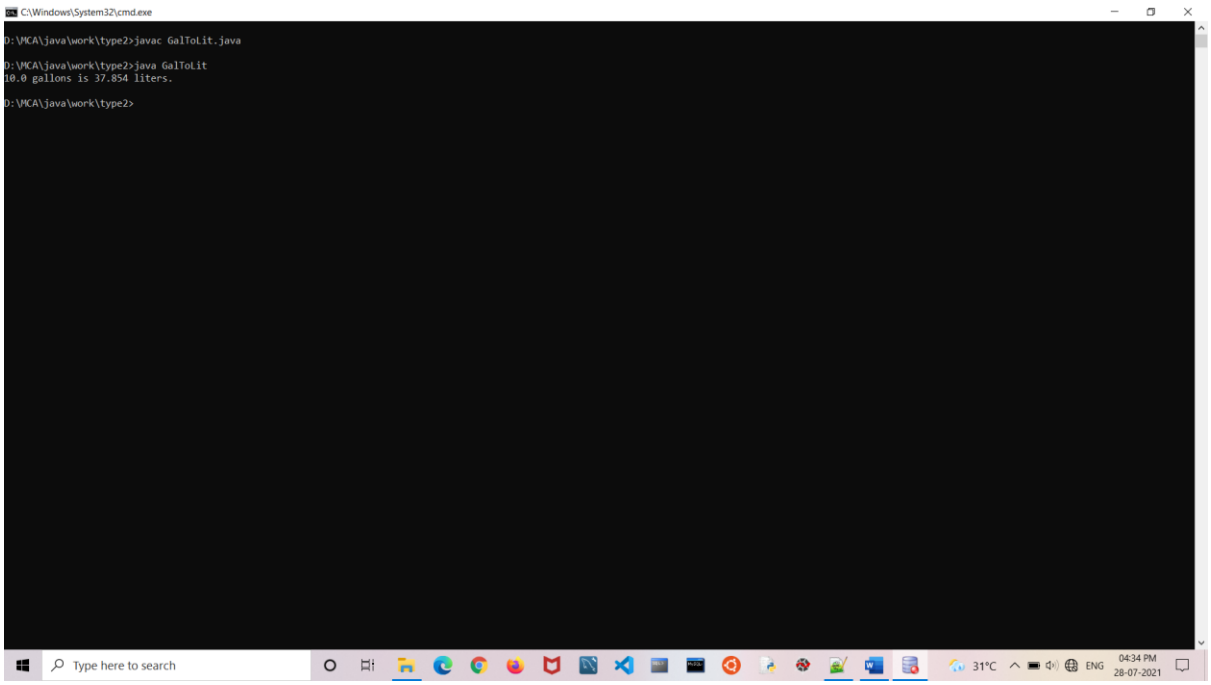
```
class first10chars
{
    public static void main(String args[])
    {
        String s="rajdip chavda daxesh chavda";
        String substring=s.substring(0,10);
        System.out.println(substring);
    }
```

}



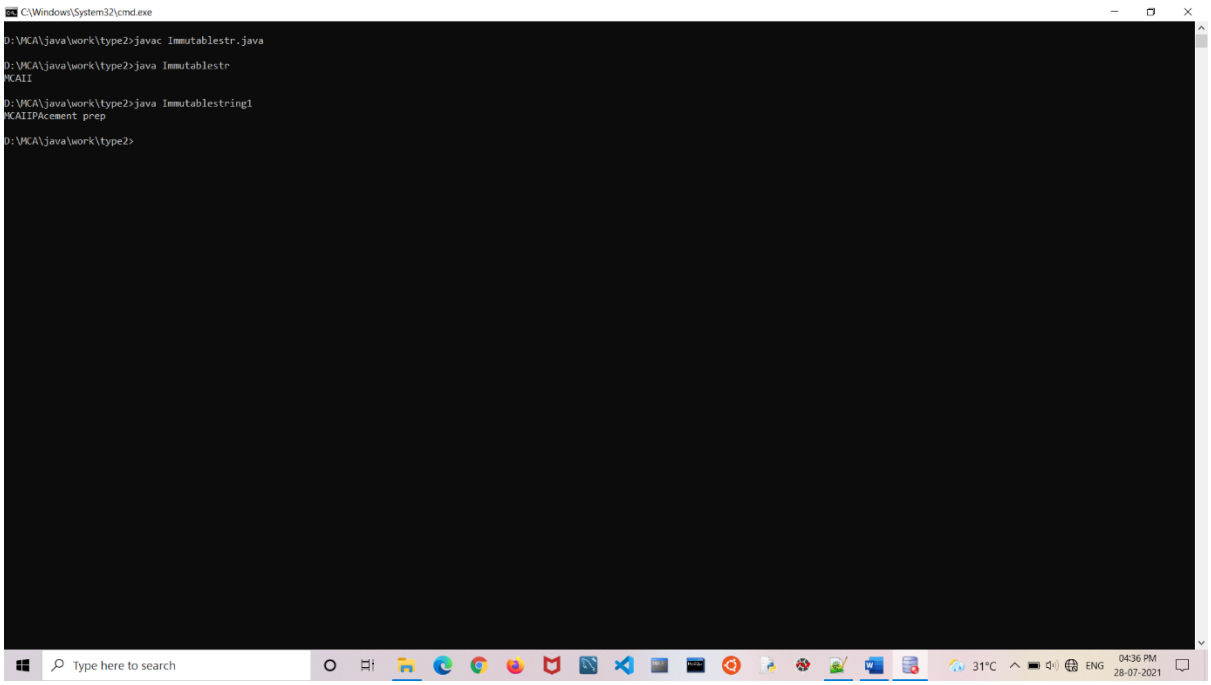
```
/*
Thios program converts gallons to liters
*/
public class GalToLit
{

      public static void main(String[]  args)
      {
     double gallons; //holds the numbers of gallons
     double liters;  //holds the numbers of liters

     gallons = 10;
```

```
        liters = gallons * 3.7854; //convert to liters
                System.out.println(gallons + " gallons is "+liters+" liters.");
         }
}
```



```
class Immutablestr{
   public static void main(String args[])
   {
      String s="MCAII";
      s.concat("PAcement prep");
      System.out.println(s);
   }
}
```

```
class Immutablestring1{
    public static void main(String args[])
    {
        String s="MCAII";
        s=s.concat("PAcement prep");
        System.out.println(s);
    }
}
```



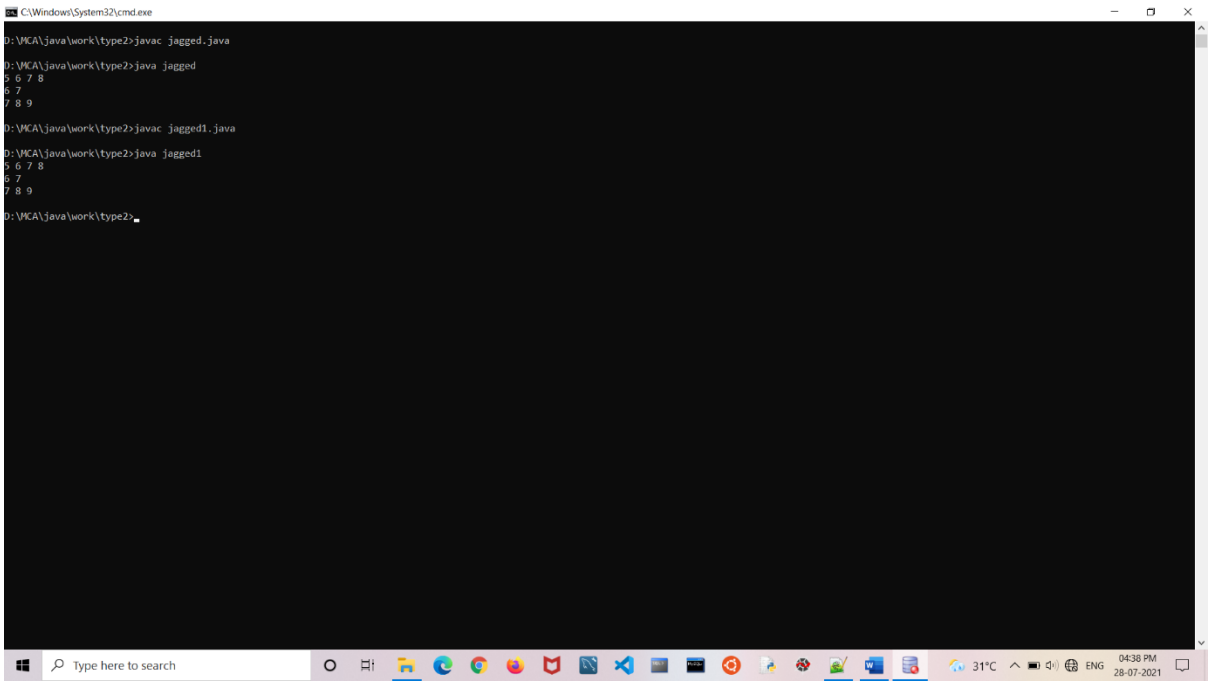
```
public class jagged
{
        public static void main(String args[])
        {
        /*      int a=5,b=6,c=7,d=8;
```

```
        int x[]={5,6,7,8};
        int y[]={6,7,8,9};
        int z[]={7,8,9,6};

        int p[][]={
                        {5,6,7,8},
                        {6,7,8,9},
                        {7,8,9,6}
        };

        for(int i[] : p)
        {
                for(int j : i )
                {
                        System.out.print(j + " ");
                }
        }
}
        System.out.println(""); */

        int a=5,b=6,c=7,d=8;
        int x[]={5,6,7,8};
        int y[]={6,7};
        int z[]={7,8,9};
```

```
        int k[][] = new int[3][];

        k[0] = new int[4];
        k[1] = new int[2];
        k[2] = new int[3];

        int p[][] ={
                        {5,6,7,8},
                        {6,7},
                        {7,8,9}
        };

        for(int i[] : p)
        {
            for(int j : i)
            {
                System.out.print(j + " ");
            }

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

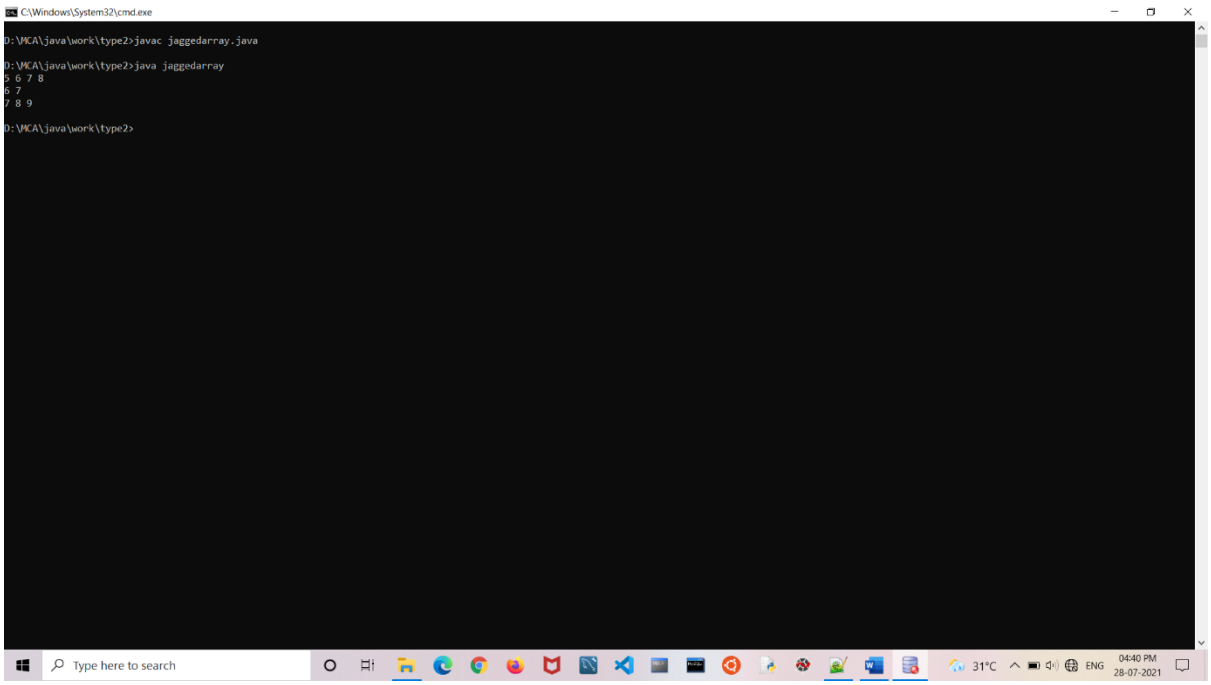
```
--------------------------------------------------------------------------------
public class jagged1{
   public static void main(String args[])
   {
      int a=5,b=6,c=7,d=8;
      int x[]={5,6,7,8};
      int y[]={6,7};
      int z[]={7,8,9};

      //create array with name K
      int k[][]=new int [3][];

      //the problem starts here ,if i know the size of my array is 3row
and 4 cols and i havve
      //we don't know number number of cols so remove
      //but how to assign value
      k[0]=new int [4];
      k[1]=new int [2];
      k[2]=new int [3];

      int p[][]={{5,6,7,8},{6,7},{7,8,9} };
      for(int i[] : p)
      {
         for(int j : i)
```

```
        {
            System.out.print(j+" ");
        }
        System.out.println();

    }

  }
}
```



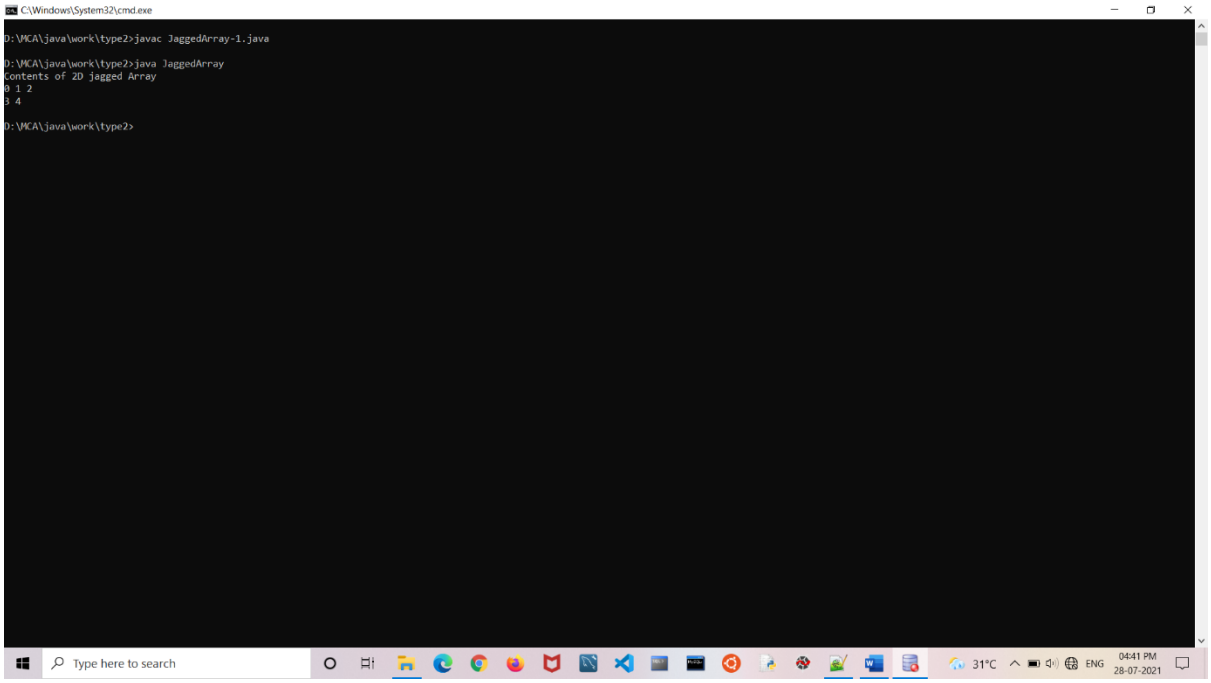
```
class jaggedarray
{
    public static void main(String args[])
    {
```

```
            int p[][]={{5,6,7,8},{6,7},{7,8,9}};
            for(int i[]:p)
            {
                  for(int j:i)
                  {
                        System.out.print(j+" ");
                  }
                  System.out.println("");
            }
      }
}
```



```
class JaggedArray
{
	public static void main(String args[])
	{
		//declaring 2D arrays with 2 rows
		int arr[][] = new int[2][];

		// Making the above array jagged

		//first row has 3 columns
		arr[0] = new int[3];

		//second rows has 2 columns
		arr[1] = new int[2];

		int count = 0;
		for(int i=0 ; i<arr.length ; i++)
			for(int j=0 ; j<arr[i].length ; j++)
				arr[i][j] = count++;

		System.out.println("Contents of 2D jagged Array");
		for(int i=0; i<arr.length ; i++)
		{
			for(int j=0 ; j<arr[i].length; j++)
```

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

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

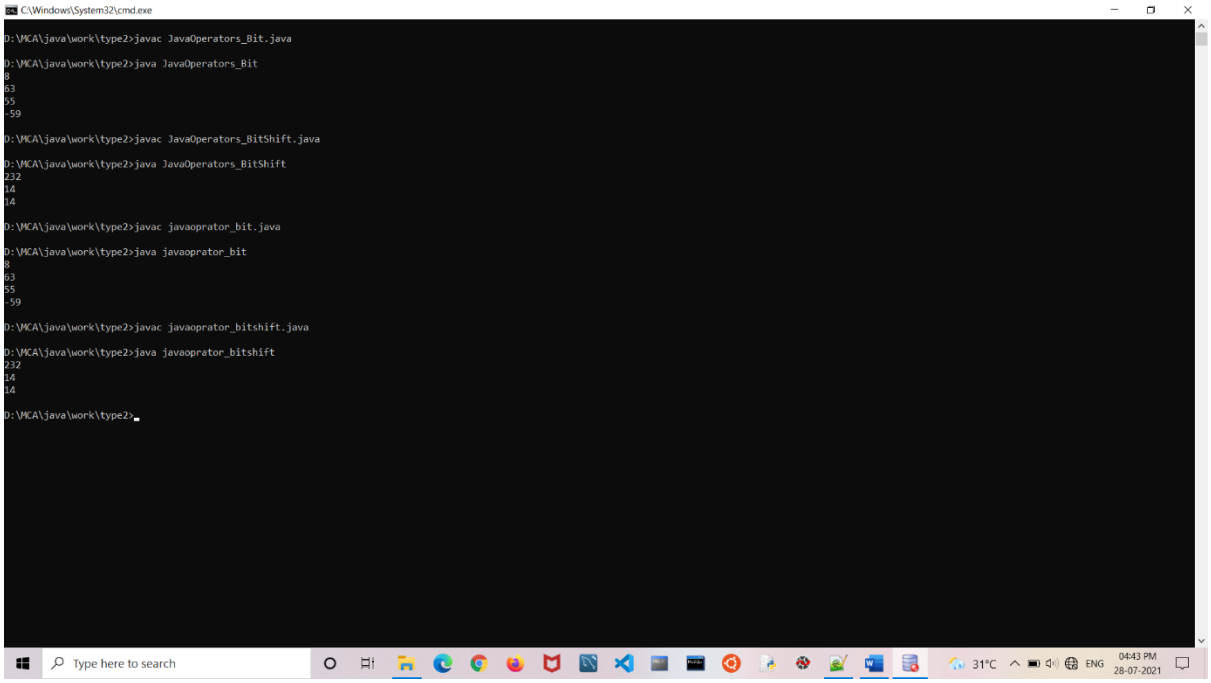


```
public class JavaOperators_Bit
{
    public static void main(String args[])
    {
        int a=58;
        int b=13;
```

```
            System.out.println(a&b);
            System.out.println(a|b);
            System.out.println(a^b); //bitwise XOR
            System.out.println(~a); // binary compliment operator
        }
}
//program for right shift , left shift and unsigned right operator
//syntax is.....  number shift_operator number_of_places_to_shift

public class JavaOperators_BitShift
{
        public static void main(String args[])
        {
            int a=58;
            System.out.println(a<<2);   //left shift
            System.out.println(a>>2);   //right shift
            System.out.println(a>>>2);  //unsigned right shift

        }
}

public class javaoprator_bit
{
```

```
	public static void main(String args[])
	{
		int a=58;
		int b=13;
		System.out.println(a&b);
		System.out.println(a|b);
		System.out.println(a^b);
		System.out.println(~a);
	}
}

public class javaoprator_bitshift
{
	public static void main(String args[])
	{
		int a=58;
		System.out.println(a<<2);
		System.out.println(a>>2);
		System.out.println(a>>>2);
	}
```

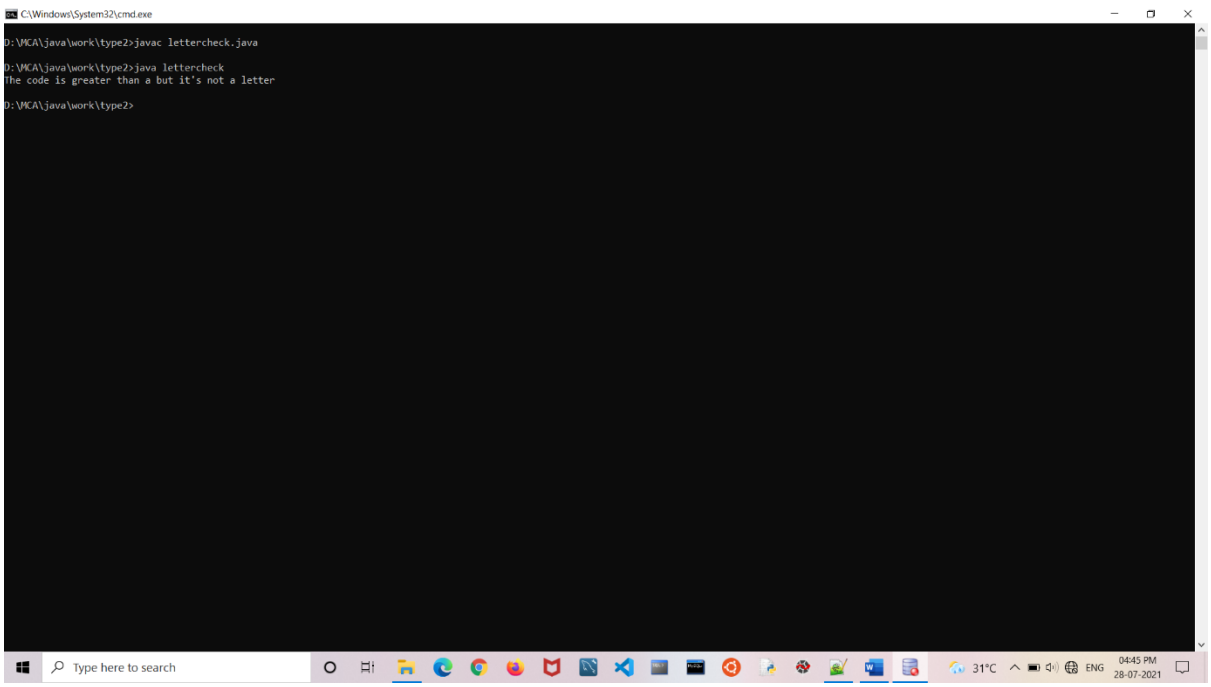
}



