1.FACTORIAL OF GIVEN NUMBER:

```
public class Factorial {
    public static void main(String[] args) {
                int n=5;
                int i;
                 int fact=1;
                for(i=1;i<=n;i++){
                fact*=i;
              }
           System.out.println("factorial of 5="+fact);
           }

}
```

OUTPUT:

factorial of 5=120

2.FIBONACCI SERIES UPTO 10:

```
public class Fibonacci{
    public static void main(String ar[]){
          int  f1=0,f2=1,f3;
          {
         System.out.print(f1);
        System.out.print(" "+f2);
         for(int i=2;i<=10;i++){
                f3=f2+f1;
          System.out.print(" "+f3);
            f1=f2;
          f2=f3;
        }
```

```
        }
    }
}
```

OUTPUT:

0 1 1 2 3 5 8 13 21 34 55

3.FIND GIVEN NUMBER ODD OR EVEN:

```
public class OddEven_3 {
    public static void main(String[] args) {
                int n=89;
                switch( n%2){
                case 0:
                System.out.println(n + " is even number");
                break;
                case 1:
                System.out.println(n +"is odd number ");
                break;
                }
        }
}
```

OUTPUT:

89is odd number

4.**FIND GIVEN NUMBER ODD OR EVEN WITHOUT USING CONDITION STATEMENT:**

```
public class OddEvenDemo {
    public static void main(String args[]) {
    int     a;
    a =5;
    int b = a%2;
```

```
        String c = b==0?"Even number":"Oddnumber";
        System.out.println(c);

        }
}
```

OUTPUT:

Oddnumber

5.CheckEvenOrOdd WithOut Using Modulus and Division.

```
public  class CheckEvenOrOdd {
        public static void main(String[] args) {
                int n=13;
                if((n&1)==1)System.out.println("Given Number is Odd.");
                else System.out.println("Given Number is Even");
          }
}
```

Output:

Given Number is Odd.

6.SWAP OF TWO NUMBER WITHOUT USING TEMPORARY VARIABLE:

```
public class swap  {
       public static void main(String args[]){
        int a=1;
         int b=2;
          int t;
//without using temporary variables.
         System.out.println(a);
         System.out.println(b);
         a=a+b;
```

```
        b=a-b;
         a=a-b;
        System.out.println(a);
         System.out.println(b);
//with using temporary variables.
        t=a;
         a=b;
         b=t;
System.out.println(a);
System.out.println(b);
}
}
```

Output:

a=1

b=2

After swap:

a=2

b=1

7.Write a program to add two number without using addition operator.

```
public class Add{
    public static void main(String args[]){
         int a = 3, b = 7;
        int c = a-(-b);
        System.out.println("add" + c);
      }
 }
```

Output : 10

8.SecondHighestNumber in An Integer Array

```
public class secondhighestintegerarray {
	public static void main(String[] args) {
		int temp,size;
		int array[]={10,20,25,63,96,57};
		size=array.length;
		for(int i=0;i<size;i++){
			for(int j=i+1;j<size;j++){
				if(array[i]>array[j]){
					temp=array[i];
					array[i]=array[j];
					array[j]=temp;
				}
			}
		}

		System.out.println("Second highest="+array[size-2]);
	}
	}
```

output

Second highest=63

9.SUM OF DIGITS OF EVEN NUMBERS:

```
 public class sumofdigit {
	public static void main(String ar[]) {
		int n=23435;
		int last;
		int sum=0;
```

```
                int even=0;
                int odd=0;
                  while(n!=0){
                last=n%10;
                if(last%2==0)
                {
                      even=even+last;
                }
                else {
                 odd=odd+last;
                }
           n=n/10;
        }
           System.out.println(even);
           System.out.print(odd);
        }
}
```

OUTPUT: 6 11

10.SUM OF DIGITS USING RECURSIVE FUNCTION

```
public class SumOfDigitsRecursive{
        private static int sumofDigits(int num){
             if (num == 0 ){
             return 0;
             }
             return num%10 + sumofDigits(num/10);
              }
            public static void main(String args[]){
                  int sum = sumofDigits(2167);
```

```
                System.out.println("SumOfDigits is = "+sum);
            }
}
```

OUTPUT:

SumOfDigits is = 16

11. SUM OF N NUMBERS IN AN ARRAY:

```
public class SumArray {

        public static void main(String[] args) {
                int nums[] = {1,2,3,4,5,6};
                int sum = 0;
                int i;
                System.out.println("nums length is"+nums.length);
                for(i=0;i<6;i++){
                        sum = sum + nums[i];

                }
                System.out.println("sum is"+sum);

        }

}
```

OUTPUT:

```
  nums length is6
 sum is21
```

12.CALCULATE SUM OF N GIVEN NUMBERS:

```
public class SumOfNnumbers{
        public static void main(String[] args) {
                int n=100;
                int a=(n*(n+1))/2;
                System.out.println("Sum of a="+a);
        //WithOut Using Formula-->.
```

```
                int sum=0;
                for(int i=1;i<=n;i++){
                        sum+=i;
                }
                System.out.println("Sum of Sum="+sum);
        }

}
```

Output:

Sum of a=5050

Sum of Sum=5050

13:CALCULATE SUM OF 1….N NUMBERS:

```
import java.util.Scanner;
class sumofnnumbers{
        public static void main(String args[]){
             long a=0;
                System.out.println("enter the numbers");
                Scanner s =  new Scanner(System.in);
                long b = s.nextLong();
                for (int i=0;i<b;i++){
                        a=a+i;
                }
        System.out.println(a);
}
}
```

Output:

input:5

output:15

14.write a program to check whether the sort-1 is prime or not.if not prime , print the factors.

```
public class SqureRoot {
public static void main(String[] args) {
double sq = Math.sqrt(1);
        System.out.println(sq);
           if(sq>=1) {
              int c = 0;
               for(int i=1; i<=sq; i++) {
                 if((sq%i) == 0)
                   System.out.println(i + " ");
                   c++;
                 }
                if( c==2)     System.out.println("prime no");
                else            System.out.println("not a prime number");
                 }
           }
}
```

output:

1.0

1

not a prime number

15.1 to 100 prime numbers and Calculate Average

```
public class printPrimeandAverage {
         public static void main(String[] args) {
                  int sum=0,a=0;float avg;
                  for(int i=2;i<=100;i++){
```

```
                boolean b=true;
                        for(int j=2;j<i;j++){
                    if(i%j==0){
                    b=false;
                    break;
                    }
                }
                if(b==true){
                        System.out.print(i+" ");
                        sum=sum+i;
                        a++;
                       }
                 }
        System.out.println(sum);
        avg=sum/a;
        System.out.println(avg);
            }
    }
```

output

2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97 1060

42.0

16:REMOVE DUPLICATE NUMBER IN ARRAY:

```
Import java.util.Arrays;
public class RemoveDuplicate1{
public static int remove(int a[],int n){
if(n==0||n==1){
return n;
      }
```

```
int[] t=new int[n];
int j=0;
for(int i=0;i<n-1;i++){
if(a[i]!=a[i+1]{
t[j++]=a[i];
}
}
t[j]=a[n-1];
//changing original array
for(int i=0;i<j;i++){
a[i]=t[i];
}
return j;
}
public static void main(String args[]){
int a[]={10,20,45,23,122,12,10,10,70};
Arrays.sort(a);
int length=a.length;
length=remove(a,length);
//printing array
for(int i=0;i<length;i++){
System.out.print(a[i]+"");
    }
}
```

OUTPUT:

10 12 20 23 43 70 122

17:COMMON ELEMENTS BETWEEN TWO ARRAY:

```
public class CommonElements_17{
```

```
public static void main(String[] args) {
                    int array1[] = {1,2,5,7,6,9,8};
                    int array2[] = {1,0,6,4,7,4,9};
                    for(int i= 0;i< array1.length;i++)
                    {
                        for(int j= 0;j< array1.length;j++)
                            {
                            if(array1[i] == array2[j])
                     System.out.println("Commen elements: "+(array1[i]));
            }
        }
    }
}
```

OUTPUT

Commen elements: 1

Commen elements: 7

Commen elements: 6

Commen elements: 9

18. ANSWER IS PROGRAM NO:8

19:PALINDROME OR NOT:

```
import java.util.*;
public class Palindrome{
        static boolean isPalindrome(String s){
                int i=0;
                int j=s.length()-1;
                while(i<j){
                        if((s.charAt(i))!=(s.charAt(j)))
                                return false;
                i++;
                j--;
                }
                return true;
        }
```

```
	public static void main(String[] args) {
		System.out.print("Enter the String: ");
		Scanner sc=new Scanner(System.in);
		String s=sc.next();
		if(isPalindrome(s))
			System.out.println("Given String is Palindrome");
		else
			System.out.println("Given String is Not aPalindrome");

	}
}
```

Output:
Enter the String: mam
Given String is Palindrome.

20:PRINT NUMBERS IN WORDS:

```
import java.util.*;

class Oneto100Words{

public static void main(String args[]){

Scanner s = new Scanner(System.in);

int n,digit1,digit2;

String a[] = {" "," ","twenty","thirty","forty","fifty","sixty","seventy","eighty","ninety","hundred"};

String b[]={","one","two","three","four","five","six","seven","Eight","nine","ten","eleven",

"twelve","thirteen","fourteen","fifteen","sixteen","seventeen",

"eighteen","nineteen"};

n = s.nextInt();

if((n<=-1)||(n>=101))

{

System.out.println("wrong input");

}

 if((n>=1)&&(n<=19))

{

System.out.println(b[n]);
```

```
}
 if((n>=20)&&(n<=100))
{
digit1=n/10;
digit2=n%10;
System.out.println(a[digit1]+" "+b[digit2]);
}
}
}
```

Output:

input:22

Twenty Two

21. S = 3+33+333+3333.....?

```
public class S {
public static void main(String[]args) {
        long a = 0;
        int n = 5;
        for(int i= 1; i<=n; i++) {
        a =(a*10)+3;
        System.out.print(a);
          System.out.print("+");
                }
           }
}
```

OUTPUT :

3+33+333+3333+33333+

22.Convert Decimal To Binary Format

```
public class decitobinary {
  public static void tobinary(int decimal){
  int binary[]=new int[10];
  int index=0;
  while(decimal>0){
            binary[index++]=decimal%2;
            decimal=decimal/2;
  }
for(int i=index-1;i>=0;i--){
System.out.print(binary[i]);
                    }
          System.out.println();
  }
  public static void main(String[] args) {
                    System.out.println("decimal of 10 is=");
                    tobinary(12);
                    System.out.println("decimal of 21 is=");
                    tobinary(2);
                    System.out.println("decimal of 31 is=");
                    tobinary(3);
         }
}
```

