
Write a program to create a Matrix (3 by 3)

```
//This program will create a 3 by 3 -matrix
public class Matrix {


     public static void main(String[] args) {

            int arr[][]=new int[3][3];

            int val=10;

            for(int i=0;i<3;i++)
            {
             for(int j=0;j<3;j++)
             {
                arr[i][j]=val;
                val+=10;
             }
            }


            for(int i=0;i<3;i++)
            {
             for(int j=0;j<3;j++)
             {
                 System.out.print(arr[i][j]+ " ");

             }
             System.out.println();

            }

     }

}
```

```
10 20 30
40 50 60
70 80 90
```

```
public class TransposeMatrix {


     public static void main(String[] args) {
           int arr[][]=new int[3][3];

            int val=10;
            for(int i=0;i<3;i++)
            {
             for(int j=0;j<3;j++)
             {
                  arr[i][j]=val;
                  val+=10;
             }
            }
            System.out.println("Before Transpose");
            for(int i=0;i<3;i++)
            {
             for(int j=0;j<3;j++)
             {
                  System.out.print(arr[i][j]+ " ");

             }
             System.out.println();

            }


            System.out.println("After Transpose");
            int temp;
            for(int i=0 ;i<3;i++)
            {
             for(int j=0;j<i;j++)
             {
                  temp=arr[j][i];
                  arr[j][i]=arr[i][j];
                  arr[i][j]=temp;

             }
            }

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

```
            for(int j=0;j<3;j++)
            {
                  System.out.print(arr[i][j]+ " ");

            }
            System.out.println();

           }


     }

}
```

Write a program to find the are of Circle,Square,Rectangle and Triangle

```
public class AreaCalculation {


     public static void main(String[] args) {

            System.out.println(getAreaOfCircle(5.5));
            System.out.println(getAreaOfRectangle(4,5));
            System.out.println(getAreaOfSquare(5));
            System.out.println(getAreaOfTriangle(5, 6));

     }

     static double getAreaOfCircle(double rad)
     {
          final double PI= 3.14;
          double area= PI * rad * rad;
          return area;
     }

     static double getAreaOfSquare(double len)
     {
          double area=len * len;
          return area;
     }

     static double getAreaOfRectangle(double len,double breadth)
```

```
    {
          double area=len * breadth;
          return area;
    }

    static double getAreaOfTriangle(double base,double height)
    {
         double area=    base * height * 1/2;
         return area;
    }
}
```

Write a program to derive Fibonacci series from 1-100

Ex: 1,1,2,3,5,8,13,21,34 …..

```
public class FibonacciSeries {


    public static void main(String[] args) {

            int first=0,second=1;
            int next=1;

           System.out.print(first + "," + second);


           int maxval=100;

            while(first+second <=maxval)
           {
               next=first + second;
               System.out.print("," + next);
               first=second;
               second=next;

           }
   }

}
```

Write a program to derive Factorial of a given number

```
public class Factorial {


     public static void main(String[] args) {

            int num= 5;
            int fact= 1;
            System.out.println("Factorial of " + num );

            for (int i= 1; i<=num; i++)
            {
                fact=fact*i;
             }
            System.out.println(fact);


     }

}
```

Write a program to verify if a given word is a palindrome
(Palindrome:characters appears same in both the directions eg: MADAM - Palindrome)

```
class Palindrome
{
   public static void main(String args[])
   {
      String original, reverse="";
      Scanner input = new Scanner(System.in);
```

```
        System.out.println("Enter a string to check if it is a
palindrome");
        original = input.nextLine();

        int length = original.length();

        for ( int i = length - 1 ; i >= 0 ; i-- )
        {
           reverse = reverse + original.charAt(i);
        }

        if (original.equals(reverse))
        {
         System.out.println("Entered string is a palindrome.");
        }
        else
        {
           System.out.println("Entered string is not a
palindrome.");
        }
        input.close();

    }
}
```

Write a program to print Prime Numbers in the given range.

```
import java.util.Scanner;


class PrimeNumber {

     public static void main(String[] args) throws Exception{

          int i;

          Scanner input = new Scanner(System.in);

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

          int num = Integer.parseInt(input.nextLine());
```

```
            System.out.println("Prime number: ");

            for (i=1; i < num; i++ ){

                  int j;

                  for (j=2; j<i; j++){
                        int n = i%j;
                        if (n==0){
                              break;
                        }

                  }

                  if(i == j){
                        System.out.print("  "+i);
                  }
      }
   }
}
```

Write a program to print Prime Numbers in the given range.