```
public class lettercheck
{
	public static void main(String[] args)
	{
		char symbol ='A';
		symbol = (char)(128.0*Math.random()); // Generate a random character
		if(symbol >='A')
		{
			// Is it A or greater?
			if(symbol <='Z')
			{
```

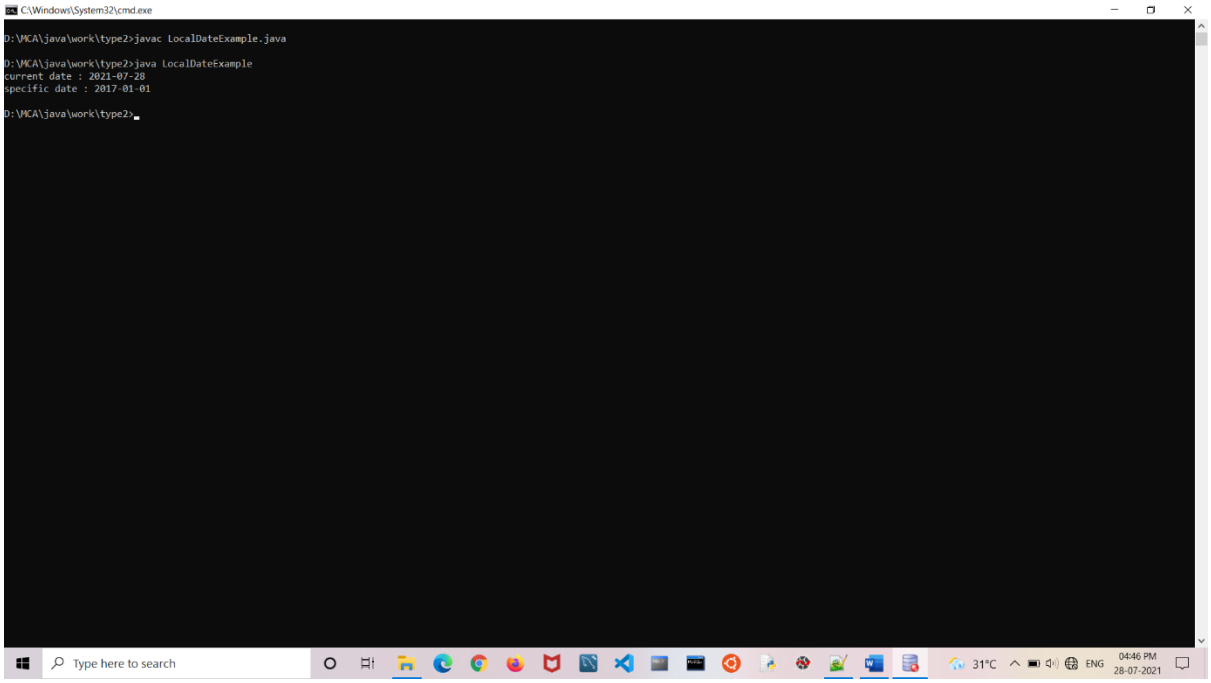
```
                    // yes, and is it Z or less?
                    // Then it is a capital letter
                    System.out.println("You have the capital letter
"+ symbol);
                }
                else
                {
                    // It is not Z or less
                    if(symbol >='A')
                    {
                        // So is it a or greater?
                        if(symbol <='Z')
                        {
                            // Yes, so is it z or less?
                            // Then it is a small letter
                            System.out.println("You have the
small letter " + symbol);
                        }
                        else
                        {
                            // It is not less than z
                            System.out.println("The code is
greater than a but it's not a letter");
                        }
                    }
```

```
                    else
                    {
                        System.out.println("The code is less than a and it's not a letter");
                    }
                }
            }
            else
            {
                System.out.println("The code is less than A so it's not a letter");
            }
        }
}
```



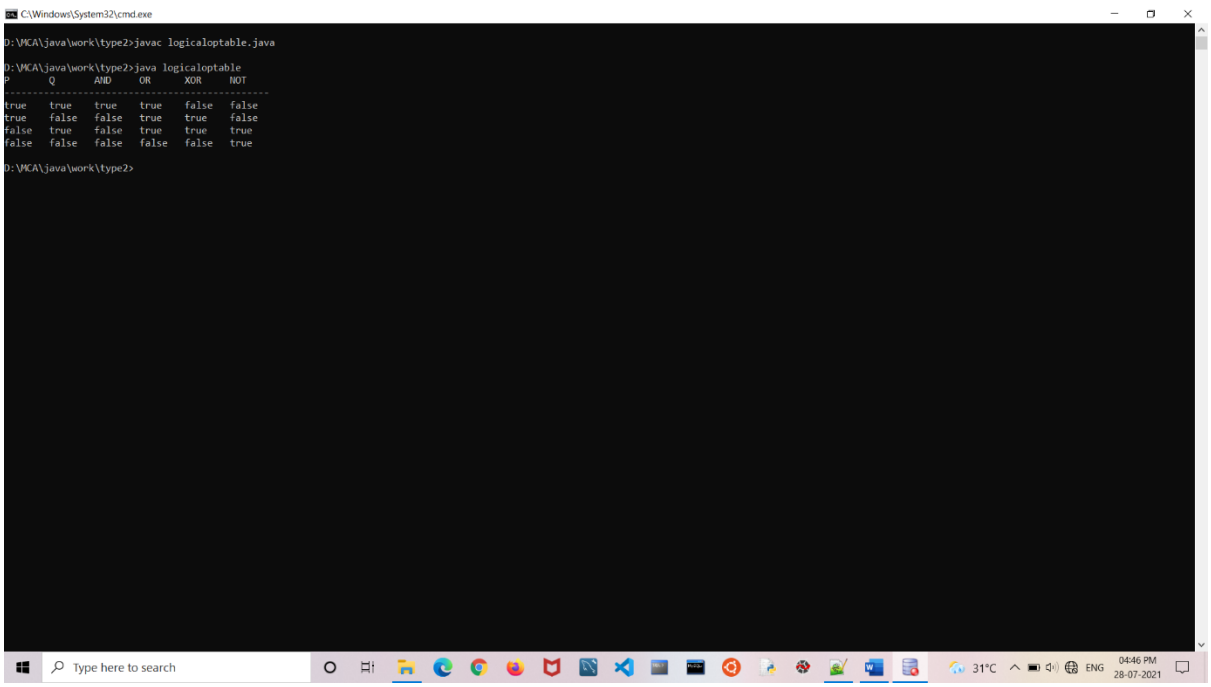
```
import java.time.LocalDate;
import java.time.Month;
import java.time.ZoneId;

public class LocalDateExample
{
	public static void main(String args[])
	{
		//current date
		LocalDate today = LocalDate.now();
		System.out.println("current date : " + today);

		//creating localdate by providing input arguments
		LocalDate firstDay_2017 =
LocalDate.of(2017,Month.JANUARY,1);
		System.out.println("specific date : " + firstDay_2017);

	}
```

}

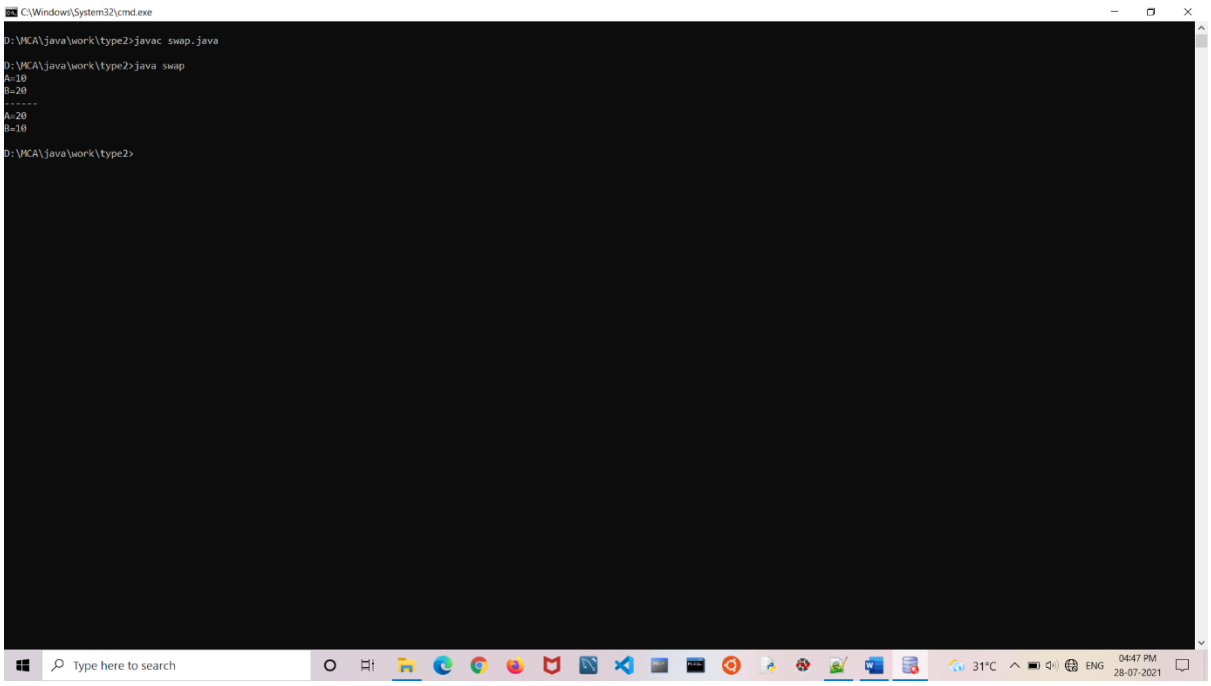


```
class logicaloptable
{
	public static void main(String args[])
	{
		boolean p,q;
		System.out.println("P\tQ\tAND\tOR\tXOR\tNOT");
		System.out.println("-----------------------------------------
	");
		p=true;q=true;

	System.out.println(p+"\t"+q+"\t"+(p&q)+"\t"+(p|q)+"\t"+(p^q)
+"\t"+(!p));
```

```
        p=true;q=false;

    System.out.println(p+"\t"+q+"\t"+(p&q)+"\t"+(p|q)+"\t"+(p^q)
+"\t"+(!p));


        p=false;q=true;

    System.out.println(p+"\t"+q+"\t"+(p&q)+"\t"+(p|q)+"\t"+(p^q)
+"\t"+(!p));


        p=false;q=false;

    System.out.println(p+"\t"+q+"\t"+(p&q)+"\t"+(p|q)+"\t"+(p^q)
+"\t"+(!p));
    }
}
```

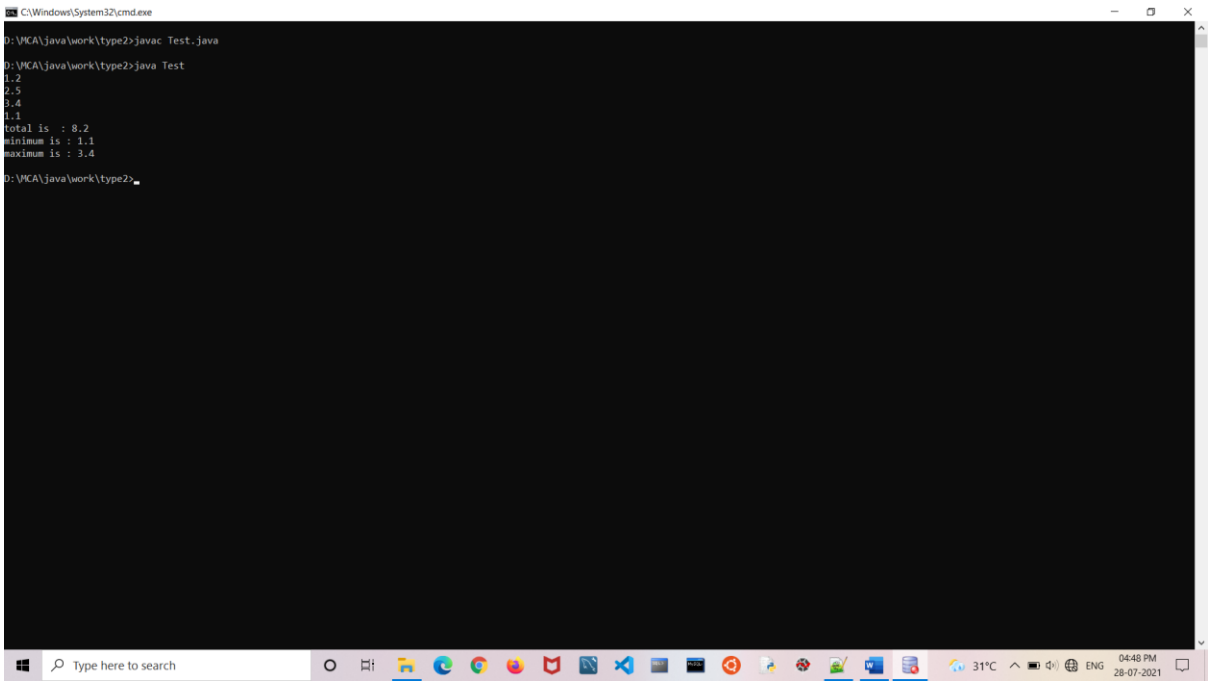


```
class swap
{
	public static void main(String args[])
	{
		int a=10,b=20,c;
		System.out.println("A="+a);
		System.out.println("B="+b);
		System.out.println("------");
		c=a;a=b;b=c;
		System.out.println("A="+a);
		System.out.println("B="+b);
	}
}
```



```
--------------------------------------------------------------------------------
import java.util.Arrays;
import java.util.*;
class Test
{
	public static void main(String args[])
	{

		double[] m = new double[4];
		m[0]=1.2;
		m[1]=2.5;
		m[2]=3.4;
		m[3]=1.1;

		System.out.println(m[0] + "\n" + m[1] + "\n"+ m[2] + "\n"
+ m[3]);

		double total;
		total = m[0] + m[1] + m[2] + m[3];
		System.out.println("total is  : " + total);



		Arrays.sort(m);
```

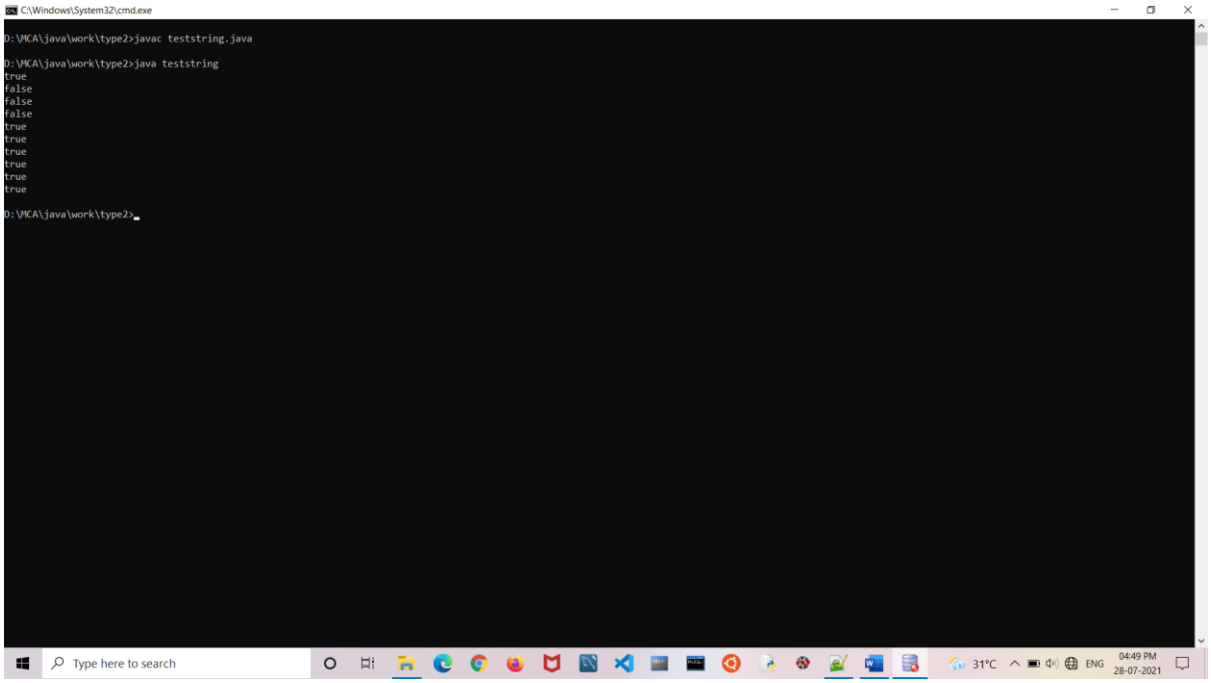
```
            System.out.println("minimum is : " + m[0]);
            System.out.println("maximum is : " + m[m.length-1]);
        }
}
```



```
class teststring
{
        public static void main(String args[])
        {
            String s1="Test";
            String s2="Test";
            String s3=new String("Test");
            final String s4=s3.intern();
```

```
            System.out.println(s1 == s2);
            System.out.println(s2 == s3);
            System.out.println(s3 == s4);
            System.out.println(s1 == s3);
            System.out.println(s1 == s4);

            System.out.println(s1.equals(s2));
            System.out.println(s2.equals(s3));
            System.out.println(s3.equals(s4));
            System.out.println(s1.equals(s4));
            System.out.println(s1.equals(s3));

        }
}

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

        }
```

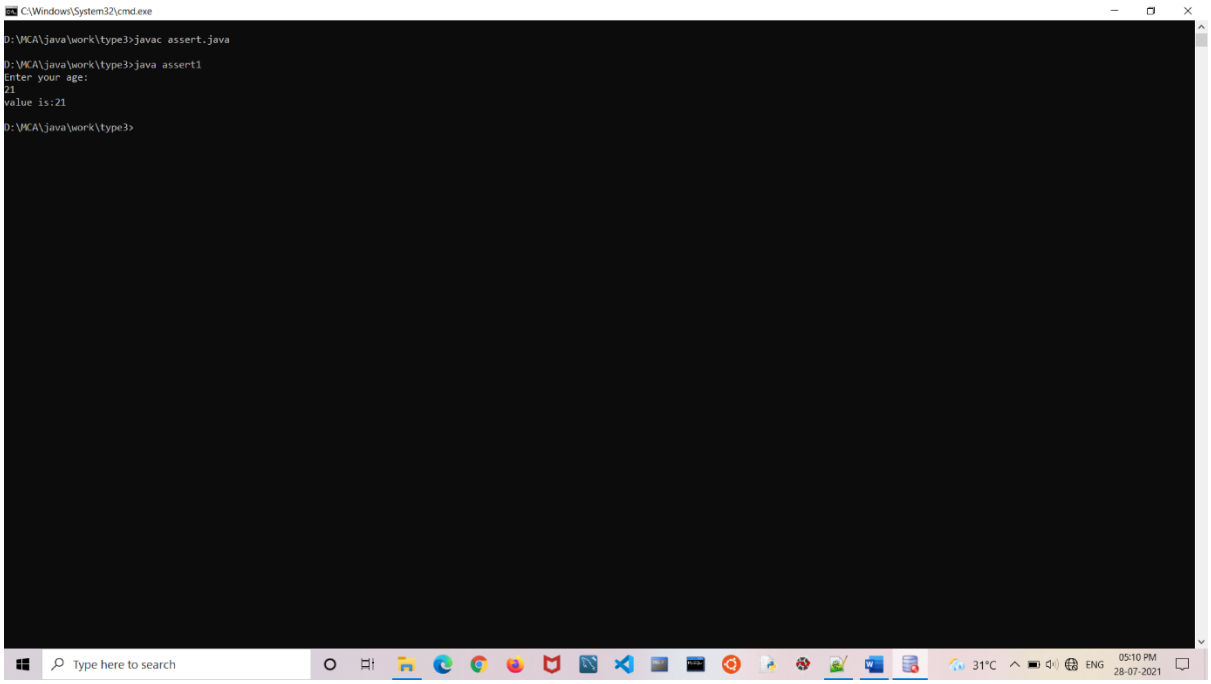
}


C:\Windows\System32\cmd.exe
D:\MCA\java\work\type2>javac teststring.java
D:\MCA\java\work\type2>java teststring
true
false
false
false
true
true
true
true
true
true
D:\MCA\java\work\type2>
Type here to search
31°C
ENG
04:49 PM
28-07-2021

Class Work – Inheritance , Scanner Class , Date and Time ,Try and Throw and String

---

```
import java.util.Scanner;
class assert1
{
	public static void main(String args[])
	{
		Scanner n=new Scanner(System.in);
		System.out.println("Enter your age:");
		int value=n.nextInt();
		assert value>=18:"not valid";
		System.out.println("value is:"+value);
	}
}
```

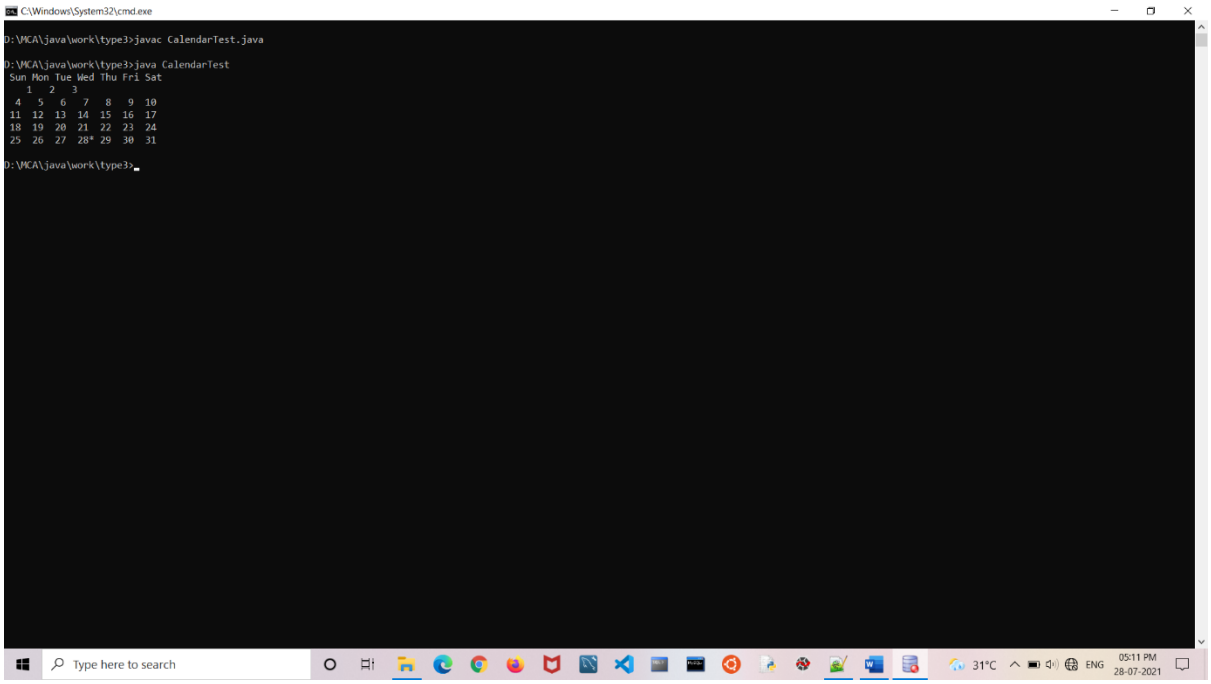