Output

```
decimal of 10 is=1100
decimal of 21 is=10
decimal of 31 is=11
```

23:SORT NUMBER IN ASCENDING ORDER USING BUBBLE SORT ALGORITHAM:

```
public class Bubblesortarray {
```

```
        public static void main(String ar[]){
                int a[]={12,32,1,322,23,1324};
                int temp=0;
                int n=a.length;
                for(int i=0;i<n;i++)
                {
                        for(int j=i+1;j<n;j++){
                                        if(a[i]>a[j]){
                                          temp=a[i];
                                          a[i]=a[j];
                                          a[j]=temp;
                                }
                          }
                          System.out.print(" "+a[i]);
                }
         }
}
```

Output : 1 12 23 32 322 1324

24: SORT NUMBER IN ASCENDING ORDER USING SHUTTLE SORT ALGORITHAM:

```
 import java.util.Arrays;
public class Shuttlesort_24 {
public static void main(String[] args) {
                int array[] = {10,21,16,3,1,25,7};
                System.out.println(Arrays.toString(array));
                for(int i = 1; i<array.length;i++){
                        if(array[i]>array[i-1]){
                         int temp = array[i];
                           array[i] = array[i-1];
```

```
                    array[i-1] =  temp;
                for(int z = i-1;(z-1)>=0;z--){
                    if(array[z] > array[z-1]){
                    temp = array[z];
                      array[z] = array[z-1];
                     array[z-1] =  temp;
                            }
            else   {
                    break;
                }
            }
        }
}
System.out.println(Arrays.toString(array));
}
}
```

OUTPUT

[10, 21, 16, 3, 1, 25, 7]

[25, 21, 16, 10, 7, 3, 1]

25. SORT ARRAY AND MERGE ARRAY

```
import java.util.Arrays;

class ArrMergeSort {
        static int[] mergeArray(int[]array1, int[]array2)
        {
                int[] mergedArray = new int[array1.length+array2.length];
                int i=0,j=0,k=0;
                while(i<array1.length)
                {
                        mergedArray[k] = array1[i];
                        i++;
                        k++;
```

```
            }
            while(j<array2.length)
            {
                  mergedArray[k] = array2[j];
                  j++;
                  k++;
            }
            Arrays.sort(mergedArray);
            return mergedArray;
      }
      public static void main(String[] args) {

int[]array1 = new int[]{1,2,5,6,9,7};
int[]array2 = new int[]{3,4,12,10};
int[]mergedArray = mergeArray(array1,array2);
System.out.println("Array1 :"+Arrays.toString(array1));
System.out.println("Array2 :"+Arrays.toString(array2));
System.out.println("Merged Array :"+Arrays.toString(mergedArray));

      }

}
```

OUTPUT:

```
Array1 :[1, 2, 5, 6, 9, 7]
Array2 :[3, 4, 12, 10]
Merged Array :[1, 2, 3, 4, 5, 6, 7, 9, 10, 12]
```

26.OVERLOADING CONCEPT:

```
public class OverLoading{

      static void meth(int a){

      System.out.println(a);

      }

      static void meth(String s){

      System.out.println(s);

      }

      public static void main(String[] args) {

      meth(10);
```

```
        meth("Ajith");
        }
}
```

Output:

10

Ajith

27:INHERITANCE CONCEPT:

```
class A{
 int a;
  int getch(){
          return a;
  }
         }
 class B extends A{
 int b;
   int show(){
     return b;
      }
   }
 class inheritance{
public static void main(String args[]){
   A a=new A();
   B b=new B();
   b.a=1;
   b.b=2;
System.out.println(b.show());
System.out.println(b.getch());
```

Output:

b=2

a=1

28. write a program to implement overriding concept?

```
class A4{
void show() {
System.out.println("superclass method");
}
}
class B4 extends A4{
void show() {
System.out.println("subclass method");
}
}
public class Override {
public static void main(String[] args) {
B4 subOb = new B4();
subOb.show();
}
}
```

OUTPUT :

subclass method

29.ABSTRACT CONCEPT:

```
abstract class A{
abstract void callme();
}
class B extends A
{
```

```
public void callme()
{
System.out.println("dynamic used in abstract");
}
}
class abstractdemo1
{
public static void main(String args[])
{
B o=new B();
A a;
a=o;
a.callme();
}
}
```

OUTPUT

```
dynamic used in abstract
```

30:PROGRAM BASED ON INTERFACE CONCEPT:

```
Import java.io.*;
interface A{
final int a=24;
void display();
}
class B implements A{
public void display(){
System.out.println(“MAASMIND”);
}
public static void main(String args[]){
```

```
B b=new B();
b.display();
System.out.print(a);
}
}
```

OUTPUT:

```
MAASMIND
24
```

31:PROGRAM TO EXPLAIN CONCEPT OF DYNAMIC DISPATCH OR RUN TIME POLYMORPHISM:

```
class A7{
        void callme(){
                System.out.println("Inside A7's callme method");
                }
        }
  class B7 extends A7{
                void callme(){
                  System.out.println("Inside  B7's callme method");
                }
        }
        class DynamicMethodDispatch_31{
                public static void main(String args[]){
                  A7 a= new A7();
                  B7 b = new B7();
                  A7 r;
                  r =a;
                  r.callme();
                  r = b;
                  r.callme();
```

```
        }
  }
```

OUTPUT

```
 Inside A7's callme method
Inside  B7's callme method
```

32:IMPLEMENT PRODUCER CONSUMER DESIGN PATTERN IN JAVA USING WAIT,NOTIFY AND NOTIFY ALL

```
class Q{
        int n;
        boolean valueSet=false;
        synchronized int get(){
                while(!valueSet){
                        try{
                                wait();

        }catch(InterruptedException e){
                System.out.println("Interrupted Exception caught");

                        }
                }
                System.out.println("Got:"+n);
                valueSet=false;
                notify();
                return n;
        }
        synchronized void put(int n){
                while(valueSet){
                        try{
                                wait();

        }catch(InterruptedException e){
        System.out.println("Interrupted exception caught");

                        }
                }
                this.n=n;
                valueSet=true;
                System.out.println("Put:"+n);
                notify();
```

```
	}
}
class Producer implements Runnable{
	Q q;
	Producer(Q q){
		this.q=q;
		new Thread(this,"Producer").start();
	}
	public void run(){
		int i=0;
		while(true){
			q.put(i++);
		}
	}

}
class Consumer implements Runnable{
	Q q;
	Consumer(Q q){
		this.q=q;
		new Thread(this,"Consumer").start();

	}
	public void run(){
		while(true){
			q.get();

		}
	}
}

public class ProducerConsumer {

	public static void main(String[] args) {
		// TODO Auto-generated method stub
Q q=new Q();
new Producer(q);
new Consumer(q);
System.out.println("Press Control-C to stop.");
	}

}
```

OUTPUT:

Put:90367

Got:90367
Put:90368
Got:90368
Put:90369

33: INPUT: CheckEvenOrOdd a[]={1,4,6,9,6,1}

```
public class CheckEvenOrOdd {
	public static void main(String[] args) {
		int a[]={1,4,6,9,6,1};
		for(int i=0;i<a.length;i++){
			if(i<3) System.out.println("Not Found");
			else System.out.println("Found");
		}
	}
}
```

Output:

Not Found

Not Found

Not Found

Found

Found

Found

34:INPUT:

int a1[]={1,2,3,4}

Int a2[]={1,2,3,4}

OUTPUT:true

```
import java.util.*;
class CompareArray{
    public static void main(String args[]){
        int a1[]={1,2,3,4};
```

```
            int a2[]={1,2,3,4};
 if(Arrays.equals(a1,a2)){
 System.out.println("true");
else{
  System.out.println("false");
      }
   }
}
```

Output: True

35.find the odd one out from the array.

Input : String str [] = {"black","white", "black", "black"}

Output : white

```
public class OddOneOutF {
public static void main(String[] args) {
String str[] = {"black","white","black","black"};
            for(String x : str) {
            (if x.equals("black")){
            System.out.print("x");
            }
else {
System.out.print("x");
                  }
        }
    }
}
```

output : white

36.Number of occurences of the array elements

```
public class occurence {
```

```
	public static void main(String[] args) {
int a[]=new int[]{1,2,3,1,1,3};
int b[]=new int[a.length];
int v=-1;
for(int i=0;i<a.length;i++){
	int count=1;
	for(int j=i+1;j<a.length;j++){
		if(a[i]==a[j]){
		count++;
		b[j]=v;
		}
	}
	if(b[i]!=v)
	b[i]=count;
}
for(int i=0;i<b.length;i++){
	if(b[i]!=v)
	System.out.println(a[i]+"-"+b[i]+"times");
}

	}
}
```

output

1-3times

2-1times

3-2times

37:ROTATE THE GIVEN INTEGER ARRAY:

INPUT:{1,2,3,4,,5,6,7}

OUTPUT1={4,5,6,7,1,2,3}

OUTPIT2={4,3,2,1,7,6,5}

```
public class ROTATE1  {
        public static void main(String args[]){
                int a[]={1,2,3,4,5,6,7};
int i,k=6;
for(i=3;i<=k;i++){
System.out.print(" "+a[i]);
if(i==6){
   i=-1;
   k=2;
    }
}
System.out.println(" ");
int i1,k1=0;
for(i1=3;i1>=k1;i1--){
System.out.print(" "+a[i1]);
if(i1==0){
   i1=7;
   k1=4;
        }
     }
    }
}
```

OUTPUT

4 5 6 7 1 2 3

4 3 2 1 7 6 5

38:PROGRAM FOR ANAGRAM:

```
import java.util.Arrays;
public class Anagramexample_38{
    public static void main(String[] args) {
                String s1 = "listen";
                String s2 = "silent";
                boolean c = isAnagram(s1,s2);
                if(c)
                        System.out.println(s1+" and "+s2+" are anagram");
                else
                        System.out.println(s1+" and "+s2+" are anagram");
}
public static boolean isAnagram(String s1,String s2){
                char c1[] = s1.toCharArray();
                char c2[] = s2.toCharArray();
                Arrays.sort(c1);
                Arrays.sort(c2);
                if(Arrays.toString(c1).equals(Arrays.toString(c2)))
                return true;
                return false;
            }
}
```

OUTPUT

listen and silent are anagram

39:FIND OUT THE PAIRS IN SUM EQUAL TO 20:

INPUT: {5,8,3,4,12,17,15,16}

OUTPUT: {5,15},{8,12},{3,17},{4,16}

```
public class ArrayPairsSum {
```

```
static void findThePairs(int a[],int num){
	System.out.println("Pairs of element whose input number"+num+"are :");
	for(int i=0;i<a.length;i++){
		for(int j=i+1;j<a.length;j++){
			if(a[i]+a[j] == num){
				System.out.println(a[i]+ "," +a[j]+ "="+num);
			}
		}
	}
}

	public static void main(String[] args) {
		findThePairs(new int[]{5,8,3,4,12,17,15,16},20);
	}

	}
```

OUTPUT:

```
Pairs of element whose input number20are :
5,15=20
8,12=20
3,17=20
4,16=20
```

40.Split The Array into Array.

```
public class SplitTwoArray{

	public static void main(String[] args) {

		int a[]={2,4,6,8,10,12,14,16};

		int b[]=new int[4];

		int c[]=new int[4];

		int j=0;

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

			if(i<4)b[i]=a[i];

			else{

				c[j]=a[i];

				j++;

			}

		}
```

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

Output:

2 4 6 8

10 12 14 16

41:REVERSE THE GIVEN ARRAY:

```
class reversearray{
public static void main(String args[]){
        int a[] = {1,2,3,4,5,6};
          for ( int x:a)
       System.out.println(x);
        System.out.println();
        int i=0;
        int j=a.length-1;
       while(i<j)
           {
              int temp = a[i];
               a[i] = a[j];
               a[j] = temp;
                  i=i+1;
```

```
                    j=j-1;
}
for(int x:a)
System.out.println(x);
}
}
```

Output:

6 5 4 3 2 1

42.Find the common elements in two arrays.

input : int a [] = {5,10,15,20,25,30}

output : int b[] {10,20,30,40,50}

```
public class CommonElements {
public static void main(String[] args) {
int a[] = {5,10,15,20,25};
int b[] = {10,20,30,40,50};
for(int i=0; i<a.length; i++) {
        for(int j=0; j<b.length; j++) {
            if(a[i] == b[j]) {
         System.out.println(b[j]);
                        }
                    }
                }
            }
}
```

Output : 10,20

43:REVERSE THE NUMBER USING RECURSION:

```
public class Recursiondemo {
	public static void main(String[] args) {
		int num=1234;
		System.out.print("reverse of the number is=");
		reversemeth(num);
  }
public static void reversemeth(int n){
	if(n<10){
		System.out.print(n);
		return;
		}
	else{
	System.out.print(n%10);
	reversemeth(n/10);
	}
	}
	}
```

Output

reverse of the number is=4321

44: FIND THE SUM OF GIVEN SERIES:

1/1!+1/2!+1/3!+1/4!+…….

```
public class factorial2 {
public static void main(String args[]){
int val=4;
double sum=0,f=1;
for(int i=1;i<=val;i++){
f=f*i;
sum=sum+(1/f);
}
```

```
System.out.print(sum);
}
}
```

output:1.7083333333333335

45:DECIMAL TO HEXADECIMAL USING STACK:

```
import java.util.Stack;
public class Dec_2_Hex_45 {
public static void main(String[] args) {
int num = 2567;
int rem = 0;
Stack<String>stk = new Stack<>();
String str = " ";
char hex[] = {'0','1','2','3','4','5','6','7','8','9','A','B','C','D','E','F'};
while(num>0){
rem = num%16;
str = hex[rem] + str;
        System.out.println("pushed element:");
        System.out.println(stk.push(str));
        num = num/16;
}System.out.println();
while(!(stk.isEmpty())){
 System.out. println("poped element:");
System.out.println(stk.pop());
}
}
```

} **OUTPUT**

```
  pushed element:7
 pushed element:  07
  pushed element:A07
```

poped element:A07

poped element:07

poped element:7

46.FIND LARGEST AND SMALLEST NUMBER IN ARRAY WITHOUT SUING SORTING METHOD:

```
public class LargestSmallest {

	public static void main(String[] args) {

		int a[]={12,3,1,56,23,95};
		int smallest=a[0];//12
		int largest=a[0];//12
		for(int i=1;i<a.length;i++){
			if(a[i]<smallest)
				smallest=a[i];//1
			 if(a[i]>largest)
				largest=a[i];//95

		}
		System.out.println("Largest numberis :"+largest);
		System.out.println("Smallest number is :"+smallest);


	}

}
```

47.Find out the Maximum Occuring Character in String.

```
import java.util.*;

public class CheckEvenOrOdd {

public static void main(String[] args) {

String s="Simple String";

int max=0;

char c[]=s.toCharArray();

String s1="";
```

```
LinkedHashMap<Character,Integer>m=new LinkedHashMap<Character,Integer>();
for(char c1:c){
if(!m.containsKey(c1)){
        m.put(c1,1);
}else{
m.put(c1, m.get(c1)+1);
max++;
}
}
for(Map.Entry<Character,Integer> m1:m.entrySet()){
if(max==m1.getValue()){
s1=s1+" "+m1.getKey();
}
}
System.out.print(s1);
}
}
```

Output :S i

48:FIND THE FACTORIAL USING RECURSION:

```
class factre{
static int getch(int n){
if(n==0){
return 1;
```

```
}else{

return n*getch(n-1);

}

}

public static void main(String args[]){

int n = 5;

System.out.println(getch(n));

    }

}
```

Output:5*4*3*2*1=120

49:PROGRAM TO EVALUATE THE GIVEN SERIES:

1/2+2/3+3/4+……+N

```
public class factor {

public static void main(String args[]){

double sum=0;

for(int i=1;i<=3;i++){

sum=sum+(i*1.0)/(i+1.0);
}

System.out.print(sum);

}
}
```

OUTPUT: 1.916666666666666

50:FIND THE YEAR IS LEAP YEAR OR NOT:

```
public class leapyear {
```

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

int year=2020;

if(((year%4==0)&&(year%100!=0))||(year%400==0))

System.out.println("2020 is leap year");

else

System.out.println("not a leap year")

        }

    }

}
```

Output:2020 is leap year

51:CONVERT DECIMAL TO BINARY USING STACK:

```
import java.util.*;

class A{

public static void main(String args[]){

int b=0;

int n=7;

int r;

Stack stk=new Stack();

while(n!=0)

{

r=n%2;

stk.push(r);

n=n/2;
```

```
}
System.out.print(" binary:");
while(!(stk.isEmpty()))
{
System.out.print(stk.pop());
}
        }
}
```

output: binary:111

52:FIND QUOTIENT AND REMAINDER WITHOUT USING MOD OR DIV:

```
public class Withoutdivmod_52 {

public static void main(String[] args) {

                int dividend = 89;
                 int divisor = 4;
                 int quotient = 0;
                 while(dividend >= divisor){
                         dividend = dividend - divisor;
                         quotient ++;
                         }
                 System.out.println("quotient is "+quotient);
                 System.out.println("reminder is "+dividend);
        }
```

```
}
```

OUTPUT

quotient is 22

reminder is 1

53. ANSWER IS PROGRAM NO:46

54.Find Union And InterSection from Two Arrays.

```
import java.util.*;
public class Check {
	public static void main(String[] args) {
		int a[]={1,2,3,4,5,6};
		int b[]={4,5,6,7,8,9};
		int c[]=new int[a.length+a.length];
		System.arraycopy(a,0,c,0,a.length);
		System.arraycopy(b,0,c,a.length,b.length);
		//Arrays.sort(c);
		System.out.print("Union : ");
		for(int i=0;i<c.length;i++){
			System.out.print(c[i]);
		}
		System.out.println("");
		System.out.print("InterSection: ");
```

```
                for(int i=0;i<a.length;i++){
                        for(int j=0;j<b.length;j++){
                                if(a[i]==b[j]) System.out.print(a[i]);
                            }
                    }
            }
}
```

Output:

Union : 123456456789

InterSection: 456

55:PROGRAM TO GENERATE HAILSTONE NYMBER STARTS FROM 7

Output:7,22,11,34,17,52,26,13,40,20,10,5,16,8,4,2,1

```
import java.util.*;
class hailstone{
  static void meth(int n){
       if(n <= 0)
          System.out.println("Invalid Input!");
        if(n == 1)
          return ;
        if(n % 2 != 0){
          int a = n * 3 + 1;
          System.out.print(a + " ");
          meth(a);
       }
```

```
        else{
            int a = n / 2;
            System.out.print(a + " ");
            meth(a);
                                }
}
public static void main(String args[]){
Scanner s = new Scanner(System.in);
System.out.println("enter the start number");
int n = s.nextInt();
meth(n);
            }
                                }
```

Output:7,22,11,34,17,52,26,13,40,20,10,5,16,8,4,2,1

56.write a program for harshad number or niven number.

Logic:Harshad number or niven number is a number which is divisible by the sum of digits .

for example ,21 is a harshad number because it is divisible by the sum of its digits . 21->sum of digits -> 2+1=3 and 21 is divisible by 3-> 21/3 =7.

```
class Harshhad {
public static int sumOfDigits(int num) {
 if (num==0) {
 return 0;
}
```

```
 return num%10+sumOfDigits(num/10);
}
public static void main(String[] args) {
  int sum = sumOfDigits(21);
  System.out.print(sum);

 if(21%sum==0) {
System.out.print("Niven no");
            }
               else {
             System.out.print("Not a Niven no");
               }
          }
}
```

Output: 3 niven no

57: .print L triangle

```
public class printbelowtriangle {
public static void main(String[] args) {
                    for(int i=0;i<=4;i++){
                              for(int j=0;j<=i;j++){
                                        System.out.print("*"+" ");
                              }
                              System.out.println();
                    }
```

}**Output**

```
*
* *
* * *
* * * *
* * * * *
```

58:Triangle:

```
public class triangle {
	public static void main(String args[]){
		for(int i=0;i<4;i++){
			for(int k=4;k>i;k--){
				System.out.print(" ");
				}
		for(int j=0;j<2*i+1;j++){
			System.out.print("*");
		}
		System.out.println();
		}}}
```

OUTPUT:

```
   *
  ***
 *****
*******
```

59:TRIANGLE

```
public class trianglr {

	public static void main(String[] args) {
		int n=5;
		for(int i=1;i<=n;i++){
			for(int j=n-1;j>=i;j--){
				System.out.print(" ");
			}
			for(int k=1;k<=i;k++){
				System.out.print("*");
			}
			System.out.println();
		}

	}

}
```

OUTPUT:

```
    *
   **
  ***
 ****
*****
```

60:PRINT BELOW MATRIX:

```
. public class TriangleNumbers {

	public static void main(String[] args) {

		int n = 4;
		for(int i=1;i<=n;i++){
			for(int j=n-1;j>=i;j--){
				System.out.print(" ");

			}
			for(int k=i-1;k>=-(i-1);k--){
				System.out.print(i-Math.abs(k));
			}
			System.out.println();
```

```
        }
    }
}
```

OUTPUT:

```
   1
  121
 12321
1234321
```

61.Print the L-Pattern.

