JAVA EXERCISE PROGRAMS

Simple Display

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

Java Variable Example: Add Two Numbers

```
public class Sivakumar{
public static void main(String[] args){
int a=30;
int b=30;
int c=a+b;
System.out.println(c);
}
}
```

Java Variable Example: Widening

```
public class Sivakumar {
public static void main(String[] args){
int a=10;
float f=a;
System.out.println(a);
System.out.println(f);
}}
```

Java Variable Example: Narrowing (Typecasting)

```
public class Sivakumar {
public static void main(String[] args){
float f=12.4539f;
//int a=f;//Compile time error
int a=(int)f;
System.out.println(f);
System.out.println(a);
}}
```

Java Variable Example: Overflow

```
class Sivakumar {
public static void main(String[] args){
//Overflow
int a=120;
byte b=(byte)a;
System.out.println(a);
System.out.println(b);
}}
```

Java Variable Example: Adding Lower Type

```
class Sivakumar {
public static void main(String[] args){
byte a=20;
byte b=10;
//byte c=a+b;//Compile Time Error: because a+b=30 will be int
byte c=(byte)(a+b);
System.out.println(c);
}}
```

Java Unary Operator Example: ++ and --

```
public class OperatorExample{
public static void main(String args[]){
int x=10;
System.out.println(x++);//10 (11)
System.out.println(++x);//12
System.out.println(x--);//12 (11)
System.out.println(--x);//10
}}
```

Display odd numbers between 1 -100

```
class OddNumber {
	public static void main(String args[]) {
		System.out.println("The Odd Numbers are:");
		for (int i = 1; i <= 100; i++) {
			if (i % 2 != 0) {
				System.out.print(i + " ");
			}
		}
	}
}
```

Sum of odd numbers between 1 -100

```
  class SumOfNum
  {
	public static void main(String args[])
	{
		int sum = 0;
		for (int i = 1; i <= 100; i++)
		{
			if (i % 2 != 0)
			{
				sum = sum + i;
			}
		}
		System.out.println("The Sum Of 100 Odd Numbers are:" + sum);
	}
  }
```

Total number of odd numbers between 1 -100

```
class TotalNumOfOddNum
{
	public static void main(String args[])
	{
		int count = 0;
		for(int i = 1;i <= 100;i++)
		{
			if(i % 2 != 0)
			{
				count++;
			}
		}
		System.out.println("The Count Of Odd Numbers are:" + count);
	}
}
```

Find sum of first n numbers

```
class SumOfNum
{
	public static void main(String args[])
	{
		int sum = 0;
		int n=10;
		for(int i = 1;i <= n;i++)
		{
			sum = sum + i;
		}
		System.out.println("The Sum Of "+n+" Numbers are:" + sum);
	}
}
```

Find the sum of the digits of a number

```
public class DigitsSum
{

	public static void main(String[] args)
	{
		int num=251025, rem = 0, sum = 0, temp;
		temp = num;

		while (num > 0)
		{
			rem = num % 10;
			sum = sum + rem;
			num = num / 10;
		}

		System.out.print("Sum of Digits of " + temp + " is " + sum);
	}
```

```
}
```

Calculate electricity bill

```
public class ElectricBill
{
        public static void main(String args[])
        {
                int units = 123;
                int bill = 0;

                if (units > 100)
                {
                        if (units >= 200)
                        {
                                if (units > 300)
                                {
                                        bill = units * 8;
                                }
                                else
```

```
                        bill = units * 6;
                }
                else
                        bill = units * 5;
        }

        System.out.println("VELLORE ELECTRICITY LTD, VELLORE");
        System.out.println("Units Consumed : " + units);
        System.out.println("Total Bill : " + bill);
    }
}
```

Java program to find Armstrong number

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

        n = 153; // input value
        arg = n;
        for (int i = 1; i < n; i++)
        {
            while (n > 0)
            {
                r = n % 10;
                sum = sum + (r * r * r);
                n = n / 10;