```
import java.text.DateFormatSymbols;
import java.util.*;

public class CalendarTest
{
	public static void main(String[]  args)
	{
		GregorianCalendar d = new GregorianCalendar ();
		int today = d.get(Calendar.DAY_OF_MONTH);
		int month = d.get(Calendar.MONTH);
		d.set(Calendar.DAY_OF_MONTH,1);

		int weekday = d.get(Calendar.DAY_OF_WEEK);
```

```
        int firstDayOfWeek = d.getFirstDayOfWeek();

        int indent = -1;

        while(weekday !=firstDayOfWeek)
        {
            indent++;
            d.add(Calendar.DAY_OF_MONTH,1);
            weekday=d.get(Calendar.DAY_OF_WEEK);
        }

        String[] weekdayNames  = new
DateFormatSymbols().getShortWeekdays();


        do
        {
            System.out.printf("%4s",weekdayNames[weekday]);
            d.add(Calendar.DAY_OF_MONTH,1);
            weekday = d.get(Calendar.DAY_OF_WEEK);
        }while(weekday !=firstDayOfWeek);
        System.out.println();

        for(int i=1;i<=indent;i++)
```

```
            System.out.print(" ");


        d.set(Calendar.DAY_OF_MONTH,1);


        do
        {
            int day=d.get(Calendar.DAY_OF_MONTH);
            System.out.printf("%3d",day);


            if(day == today) System.out.print("*");
            else System.out.print(" ");

            d.add(Calendar.DAY_OF_MONTH,1);
            weekday = d.get(Calendar.DAY_OF_WEEK);

            if(weekday == firstDayOfWeek)
System.out.print("\n");
        }while(d.get(Calendar.MONTH)==month);

        if(weekday !=firstDayOfWeek) System.out.println();
```

```
        }
}
```

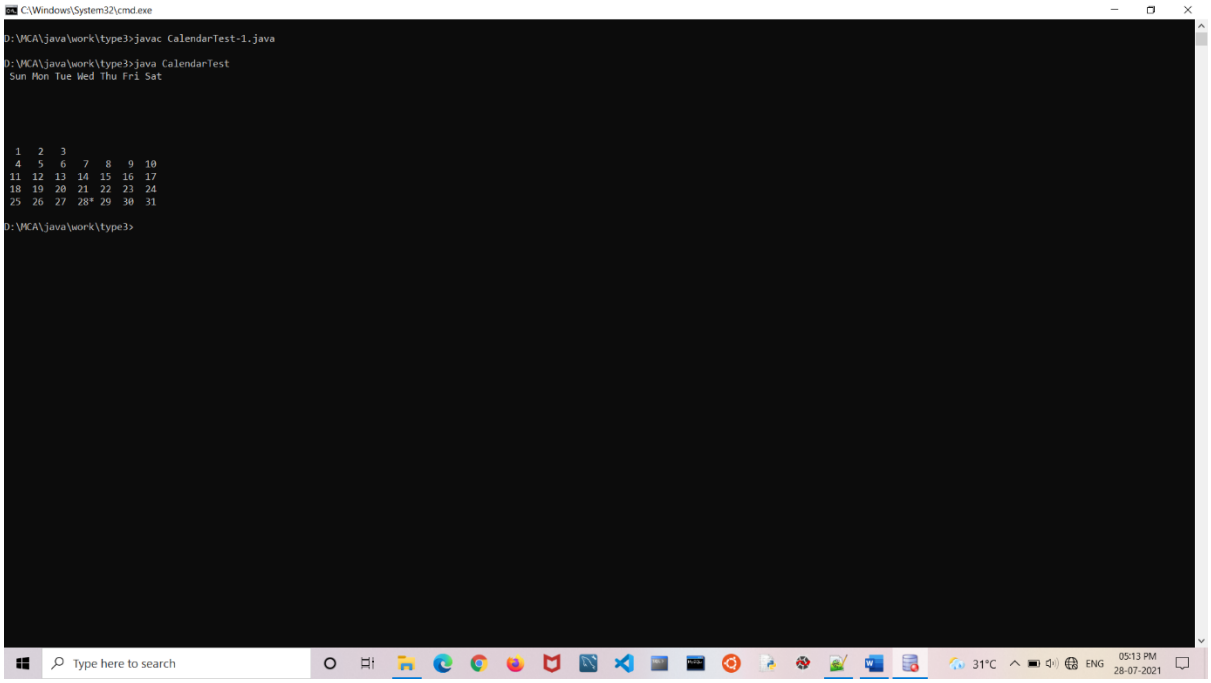


```
import java.text.DateFormatSymbols;
import java.util.*;
//import java.util.Calendar;

class CalendarTest
{
        public static void main(String[] args)
        {
                GregorianCalendar d = new GregorianCalendar();

                int today = d.get(Calendar.DAY_OF_MONTH);
```

```
            int month = d.get(Calendar.MONTH);

            d.set(Calendar.DAY_OF_MONTH,2);

            int weekday = d.get(Calendar.DAY_OF_WEEK);

            int firstDayOfWeek = d.getFirstDayOfWeek();

            int indent = 0;
            while (weekday != firstDayOfWeek)
            {
                  indent++;
                  d.add(Calendar.DAY_OF_MONTH,-1);
                  weekday = d.get(Calendar.DAY_OF_WEEK);
            }

            String[] weekdayNames = new
DateFormatSymbols().getShortWeekdays();

            do
            {
                  System.out.printf("%4s",weekdayNames[weekday]);
                  d.add(Calendar.DAY_OF_MONTH,1);
                  weekday=d.get(Calendar.DAY_OF_WEEK);
```

```
}
while(weekday != firstDayOfWeek);
System.out.println();

for (int i = 1;i<=indent;i++)
System.out.println(" ");

d.set(Calendar.DAY_OF_MONTH, 1);
do
{
        int day = d.get(Calendar.DAY_OF_MONTH);
        System.out.printf("%3d",day);

        if(day == today) System.out.print("*");
        else System.out.print(" ");

        d.add(Calendar.DAY_OF_MONTH, 1);
        weekday = d.get(Calendar.DAY_OF_WEEK);

        if(weekday == firstDayOfWeek) System.out.println();

}
while(d.get(Calendar.MONTH) == month);
```

```
            if(weekday != firstDayOfWeek) System.out.println();
        }
}
```

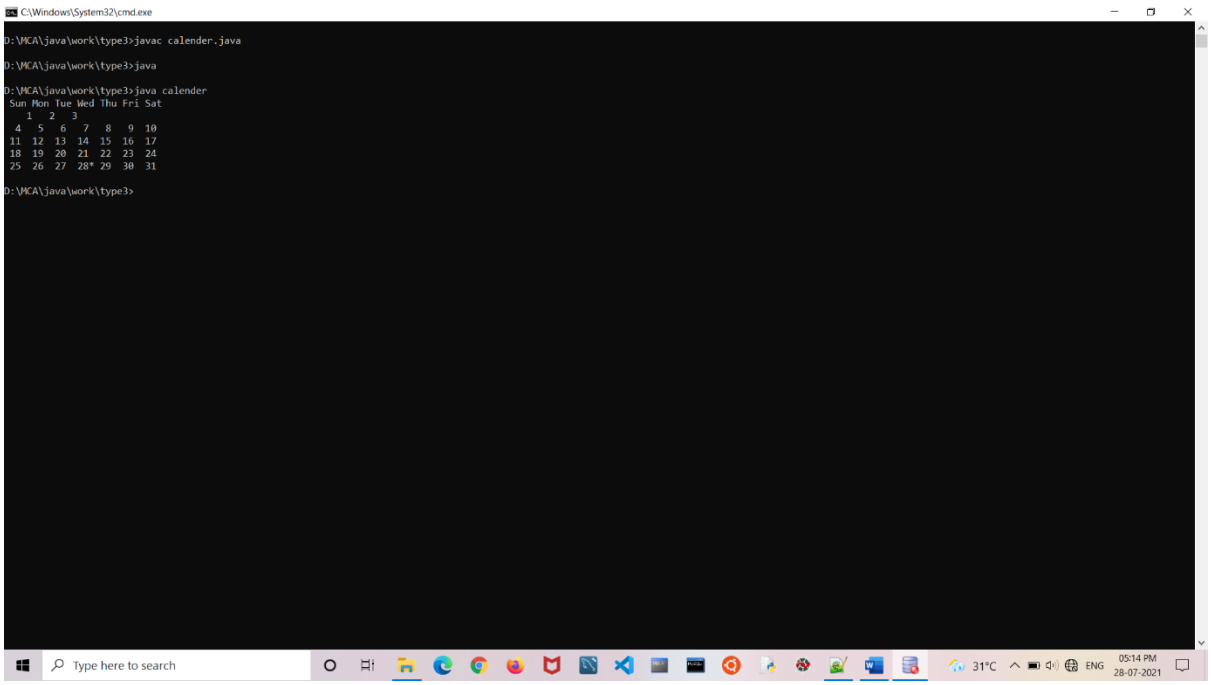


```
import java.text.DateFormatSymbols;
import java.util.*;
class calender
{
        public static void main(String[]  args)
        {
                GregorianCalendar d = new GregorianCalendar ();
                int today = d.get(Calendar.DAY_OF_MONTH);
                int month = d.get(Calendar.MONTH);
                d.set(Calendar.DAY_OF_MONTH,1);
```

```
        int weekday = d.get(Calendar.DAY_OF_WEEK);
        int firstDayOfWeek = d.getFirstDayOfWeek();

        int indent = -1;

        while(weekday !=firstDayOfWeek)
        {
            indent++;
            d.add(Calendar.DAY_OF_MONTH,1);
            weekday=d.get(Calendar.DAY_OF_WEEK);
        }


        String[] weekdayNames  = new
DateFormatSymbols().getShortWeekdays();


        do
        {
            System.out.printf("%4s",weekdayNames[weekday]);
            d.add(Calendar.DAY_OF_MONTH,1);
            weekday = d.get(Calendar.DAY_OF_WEEK);
        }while(weekday !=firstDayOfWeek);
```

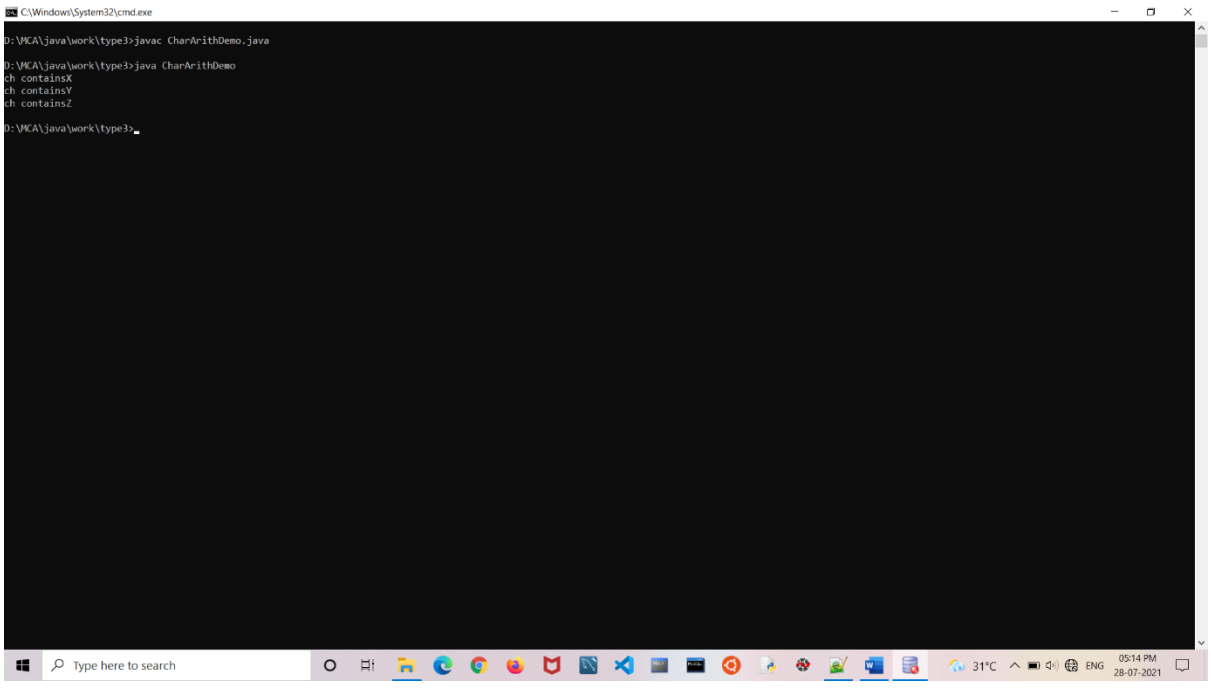
```
System.out.println();

for(int i=1;i<=indent;i++)
        System.out.print(" ");


d.set(Calendar.DAY_OF_MONTH,1);


do
{
        int day=d.get(Calendar.DAY_OF_MONTH);
        System.out.printf("%3d",day);


        if(day == today)
                System.out.print("*");
        else
                System.out.print(" ");

        d.add(Calendar.DAY_OF_MONTH,1);
        weekday = d.get(Calendar.DAY_OF_WEEK);

        if(weekday == firstDayOfWeek)
```

```
                System.out.print("\n");
            }while(d.get(Calendar.MONTH)==month);

            if(weekday !=firstDayOfWeek)
                System.out.println();

        }
}
```



```
class CharArithDemo
{
    public static void main(String args[])
    {
        char ch;
```

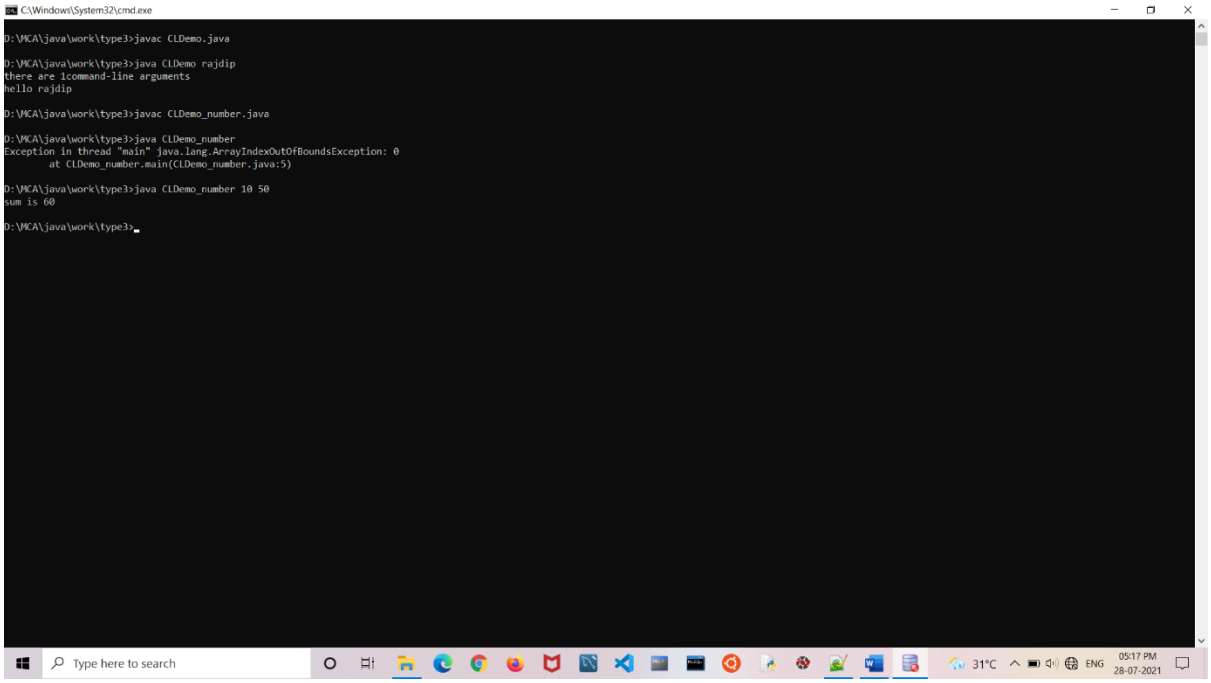
```
        ch='X';
        System.out.println("ch contains" + ch);

        ch++;
        System.out.println("ch contains" + ch);

        ch=90;
        //ch++;
        System.out.println("ch contains" + ch);

    }
}
```



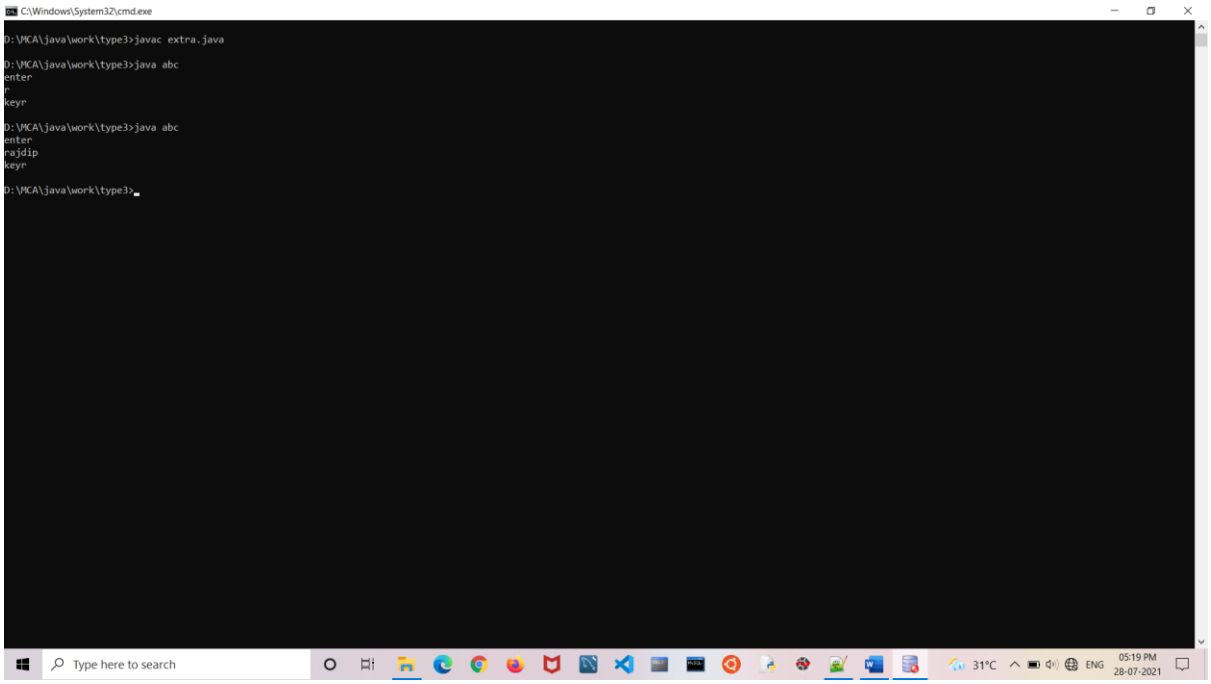
//program for command line argument usinf for loop , program 3

```
class CLDemo
{
	public static void main(String args[])
	{
		System.out.println("there are " + args.length +
"command-line arguments");
		//System.out.println("they are");
		for(int i=0; i<args.length; i++)
		//System.out.println("args[" + i + "]:" +args[i]);
		System.out.println("hello " + args[i]);
	}
}
class CLDemo_number
{
	public static void main(String args[])
	{
		System.out.println("sum is " + (Integer.parseInt(args[0]) +
Integer.parseInt(args[1])));
	}
```

}

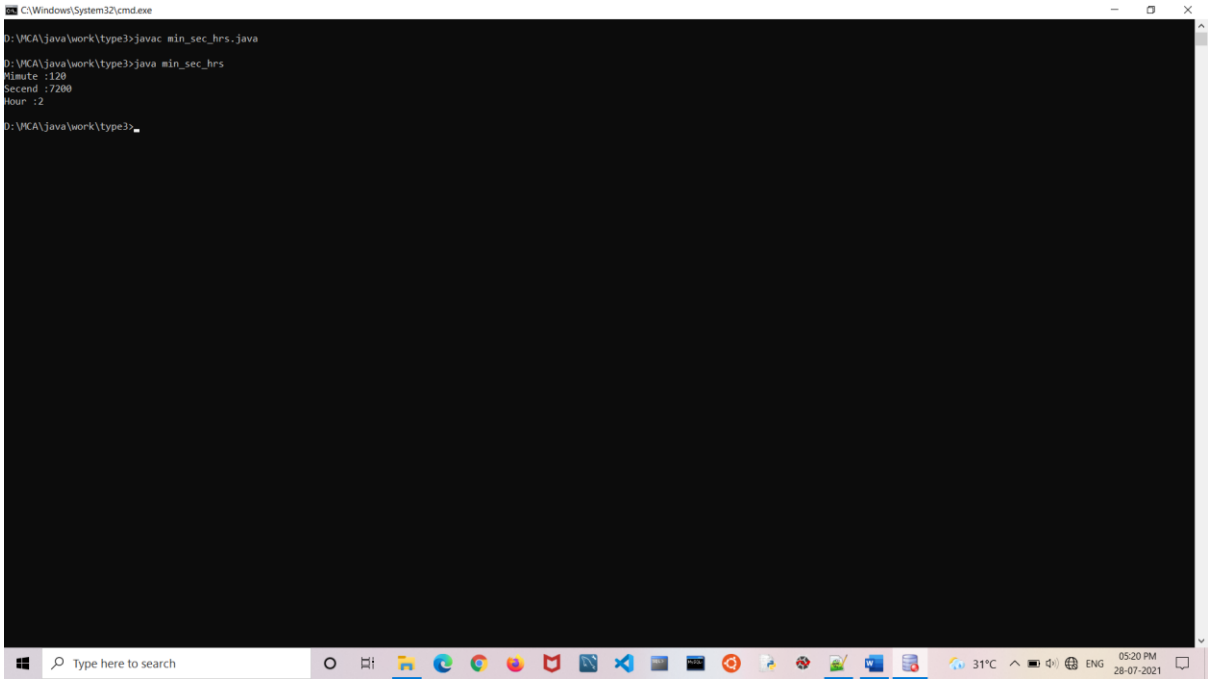