```
public class Lpattern{
    public static void main(String[] args) {
        int a=9;
        for(int i=0;i<4;i++){
            for(int j=0;j<=i;j++){
                System.out.print(a);
                a--;
            }
            System.out.println();
        }
    }
}
```

Output:

```
9
87
```

654

3210

62: PRINT BELOW MATRIX:

```
class pattern62{
public static void main(String args[]){
        int a = 0;
       for(int i=0;i<5;i++){
      for(int j=0;j<=i;j++){
              if(a==4){
             a=0;
             }
  System.out.print(a);
  a++;
           }
System.out.println();
}
}
}
```

Output:

0

1 2

3 0 1

2 3 0 1

2 3 0 1 2

63. Print the * in diamond program

```
public class Diamond {
 public static void main(String args[])
int n=5;
 for(int r=1; r<=(n+1)/2; r++ ) {
      for(int c=1; c<=(n+1)/2-r; c++) {
            System.out.print("  ");
            }
for(int i=1; i<=r; i++) {
 System.out.println("*  ");
 }
System.out.println();
}
 for(int r=(n-1)/2; r>0; r--) {
for(int c=1; c<=(n-1)/2-r; c++) {
System.out.print("  ");
}
for(int i=1; i<r; i++) {
System.out.print("  *");
}
System.out.println();
 }
 }
```

```
}
```

OUTPUT:

```
        *
   *         *
*       *       *
  *           *
          *
```

64.ALLAHABAD

```
public class ALLAHABAD1 {
  public static void main(String[] args){
  char c[]={'A','L','L','A','H','A','B','A','D'};
 int z=0;
 for(int i=0;i<3;i++){
            for(int j=2;j>i;j--){
                        System.out.print(" ");
                        }
            for(int k=0;k<=i;k++){
                        System.out.print(c[z]+" ");
                        z++;
                        }
            System.out.println(" ");
            }
           for(int i=2;i>0;i--){
                        for(int j=2;j>=i;j--){
```

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

OUTPUT

```
   A
 L   L
A H  A
 B   A
   D
```

65:PRINT THE PATTERN:

```
public class triangle {
        public static void main(String args[]){
            int n=1;
            for(int i=0;i<4;i++){
                for(int k=5;k>i;k--){
                    System.out.print(" ");
                    }
```

```
                for(int j=0;j<2*i+1;j++){
                        if(j==i||j==0){
                                if(j==i)
                                System.out.print(" "+(n-i));
                        else  System.out.print(" "+(n+i));
                }
                else{
                        System.out.print(" "+n);
                }
                n++;
                }
                System.out.println();
                }
            }
}
```

OUTPUT:

```
      I
   3  2    4
7  6   5   8   9
```

66: Printpattern

```
public class PrintpatternChar_66 {
        public static void main(String args[]){
           int k=97;
         char c=(char)k;
```

```
            for(int i = 0;i<3;i++){
             for(int j=1;j<=5;j++){
                        if((j%2) != 0 ){
                         System.out.print(c);
                          c++;
                         System.out.print(c);
            c++;
               }
            else{
             c++;
             System.out.print(" ");
             }
             }
          System.out.println();
          c++;
        }
      }
}
```

OUTPUT:

ab de gh

jk mn pq

st vw yz

67:PRINT THE PATTERN:

```
public class StartTriangle {
```

```
public static void main(String[] args) {
		int i,j,k,s=2,c=2,p=1;
		for(i=1;i<=4;i++){
			for(j=4;j>i;j--){
				System.out.print("  ");


			}
			for(k=1;k<=i*2-1;k++){
			if(k==i){
				System.out.print(""+p+" ");
				p=0;
				c--;
			}else{
				System.out.print(""+c+" ");
			}
			p=p+s*3;
			s++;
			c++;
			}
			System.out.println();
			}
		 }


	}
```

OUTPUT:

```
       1
    2 15 3
  4 5 66 6 7
8 9 10 189 11 12 13
```

68:.Print the Character pyramid.

```
public class PrintPyramid {
        public static void main(String[] args) {
                char a='A';
                int n=4;
                int sum=0;
                for(int i=0;i<5;i++){
                        for(int j=n;j>0;j--){
                                System.out.print(" ");
                        }
                        for(int k=0;k<=sum;k++){
                                System.out.print(a);
                        }
                        n--;
                        a++;
                        sum+=2;
                        System.out.println();
```

```
                }
        }

}
```

Output:

```
         A
       B B B
     C C C C  C
   D D D D D D D
E  E E  E  E E E E E
```

69: PRINT SPIRAL MATRIX:

```
import java.util.*;
class Circular_Matrix{
    public static void main(String args[]){
         Scanner sc = new Scanner(System.in);
        System.out.print("Enter the number of elements : ");
        int n = sc.nextInt();
         int A[][] = new int[n][n];
        int k=1, c1=0, c2=3, r1=0, r2=3;
        while(k<=n*n)
           {
              for(int i=c1;i<=c2;i++)
              {
                 A[c1][i]=k++;
```

```
        }
        for(int j=r1+1;j<=r2;j++)
        {
            A[j][r2]=k++;
          }
         for(int i=c2-1;i>=c1;i--)
        {
            A[c2][i]=k++;
        }  for(int j=r2-1;j>=r1+1;j--)
        {
            A[j][r1]=k++;
        }
      c1++;
      c2--;
      r1++;
      r2--;
     }

/* Printing the Circular matrix */
System.out.println("The Circular Matrix is:");
for(int i=0;i<n;i++)
    {
        for(int j=0;j<n;j++)
            {
```

```
                    System.out.print(A[i][j]+ "\t");
                }
             System.out.println();
            }
        }
}
```

Output:

```
1  2  3  4
12 13 14 5
11 16 15 6
10  9  8 7
```

70.Write a program for matrix multiplication.

```
public class MatrixMultiplication {
   public static void main(String args[]) {
   int a[] [] = {{1,1,1},{2,2,2},{3,3,3}};
   int b[] [] = {{1,1,1},{2,2,2},{3,3,3}};
int c[][] = new int[3][3];
 for(int i=0; i<3; i++) {
for(int j=0; j<3; j++) {
 c[i][j]=0;
 for(int k=0; k<3; k++) {
   c[i][j]+=a[i][k]*b[k][j];
                    }
```

```
 System.out.print(c[i][j]+" ");
            }
 System.out.println();
        }
    }
}
```

output:

6 6 6

12 12 12

18 18 18

71.Symmetric Matrix

```
public class SymmetricMatric {
   public static void main(String[] args){
                    int a[][]={{1,2,3},{4,5,6},{7,8,9}};
                    System.out.println("original matrix:");
                    for(int i=0;i<3;i++){
                            for(int j=0;j<3;j++){
                                    System.out.print(a[i][j]+" ");
                            }
                            System.out.println();
                    }
                   System.out.println("original matrix converted into symmetric matrix:");
                    for(int i=0;i<3;i++){
                            for(int j=0;j<3;j++){
```

```
                    System.out.print(a[j][i]+" ");
                }
                System.out.println();
            }
        }
}
```

OUTPUT

original matrix:

1 2 3

4 5 6

7 8 9

original matrix converted into symmetric matrix:

1 4 7

2 5 8

3 6 9

72:INPUT:MA@$AS23MINDI OUTPUT:MAASMIND@$6

```
import java.util.*;
public class Arrangestring {
public static void main(String[] args) {
String str="MA@$AS11MIND1";
   int temp=0,sum=0,res=0;
        StringBuffer a=new StringBuffer();
        StringBuffer num=new StringBuffer();
        StringBuffer spl=new StringBuffer();
```

```
 for(int i=0;i<str.length();i++)
 {
 if(Character.isDigit(str.charAt(i)))
            {
num.append(str.charAt(i));
                    String s1=new String(num);
                    int v=Integer.parseInt(s1);
                if(v!=0){
                        temp=v%10;
                    sum=sum+temp;
                    v=v/10;
                    }
                }
            else if(Character.isLetter(str.charAt(i)))
            a.append(str.charAt(i));
            else
                    spl.append(str.charAt(i));
                   }
 System.out.print(" "+a+spl+sum);
        }
    }
```

output: MAASMIND@$ 6

73:REVERSE THE WORDS OF STATEMENT:

```
public class Reversewords_73 {
```

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

                String s1 = "How are you?";

                String s[] = s1.split(" ");

                String ans = " ";

                System.out.println(s1);

                for(int i = s.length-1;i>=0;i--){

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

                    }

                }

            }
```

OUTPUT:

How are you?

you? are How

74:I/P: Ka2s^%67$ OUTPUT:15$%^

```
public class RemoveSumReverse {

        public static void main(String[] args) {
                String str="Ka2s^%67$";
                int temp=0,sum=0;
                StringBuffer a=new StringBuffer();
                StringBuffer num=new StringBuffer();
                StringBuffer spll=new StringBuffer();
                for(int i=0;i<str.length();i++){
                        if(Character.isDigit(str.charAt(i))){
                                num.append(str.charAt(i));
                                String s1=new String(num);
                                int v=Integer.parseInt(s1);
                                if(v!=0){
                                        temp=v%10;
                                        sum=sum+temp;
                                        v=v/10;
                                }
```

```
                }
                        else if(Character.isLetter(str.charAt(i))){
                }
                        else
                                spll.append(str.charAt(i));

        }
System.out.println(" "+sum+spll.reverse());
        }
        }
```

OUTPUT:

15$%^

75: Replace with $ for continious two occurance of character in a string.

```
public class Check {

        public static void main(String[] args) {

                String s="Annual Offer";

                String c="";

                int l=s.length();

                for(int i=0;i<l-1;i++){

                        if(s.charAt(i)==s.charAt(i+1)){

                                c=c+"$";

                                i++;

                        }

                        else c=c+s.charAt(i);

                }

            if(s.charAt(l-2)==s.charAt(l-1)) c=c+"$";

                else c=c+s.charAt(l-1);

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

```
        }
}
```

Output: A$ual O$er

76:REVERSE THE WORDS IN ODD POSITION:

```
import java.util.*;
 class ReverseWords{
      public static void main(String args[]){
          String a="object oriented high level programming language";
           String a1[]=a.split(" ");
          for(int i=0;i<a1.length;i++){
                  if(i%2==0){
                    System.out.println(new StringBuffer(a1[i]).reverse());
           }
          else{
                    System.out.println(a1[i]);
                    }
               }
          }
}
```

Output: input:object oriented high level programming language

output:tcejbo oriented hgih level gnimmargorp language

77:

78: Separate String without using Split method