```
import java.util.Scanner;

public class PascalsTriangle {

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


          System.out.println("Please enter the size of the
triangle you want");

          int size = sc.nextInt();

          int[][] myArray = new int[size][size];

          myArray = fillArray(myArray);

          //myArray = calculateArray(myArray);
```

```
        printArray(myArray); //prints the array

    }


   private static int[][] fillArray(int[][] array)
    {
          array[0][1] = 1;

            for (int i = 1; i < array.length; i++)
            {
                for (int j = 1; j < array[i].length; j++)
                {
                array[i][j] = array[i-1][j-1] + array[i-1][j];
                }
            }

            return array;
        }

        private static void printArray(int[][] array)
        {
            for (int i = 0; i < array.length; i++)
            {
                for (int j = 0; j < array[i].length; j++)
                {
                    if(array[i][j] != 0)
                    System.out.print(array[i][j] + " ");
                }
                System.out.println();
            }

        }

}
```

Write a program to take a String as input and reverse it.

```
public class StringReverse {
```

```
	public static void main(String[] args) {
		 String original, reverse = "";
		Scanner in = new Scanner(System.in);

		System.out.println("Enter a string to reverse");
		original = in.nextLine();

		int length = original.length();

		for ( int i = length - 1 ; i >= 0 ; i-- )
		{
		   reverse = reverse + original.charAt(i);
		}
		System.out.println("Reverse of entered string is:
"+reverse);

	}

}
```

Write a program to reverse a number

```
public class NumberReverse {


	public static void main(String[] args) {
		 int original=12345;
		StringBuffer reverse=new StringBuffer();
		String str= Integer.toString(original);
		int length = str.length();

		for ( int i = length - 1 ; i >= 0 ; i-- )
		{
		   reverse = reverse.append(str.charAt(i));
		}
		System.out.println("Reverse of the number " +
reverse.toString());


	}


}
```

Write a program to print Floyd's Triangle

```
public class FloydsTriangle {

     public static void main(String args[])
     {
     int i, j, n;
     for( i = 1; i <= 5; i++)
     {
        for( j = i, n = 1; n <= i; n++, j++)
        {
         System.out.print(j%2 + " ");
        }
         System.out.println(" ");
     }
 }
}
```

Write a program to print highest of 3 numbers

```
class LargestOfThreeNumbers
{
   public static void main(String args[])
   {
      int x, y, z;
      System.out.println("Enter three integers ");
      Scanner in = new Scanner(System.in);

      x = in.nextInt();
      y = in.nextInt();
      z = in.nextInt();

      if ( x > y && x > z )
         System.out.println("First number is largest.");
      else if ( y > x && y > z )
         System.out.println("Second number is largest.");
      else if ( z > x && z > y )
         System.out.println("Third number is largest.");
      else
         System.out.println("Entered numbers are not
distinct.");
   }
}
```

```
/*An Armstrong number is a number such that the sum
! of its digits raised to the third power is equal to the number
! itself.  For example, 371 is an Armstrong number, since
! 3**3 + 7**3 + 1**3 = 371.*/
```

Write a program to find in a given number is Armstrong number

```
class ArmstrongNumber
{
   public static void main(String args[])
   {
      int n, sum = 0, temp, r;

      Scanner in = new Scanner(System.in);
      System.out.println("Enter a number to check if it is an
armstrong number");
      n = in.nextInt();

      temp = n;

      while( temp != 0 )
      {
         r = temp%10;
         sum = sum + r*r*r;
         temp = temp/10;
      }

      if ( n == sum )
         System.out.println("Entered number is an armstrong
number.");
      else
         System.out.println("Entered number is not an armstrong
number.");
   }
}
```

Write a program to print Armstrong number in a given range like 100 to 1000

```
public class ArmstrongNumberInRange {
```

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


        for(int num=100;num<=1000;num++)
        {
          int sum = 0, r=0,  temp=num;

          while( temp > 0 )
          {
             r = temp%10;
             sum = sum + (r*r*r);
             temp =temp/10;

          }

          if ( num == sum )
             System.out.println("Number " + num + " is an
armstrong number");
        }

    }


}
```

Write a program for simple number sort

```
public class SimpleNumberSort {

    public static void main(String args[])
   {

        int[] arr=new int[5];

         arr[0]=10;
         arr[1]=30;
         arr[2]=44;
         arr[3]=50;
         arr[4]=25;
         Arrays.sort(arr);

        for(int i=0;i<arr.length;i++)
        {
                System.out.println(arr[i]);
```

```
        }

    }
}
```

Sorting in descending order, however, is only possible either by writing your own sorting code, or converting your array to Integer objects then importing from the Collections library