```
class abc
{
	public static void main(String args[])
		throws java.io.IOException
		{
			char ch;
			System.out.println("enter");
			ch=(char)System.in.read();
			System.out.println("key"+ch);
		}
}
```



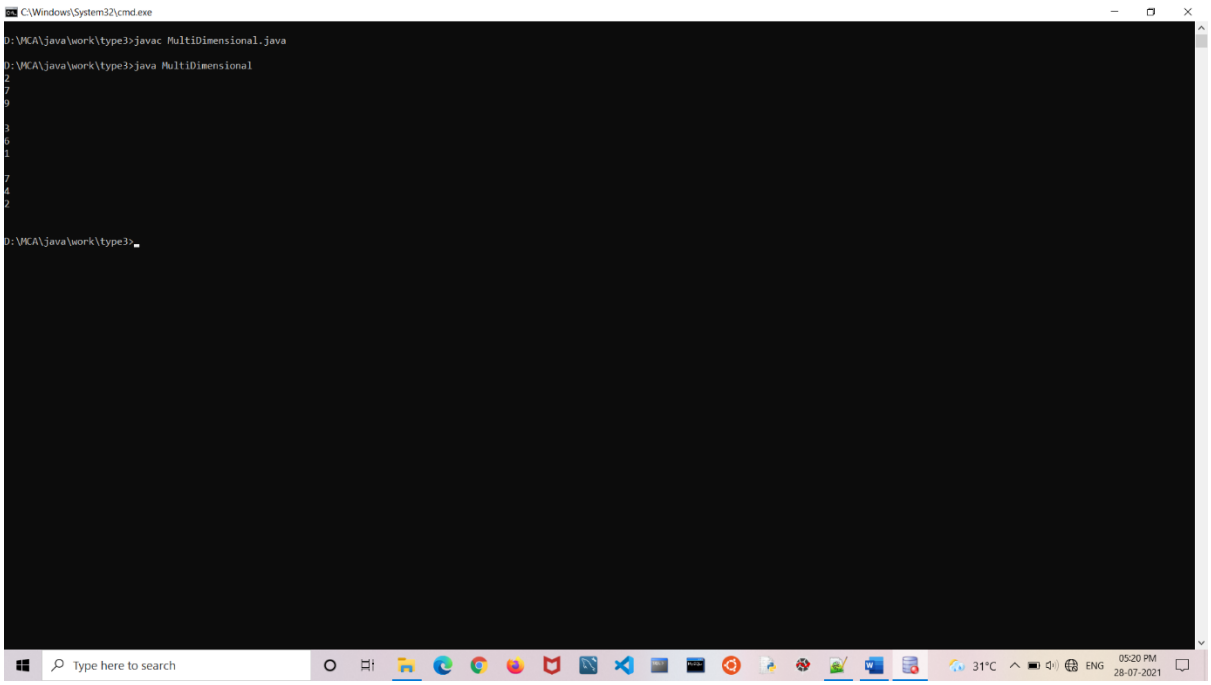
```
class min_sec_hrs
{
	public static void main(String args[])
	{
		int min=120;
		int sec,hrs;
		sec=min*60;
		hrs=min/60;
		System.out.println("Mimute :"+min);
		System.out.println("Secend :"+sec);
		System.out.println("Hour :"+hrs);
	}
```

}



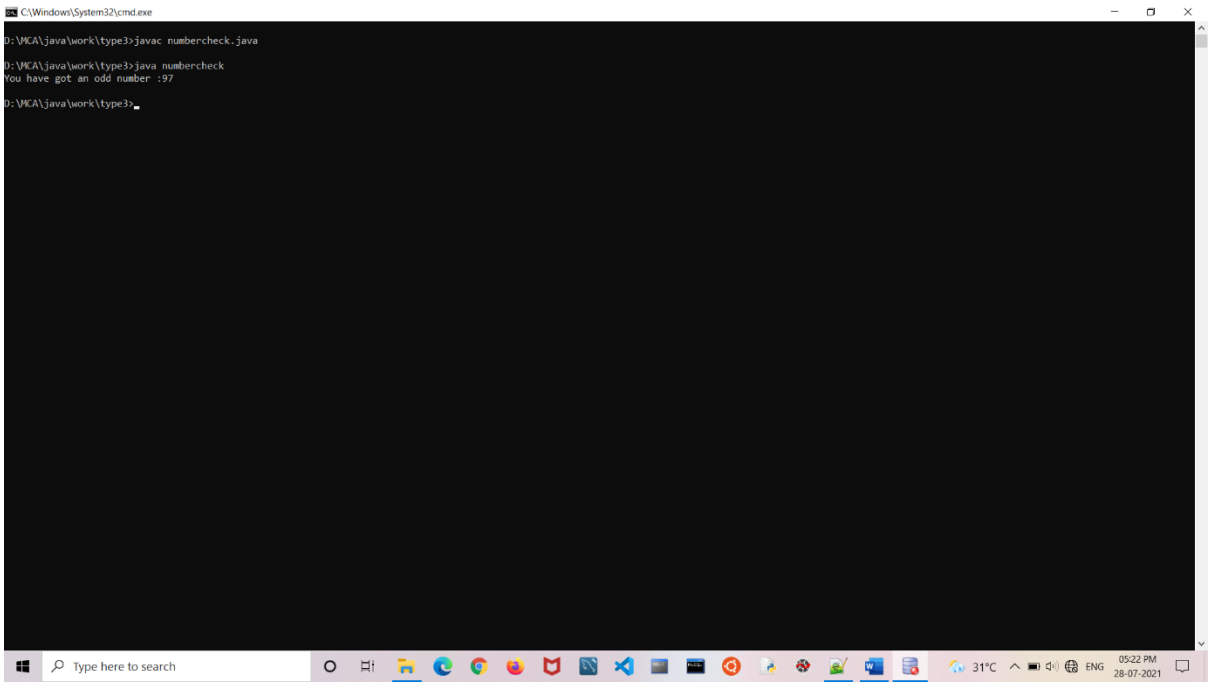
```
class MultiDimensional
{
	public static void main(String args[])
	{
		int arr[][] = { {2,7,9} , {3,6,1} , {7,4,2} };

		//printing 2D array
		for(int i=0; i<3 ; i++)
		{
			for(int j=0 ; j<3 ; j++)
				System.out.println(arr[i][j] + " " );
```

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



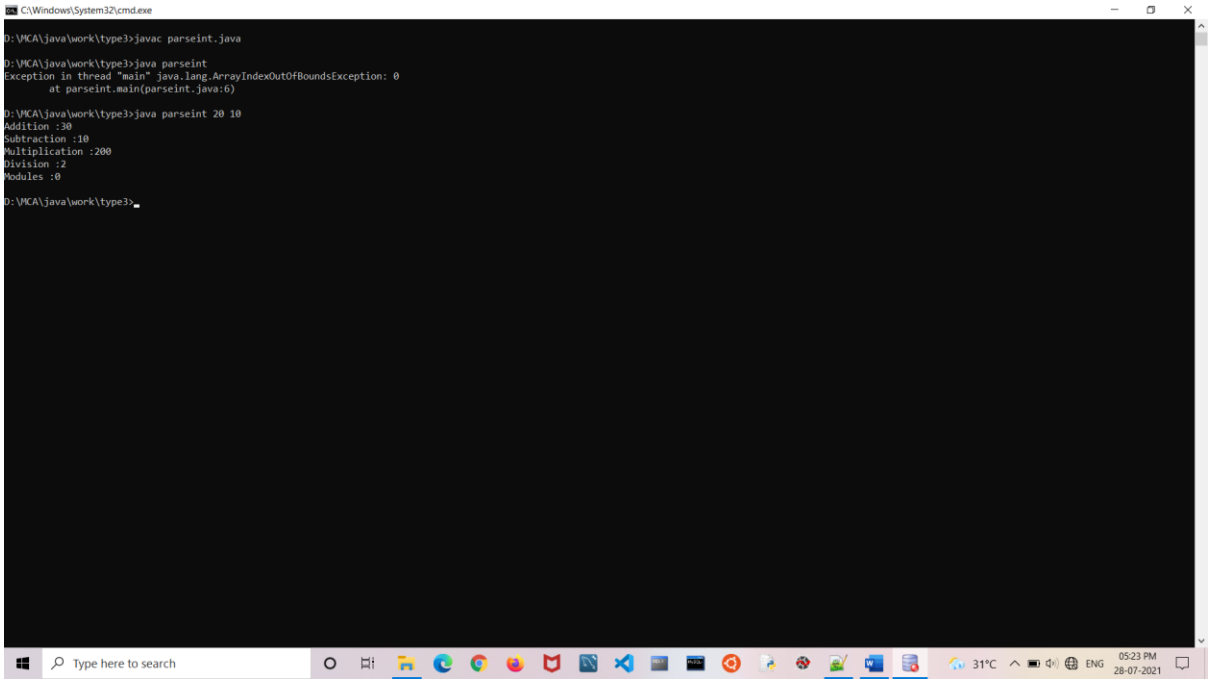
```
import java.io.*;
public class numbercheck
{
    public static void main(String[] args)
    {
        int number = 0;
        number = 1+(int)(100*Math.random()); // Get a random integer between 1 & 100
        if(number%2 == 0)
```

```
        {
            // Test if it is even
            System.out.println("You have got an even number :"+number); // It is even
        }
        else
        {
            System.out.println("You have got an odd number :"+number); // It is odd
        }
    }
}
```



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

		int a=Integer.parseInt(args[0]);
		int b=Integer.parseInt(args[1]);
		int add=a+b;
		int sub=a-b;
		int mul=a*b;
		int div=a/b;
		int mod=a%b;
		System.out.println("Addition :"+add);
		System.out.println("Subtraction :"+sub);
		System.out.println("Multiplication :"+mul);
		System.out.println("Division :"+div);
		System.out.println("Modules :"+mod);
		//System.out.println("sum is :"
+(Integer.parseInt(args[0])+Integer.parseInt(args[1])));
	}
```

}



```
package abstractClasses;
public class PersonTest
{
	public static void main(String[] args)
	{
		Person[] people=new Person[2];

		people[0]=new Employee("rajdip",50000,2000,10,1);
		people[1]=new Student("daxesh","computer");

		for(Person p : people)
```

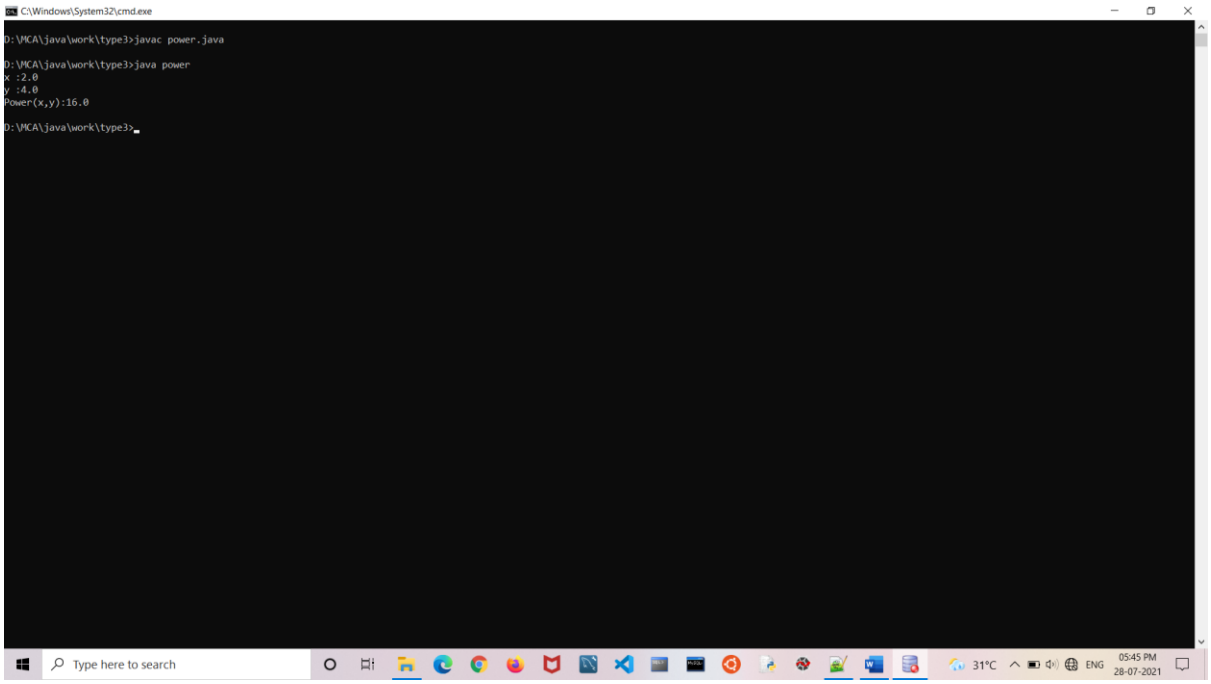
```
            {
                System.out.println(p.getName()+ " ,
"+p.getDescription());
            }
        }
}

package abstractClasses;

public abstract class Person
{
        public abstract String getDescription();
        private String name;

        public Person(String n)
        {
        name=n;
        }
        public String getName()
        {
        return name;
        }
}
```

```
package abstractClasses;
import java.util.Date;
import java.util.GregorianCalendar;

public class Employee extends Person
{
	private double salary;
	private Date hireDay;

	public Employee(String n,double s,int year,int month,int day)
	{
		super(n);
		salary=s;
		GregorianCalendar calender=new
GregorianCalendar(year,month-1,day);
		hireDay=calender.getTime();

	}
	public double getSalary()
	{
		return salary;
	}
	public Date getHireDay()
```

```
	{
		return hireDay;
	}
	public String getDescription()
	{
		return String.format("an employee with a salary of $%.2f",salary);
	}

	public void raiseSalary(double byPercent)
	{
		double raise=salary*byPercent/100;
		salary+=raise;
	}
}
	package abstractClasses;

public class Student extends Person
{
	private String major;
	public Student(String n,String m)
	{
		super(n);
		major=m;
```

```
	}
	public String getDescription()
	{
		return "a Student majoring in " +major;
	}
}
```

```
class power
{
	public static void main(String args[])
	{
		double x=2,y=4;
		double c;
		System.out.println("x :"+x);
		System.out.println("y :"+y);
		c=Math.pow(x,y);
		System.out.println("Power(x,y):"+c);
	}
}
```

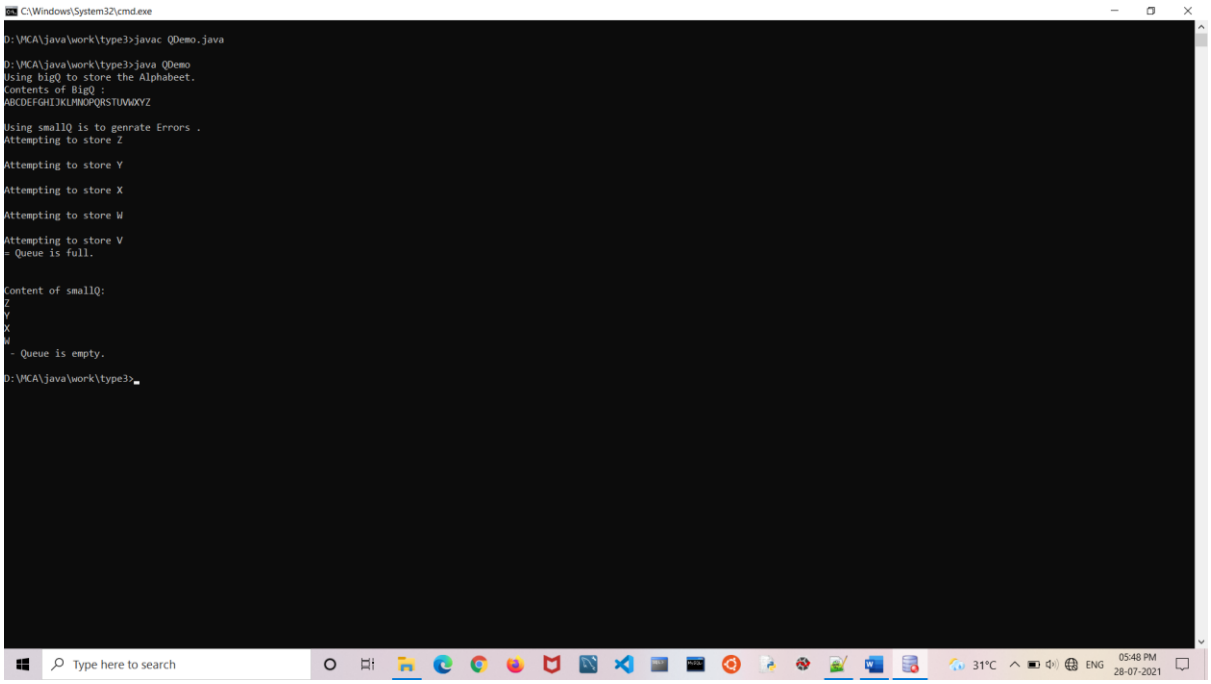



```
/* try this 5-2
A queue class for Characters
*/
 class Queue
 {
    char q[];// this array holds the queue
    int putloc,getloc;
    // this put and get induces

   Queue(int size)
   {
      q =new char[size];// allocate memory for queue
      putloc = getloc =0;
   }
```

```
  // put a character into the queue
  void put(char ch)
  {
     if(putloc == q.length)
     {
        System.out.println("= Queue is full.");
        return;
     }
     q[putloc++]=ch;
  }
  //get a charcter from the queue
  char get()
  {
     if(getloc== putloc)
     {
        System.out.println(" - Queue is empty. ");
        return (char) 0;
     }
     return q[getloc++];
  }
}
 // Demonatrate the queue class
   class QDemo{
      public static void main(String args[])
```

```
{
  Queue bigQ=new Queue(100);
  Queue smallQ=new Queue(4);
  char ch;
  int i;

  System.out.println("Using bigQ to store the Alphabeet. ");
  //put some number into bigQ
  for(i=0;i<26;i++)
     bigQ.put((char) ('A' + i));

  //retrieve and display elements from bigQ
  System.out.println("Contents of BigQ :");
  for(i=0;i<26;i++)
  {
     ch=bigQ.get();
     if(ch != (char) 0)
        System.out.print(ch);
  }
  System.out.println("\n");

  System.out.println("Using smallQ is to genrate Errors .");
  //Now ,use small! to generate some errors
  for(i=0;i<5;i++)
```

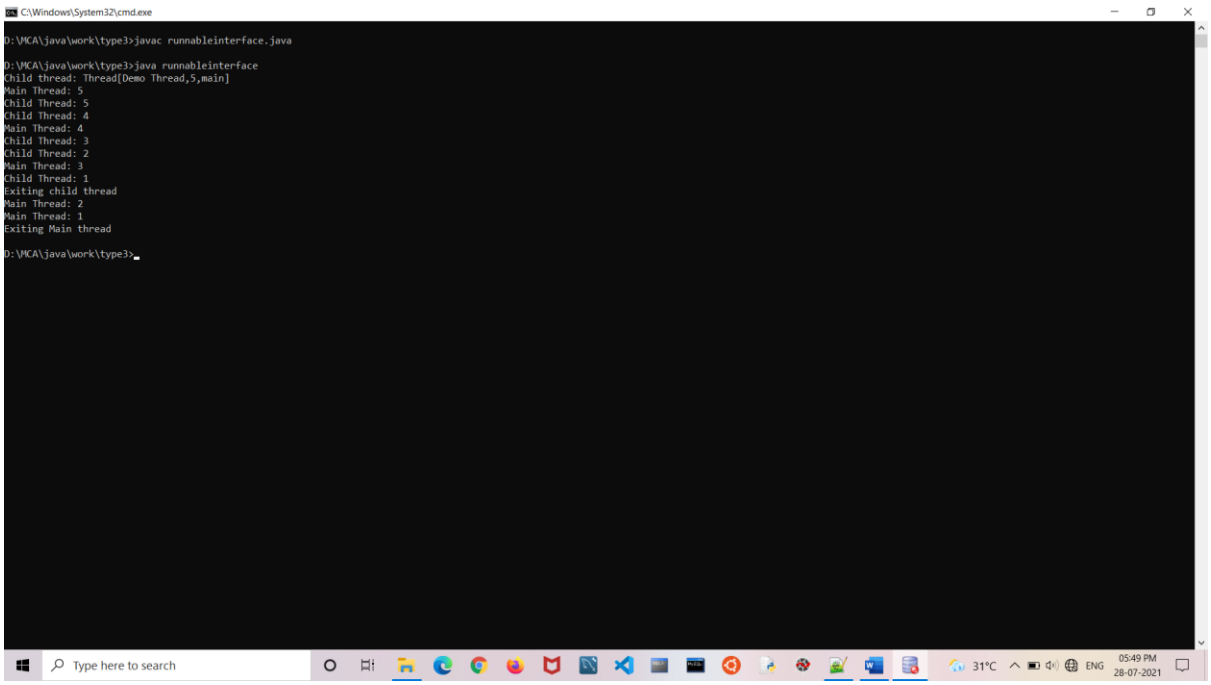
```
        {
            System.out.println("Attempting to store " + (char) ('Z' - i ) );

            smallQ.put((char) ('Z' - i));
            System.out.println();
        }
        System.out.println();

        //more erroes on smallQ
        System.out.println("Content of smallQ:");
        for(i=0;i<5;i++)
        {
            ch=smallQ.get();

            if(ch != (char) 0)
                System.out.println(ch);
        }
    }
}
```



```
class newthread implements Runnable{
        Thread t;
        newthread(){
                t = new Thread(this,"Demo Thread");
                System.out.println("Child thread: "+t);
                t.start();
        }

        public void run(){
                try{
                        for(int i=5;i>0;i--){
                                System.out.println("Child Thread: " + i);
                                Thread.sleep(500);
```

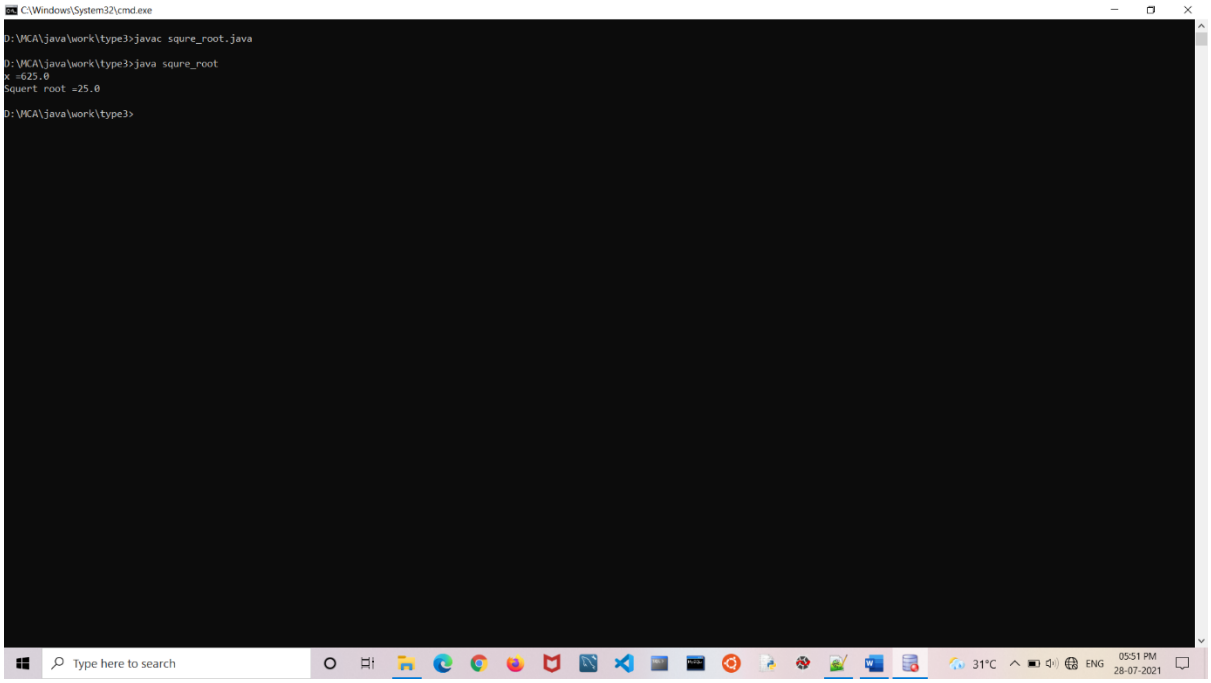
```
                }
            }catch(InterruptedException e){
                System.out.println("Child interrupted");
            }
            System.out.println("Exiting child thread");
        }


}
class runnableinterface{
        public static void main(String args[]){
            new newthread();

            try{
                for(int i=5;i>0;i--){
                    System.out.println("Main Thread: " + i);
                    Thread.sleep(1000);

                }
            }catch(InterruptedException e){
                System.out.println("Main interrupted");
            }
            System.out.println("Exiting Main thread");
```

```
        }

}
```