```
import java.util.StringTokenizer;
```

```
public class StringwithoutsplitMethod {
	public static void main(String[] args) {
			StringTokenizer str=new StringTokenizer("Red;Green;Blue;Black;White",";");
			while(str.hasMoreTokens()){
			System.out.println(str.nextToken());
			}
		}
	}
```

OUTPUT

```
Red
Green
Blue
Black
White
```

79: CONVERT STRING TO INTEGER AND INTEGER TO STRING:

```
public class conversion {
public static void main(String[] args) {
	conversion a=new conversion();
		a.intToString();
		a.stringToInt();
		    }
	void intToString(){
		int x=5;
```

```
        String s1=Integer.toString(x);
        System.out.println(s1);
        String s2=String.valueOf(x);
        System.out.println(s2);
    }
    void stringToInt(){
        String s="3";
        int num=Integer.parseInt(s);
        System.out.println(num);
        int num1=Integer.valueOf(num);
        System.out.println(num1);
    }

}
```

Output:

5

5

3

3

80:PROGRAM THAT PRINT THE NUMBER FROM 1-100(FIZZ –BUZZ)

```
public class fizzbuzz_80 {
    public static void main(String[] args) {
        int n = 100;
        for(int i = 1; i<n ; i++){
```

```
                        if(i%5 == 0 & i%3 ==0 )
                                System.out.print("FizzBuzz" + " ");
                        else if(i%5 == 0)
                                System.out.print("Buzz" + " ");
                        else if(i%3 == 0)
                           System.out.print("Fizz" + " ");
                        else
                                System.out.print( i + " ");
                }
        }
}
```

OUTPUT

1 2 Fizz 4 Buzz Fizz 7 8 Fizz Buzz 11 Fizz 13 14 FizzBuzz 16 17 Fizz 19 Buzz

Fizz 22 23 Fizz Buzz 26 Fizz 28 29 FizzBuzz 31 32 Fizz 34

Buzz Fizz 37 38 Fizz Buzz 41 Fizz 43 44 FizzBuzz 46 47

Fizz 49 Buzz Fizz 52 53 Fizz Buzz 56 Fizz 58 59 FizzBuzz 61 62 Fizz 64

Buzz Fizz 67 68 Fizz Buzz 71 Fizz 73 74 FizzBuzz 76 77 Fizz 79 Buzz

Fizz 82 83 Fizz Buzz 86 Fizz 88 89 FizzBuzz 91 92 Fizz 94 Buzz Fizz 97 98 Fizz

81:COUNT THE NUMBER OF OCCURENCES OF EACH CHARACTER IN WORD:

```
import java.util.Scanner;
public class NumberOfOccurences {
        public static void main(String[] args) {
                System.out.println("Enter a word");
```

```
Scanner sc=new Scanner(System.in);
String s=sc.next();
int c=0;
for(char i='A';i<='Z';i++){
        c=0;
        for(int j=0;j<s.length();j++){
                if(i==s.charAt(j)){
                        c++;

                }
        }
        if(c>0){
                System.out.println(i+"-"+c+"times");
              }
      }
        }

}
```

OUTPUT:

```
Enter a word
COCOCOLA
A-1times
C-3times
L-1times
O-3times
```

82.Count the number of occurence of each word in a Sentence.

```
public class CountWord{

        public static void main(String[] args) {

                 String s1[]= {"Learn", "something", "about", "everything", "and", "everything",
"about", "something"};

                //String s2[]=s.split(" ");

                int n=1;

                for(int i=0;i<s1.length;i++){

                        if(s1[i]=="*") break;
```

```
                for(int j=i+1;j<s1.length;j++){
                        if(s1[i]==s1[j]){
                                n+=1;
                                s1[j]="*";
                        }
                }
                System.out.println(s1[i]+" "+n+"-Times ");
                n=1;
            }
        }

}
```

OUTPUT:

Learn-1-Times

Something -2-times

About-2 times

Everything -2 times

And- 1 times

83: FIND THE VALUE OF STRING

INPUT:JAVA OUTPUT: 34

```
class ValueofString{
        public static void main(String args[]){
            String a="ABCDEFGHIJKLMNOPQRSTUVWXYZ";
                String a1="JAVA";
```

```
                int b=0;
                for(int i=0;i<a1.length();i++){
                 b=b+(a.indexOf(a1.charAt(i))+1);
          }
System.out.println(b);
 }
}
```

Output:

input:JAVA

Output:34

84: Write a program for singleton class.

```
import java.util.*;
public class Singleton {
private static Singleton myObj1;
 static {
 myObj1=new Singleton();
 }
private Singleton() {
}
public static Singleton getInstance() {
return myObj1;
}
public void testMe() {
```

```
System.out.println("hey");
}
public void testMe2() {
System.out.println("hiii");
}
public static void main(String[] args) {
Singleton ms = getInstance();
ms.testMe();
ms.testMe2();
    }
}
```

output:

```
hey
hii
```

85:.Unique words

```
import java.util.LinkedHashMap;
import java.util.Map;
public class UNIQUEWORDS {
                    public static void main(String[] args) {
    String str="guitar is instrument and piano is instrument";
            String[] strArray=str.split("\\s+");
            Map<String, String> hMap=new LinkedHashMap<String,String>();
            for(int i=0;i<strArray.length;i++){
```

```
                if(!hMap.containsKey(strArray[i])){

                        hMap.put(strArray[i],"Unique");

                }

              }

        System.out.println(hMap);

  }



        }
```

OUTPUT

{guitar=Unique, is=Unique, instrument=Unique, and=Unique, piano=Unique}

86:FIND AGE BY GIVING DATE OF BIRTH:

```
import java.util.*;
import java. text.SimpleDateFormat;
import java .util.Calendar;
        public class factor{
                public static void main(String args[])throws Exception{
                        System.out.print("enter date d birth");
                        Scanner s=new Scanner(System.in);
                        String input=s.nextLine();
                        s.close();
                        SimpleDateFormat sdf=new SimpleDateFormat("yyyy-MM-DD");
                        Calendar dob=Calendar.getInstance();
                        dob.setTime(sdf.parse(input));
                        System.out.println("age is:"+  getAge(dob));
                        }

                public static int getAge(Calendar dob) throws Exception{

                        Calendar today=Calendar.getInstance();
                        int curyear=today.get(Calendar.YEAR);
                        int dobYear=dob.get(Calendar.YEAR);
                        int age=curyear-dobYear;
                return age;
                }
        }
```

OUTPUT:
enter date d birth 1998-11-11
age is:22

87.FIND OUT THE NUMBER OF DAYS INBETWEEN TWO GIVEN DATES:

```
import java.util.Date;

import java.text.ParseException;

import java.text.SimpleDateFormat;

 public class NumberOfDays_87 {

        public static void main(String[] args) {

                SimpleDateFormat For = new SimpleDateFormat("dd MM yyyy");

                try{

                   Date dateBefore = For.parse("31 01 2021");

                   Date dateAfter  = For.parse("01 02 2021");

                   long difference = dateAfter.getTime()-dateBefore.getTime();

                   float daysBetween = (difference/(1000*60*60*24));

                   System.out.println("Number of days ="+ daysBetween);

                }

                catch(Exception e){

                        e.printStackTrace();

                }

         }

}
```

OUTPUT

```
Number of days =1.0
```

88.PROGRAM TO INSERT AND RETERIVE DATA FROM LIST:

```
import java.util.ArrayList;
import java.util.Iterator;
import java.util.List;
```

```
public class list {
	public static void main(String args[]){
		List <String>l=new ArrayList<String>();
		l.add("SANGEETHA");
		l.add("RAMYA");
		l.add("kEERTHANA");
		l.add("MONISHA");
		Iterator itr=l.iterator();
		while(itr.hasNext()){
			System.out.println(itr.next());
		}

	}

}
```

OUTPUT:

SANGEETHA
RAMYA
kEERTHANA
MONISHA

89..Insert and retrive data from Set.

```
import java.util.*;

class LinkedHashSet2{

	public static void main(String args[]){

		LinkedHashSet<String> al=new LinkedHashSet<String>();

		al.add("Ravi");

		al.add("Vijay");

		al.add("Ravi");

		al.add("Ajay");

		Iterator<String> itr=al.iterator();

		while(itr.hasNext()){

			System.out.println(itr.next());

		}
```

```
        }
}
```

Output:

Ravi

Vijay

Ajay

90. WRITE A PROGRAM TO RETRIEVE DATA FROM MAP

```
import java.util.*;
class MAP{
public static void main(String args[]){
Map<String,Character> m =new HashMap<String,Character>();
m.put("shameem",'s');
m.put("Ajith",'a');
m.put("Rafiq",'r');
m.put("Kaali",'k');
char a=m.get("shameem");
char b=m.get("Ajith");
System.out.println(a);
System.out.println(b);
          }
}
```

Output: s a

91.REMOVE DUPLICATE FROM ARRAYLIST USING COLLECTION:

```
import java.util.*;

public class arra{

public static <T>ArrayList<T>removeDuplicates(ArrayList<T>list){

ArrayList<T>newList = ArrayList<T>();

for(T element:list)

if(!newList.contains(element)){

 newList.add(element);

        }

   }

return newList;

}

public static void main(String args[]){

ArrayList<Integer> list= new ArrayList< >(Arrays.asList(1,10,1,2,3,3,4,5,6,7,8,8));

System.out.println(list);

ArrayList<Integer> newList =remove Duplicates(list);

System.out.println("Arraylist with duplicates removed:" + newList);

}
```

Output:

with duplicates : 1,10,2,3,3,4,5,6,7,8,8

removed duplicates : 1,10,2,3,4,5,6,7,8

92.Same AS NO-16

93.JDBC PROGRAM TO INSERT AND SELECT A RECORD FROM EMPLOYEE TABLE:

```
import java.sql.*;
```

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

Connection con=null;
try{
Class.forName("oracle.jdbc.driver.OracleDriver");
}
catch(Exception e)
{
System.out.println("fail to load");
}
try{

con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:XE","moni","moni");
        PreparedStatement st=con.prepareStatement("insert into
emp(empid,empname,empno)values(?,?,?)");
st.setInt(1,4);
st.setString(2,"SIVA");
st.setInt(3,3335555);
st.executeUpdate();
System.out.println("one record inserted");


}

catch(SQLException e){
System.out.println(e);
}
}
}
```

OUTPUT: ONE RECORD INSERTED

SQL DETAILS:

```
SQL> create table emp(empid number(20),empname varchar2(20),empno number(20));

Table created.

SQL> insert into emp values(1,'Moni',777777777);

1 row created.

SQL> insert into emp values(2,'Ramya',999999999);

1 row created.

SQL> insert into emp values(3,'Keerthana',876666222);

1 row created.

SQL> comit;
```

SQL> select * from emp;

EMPID	EMPNAME	EMPNO
1	Moni	777777777
2	Ramya	999999999
3	Keerthana	876666222
4	SIVA	984044444

94.