Write a program for number sorting using bubble sort for desceding order

```
public class IntegerSorting {

    public static void main(String[] args) {

        int temp;

        int num[]={5,8,2,1,9};

        for(int i=0;  i < num.length;  i++ )
        {
            for(int j=i+1;  j < num.length;  j++ )
            {
             // to get ascending order change it >

              if ( num[i] < num[j] )
              {
                 temp = num[ i ];
                 num[ i ] = num[ j ];
                 num[ j ] = temp;
               }
            }
        }
```

```
            for(int k=0;   k < num.length;   k++ )
            {
              System.out.println(num[k]);
            }


      }

}
```

Please solve this also

1 Write a program to print odd numbers b/w 1-100
2.Write a program to print even numbers b/w 1-100
3.Write a program to print sum of 100 numbers
4. Write a program to print product of first 10 numbers
5Write a Java program to compare two numbers
6. **Write a Java Program to list all even numbers between two numbers**
Write a program to print the below Triangle

1
23
456
78910
7 Write a program to 10 -1 in reverse order

8 Write a program to print

1

22
333
4444
55555

9 Write a program to find if two integers are both even or both odd none

10 Write a program to print all odd numbers from 10 -50

11Write a program to find the sum of all the numbers from 10-50 that are divisible by 3

Write a program to get the following

input str1="Water,str2="Bottle"
o/p-WatBottleer

```
public class StringManipulation1 {


     public static void main(String[] args) {
           String str1="Water";
          String str2="Bottle";


           //str2.replaceFirst("", str1.substring(0, 3));
           //o/p WatBottle
```

```
            //str1.substring(str1.length()-2, str1.length());
             //o/p er
            System.out.println(str2.replaceFirst("",
str1.substring(0, 3))+ str1.substring(str1.length()-2,
str1.length()));


      }

}
```

Write a Program to print average of the integer array elements and also to print the mean base on odd or even number of elements in the array

```
public class ArrayAverage {

       public static void main(String[] args) {


          int[] numbers = new int[]{10,20,15,25,16,60,100,5,7};

                //to print the average of array elements
                int sum = 0;

                for(int i=0; i < numbers.length ; i++)
                        sum = sum + numbers[i];


                double average = sum / numbers.length;

                System.out.println("Average value of array
elements is : " + average);

        //to give you the mean based on odd or even elements
                // in the array
                if (numbers.length % 2==0)
                {
                     int num1pos=numbers.length/2;
                     int num2pos=num1pos +1;
                     double mean=(numbers[num1pos-
1]+numbers[num2pos-1])/2;
                     System.out.println(mean);
```

```
                }
                else
                {
                    int num1pos=numbers.length/2;
                    System.out.println(numbers[num1pos]);
                }

        }
}
```

Write a program to divide a number without using / operator

```
public class DivideWithOutOperator {


    public static void main(String[] args) {
        int number = 26;
        int divisor = 5;
        int result = 0;

        while((number-divisor)>=0){
          result++;
          number = number - divisor;
        }

        System.out.println(result);
    }

}
```

Write a program to multiply 2 numbers without using number without using * multiplication operator

```
public class MultiplyWithoutOperator {


    public static void main(String[] args) {
        int number1 = 10;
        int number2 = 5;
        int result = 0;

        for(int i=1;i<=number2;i++)
        {
```

```
                result=result + number1;
            }

            System.out.println(result);

        }

}
```

Write a program to sort numbers and digits in a given String

```
public class SortingNumberAndDigits {


    public static void main(String[] args) {
        String str="abcd123efgh456";
        char[] charArray = str.toCharArray();
        StringBuffer str1=new StringBuffer();
        StringBuffer str2=new StringBuffer();
        for(char ch: charArray)
        {
            if (Character.isDigit(ch))
            {
              str1=str1.append(ch);
            }
            else
            {
                str2=str2.append(ch);
            }
        }
        System.out.println(str1);
        System.out.println(str2);


    }
}
```

Write a program to print A-Z and a-z

```
public class PrintA2Z {

    public static void main(String[] args) {
        for(char ch='a';ch<='z';ch++){
          System.out.print(ch+" ");
       }
       System.out.println();
```

```
        for(char ch='A';ch<='Z';ch++){
            System.out.print(ch+" ");
        }

    }

}
```

Write a program to reverse a String and also Sort the string characters alphabetically.

```
public class ReverseAndSort {


    public static void main(String[] args) {
        String str="Hello Chennai";
        StringBuffer str1 = new StringBuffer(str);
        System.out.println(str1.reverse());
        //to put it in a string
        str=str1.reverse().toString();
        System.out.println(str);

        //code to sort
        char[] charArray = str.toCharArray();
        Arrays.sort(charArray);
        str=new String(charArray);
        System.out.println(str);

    }

}
```