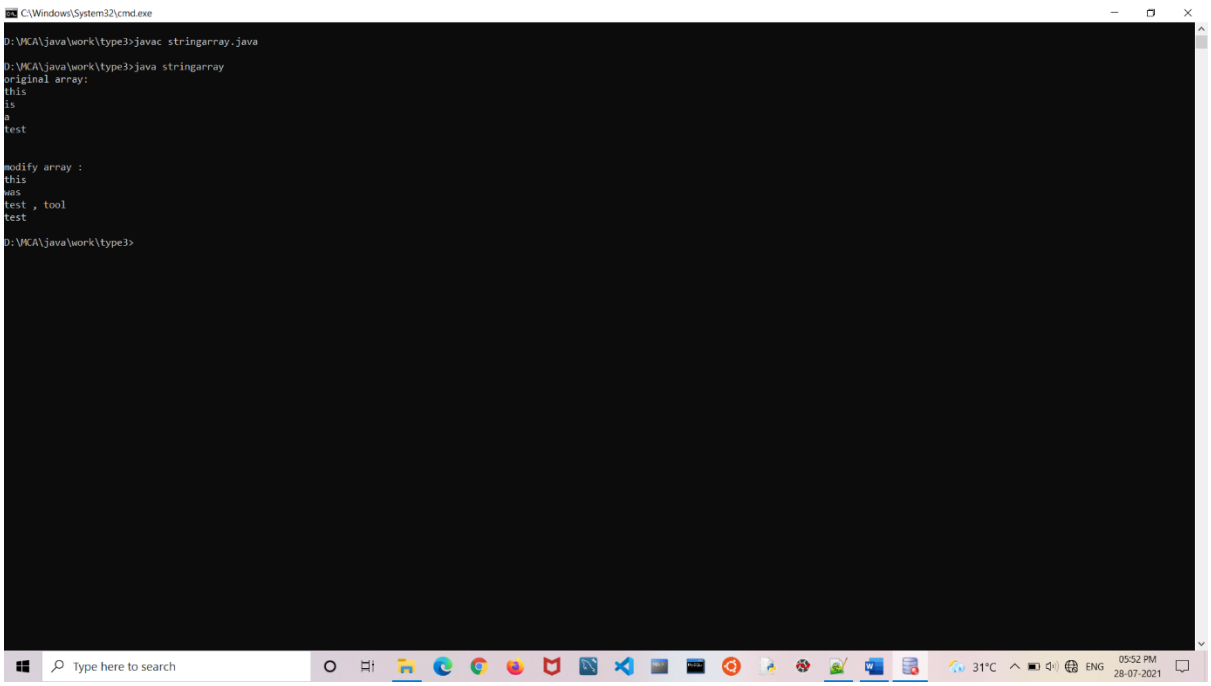
```
class squre_root
{
        public static void main(String args[])
        {
                double x=625,y;
                System.out.println("x ="+x);
                y=Math.sqrt(x);
                System.out.println("Squert root ="+y);
        }
```

}



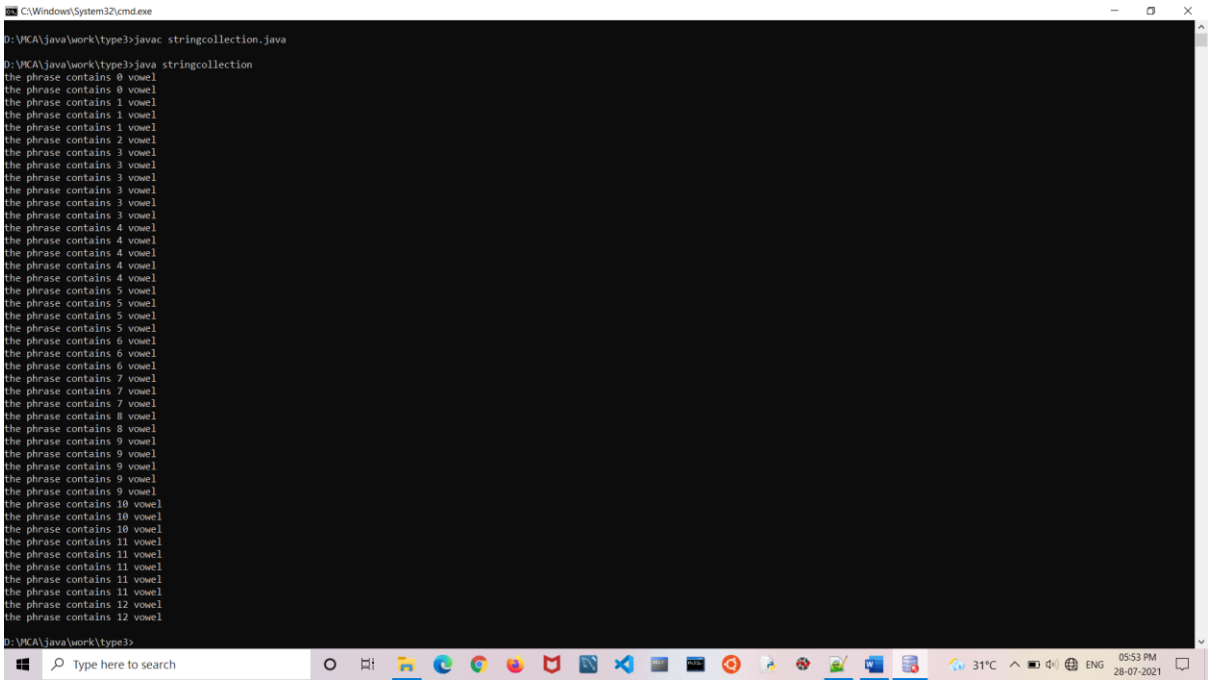
```
class stringarray
{
	public static void main(String args[])
	{
		String str[]={"this","is","a","test"};

		System.out.println("original array: ");
		for(String s:str)
			System.out.println(s+" ");
		System.out.println("\n");

		str[1]="was";
```

```
            str[2]="test , tool";

            System.out.println("modify array :");
            for(String s:str)
                System.out.println(s+" ");

        }
}
```



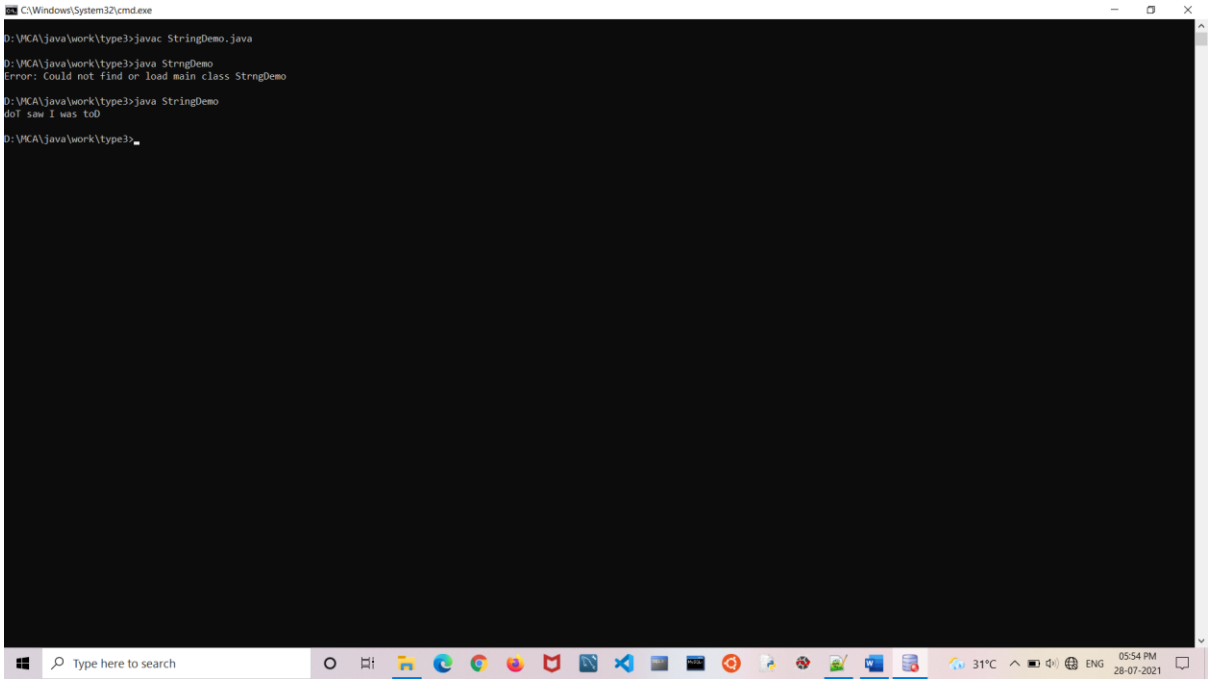
```
class stringcollection
{
        public static void main(String args[])
        {
```

```
        String phrase="the quick brown for jumped over the lary dog";
        int vowel=0;
        for(char ch:phrase.toCharArray())
        {
            ch=Character.toLowerCase(ch);
            if(ch=='a'||ch=='i'||ch=='e'||ch=='o'||ch=='u')
            {
                ++vowel;
            }
            System.out.println("the phrase contains "+vowel+" vowel");
        }
    }
}
```



```
public class StringDemo
{
	public static void main(String args[])
	{
		String palindrome = "Dot saw I was Tod";
		int len = palindrome.length();

		char[] tempCharArray = new char[len];
	char[] charArray = new char[len];

		//put original string in an array of char
		for(int i = 0 ; i<len ; i++)
		{
			tempCharArray[i] = palindrome.charAt(i);
		}

		//reverse array of char
		for(int j = 0 ; j< len; j++)
		{
			charArray[j] = tempCharArray[len - 1 - j];
		}

		String reversePalindrome = new String(charArray);
```

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

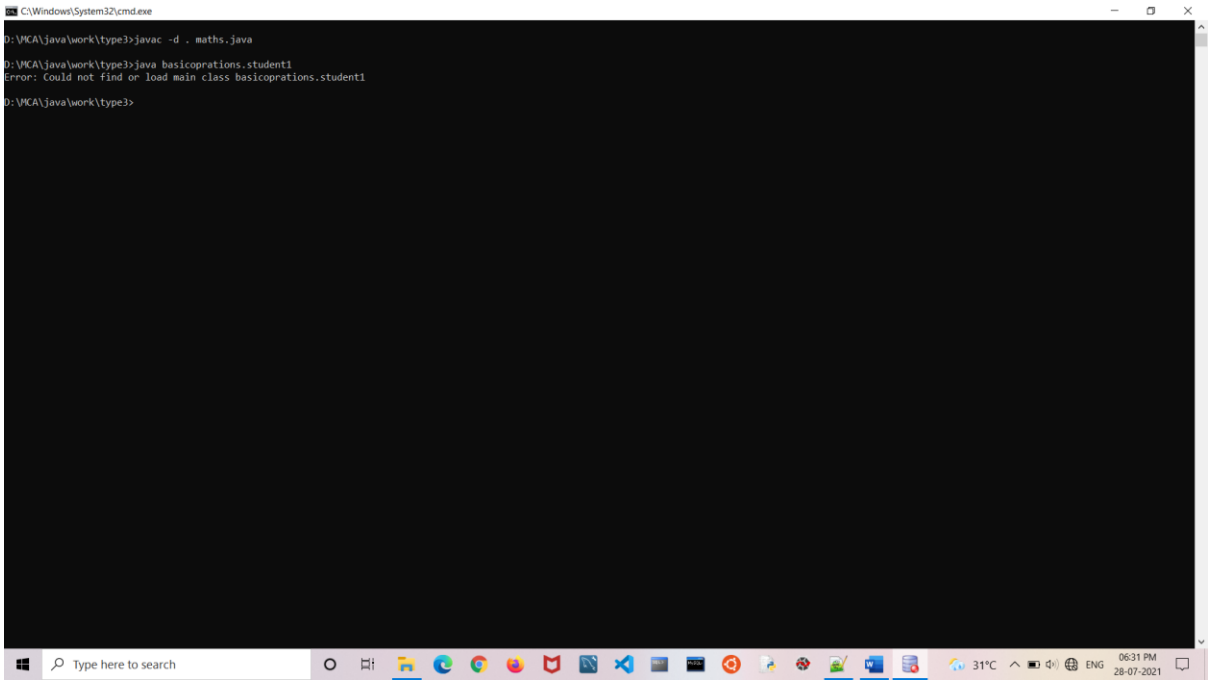


```
package basicoperations;
public interface maths
{
        public void add();
        public void sub();
        public void mul();
        public void div();
}
package basicoprations;
import java.util.Scanner;
```

```
class student1 implements maths
{
	@Override
	public void add()
	{
		Scanner kb=new Scanner(System.in);
		System.out.println("enter any two integer value to perform addition");
		int a=kb.nextInt();
		int b=kb.nextInt();
		int s=a+b;
		System.out.println("Addition is :"+s);
	}
	@Override
	public void sub()
	{
		Scanner kb=new Scanner(System.in);
		System.out.println("enter any two integer value to perform subtraction");
		int a=kb.nextInt();
		int b=kb.nextInt();
		int s=a-b;
		System.out.println("Subtraction is :"+s);
	}
```

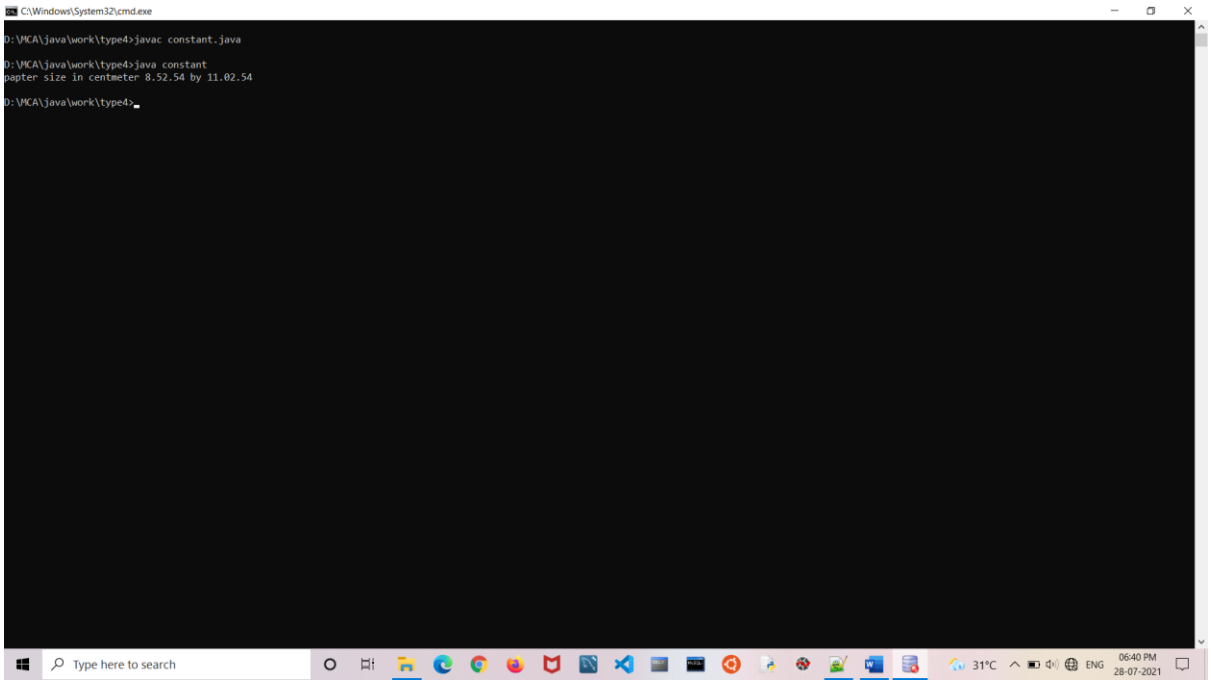
```
    @Override
    public void mul()
    {
        Scanner kb=new Scanner(System.in);
        System.out.println("enter any two integer value to
perform multipication");
        int a=kb.nextInt();
        int b=kb.nextInt();
        int s=a*b;
        System.out.println("multipication is :"+s);
    }
    @Override
    public void div()
    {
        Scanner kb=new Scanner(System.in);
        System.out.println("enter any two integer value to
perform divition");
        int a=kb.nextInt();
        int b=kb.nextInt();
        int s=a/b;
        System.out.println("divition is :"+s);
    }
    public static void main(String args[])
    {
```

```
            student1 std=new student1();
            std.add();
            std.sub();
            std.mul();
            std.div();
        }
}
```



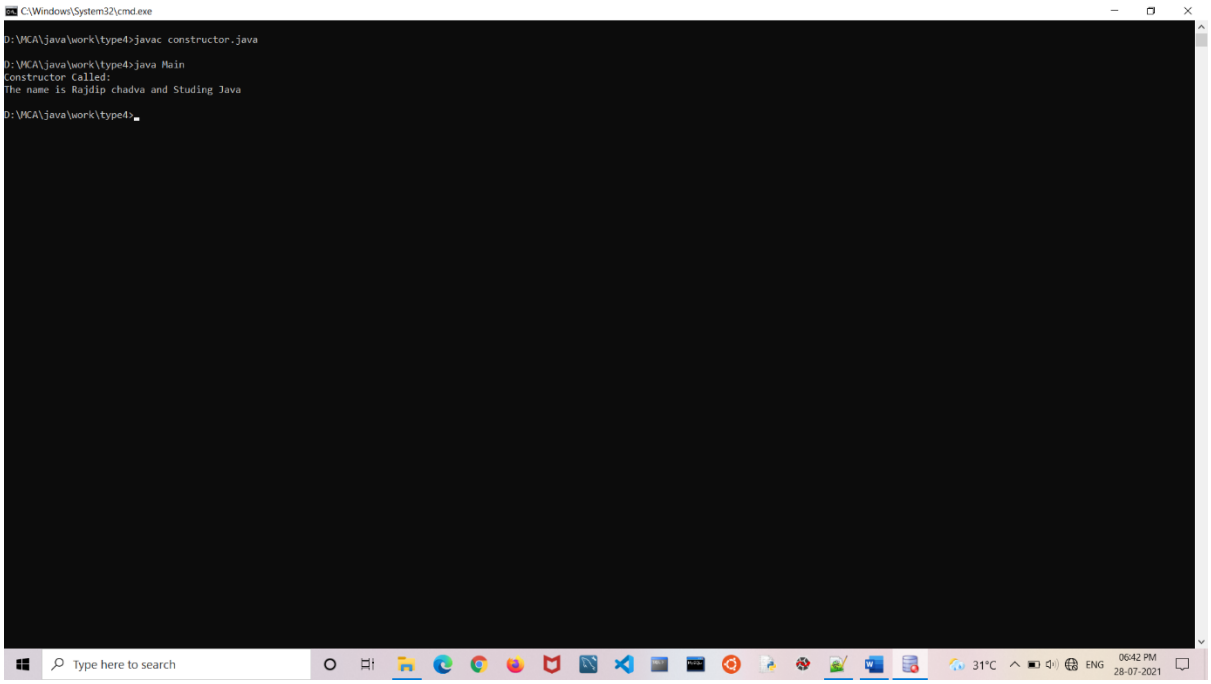
Class Work – Package , Thread , Try and Catch and Throw

```
public class constant
{
	public static final double cm_per_inch=2.54;

	public static void main(String args[])
	{
		double papterwidth=8.5;
		double papterheight=11;
		System.out.println("papter size in centmeter "+
papterwidth + cm_per_inch + " by " + papterheight + cm_per_inch );
	}
}
```



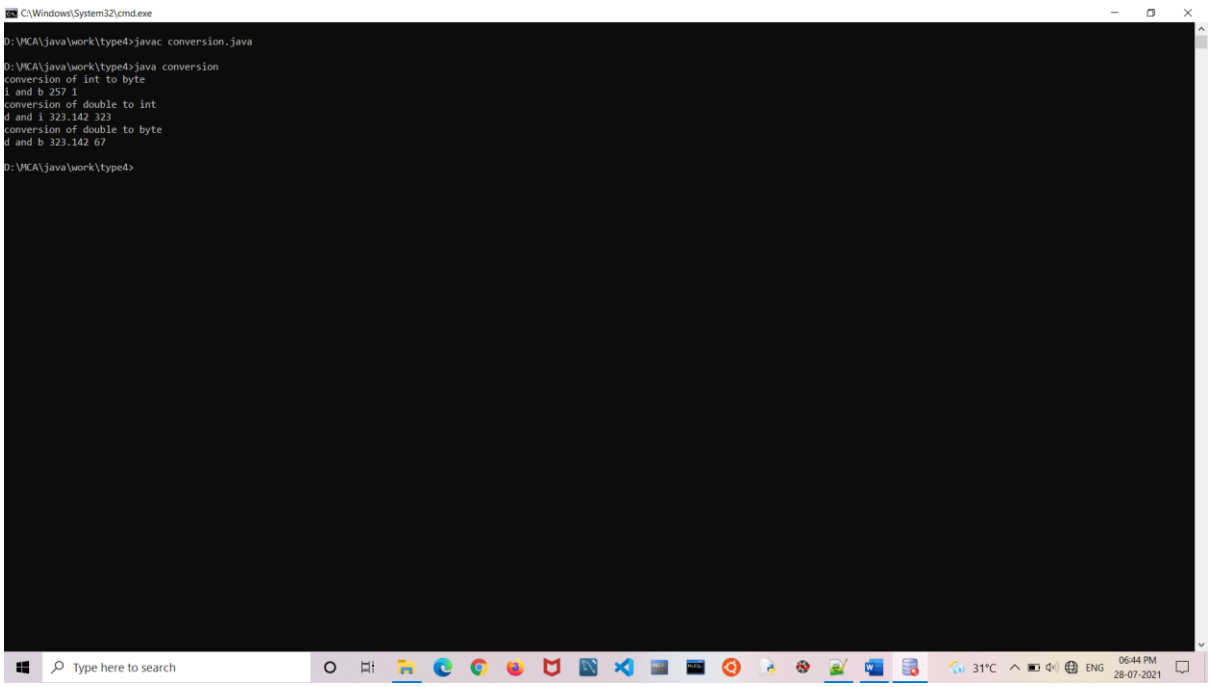
```
class Main
{
  private String name;

  // constructor
  Main() {
    System.out.println("Constructor Called:");
    name = "Rajdip chadva and Studing Java";
  }

  public static void main(String[] args) {
    // constructor is invoked while
    // creating an object of the Main class
```

```
    Main obj = new Main();
    System.out.println("The name is " + obj.name);
  }
}
```