95.PATTERN PROGRAM

```
public class Pattern95 {
public static void main(String[] args) {
                int n=1;
                int l=4;
                for(int i=0;i<4;i++){
                        for(int j=1;j<=n;j++){
                                System.out.print("*");
                        }
                        System.out.println("");
                        for(int k=1;k<=l;k++){
                                if(l==16) break;
                                System.out.print("*");
                        }
                        n=n+2;
```

```
                    l=l+4;
                    System.out.println("");
                }
        }
    }
```

OUTPUT:

```
*
****
***
********
*****
**********
*******
```

96. Print the Diagonal Numbers in Matrix.

```
class PrintDiagonal{
   static void printPrincipalDiagonal(int mat[][], int n){
      System.out.print("Principal Diagonal: ");
          for (int i = 0; i < n; i++) {
         for (int j = 0; j < n; j++) {
            if (i == j) {
               System.out.print(mat[i][j] + ", ");
            }
         }
      }
```

```
        System.out.println("");
    }
    static void printSecondaryDiagonal(int mat[][], int n) {
        System.out.print("Secondary Diagonal: ");
             for (int i = 0; i < n; i++) {
            for (int j = 0; j < n; j++) {
                if ((i + j) == (n - 1)) {
                    System.out.print(mat[i][j] + ", ");
                }
            }
        }
        System.out.println("");
    }
    public static void main(String args[]) {
        int n = 3;
        int a[][] = { { 1, 2, 3},  { 4, 5, 6},   { 7, 8, 9} };
         printPrincipalDiagonal(a, n);
        printSecondaryDiagonal(a, n);
    }
}
```

Output:

Principal Diagonal: 1, 5, 9,

Secondary Diagonal: 3, 5, 7,

97. ConsiderTwoArrays:

```
 class ConsiderTwoArrays{
public static void main(String args[]){
int bob[] = {1,2,3};
int alice[] = {4,2,6};
int bobScore = 0;
int aliceScore = 0;
int i = 0;
for(i=0;i<3;i++){
if(bob[i]<alice[i]){
aliceScore++;
}
else if(bob[i]>alice[i]){
bobScore++;
  }
}
System.out.println(bobScore);
System.out.println(aliceScore);
}
}
```

Output:

aliceScore=1

bobScore=0

98.WRITE A PROGRAM TO VALIDATE NAME ,AGE,PHONENUMBER,EMAIL ID:

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>student information form</title>
<script type ="text/javascript">
function valid()
{
   var na = document.getElementById("nm").value;
   var ag = document.getElementById("age").value;
   var em = document.getElementById("eid").vaule;
    var mno = document.getElementById("pno").value;
      if(na==""||na==null){
          alert("enter the name");
          return false;
}
else if(isNaN(ag)||ag<1||ag>100){
         alert("the age must be a number between 1 and 100");
          return false;
}
else if(em==-1){
alert("E-mail ID is not valid");
return false;
}
```

```
else if(mno==""||mno==null){
alert("enter the phone number");
return false;
}
else if(isNaN(mno)||mno.length>10||mno.length<10){
alert("the mobile.no always has 10 digit numerical value");
return false;
}
else
alert("the students information submitted successfully");
}
</script>
</head>
<body>
<center>
 <b>
<h3> student information form</h3>
<form action="pro.html" method="post">
Name: <input type="text" id="nm"><br><br>
Age: <input type="text" id="age" ><br><br>
<!-- onkeyPress="return vaildnum()" -->
Email id: <input type="text" id="eid"><br><br>
Phone Number: <input type="text" id="pno"><br><br>
<input type="submit" value="submit" onclick="return  valid()" >
```

```
</form>
</b>
</center>
</body>
</html>
```

99. .SIMPLE CALCULAT0R

```
import java.util.Scanner;
public class Calci {
        public static void main(String[] args) {
        Scanner ob=new Scanner(System.in);
        System.out.print("Enter two numbers:");
         double first=ob.nextDouble();
         double second=ob.nextDouble();
           System.out.print("Enter an operator(+,-,*,/):");
           char operator=ob.next().charAt(0);
           double result;
          switch(operator){
           case'+':
           result=first+second;
           break;
             case '-':
                    result=first-second;
                    break;
```

```
            case '*':
                    result=first*second;
                    break;
                case '/':
                    result=first/second;
                    break;
            default:
            System.out.println("ERROR OPERATOR IS NOT CORRECT");
                        return;
                }
        System.out.println(first+" "+operator+" "+second+"="+result);
}
        }
```

OUTPUT

```
Enter two numbers:23 2
Enter an operator(+,-,*,/):/
23.0 / 2.0=11.5
```

100.FIND HCF AND LCM:

```
import java.util.Scanner;
public class LCMAnd(HCF/GCD) {
        public static void main(String args[]){
                int a,b,lcm,hcf=0;
                Scanner s=new Scanner(System.in);
```

```
                System.out.print("enter first number:");
                a=s.nextInt();
                System.out.print("enter second number:");
                b=s.nextInt();
                for(int i=1;i<=a&&i<=b;i++){
                        if(a%i==0&&b%i==0){
                            hcf=i;
                        }
                }
                System.out.println("Hcf of numbers:"+hcf);
                lcm=(a*b)/hcf;
                System.out.print("Lcm of numbers:"+lcm);
        }
}
```

OUTPUT:

```
enter first number:5
enter second number:10
```

101.PROGRAM FOR FIND NCR AND NPR:

```
import java.util.Scanner;
public class NCR_NPR_101 {
        public static int fact(int num){
                int fact=1,i;
                for(i=1;i<=num;i++){
```

```
                fact=fact*i;
            }
            return fact;
        }
        public static void main(String[] args) {
            int n,r;
            Scanner scanner=new Scanner(System.in);
            System.out.print("Enter value of n ;");
            n=scanner.nextInt();
            System.out.print("Enter value of r ;");
            r=scanner.nextInt();
            System.out.println("NCR is "+(fact(n)/(fact(n-r)*fact(r))));
            System.out.println("PNR is "+ (fact(n/(fact(n-r)))));
        }
}
```

OUTPUT:

Enter value of n ;3

Enter value of r ;2

NCR is 3

PNR is 6

102.CALCULATE AREA ,PERIMETER AND CIRCUMFERENCE:

```
public class AreaPeriCircum {
    public static void main(String[] args) {
        int r=3;
        int l=5;
```

```
                int b=6;
                double area= Math.PI*(r*r);
                System.out.println("The area of circle is:" +area);
                double circumference= Math.PI*2*r;
                System.out.println("The circumference is:" +circumference);
                area = l*b;
                double perimeter = 2*(l+b);
                System.out.println("area of Rectangle:" +area);
                System.out.println("perimeter of Rectangle:" +perimeter);


                }

}
```

OUTPUT:

```
The area of circle is:28.274333882308138
The circumference is:18.84955592153876
area of Rectangle:30.0
perimeter of Rectangle:22.0
```

103.Convert Farhenhit To Celcius(or)Centigrade.

```
import java.uti.*;
public class FarhenhitToCelcius{
        public static void main(String args[]){
                System.out.print("Enter the Farhenhit value=");
                Scanner sc=new Scanner(System.in);
                int a=sc.nextInt();
                double b=(a-32)/1.8;
                System.out.println("Celcius="+b);
        }
}
```

Output:

```
Enter the Farhenhit value=50
Celcius=10;
```

104 MATRIX MULTIPLICATION ,ADDITION AND SUBTRACTION:

```
public class array {
```

```
public static void main(String args[]){
        int a[][]={{1,1,1},{2,2,2},{3,3,3}};
        int b[][]={{1,1,1},{2,2,2},{3,3,3}};
        int c[][]=new int[3][3];
        System.out.println("Matrix Multiplication");
        for(int i=0;i<3;i++){
                for(int j=0;j<3;j++){
                        c[i][j]=0;
                        for(int k=0;k<3;k++){
                                c[i][j]+=a[i][k]*b[k][j];
                        }
                        System.out.print(c[i][j]+" ");
                        }
                System.out.println();
                }
        System.out.println("addition Matrix");
        for(int i=0;i<3;i++){
                for(int j=0;j<3;j++){
                        c[i][j]=0;
                        for(int k=0;k<3;k++){
                                c[i][j]=a[i][j]+b[i][j];
                        }
                        System.out.print(c[i][j]+" ");
                        }
```

```
                System.out.println();
                }
        System.out.println("Subtraction Matrix");
        for(int i=0;i<3;i++){
                for(int j=0;j<3;j++){
                        c[i][j]=0;
                        for(int k=0;k<3;k++){
                                c[i][j]=a[i][j]-b[i][j];
                        }
                        System.out.print(c[i][j]+" ");
                        }
                System.out.println();
                }
        }
}
```

OUTPUT:

```
Matrix Multiplication
6 6 6
12 12 12
18 18 18
Addition Matrix
2 2 2
4 4 4
6 6 6
Subtraction Matrix
0 0 0
0 0 0
0 0 0
```

106.SAME AS NUM 71

107:CHECK GIVEN NUMBER IS PERFECT OR NOT:

```
import java.util.Scanner;
public class Prefectnumber {
	public static void main(String args[]){
		int num,sum=0;
	   Scanner s=new Scanner(System.in);
		System.out.print("enter first number:");
		num=s.nextInt();
		for(int i=1;i<num;i++){
			if(num%i==0){
				sum+=i;
			  }
			}
			if(sum==num){
				System.out.print("perfect number");
			}
			else System.out.print("not perfect number");

	}

}
```

OUTPUT:

```
enter first number:28
perfect number
```

108:FIND GIVEN NUMBER IS ARMSTRONG OR NOT:

```
public class ArmstrongNum_108 {
        public static void main(String args[]){
            int a = 0,b,temp;
            int n = 153;
            temp = n;
            while(n>0){
            b = n%10;
            n = n/10;
            a = a+(b*b*b);
            }
              if(temp == a)
              System.out.println(temp+" is armstrong number");
              else
              System.out.println( temp+" is not a armstrong number");
        }
    }
```

Output

153 is armstrong number

109.

110.VALIDATE THE EMAIL ID:

```
<html>
	<head>
		<script>
			function nullCheck(){
				var email=document.frm.email.value;
				if(email==null||email==""){
					alert("email can't be blank");
					return false;
				}
			}
			function checkmail(){
				var email=document.frm.email.value;
				var mail=/^\w+@[a-zA-Z]+?\.[a-zA-Z]{2,3}$/;
				if(!email.match(mail)){
					alert("email must contain @ and .");
					return false;
				}
			}
			function validSubmit(){
				var retvalue;
				retvalue=nullCheck();
				if(retvalue==false){
					return retvalue;
				}
```

```
                    retvalue=checkmail();
                    if(retvalue==false){
                        return retvalue;
                    }
                }
            </script>
        </head>
        <body>
            <form name="frm" action="welcome.html">
                email:<input type="text" name="email"/></br>
                <input type="submit" value="login" onClick="return validSubmit()"/>
            </form>
        </body>
</html>
```

111.VALIDATE THE PHONE NUMBER:

```
<html>
    <head>
        <script type = "text/javascript">
                    function nullcheck(){
                    var phoneno = document.frm.phoneno.value;
 if(phoneno==isNaN()||phoneno == null || phoneno == " "){
alert("phoneno cant be blank");
return false;
}
```

```
else if(phoneno.length != 10){
alert("need numbers");
return false;
        }
}
</script>
</head>
<body>
<form name="frm" action="welcome.html">
Phone No:<input type="text" name="phoneno"/>
<input type="submit" name="login" onClick="return nullcheck()"/>
</form>
</body>
</html>
```

112.VALIDATE THE GIVEN DATE:

```
<Html>
    <head>
    <script>
        function isDate(){
            var currVal =document.frm.Date.value;
            if(currVal == ''){
                alert("Date is Can't be Blank");
                return false;
            }
              var rxDatePattern = /^(\d{1,2})(\/|-)(\d{1,2})(\/|-)(\d{4})$/; //Declare
```

Regex

```
                    var dtArray = currVal.match(rxDatePattern); // is format OK?
                        if (dtArray == null){
                    alert("Please Enter Valide Date");
                    return false;
                }
                //Checks for mm/dd/yyyy format.
                dtMonth = dtArray[1];
                dtDay= dtArray[3];
                dtYear = dtArray[5];
                  if (dtMonth < 1 || dtMonth > 12){
                    alert("Please Enter valid Date");
                    return false;
                }
                else if (dtDay < 1 || dtDay> 31){
                    alert("Please Enter valid Date");
                    return false;
                }
        else if ((dtMonth==4 || dtMonth==6 || dtMonth==9 || dtMonth==11) && dtDay ==31) {
                    alert("Please Enter valid Date");
                    return false;
                }
                else if (dtMonth == 2)
                {
```

```
                var isleap = (dtYear % 4 == 0 && (dtYear % 100 != 0 || dtYear % 400 ==0));
                    if (dtDay> 29 || (dtDay ==29 && !isleap)){
                        alert("Please Enter valid Date");
                            return false;
                    }
                }
                return true;
            }
            </script>
            <title> Date Validation</title>
            </head>
            <body>
            <form name="frm" onsubmit="return isDate()" action="welcome.html">
            Date<input type="text" name="Date" value="mm-dd-yyyy"/>
            <input type="submit" value="login" />
            </form>
            </body>
            </html>
```

LOGIN SUCCES PAGE:

```
<html>
    <head>
        <title>
            loginApp
        </title>
```

```
        </head>
        <body>
            <b>validation success..
        </body>
</html>
```

113.VALIDATE SIMPLE INTEREST:

```
public class SIMPLEINTREST {
public static void main(String args[]){
        float si, p=5000,r=15,t=1;
        //p=principle, r=rate, t=times
        si=(p*r*t)/100;
        System.out.println("Simple interest is="+si);
}
}
```

OUTPUT

Simple interest is=750.0

114.VALIDATE COMPOUND INTEREST:

```
public class CompoundInterest {
    public void compound(int p,int t,double r,int n){
                double amount=p* Math.pow(1+(r/n),n*t);
                double cinterest=amount-p;
                System.out.println("compund interst after "+t+" years:"+cinterest);
                System.out.print("Amount after "+t+" years:"+amount);
```

```
        }
        public static void main(String[] args){
                CompoundInterest ob=new CompoundInterest();
                 ob.compound(2000,5,0.08,12);
        }
}
```

OUTPUT:

compund interst after 5 years:979.6914166032102

Amount after 5 years:2979.69141660321

115:GENERATE 5 DIGIT RANDOM NUMBER:

```
public class Random_115 {
        public static void main(String[] args) {
          System.out.println(Math.round(Math.random()*100000));
                }
}
```

OUTPUT:

59787

116:PROGRAM FOR Celsius To Farenheit

```
import java.util.Scanner;
public class CelsiusToFarenheit {
        public static void main(String[] args) {
                System.out.println("Enter the celcius");
                Scanner sc=new Scanner(System.in);
                double a=sc.nextDouble();
```

```
                double b=((a*9/5)+32);
                System.out.println(b);

        }

}
```

OUTPUT:

Enter the celsius
45
113.0

117.CONVERT FARENHEIT TO CELSIUS:

```
import java.uti.*;

public class FarhenhitToCelcius{

        public static void main(String args[]){

                System.out.print("Enter the Farhenhit value=");

                Scanner sc=new Scanner(System.in);

                int a=sc.nextInt();

                double b=(a-32)/1.8;

                System.out.println("Celcius="+b);

        }

}
```

Output:

Enter the Farhenhit value=50

Celcius=10;

118:REPLACE STRING:

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

```
class ReplacetheString{
    public static void main(String args[]){
        Scanner sc=new Scanner(System.in);
            System.out.println("enter the string");
            String a=sc.nextLine();
            System.out.println("replaceable word");
            String b=sc.nextLine();
            System.out.println("new word");
            String c=sc.nextLine();
            if(a.contains(b)){
                System.out.println(a.replace(b,c));
            }
           sc.close();
        }
    }
```

Output:

```
enter the string:  this the time say the hello
 replaceable word:   the
  new word:    a
   this a time say a hello
```

119: PRINT EVEN NUMBER IN ASCNDING AND ODD NUMBER IN DESCENDING:

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

```
int a[]={2,8,4,9,3,6,1,7};
Arrays.sort(a);
for(int i=0;i<a.length;i++)
if(a[i]%2==0)
        System.out.print(a[i]);

for(int i=a.length-1;i>=0;i--)
        if(a[i]%2!=0)
                System.out.print(a[i]);

        }
}
```

OUTPUT:

24689731

120. EVEN AND ODD IN ASCENDING

```
import java.util.Arrays;
public class oddeveninAscending {
public static void main(String[] args) {
int a[]={2,8,4,9,3,6,1,7};
Arrays.sort(a);
for(int i=0;i<a.length;i++)
if(a[i]%2==0)
        System.out.print(a[i]);

for(int i=0;i<a.length;i++)
        if(a[i]%2!=0)
                System.out.print(a[i]);

        }
}
```

OUTPUT:

24681379

121.ADD 1 NUMBER IF GIVEN IS EVEN NUMBER IS EVEN ELSE SUBTRACT 1 FROM THE NUMBER:

```
import java.util.*;

public class addorsub{

        public static void main(String args[]){

                int a[]={2,3,1,6,7};

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

                        if(a[i]%2==0){
```

```
                    System.out.print(a[i]+1);
                    }
                        else    System.out.print(a[i]-1);
                }
            }
        }
```

Output:32076

122: AIR ROUTES

INPUT :4 OUTPUT :6

```
public class Airroutecount_122 {
    public static void main(String[] args) {
        int a=0;
        int n=4;
        for(int i=1;i<n;i++){
            for(int j=1;j<i+1;j++){
                a++;
            }
        }
        System.out.println("output="+a);
    }
}
```

OUTPUT=6

123:FIRST REPEATING WORD:

```
public class FirstRepeat {
```

```
	public static void main(String[] args) {
		String str="ICECREAM";
		int k=0;
		char a[]=str.toCharArray();
		for(int i=0;i<str.length();i++){
			for(int j=i+1;j<str.length();j++){
				if(str.charAt(i)==str.charAt(j)){
					k++;
					break;

				}
			}
			if(k==1){
				System.out.println(str.charAt(i));
				break;
			}
		}
	}

}
```

OUTPUT:

C

124: PRINT THE NUMBER PATTERN:

```
import java.util.Scanner;

public class Check {

    public static void main(String args[]) {

        System.out.print("Enter the Number :");

            Scanner s=new Scanner(System.in);

            int n=s.nextInt();

            for(int i=0;i<3;i++){

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

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

```
                }
            System.out.println("");
            n--;
            }
            for(int i=0;i<2;i++){
                for(int j=1;j>=i;j--){
                    System.out.print(n);
                }
                System.out.println("");
                n--;
            }
        }
}
```

Output:

```
Enter the Number :5
5
44
333
22
1
```

125: WRITE A PROGRAM TO CONVERT ROMAN DIGIT TO DECIMAL NUMBER:

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

```
                Scanner sc = new Scanner(System.in);
                System.out.println("enter roman letter");
                String s=sc.next();
Map<Character,Integer>m = new LinkedHashMap<Character,Integer>();
m.put('I',1);
m.put('V',5);
m.put('X',10);
m.put('L',50);
m.put('C',100);
m.put('D',500);
m.put('M',1000);
if(s.length()>2){
int temp=m.get(s.charAt(s.length()-1));
for(int i=0;i<s.length()-1;i++) {
    if(m.get(s.charAt(i))>=m.get(s.charAt(i+1))) {
     temp = temp+ m.get(s.charAt(i));
     }
else {
temp = temp - m.get(s.charAt(i));
       }
   }
System.out.println(temp);
}
else {
```

```
        int temp = 0;
        for(int i = 0;i<s.length()-1;i++) {
                if(m.get(s.charAt(i))>=m.get(s.charAt(i+1))) {
                        temp = temp+m.get(s.charAt(i));
                }
                else {
                        temp = temp - m.get(s.charAt(i));
                    }
                temp = temp+m.get(s.charAt(s.length()-1));
            }
        System.out.println(temp);
      }
  }
}
```

Output:

input:VI

output:6

126:FIND FIRST NON REPEATE CHARACTER:

```
import java.util.HashMap;
public class Repeated_Nonrepeated {
        public static void main(String args[]){
        String inp="BOOK";
        HashMap<Character,Integer> count=new HashMap<Character,Integer>();
        char[] strArray=inp.toCharArray();
        for(char c:strArray){
                if(count.containsKey(c)){
                        count.put(c,count.get(c)+1);
                }
                else {
                        count.put(c,1);
                }
```

```
        }
        for(char c:strArray){
                if(count.get(c)==1){
                        System.out.println("Non repeated character:"+c);
                        break;
                        }

        }
  }
}
```