Write a program to print a the following Triangle

```
    1
   1 1
  1 1 1
 1 1 1 1
1 1 1 1 1
```

```
public class TriangleOne {

     public static void main(String[] args) throws IOException {
     System.out.println("Enter the number of rows");
             Scanner in = new Scanner(System.in);

             int numRow = in.nextInt();
             for (int i = 1; i <= numRow; i++) {
                 // Prints the blank spaces
                 for (int j = 1; j <= numRow - i; j++) {
                     System.out.print(" ");
                 }
                 // Prints the value of the number
                 for (int k = 1; k <= i; k++) {
                     System.out.print("1 ");
                 }
                 System.out.println();
             }
    } }
```

Write a program to print a the following Triangle

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

```
public class RowNumberIncrementTriangle {

      public static void main(String[] args) throws IOException
{
          System.out.println("Enter the number of rows");
                  Scanner in = new Scanner(System.in);

                  int numRow = in.nextInt();
                  for (int i = 1; i <= numRow; i++) {
                      // Prints the blank spaces
                      for (int j = 1; j <= numRow - i; j++) {
                          System.out.print(" ");
                      }
                      // Prints the value of the number
                      for (int k = 1; k <= i; k++) {
                          System.out.print(i +" ");
                      }
                      System.out.println();
                  }
          }
 }
```

Write a program to print a the following Triangle

```
    1
   32
  654
```

10987

```
public class FlippedTriangle
{
     public static void main(String[] args)
     {
        int rows=4;
        int cntr=1;
        int start;
        int val;
       for(int i=1;i<=rows;i++)
       {

        for(int k=rows-i;k>=1;k--)
        {
             System.out.print(" ");
        }
        start=cntr + i-1;
             val=start;
        for(int j=1;j<=i;j++)
        {

          System.out.print(start);
          start--;
          cntr++;
        }
        System.out.println();
       }

     }

}
```

Write a program to print the next characters in a given String
Ex:
String s1=”Selenium”
o/p should be- `Tfmfojvn`

```
public class SetNextCharForString {

	public static void main(String[] args) {
		String str="Selenium";
		StringBuffer str1=new StringBuffer();
		char arr[]=str.toCharArray();

		for(int i=0;i<=arr.length-1;i++)
		{
			char ch=arr[i];
			str1=str1.append(++ch);
		}
		System.out.println(str1);

	}

}
```

Write a program to print the perfect numbers b/w 1-500

Ex:

The number 6 is said to be a perfect number because it is equal to the sum of all its exact divisors (other than itself).

6 = 1 + 2 + 3

```
public class PerfectNumber{

    public static void main(String[]args){

     int sum=0, x=0;

     for(int num=1;num<500;num++)
     {
        for(int i=1;i<num;i++)
        {
          x=num%i;
          if(x==0)
            sum=sum+i;
        }
        if(sum==num)
        {
         System.out.println("Perfect Number is: "+num);
         System.out.println("Factors are: ");
```

```
          for(int i=1;i<num;i++)
          {
           x=num%i;
           if(x==0)
           System.out.println(i);
          }
         }
          sum=0;
    }
 }
}
```

Write a program to print the adams number

If the reverse square root of the reverse of square of a number is the number itself then it is Adam Number.
12 and 21
Take 12
square of 12 = 144
reverse of square of 12 = 441
square root of the reverse of square of 12 = 21
The reverse square root of the reverse of square of 12 = 12, then number itself.
Such number is called Adam Number.

```
class AdamsNumber
{


      public static void main(String[] args)
      {

         AdamsNumber an = new AdamsNumber();
         int i, n, rn;
         int sn, rsn, rrsn;
         System.out.println("List of Adam Numbers under 1000");
         for (i = 10; i < 1000; i++)
         {
            n = i;
            rn = an.ReverseNumber(i);
            if (n == rn)
               continue;
            sn = n * n;
            rsn = rn * rn;
```

```
            rrsn = an.ReverseNumber(rsn);
            if (rrsn == sn)
            {
               System.out.println(n);
            }
         }
      }

      int CountNumberOfDigits(int n)
      {
         int numdgits = 0;
         do
         {
            n = n / 10;
            numdgits++;
         }
         while (n > 0);
         return numdgits;
      }

      int ReverseNumber(int n)
      {
         int i = 0, result = 0;
         int numdigits = CountNumberOfDigits(n);
         for (i = 0; i < numdigits; i++)
         {
            result *= 10;
            result += n % 10;
            n = n / 10;
         }
         return result;
      }
}
```