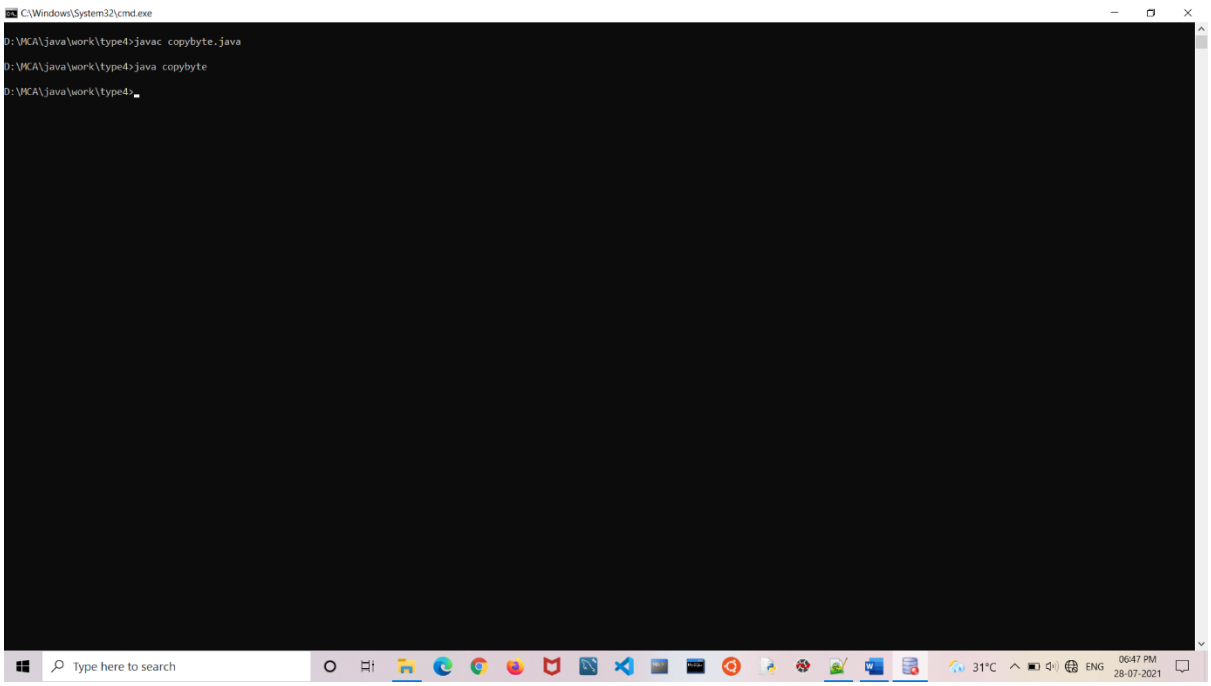
```
class conversion
{
        public static void main(String args[])
        {
                byte b;
                int i=257;
                double d=323.142;

                System.out.println("conversion of int to byte ");
```

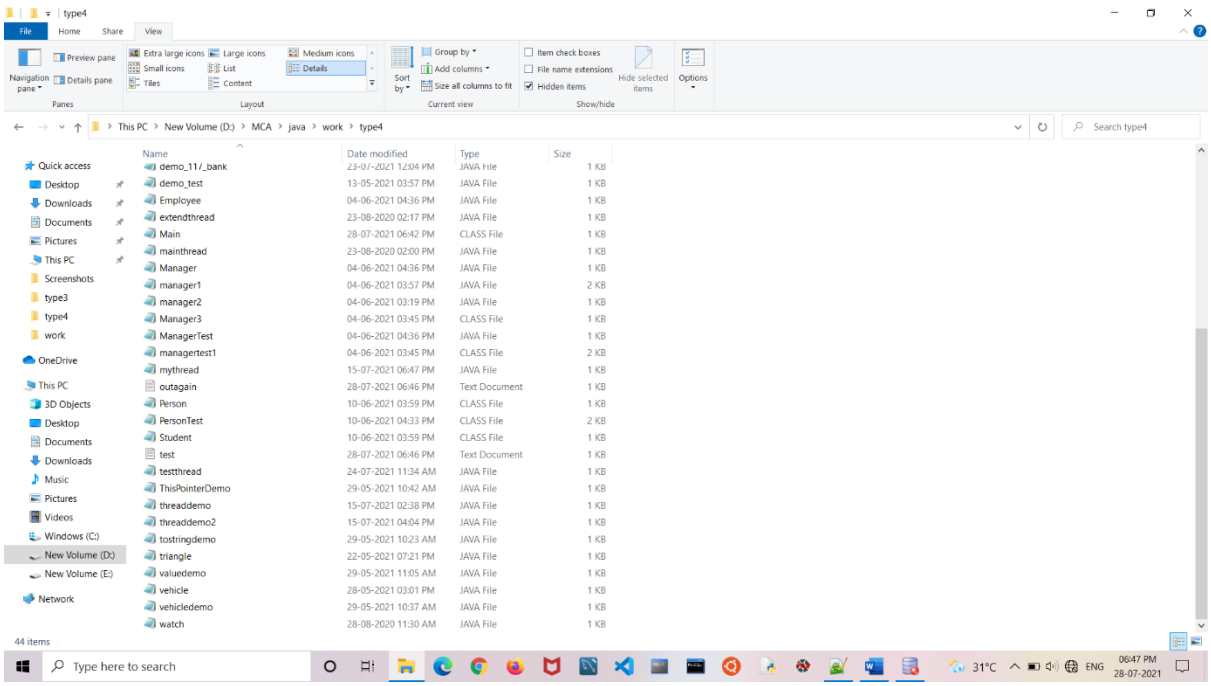
```
            b=(byte)i;
            System.out.println("i and b "+i+" "+b);

            System.out.println("conversion of double to int ");
            i=(int)d;
            System.out.println("d and i "+d+" "+i);

            System.out.println("conversion of double to byte ");
            b=(byte)d;
            System.out.println("d and b "+d+" "+b);
        }
}
```



```
import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.IOException;

public class copybyte
{
	public static void main(String args[]) throws IOException
	{
		FileInputStream in=null;
		FileOutputStream out=null;

		try
		{
			in=new FileInputStream("test.txt");
			out=new FileOutputStream("outagain.txt");
			int c;
			while ((c=in.read())!=-1)
			{
				out.write(c);
			}
		}
		finally
		{
			if(in != null)
```

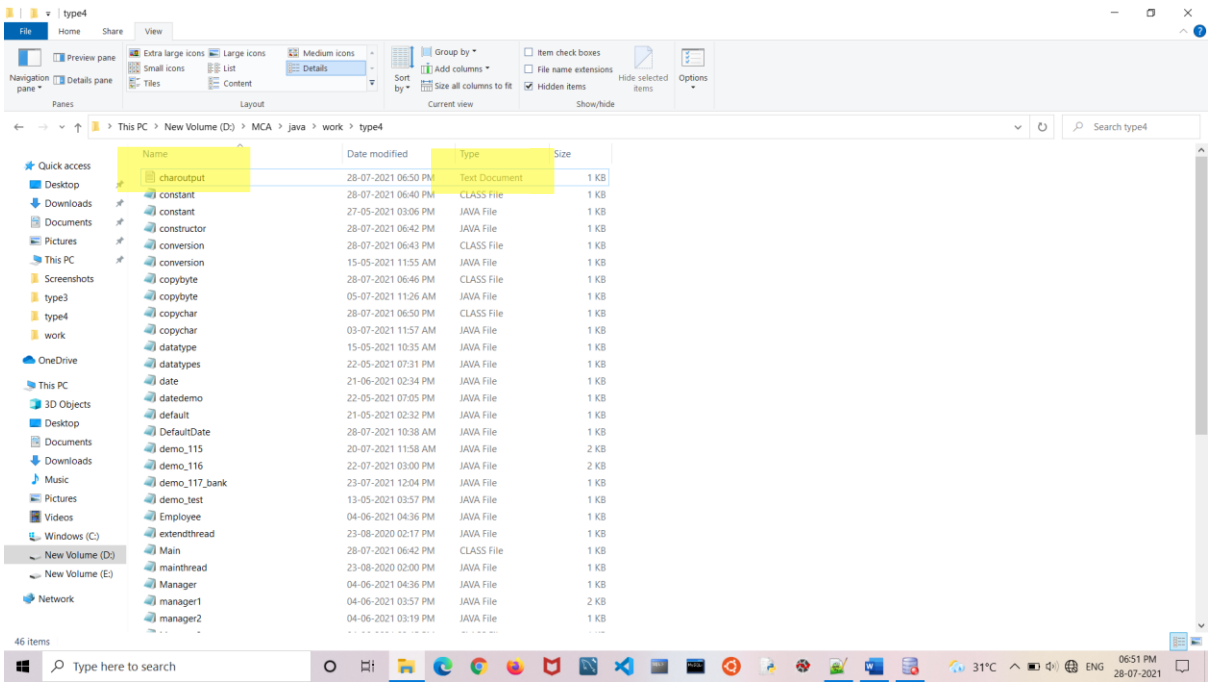
```
                {
                        in.close();
                }
                if(out!=null)
                {
                        out.close();
                }
        }
    }
}
```





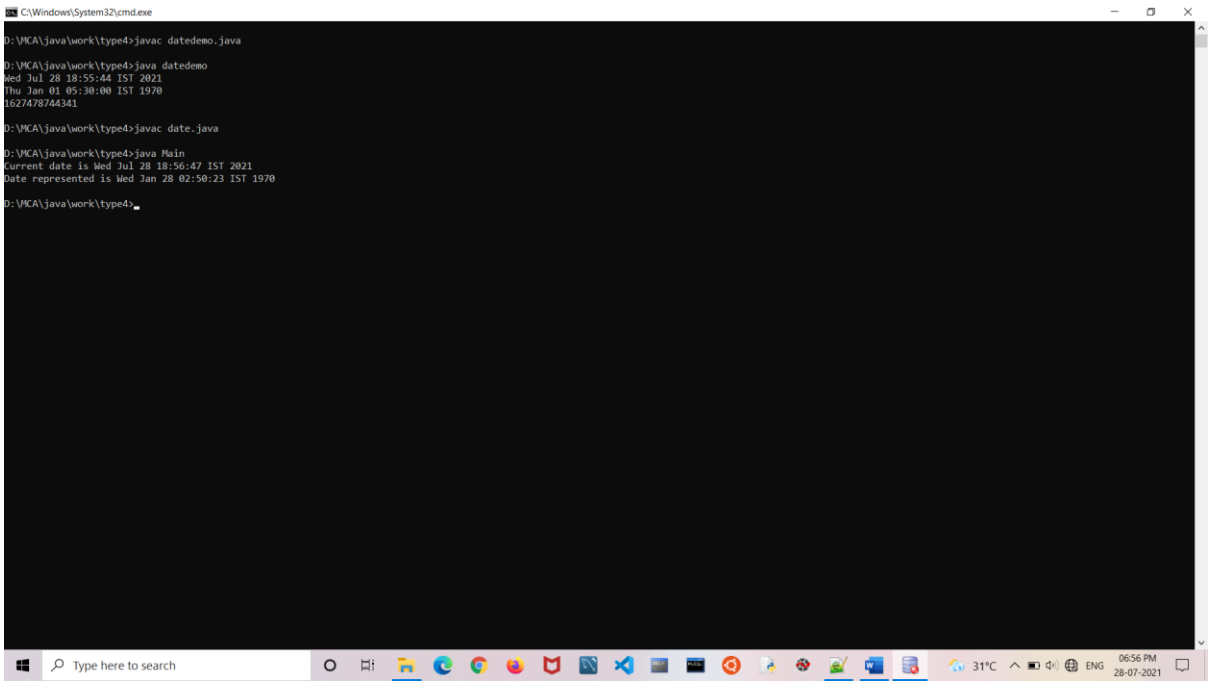
```
import java.io.FileReader;
import java.io.FileWriter;
import java.io.IOException;


public class copychar
{
	public static void main(String args[]) throws IOException
	{
		FileReader instream=null;
		FileWriter outstream=null;

		try
```

```
        {
            instream=new FileReader("test.txt");
            outstream=new FileWriter("charoutput.txt");
            int c;
            while ((c=instream.read())!=-1)
            {
                outstream.write(c);
            }
        }
        finally
        {
            if(instream != null)
            {
                instream.close();
            }
            if(outstream!=null)
            {
                outstream.close();
            }
        }
    }
}
```



```
import java.util.*;
class Main
{
    public static void main(String[] args)
    {
        Date d1 = new Date();
        System.out.println("Current date is " + d1);
        Date d2 = new Date(2323223232L);
        System.out.println("Date represented is "+ d2 );
    }
}
```

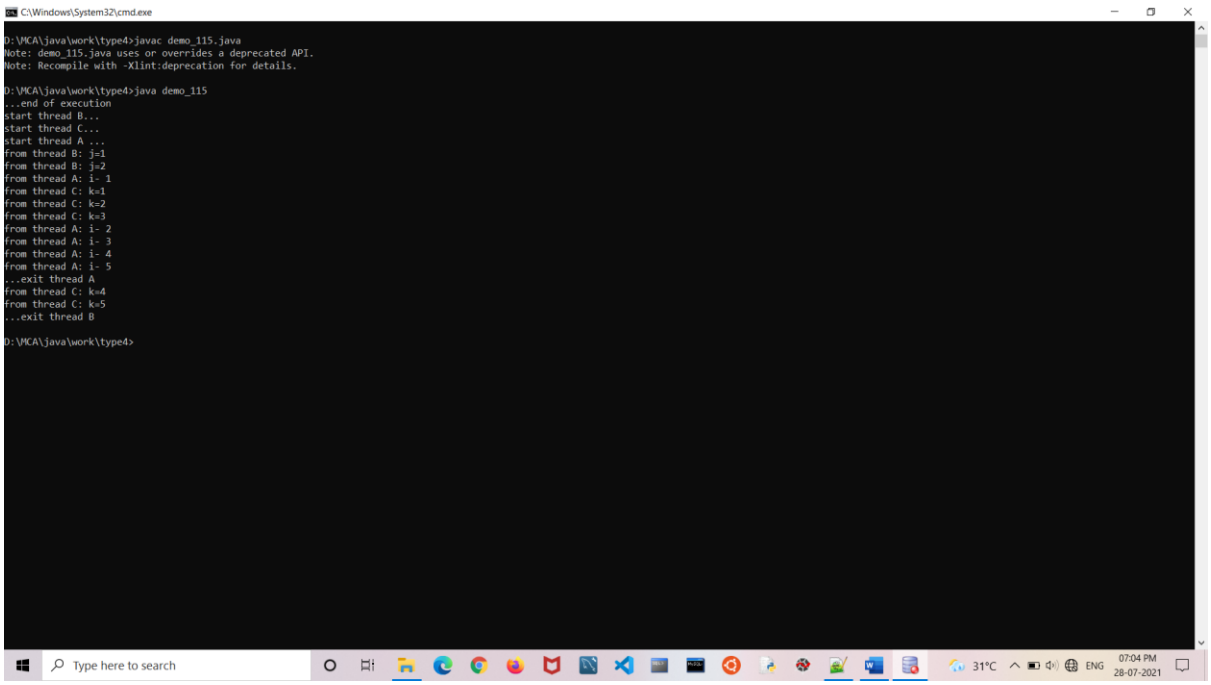
```
import java.util.*;
class datedemo
{
	public static void main(String args[])
	{
		Date currdate=new Date();
		System.out.println(currdate);
		Date epoch=new Date(0);
		System.out.println(epoch);
		long millis=System.currentTimeMillis();

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



```
class ClassA extends Thread
{
	public void run()
	{
		System.out.println("start thread A ...");
		for(int i=1;i<=5;i++)
		{
			if(i==1)yield();
			System.out.println("from thread A: i- "+i);
		}
		System.out.println("...exit thread A");
	}
}
class ClassB extends Thread
{
	public void run()
	{
		System.out.println("start thread B...");
		for(int j=1;j<=5;j++)
		{
			System.out.println("from thread B: j="+j);
			if(j==2) stop();
```

```
            }
            System.out.println("...exit thread B");
        }
}
class ClassC extends Thread
{
        public void run()
        {
            System.out.println("start thread C...");
            for(int k=1;k<=5;k++)
            {
                System.out.println("from thread C: k="+k);
                if(k==3)
                {
                    try
                    {
                        sleep(1000);
                    }
                    catch(Exception e){}
                }
            }
            System.out.println("...exit thread B");
        }
}
```

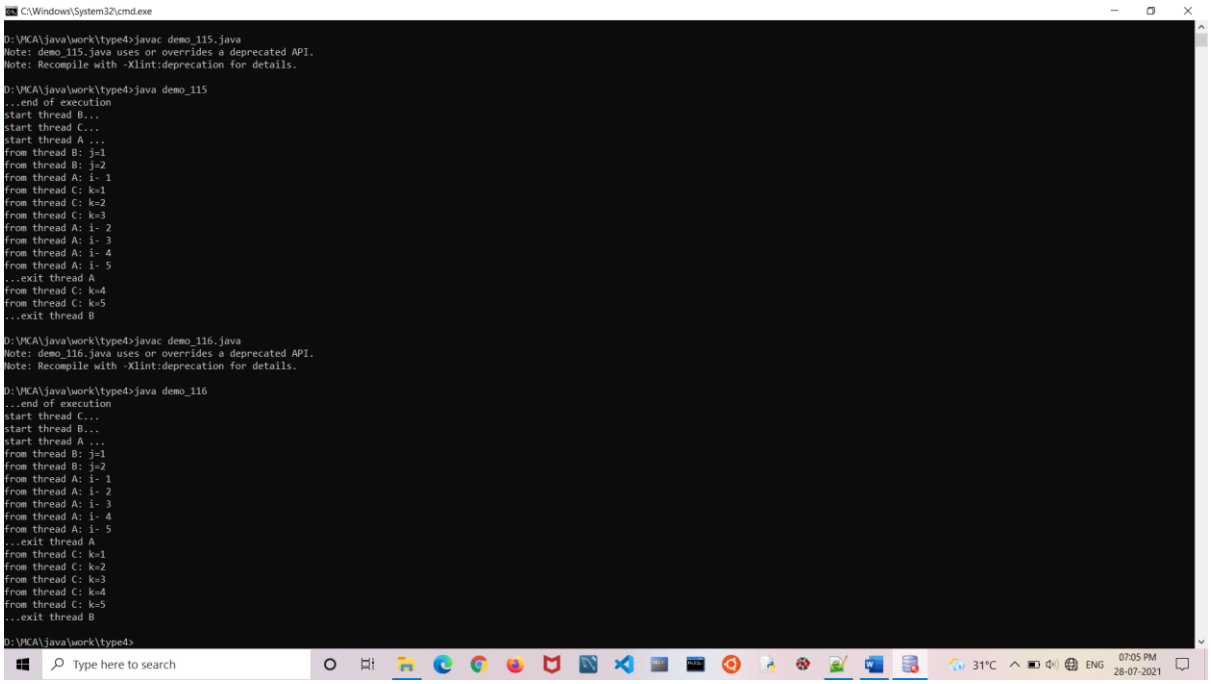
```
public class demo_115
{
	public static void main(String args[])
	{
		ClassA t1=new ClassA();
		ClassB t2=new ClassB();
		ClassC t3=new ClassC();
		t1.start();
		t2.start();
		t3.start();
		System.out.println("...end of execution");
	}
}
```



```
--------------------------------------------------------------------------------
class ClassA extends Thread
{
	public void run()
	{
		System.out.println("start thread A ...");
		for(int i=1;i<=5;i++)
		{
			if(i==1)yield();
			System.out.println("from thread A: i- "+i);
		}
		System.out.println("...exit thread A");
	}
}
class ClassB extends Thread
{
	public void run()
	{
		System.out.println("start thread B...");
		for(int j=1;j<=5;j++)
		{
			System.out.println("from thread B: j="+j);
			if(j==2) stop();
		}
```

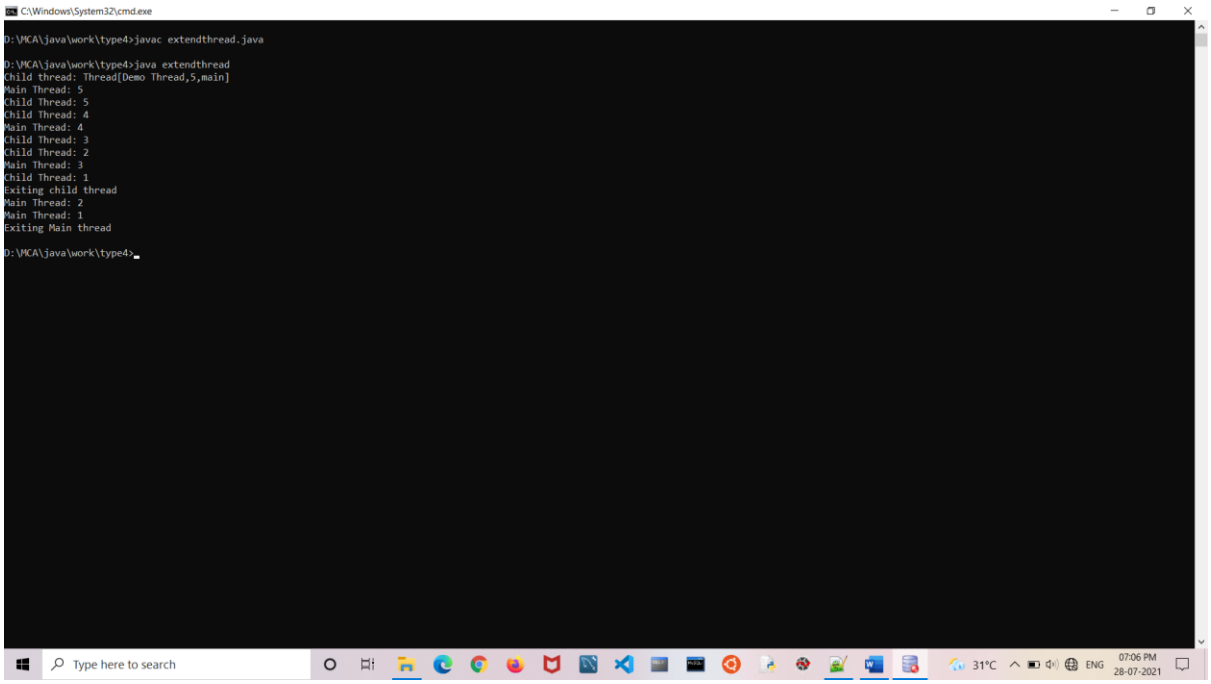
```
            System.out.println("...exit thread B");
        }
}
class ClassC extends Thread
{
        public void run()
        {
            System.out.println("start thread C...");
            for(int k=1;k<=5;k++)
            {
                System.out.println("from thread C: k="+k);
                if(k==3)
                {
                    try
                    {
                        sleep(1000);
                    }
                    catch(Exception e){}
                }
            }
            System.out.println("...exit thread B");
        }
}
```