OUTPUT: Non repeated character:B

127:ARRANGE THE LEFT MOST BIT BY ASCENDING ORDER:

```
import java.util.Arrays;

public static void main(String args[]){

int a={3,52,10,25,456};

String s1[]=new String[a.length];

for(int i=0;i<s1.length;i++){

s[i]=Integer.toString(a[i]);

}

Arrays.sort(s1);

for(int i=0;i<s1.length;i++){

a[i]=Integer.parseInt(S1[i]);

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

      }

    }

}
```

OUTPUT: 10 25 3 456 52

128: CHANGE UPPERCASE TO LOWERCASE AND VICEVERSA

```
import java.util.*;

import java.util.Scanner;
```

```
public class UppercaseLowercase {
	public static void main(String args[]){
		Scanner sc=new Scanner(System.in);
		String s=null;
		char c[]=null;
		System.out.print("enter first String:");
		s=sc.nextLine();
		c=s.toCharArray();
		for(int i=0;i<c.length;i++){
			if((c[i]>='A'&& c[i]<='Z')||(c[i]>='a'&& c[i]<='z')){
			  if(c[i]>='A'&& c[i]<='Z')
				c[i]=(char) (c[i]+32);
			   else if(c[i]>='a'&& c[i]<='z')
				c[i]=(char) (c[i]-32);
			}
			}
	System.out.print(c);
	}
}
```

OUTPUT:

```
enter first String:MASSmind JaVA
massMIND jAva
```

129:CALCULATE SUM,MAXIMUM,MINIMUM OF ARRAY

```
 import java.util.Arrays;
public class SumOfArray_129{
```

```
    public static void main(String[] args) {
            int a[]={3,8,9,1,2,0,4,7};
            int sum=0;
            for(int i=0;i<a.length;i++){
                    sum=sum+a[i];
            }
            System.out.println("sum of Array ="+sum);
            float Ave =(sum/a.length);
            System.out.println("Avarage of array= "+Ave);
            Arrays.sort(a);
            System.out.println("min of array ="+a[0]);
            System.out.println("max of array ="+a[(a.length -1)]);
            }
    }
```

OUTPUT:

sum of Array =34

Avarage of array= 4.0

min of array =0

max of array =9

130:PRINT THE NUMBER PATTERN:

```
public class Pattern {
    public static void main(String[] args) {
            for(int i=1;i<=4;i++){
                    for(int j=1; j<=i;j++){
                            System.out.print(i);
```

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

        }

        for(int i=5;i<=7;i++){
            for(int j=7;j>=i;j--){
                System.out.print(i);
                }
            System.out.println();

            }
      }

    }
```

OUTPUT:

```
1
22
333
4444
555
66
7
```

131: .Find the Position of Character.

```
import java.util.*;

public class FindPosition{

    public static void main(String args[]) {

         int z=1;

        String s="MATHMATICS";

        char c[]=s.toCharArray();

        Map<Integer,Character> m=new LinkedHashMap<Integer,Character>();

        for(int i=0;i<20;i++){

            for(int j=0;j<c.length;j++){
```

```
                    m.put(z,c[j]);
                    z++;
                }
            }
            System.out.print("Enter the number : ");
            Scanner sc=new Scanner(System.in);
            int n=sc.nextInt();
            System.out.println(m.get(n));
        }
    }
```

Output:

Enter the number : 14

H

132:MINIMUM NUMBER TO ADD OR SUBTRACT TO GET PRIME NUMBER:

```
import java.util.*;
public class Prime2 {
    static Scanner sc = new Scanner(System.in);
    static int a=sc.nextInt();
    static int n=0;
    static int n1=0;
    static boolean meth(){
     int num = a+n;
      boolean flag = false;
        for(int i=2;i<=num/2;i++){
```

```
                    if(num%i==0){
                     flag=true;
                     break;
                    }
   }
  return flag;
}
static boolean meth1(){
              int num = a+n1;
              boolean flag = false;
             for(int i=2;i<=num/2;i++) {
                    if(num%i==0) {
                     flag=true;
                     break;
                    }
                  }
  return flag;
}
public static void main(String[] args) {
           for(int i=0;i<10;i++){
                    if(!meth()||!meth1()){
                              if(!meth()){
                              System.out.println(n);
                              break;}
```

```
                    if(!meth1()){
                        System.out.println(n1);
                        break;
                            }
                    }
                     else {
                        n++;
                        n1--;
                       meth();
                      meth1();
                        }
                }
            }
```

Output:

input: 13

output:0

input: 14

output:-1

input: 27

output:2

133:FIND MOST OCCURING WORD IN STRING:

134:FIND THE HIGHEST PRIME FACTOR GIVEN NUMBER:

```
public class Highestprimefactor {
```

```
public static int highestpf(long num){
        int i;
        for(i=2;i<=num;i++){
                if(num%i==0){
                        num/=i;
                        i--;
                        }
                  }
         return i;
            }
public static void main(String[] args) {
          System.out.println(highestpf(60));
     }
}
```

OUTPUT: 5

135: SHUFFLE THE DIGIT IN THE GIVEN NUMBER AND FIND HIGHEST NUMBER:

```
public class HighestNumber {
        public static void main(String args[]){
        int num=123;
        int result=0,mul=1;
        int count[]=new int[10];
        String str=String.valueOf(num);
        for(int i=0;i<str.length();i++){
                count[str.charAt(i)-'0']++;
```

```
        }
        for(int i=0;i<=9;i++){
                while(count[i]>0){
                        result=result+(i*mul);
                        count[i]--;
                        mul=mul*10;
                }
        }
        System.out.print(result);
        }
}
```

Output:

321

136:

137:

138: Programe to arrange the same numbers near by Position.

```
public class Check {
    public static void main(String args[]) {
         int a[]={8,7,4,1,0,-2,-1,-4,-6};
        int k=0;
        for(int i=0;i<a.length;i++){
                k=0;
                for(int j=i+1;j<a.length;j++){
```

```
                    if(a[i]=='*'){
                        k++;
                        break;
                    }
                    if(Math.abs(a[i])==Math.abs(a[j])){
                        System.out.print(a[i]+" "+a[j]+" ");
                        a[j]='*';
                        k++;
                    }
                }
                if(k==0) System.out.print(a[i]+" ");
            }
        }
}
```

Output:

8 7 4 -4 1 -1 0 -2 -6

139:FIND THE HIGHEST SEQUENCE OF REPEATED ELEMENT:

```
public class HighestSequence {
        public static void main(String args[]){
         char array[]={'a','5','c','s','e','9','A','b','a','5','c','s','e'};
         for(int i=0;i<array.length;i++){
          for(int j=i+1;j<array.length;j++){
              if(array[i]==array[j]){
                  System.out.print(i);
```

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

OUTPUT:0 ,1 ,2 ,3 and 8 ,9 ,10 ,11 ,12

140:PRINT THE PATTERN:

```
class pattern140_2{
        public static void main(String args[]){
                int n = 2, a=4;
                for(int i = 0 ; i<a ; i++){
                        if(i == 2){
                                n--;
                            }
                   System.out.print(n);
                }
                n=2;
                System.out.println();
                for(int i = 0 ; i<a-2 ; i++){
                        if(i == 1){
                        n--;
                        }
                        System.out.print(n);
                    }
```

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

OUTPUT:

333222111

332211

321

141:PASCAL TRIANGLE:

```
public class pascal141 {
        public static int fact(int num){
                        int f=1,i=1;
                        while(i<=num) {
                                f=f*i;
                                i++;
                               }
                        return f;
                         }
         public static void main(String[] args) {
                int line=4;
                int i,j;
                for(i=0;i<line;i++){
                        for(j=0;j<=i;j++){
                                System.out.print(fact(i)/(fact(j)*fact(i-j)));
                        }
```

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

OUTPUT:

1

1 1

1 2 1

1 3 3 1

142:FIRST REPEATED AND NON REPEATED CHARACTER IN A GIVEN:

```
import java.util.HashMap;
public class Repeated_Nonrepeated {
    public static void main(String args[]) {
    String inp="MAASMIND";
    HashMap<Character,Integer> count=new HashMap<Character,Integer>();
    char[] strArray=inp.toCharArray();
    for(char c:strArray){
        if(count.containsKey(c)){
            count.put(c,count.get(c)+1);
        }
        else {
            count.put(c,1);
        }
    }
```

```
        for(char c:strArray){
                if(count.get(c)==1){
                        System.out.println("Nonrepeated:"+c);
                        break;
                }
        }
        for(char c:strArray){
                if(count.get(c)>1){
                        System.out.println("Repeated:"+c);
                        break;
                }
           }
       }
}
```

OUTPUT:

Nonrepeated: S

Repeated: M

143: PRINT THE PATTERN:

```
public class starpatternNum_143 {
        public static void main(String[] args) {
                int n=5;
                for(int i=1;i<=n;i++){
                        for(int j=1;j<=n;j++){
                                if(i==j || i+j==n+1){
```

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

OUTPUT:

```
1       5
 2     4
   3
  2   4
1       5
```

144:PRINT HELLO PATTERN:

```
import java.util.Scanner;

public class HELLOPattern1 {

        public static void main(String[] args) {
                System.out.println("Enter the String");
                Scanner sc=new Scanner(System.in);
                String s1=sc.next();
                int n=10;
```

```
                String s=s1+new StringBuffer(s1).reverse();
                char c[]=s.toCharArray();
                char c1[][]={c,c,c,c,c,c,c,c,c,c};
                for(int i=0;i<n;i++){
                        for(int j=0;j<n;j++){
                                if(i==j)
                                        System.out.print(c1[i][j]);
                                else if((i+j)==(n-1))
                                        System.out.print(c1[i][j]);
                                else System.out.print(" ");
                        }
                        System.out.println("");
                }
        }

}
```

OUTPUT:

```
Enter the String
HELLO
H       H
 E     E
  L    L
   L  L
    OO
    OO
   L  L
  L    L
 E     E
H       H
```

144.b Hello Pattern.

```
public class Check {

    public static void main(String args[]) {

        String s1="Hello";

                String s=new StringBuffer(s1).deleteCharAt(0).reverse()+s1;

                int k=s.length();
```

```
                int k1=s.length()-1;
                char c[]=s.toCharArray();
                for(int i=0;i<k;i++){
                        for(int j=0;j<k;j++){
                                if(i==j) System.out.print(c[i]);
                                else if(k1==j) System.out.print(c[k1]);
                                else System.out.print(" ");
                        }
                        k1--;
                        System.out.println("");
        }
    }
}
```

OutpUT:

```
o     o
 I   I
  I I
   e e
    H
   e e
  I  I
 I    I
o      o
```