```
public class demo_116
{
	public static void main(String args[])
	{
		ClassA t1=new ClassA();
		ClassB t2=new ClassB();
		ClassC t3=new ClassC();

		t3.setPriority(Thread.MAX_PRIORITY);
		t2.setPriority(t2.getPriority()+1);
		t1.setPriority(Thread.MIN_PRIORITY);

		t1.start();
		t2.start();
		t3.start();
		System.out.println("...end of execution");
	}
}
```



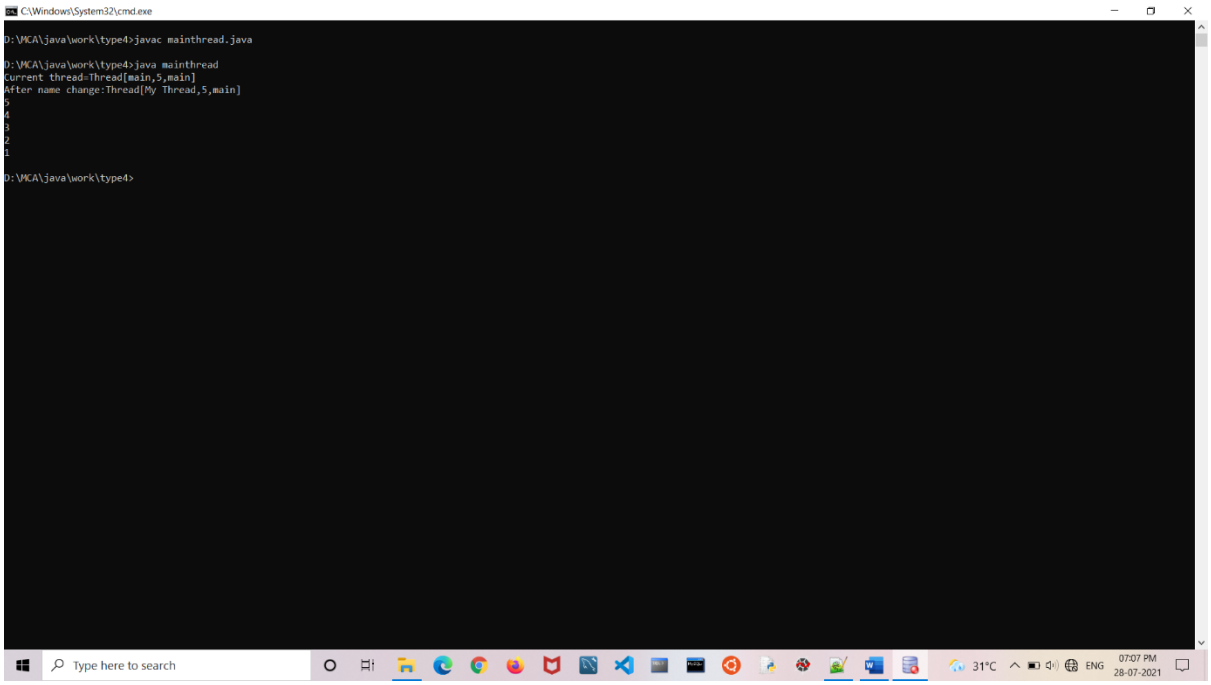
```
class newthread extends Thread{
        newthread(){
                super("Demo Thread");
                System.out.println("Child thread: " + this);
                start();
        }

        public void run(){
                        try{
                        for(int i=5;i>0;i--){
                                System.out.println("Child Thread: " + i);
                                Thread.sleep(500);
```

```
                }
            }catch(InterruptedException e){
                System.out.println("Child interrupted");
            }
            System.out.println("Exiting child thread");
        }
}
class extendthread{
        public static void main(String args[]){
            new newthread();

            try{
                for(int i=5;i>0;i--){
                    System.out.println("Main Thread: " + i);
                    Thread.sleep(1000);

                }
            }catch(InterruptedException e){
                System.out.println("Main interrupted");
            }
            System.out.println("Exiting Main thread");
            }

}
```



```
class mainthread{
        public static void main(String arg[]){
                Thread t = Thread.currentThread();
                System.out.println("Current thread=" + t);

                t.setName("My Thread");

                System.out.println("After name change:" + t);

                try{
                        for(int i=5;i>0;i--){
                                System.out.println(i);
                                Thread.sleep(1000);
```

```
                }
        }catch(InterruptedException e){
                System.out.println("Main Thread Interrupted");

        }

    }

}
```

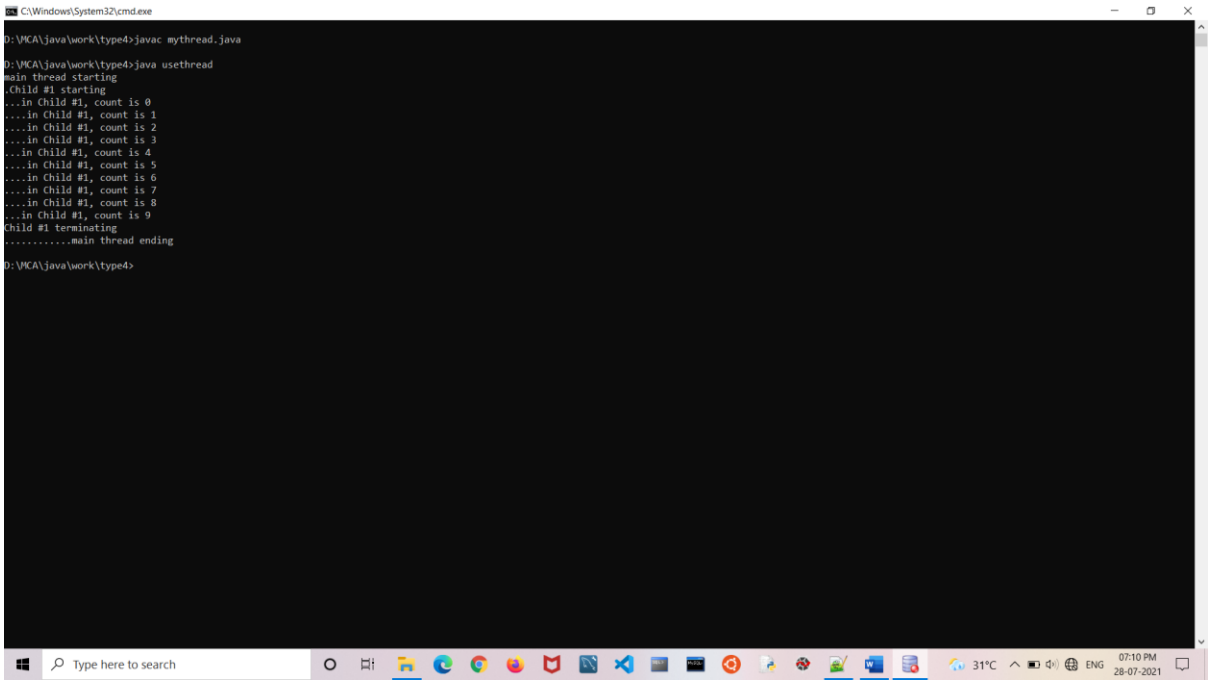


```
class mythread implements Runnable
{
        String thrdname;
        mythread(String name)
```

```
	{
		thrdname=name;
	}
	public void run()
	{
		System.out.println(thrdname + " starting ");
		try
		{
			for(int count=0;count<10;count++)
			{
				Thread.sleep(400);
				System.out.println("in "+thrdname+", count is
"+count);

			}
		}
		catch(InterruptedException exc)
		{
			System.out.println(thrdname+"interrupted.");
		}
		System.out.println(thrdname+" terminating");
	}
}
class usethread
```

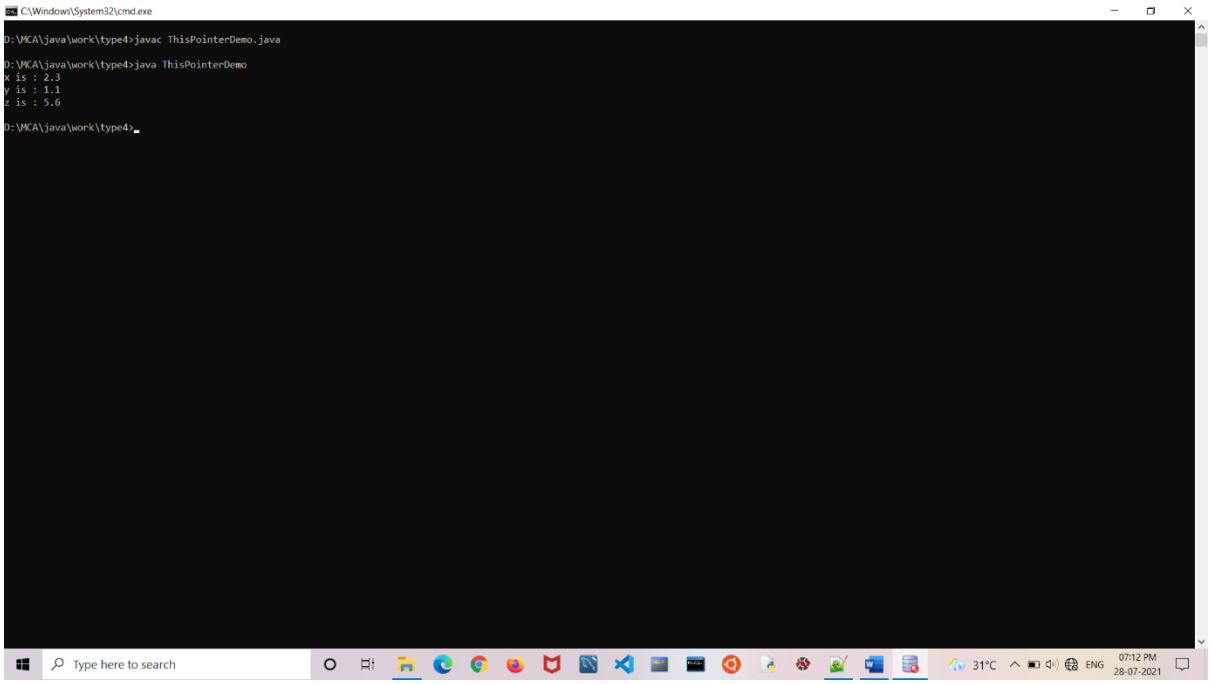
```
{
	public static void main(String args[])
	{
		System.out.println("main thread starting");

		mythread mt=new mythread("Child #1");

		Thread newThrd = new Thread(mt);

		newThrd.start();

		for(int i=0;i<50;i++)
		{
			System.out.print(".");
			try
			{
				Thread.sleep(100);
			}
			catch(InterruptedException exc)
			{
				System.out.println("main thread
interrupted.");
			}
		}
```

```
            System.out.println("main thread ending");
        }
}
```



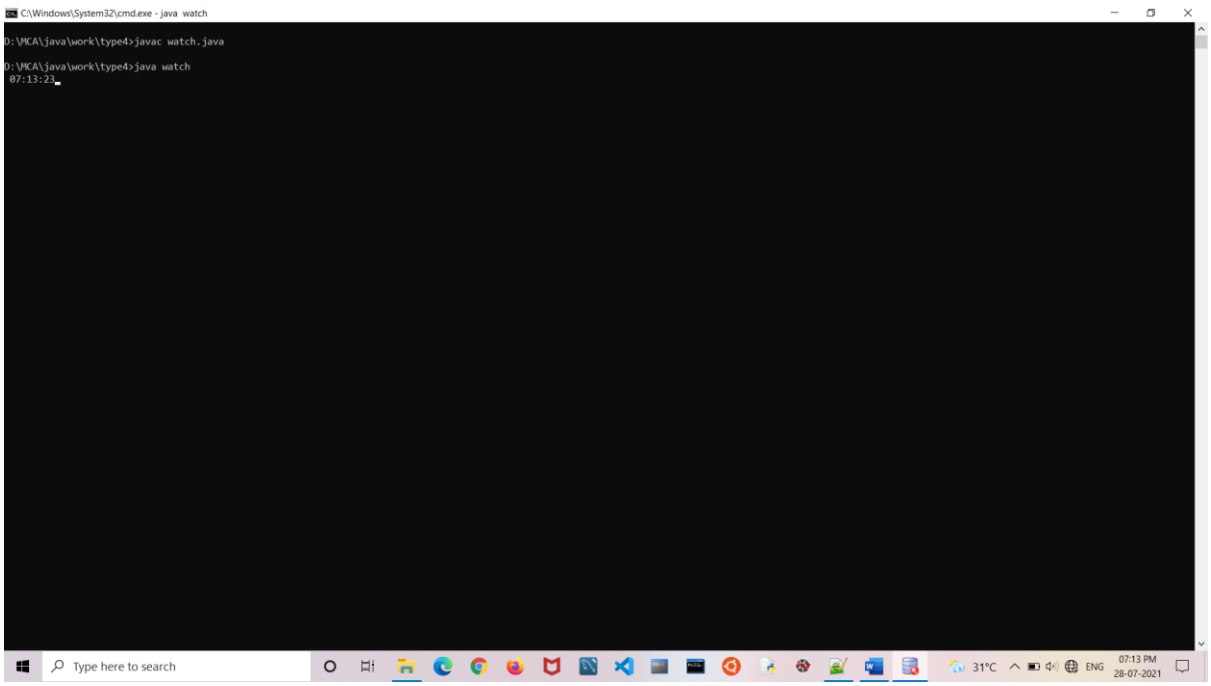
```
//demonstration of this pointer
//java can have more than one class


class Point3D
{
        double x;
        double y;
        double z;

        Point3D(double x,double y,double z)
```

```
        {
                this.x=x;
                this.y=y;
                this.z=z;
        }
}

class ThisPointerDemo
{
        public static void main(String args[])
        {
        Point3D p = new Point3D(2.3,1.1,5.6);

        System.out.println("x is : " + p.x);
        System.out.println("y is : " + p.y);
        System.out.println("z is : " + p.z);
        }
}
```



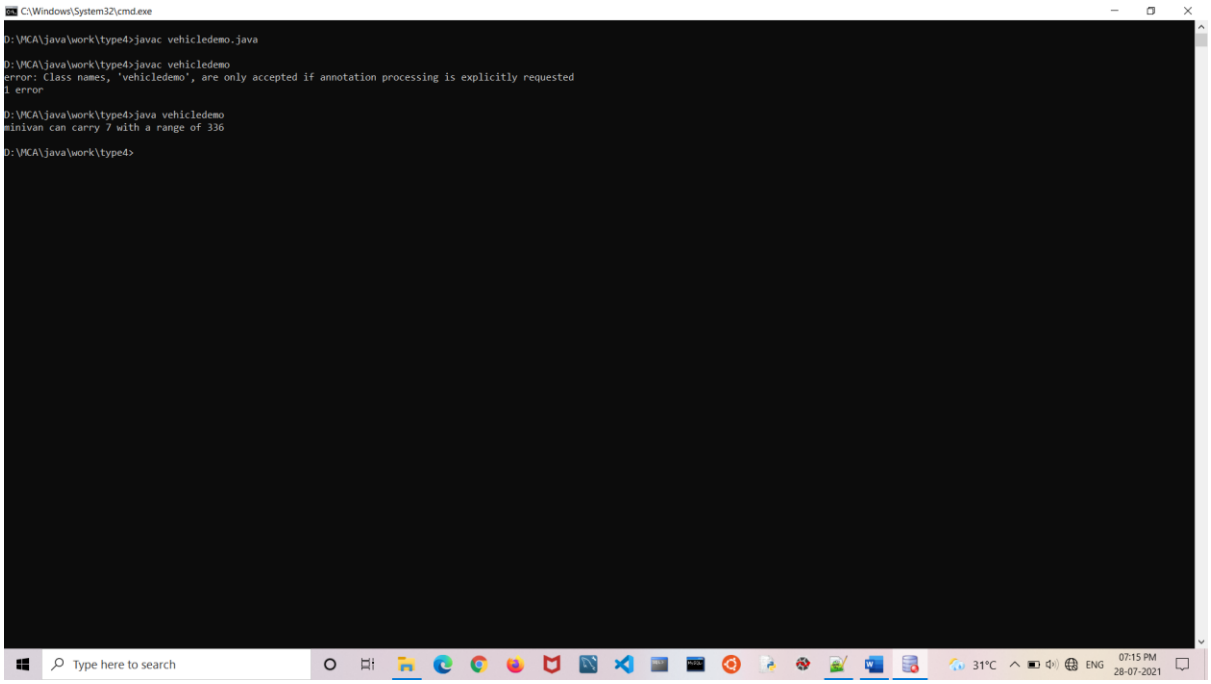
```
import java.text.DateFormat;
import java.text.SimpleDateFormat;
import java.util.Calendar;
import java.util.Date;

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

        while (true) {
            Date date = Calendar.getInstance().getTime();
            DateFormat formatter = new SimpleDateFormat("hh:mm:ss");
            String today = formatter.format(date);
            System.out.print("\r " + today);
```

```
            try {
                Thread.sleep(1000);
            } catch (InterruptedException e) {
                e.printStackTrace();
            }
        }
    }
}
```



```
/* A program that uses the vechicle class.
   call this file vehicledemo.java
*/
class vechicle{
```

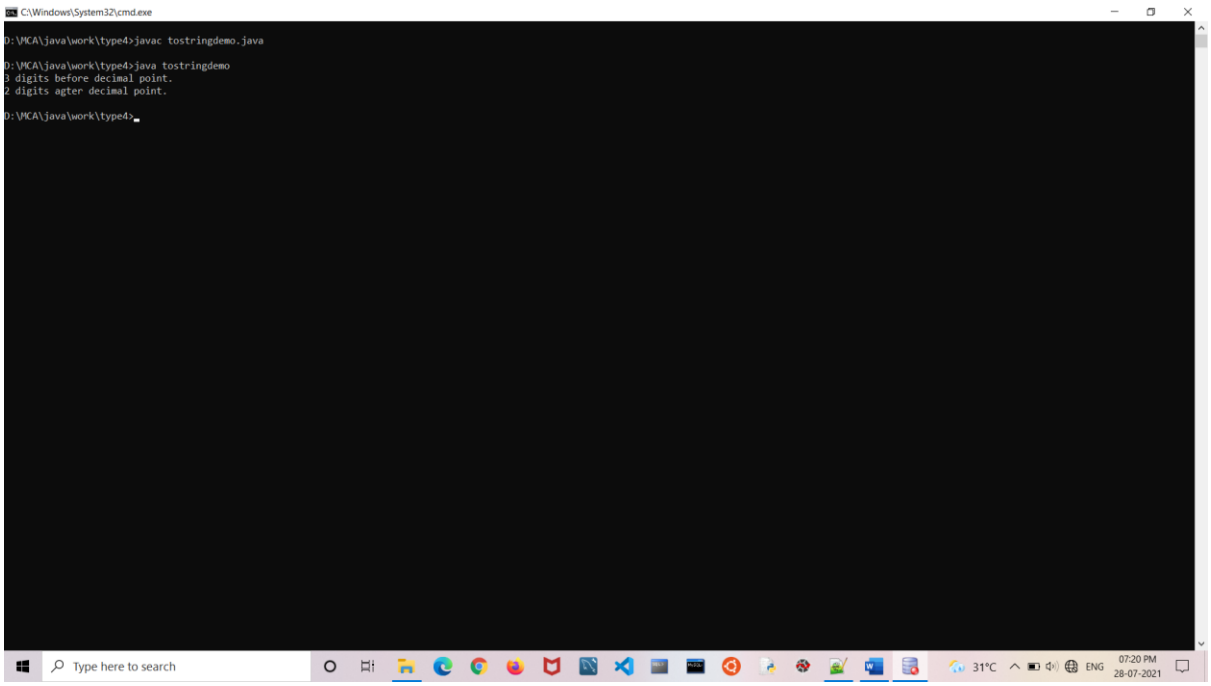
```
    int passengers;//number of passengers
    int fulecap;//fuel capacity in gallons
    int mpg;// fuel consumption in miles per gallon
}
//this class declares an object of type vehicle
class vehicledemo{
    public static void main(String args[])
    {
        vechicle minivan=new vechicle();
        int range;
        //assign values to fields in minivan
        minivan.passengers=7;
        minivan.fulecap=16;
        minivan.mpg=21;
        //computer the range assuming a full tank of gas
        range=minivan.fulecap * minivan.mpg;
        System.out.println("minivan can carry " + minivan.passengers + "
with a range of "+ range);
    }
}
```



```
class vehicle
{
        private int passenges;
        private int fuelcap;
        private int mpg;

        vehicle(int p,int f,int m)
        {
                passengers=p;
                fuelcap=f;
                mpg=m;
        }
        int range()
```

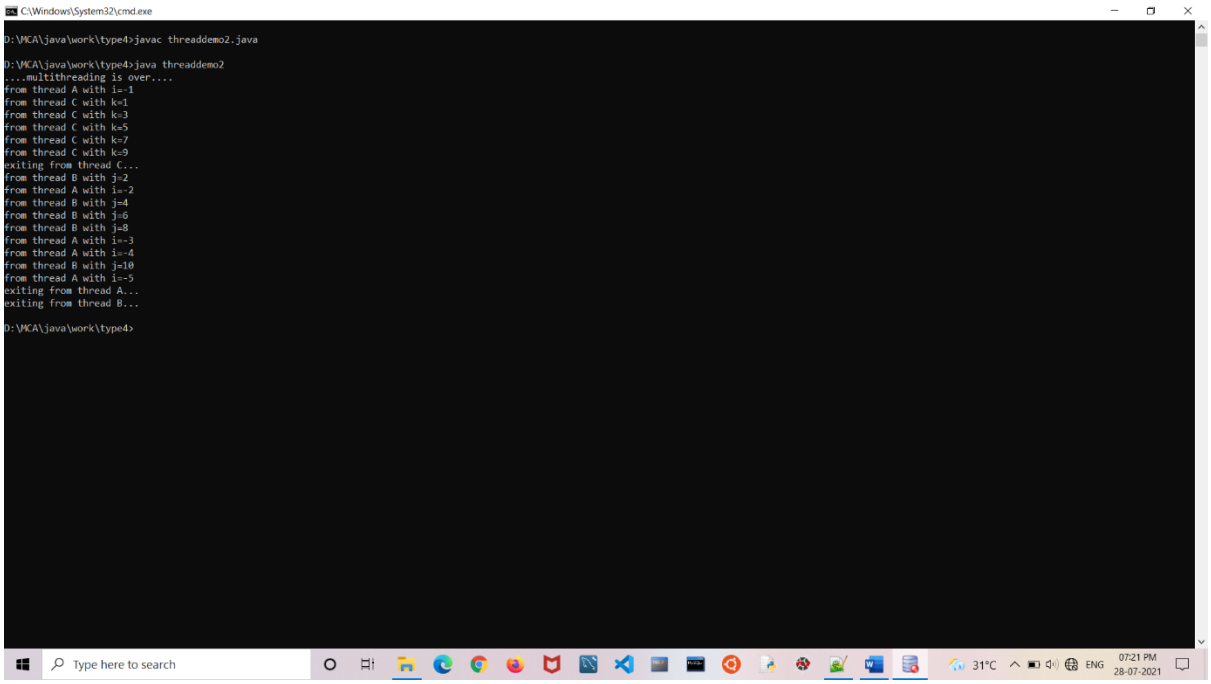
```
        {
                return mpg*fuelcap;
        }
        double fuelneeded(int miles)
        {
                return() miles/mpg;
        }
        int getpassengers()
        {
                return passengers;
        }
        void setpassengers(int p)
        {

        }
        int getFuelcap()
        {


        }
}

public class tostringdemo
{
```

```
	public static void main(String args[])
	{
		double d=858.25;
		String s=Double.toString(d);

		int dot=s.indexOf('.');

		System.out.println(dot+" digits "+"before decimal point.");
		System.out.println((s.length()-dot-1)+" digits agter decimal point.");
	}
}
```



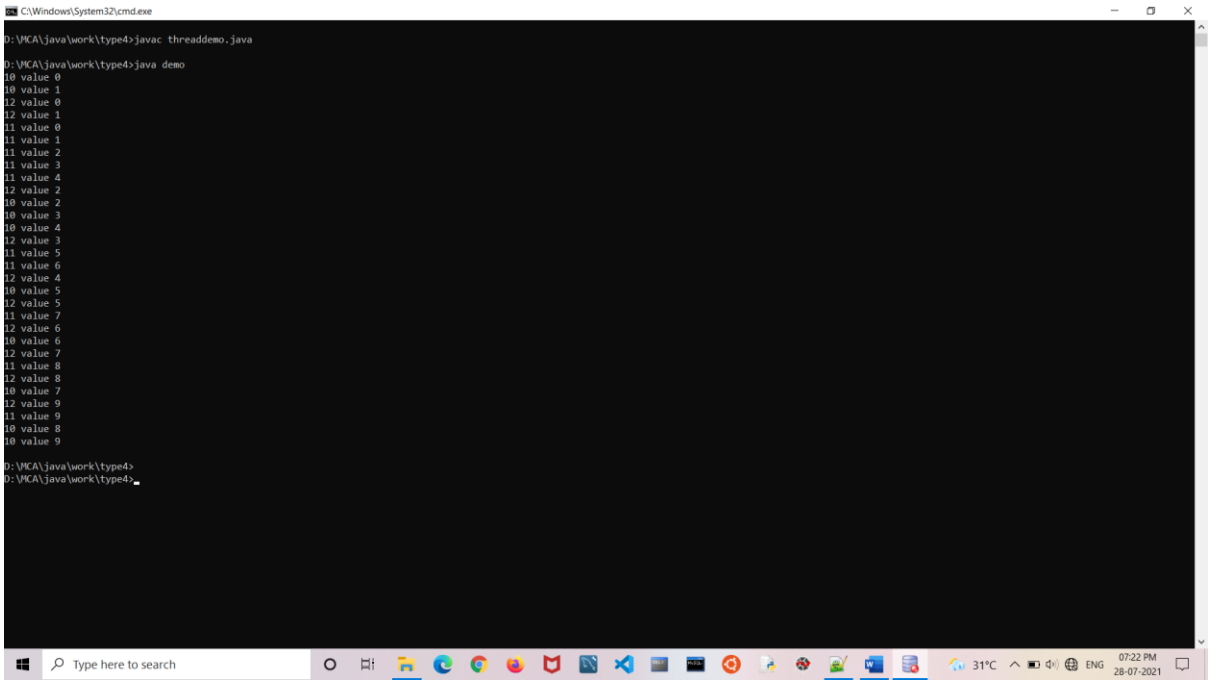
```
class threadA extends Thread
{
	public void run()
	{
		for(int i=1;i<=5;i++)
		{
			System.out.println("from thread A with i=" + -1*i);
		}
		System.out.println("exiting from thread A...");
	}
}
class threadB extends Thread
{
	public void run()
	{
		for(int j=1;j<=5;j++)
		{
			System.out.println("from thread B with j=" + 2*j);
		}
		System.out.println("exiting from thread B...");
	}
}
class threadC extends Thread
```

```
{
	public void run()
	{
		for(int k=1;k<=5;k++)
		{
			System.out.println("from thread C with k="+ (2*k-
1));
		}
		System.out.println("exiting from thread C...");
	}
}
class threaddemo2
{
	public static void main(String args[])
	{
		threadA a=new threadA();
		threadB b=new threadB();
		threadC c=new threadC();
		a.start();
		b.start();
		c.start();
		System.out.println("....multithreading is over....");
	}
}
```



```
class myclass extends Thread
{
	public void run()
	{
		for(int i=0;i<10;i++)
		{
			System.out.println(Thread.currentThread().getId()+" value "+i);
		}
		try
		{
			Thread.sleep(0);
		}
```

```
            catch(InterruptedException e)
            {
                  e.printStackTrace();
            }
      }
}
class demo
{
      public static void main(String args[])
      {
            myclass m1=new myclass();
            m1.start();

            myclass m2=new myclass();
            m2.start();

            myclass m3=new myclass();
            m3.start();
      }
}
```



```
package inheritance;

 import java.util.Date;
 import java.util.GregorianCalendar;

 public class Employee
 {
    private String name;
    private double salary;
    private Date hireDay;

    public Employee(String n, double s, int year, int month, int day)
    {
```

```
   name = n;
   salary = s;
   GregorianCalendar calendar = new GregorianCalendar(year, month
-1, day);
   hireDay = calendar.getTime();
   }

   public String getName()
   {
   return name;
   }
   public double getSalary()
   {
   return salary;
   }

   public Date getHireDay()
   {
   return hireDay;
   }

   public void raiseSalary(double byPercent)
   {
   double raise = salary * byPercent / 100;
```

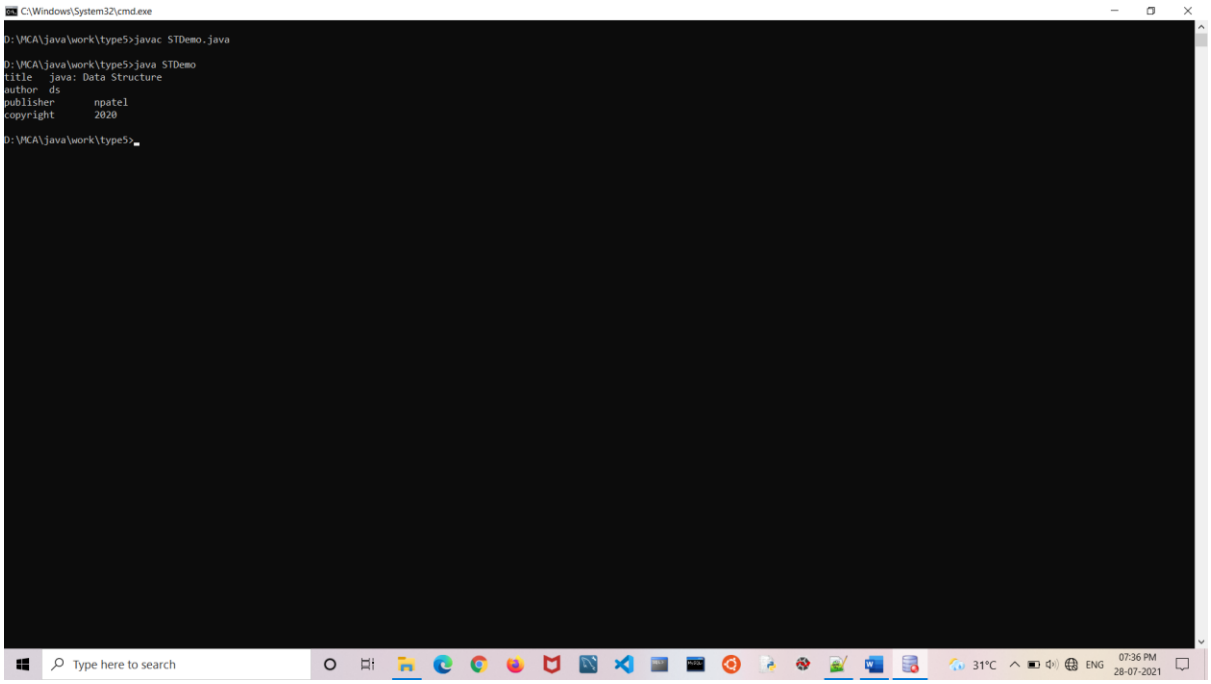
```
   salary += raise;
   }
 }

package inheritance;
public class Manager extends Employee
{
   private double bonus;

   public Manager(String n, double s, int year, int month, int day)
   {
      super(n, s, year, month, day);
      bonus = 0;
   }

   public double getSalary()
   {
      double baseSalary = super.getSalary();
      return baseSalary + bonus;
   }

   public void setBonus(double b)
   {
      bonus = b;
```

```
    }
}

package inheritance;

public class ManagerTest
{
    public static void main(String[] args)
    {
    // construct a Manager object
    Manager boss = new Manager("Carl Cracker", 80000, 1987, 12,
15);
    boss.setBonus(5000);
    Employee[] staff = new Employee[3];
    // fill the staff array with Manager and Employee objects
    staff[0] = boss;
    staff[1] = new Employee("Harry Hacker", 50000, 1989, 10, 1);
    staff[2] = new Employee("Tommy Tester", 40000, 1990, 3, 15);
    // print out information about all Employee objects
    for (Employee e : staff)
        System.out.println("name=" + e.getName() + ",salary=" +
    e.getSalary());
    }
 }
```

Class Work – Synchronization , Exception , LocalDate and LocalTime , StringTokenizer

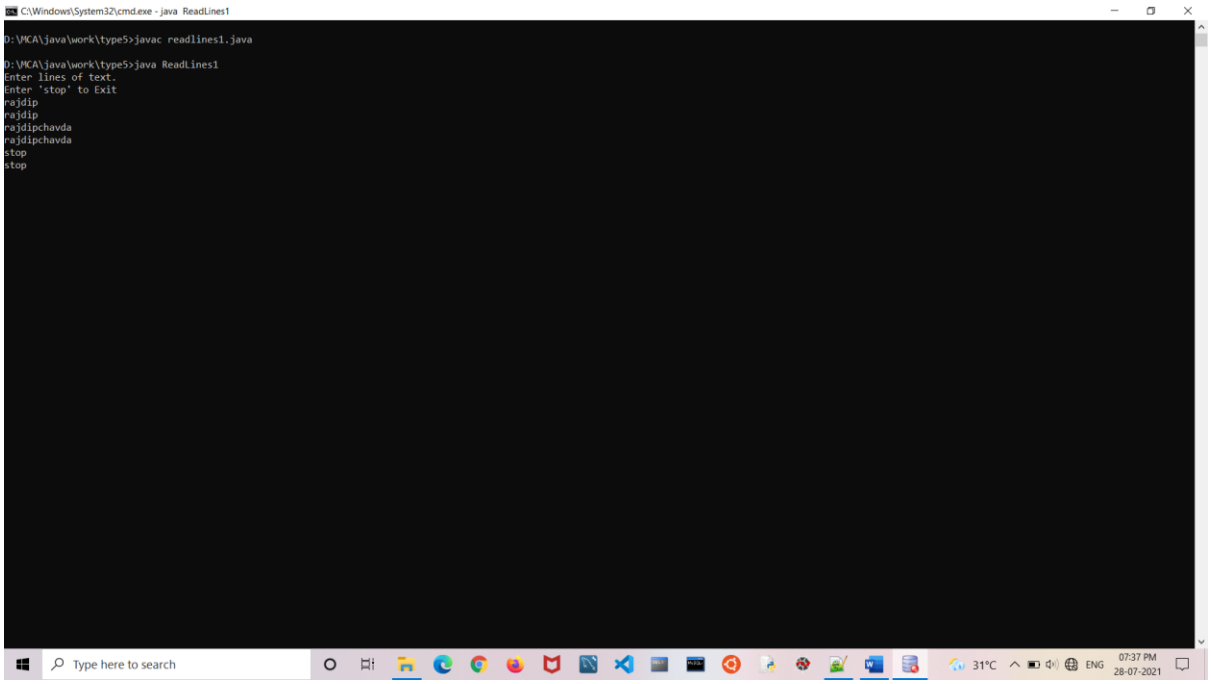
---

```
import java.util.StringTokenizer;
class STDemo
{
	static String in="title=java: Data Structure;"+"author=ds;"+"publisher=npatel;"+"copyright=2020";

	public static void main(String args[])
	{
		StringTokenizer st=new StringTokenizer(in,"=;");
		while(st.hasMoreTokens())
		{
			String key=st.nextToken();
			String val=st.nextToken();
			System.out.println(key+"\t"+val);
		}
	}
}
```



```
import java.io.*;
class ReadLines1
{
	public static void main(String args[]) throws IOException
	{
		BufferedReader br=new BufferedReader (new InputStreamReader(System.in));
		String str;

		System.out.println("Enter lines of text.");
		System.out.println("Enter 'stop' to Exit");
			str="a";
			while ((!str.equals ("stop")) || (!str.equals ("STOP")))
```

```
                {
                        str=br.readLine();
                        System.out.println(str);
                }
        }
}
```



```
class Q
{
        int n;
        boolean flag=false;
        synchronized void put(int n)
        {
                if(flag)
```

```
        {
                try
                {
                        wait();
                }
                catch(InterruptedException e)
                { };
        }
        this.n=n;
        System.out.println("Produce :"+n);
        flag=true;
        notify();
}
synchronized int get()
{
        if(! flag)
        {
                try
                {
                        wait();
                }
                catch(InterruptedException e)
                { };
        }
```

```
        }
}
class Consumer implements Runnable
{
        Q q;
        Consumer (Q q )
        {
                this.q =q;
                new Thread (this).start( );
        }
        public void run()
        {
                while(true)
                {
                        q.get();
                }
        }
}
class public_120
{
        public static void main(String args[])
        {
                Q q=new Q();
                new Producer(q);
```

```
            new Consumer(q);


        }
}
```

```
import java.io.IOException;


public class jumbalename implements Runnable
{
        public jumbalename(String firstname,String secendname,long
delay)
        {
                this.firstname=firstname;
                this.secendname=secendname;
                aWhile=delay;
        }
        public void run()
        {
                try
                {
                        while(true)
                        {
                                System.out.println(firstname);
                                Thread.sleep(aWhile);
```

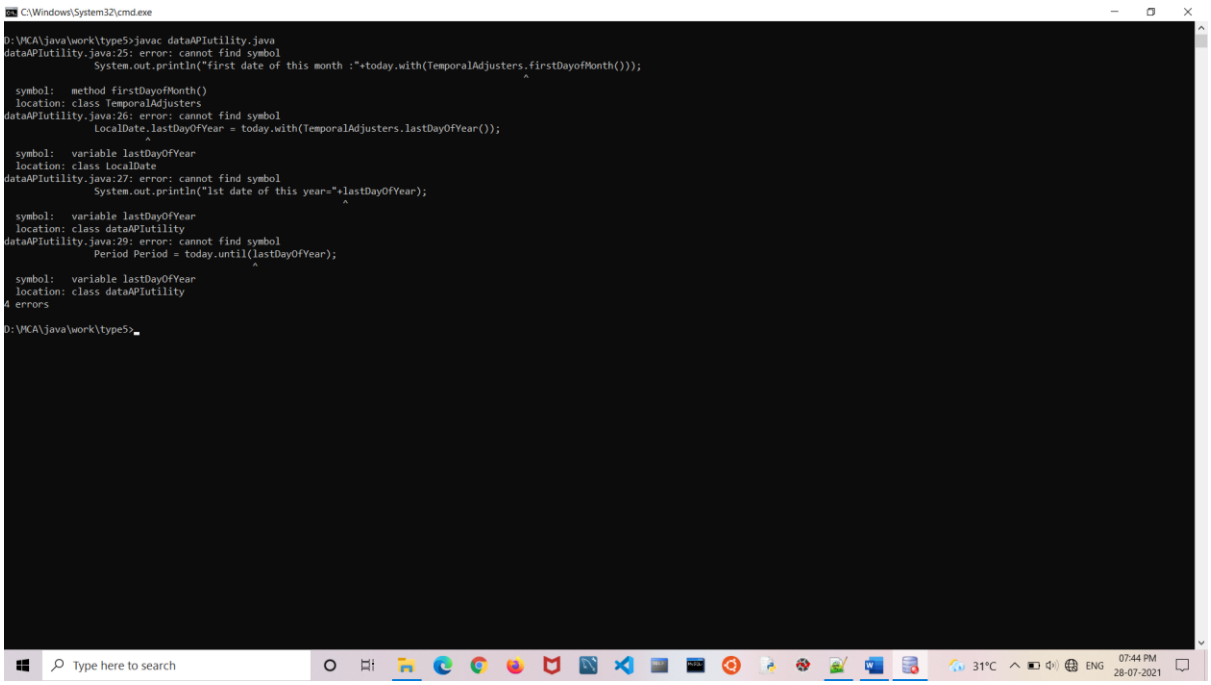
```
                    System.out.println(secendname+"\n");
                }
            }
            catch(InterruptedException e)
            {
                system.out.println(firstname+secendname+e);
            }
        }
        public static void main(String args[])
        {
            Thread first =new Thread(new
jumbalename("rajdip","chavda",200L));
            Thread secend =new Thread(new
jumbalename("abhiraj","makwana",300L));
            Thread third =new Thread(new
jumbalename("prajval","rai",500L));

            first.setDaemon(true);
            secend.setDaemon(true);
            third.setDaemon(true);

            System.out.println("press enter when you have had
enough...\n");

            first.start();
```

```
                secend.start();
                third.start

                try
                {
                        System.in.read();
                        System.out.println("Enter pressed...\n");
                }
        }
}
```

```
import java.time.LocalDate;
import java.time.LocalTime;
import java.time.Period;
import java.time.temporal.TemporalAdjusters;

public class dataAPIutility
{
        public static void main(String args[])
        {
                LocalDate today=LocalDate.now();
                System.out.println("year "+today.getYear()+" is leap Year
? "+ today.isLeapYear());
```

```
System.out.println("Today is before
01/01/2015?"+today.isBefore(LocalDate.of(2015,1,1)));

System.out.println("current
time="+today.atTime(LocalTime.now()));

System.out.println("10 days after today will be
"+today.plusDays(10));
System.out.println("3 weeks after today will be
"+today.plusDays(3));
System.out.println("20 months after today will be
"+today.plusDays(20));

System.out.println("10 days before today will be
"+today.minusDays(10));
System.out.println("3 weeks before today will be
"+today.minusDays(3));
System.out.println("20 months before today will be
"+today.minusDays(20));

System.out.println("first date of this month
:"+today.with(TemporalAdjusters.firstDayofMonth()));
LocalDate.lastDayOfYear =
today.with(TemporalAdjusters.lastDayOfYear());
System.out.println("lst date of this
year="+lastDayOfYear);
```

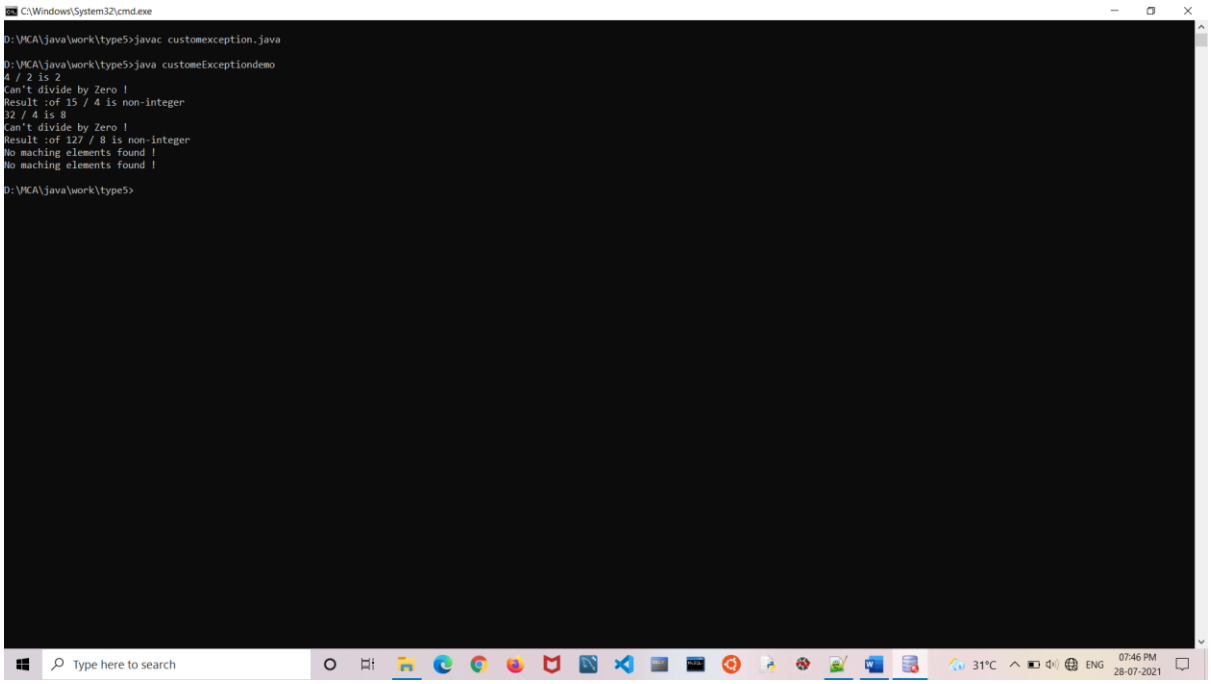
```
        Period Period = today.until(lastDayOfYear);
        System.out.println("period format "+Period);
        System.out.println("months remainnig in the year
"+Period.getMonths());

    }
}
```



```
class NonIntResultException extends Exception
{
    int n;
    int d;
    NonIntResultException(int i,int j)
```

```
	{
		n=i;
		d=j;
	}
	public String toString()
	{
		return "Result :of " +n+" / "+d+" is non-integer ";
	}
}
class customeExceptiondemo
{
	public static void main(String args[])
	{
		int numer[]={4,8,15,32,54,127,256,512};
		int denom[]={2,0,4,4,0,8};
		for(int i=0;i<numer.length;i++)
		{
		try
		{
			if((numer[i]%2) != 0)
			{
				throw new
NonIntResultException(numer[i],denom[i]);
			}
```

```
                System.out.println(numer[i] + " / "+denom[i] + " is "
+ numer[i]/denom[i]);
            }
            catch( ArithmeticException exc)
            {
                System.out.println("Can't divide by Zero !");
            }
            catch(ArrayIndexOutOfBoundsException exc)
            {
                System.out.println("No maching elements found !");
            }
            catch(NonIntResultException exc)
            {
                System.out.println(exc);
            }
            }
        }
}
```



```
class Account
{
	public int balance;
	public int accountNo;
	void displayBalance()
	{
		System.out.println(" Accouunt Number "+ accountNo + " Balance "+ balance);
	}

	synchronized void deposit(int amount)
	{
		balance=balance+amount;
```

```
            System.out.println(amount+ " is deposited ");
            displayBalance();
        }
        synchronized void withdraw(int amount)
        {
            balance=balance-amount;
            System.out.println(amount+ " is withdrawn ");
            displayBalance();
        }
}
class TransactionDeposit implements Runnable
{
        int amount;
        Account accountX;
        TransactionDeposit(Account x,int amount)
        {
            accountX=x;
            this.amount=amount;
            new Thread(this).start();
        }
        public void run()
        {
            accountX.deposit(amount);
        }
```

```
}
class TransactionWithdraw implements Runnable
{
	int amount;
	Account accountY;
	TransactionWithdraw(Account y,int amount)
	{
		accountY=y;
		this.amount=amount;
		new Thread(this).start();
	}
	public void run()
	{
		accountY.deposit(amount);
	}
}
class SynchronizedDemo
{
	public static void main(String[] args) {
		Account Abc=new Account();
		Abc.balance=1000;
		Abc.accountNo=111;
		TransactionDeposit t1;
		TransactionWithdraw t2;
```

```
            t1=new TransactionDeposit(Abc,500);
            t2=new TransactionWithdraw(Abc,900);
        }
}
```