            }

        }
        if (arg == sum)
        {
            System.out.println("Given number is armstrong number: " + arg);
        }
        else
        {
            System.out.println("Given number is not armstrong number: " + arg);
        }
    }

}
```

Program to print Armstrong number between 1 to 1000

```
public class ArmstrongNumbers
{

    public static void main(String[] args)
    {
        int num,rem,limit=1000, sum = 0;
        System.out.print("Armstrong numbers from 1 to N:");
        for (int i = 1; i <= limit; i++)
```

```
                {
                        num = i;
                        while (num > 0)
                        {
                                rem = num % 10;
                                sum = sum + (rem*rem*rem);
                                num = num / 10;
                        }

                        if (sum == i)
                        {
                                System.out.print(i + " ");
                        }
                        sum = 0;
                }

        }

}
```

Print given number in words

```
   public class NumberToWords
   {

        public void pw(int n, String ch)
        {
   String one[] = { " ", " One", " Two", " Three", " Four", " Five", " Six", " Seven", " Eight", "
Nine", " Ten"," Eleven", " Twelve", " Thirteen", " Fourteen", "Fifteen", " Sixteen", " Seventeen",
" Eighteen"," Nineteen" };

   String ten[] = { " ", " ", " Twenty", " Thirty", " Forty", " Fifty", " Sixty", "Seventy", " Eighty",
" Ninety" };

                if (n > 19)
                {
                        System.out.print(ten[n / 10] + " " + one[n % 10]);
                }
                else
                {
                        System.out.print(one[n]);
                }
                if (n > 0)
                        System.out.print(ch);
        }

        public static void main(String[] args)
        {

                int n=28;
                System.out.print(n);
                if (n <= 0)
                {
```

```
                System.out.println("Enter numbers greater than 0");
            }
            else
            {
                NumberToWords a = new NumberToWords();
                a.pw((n / 1000000000), " Hundred");
                a.pw((n / 10000000) % 100, " crore");
                a.pw(((n / 100000) % 100), " lakh");
                a.pw(((n / 1000) % 100), " thousand");
                a.pw(((n / 100) % 10), " hundred");
                a.pw((n % 100), " ");
            }
        }
    }
```

Program to check the given number is Palindrome or not

```
public class PalindromeNumberCheck
{

    public static void main(String[] args)
    {
        int n=121,pal,r,rev=0;
        pal = n;

        while (n > 0)
        {
            r = n % 10;
            rev = rev * 10 + r;
            n = n / 10;

        }

        if (rev == pal)
        {
            System.out.println(" The given no is  palindrome "+ rev);
        }
        else
        {
            System.out.println("The given no is not palindrome " + rev);
        }

    }

}
```

Program to print palindrome number upto N numbers

```
public class PalindromeUptoN
{

    public static void main(String[] args)
    {
        int n, b, rev = 0;
        int limit=50;
```

```
                System.out.print("Palindrome numbers from 1 to N:");
                for (int i = 1; i <= limit; i++)
                {
                        n = i;
                        while (n > 0)
                        {
                                b = n % 10;
                                rev = rev * 10 + b;
                                n = n / 10;
                        }
                        if (rev == i)
                        {
                                System.out.print(i + " ");
                        }
                        rev = 0;
                }

        }

}
```

Program to print N prime numbers and find sum and average

```
public class PrimeNumberUptoN
{

        public static void main(String[] args)
        {
                int num =0, i =0;
                System.out.println("Prime numbers from 1 to 100 are :");
           for (i = 1; i <= 100; i++)
           {
             int counter=0;
               for(num =i; num>=1; num--)
               {
                   if(i%num==0)
                      {
                   counter = counter + 1;
                      }
                   }
                   if (counter ==2)
                    {
                            System.out.print(i+" ");
                    }
            }
          }
        }
```

Program to print patterns of numbers and stars

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

```
public class PyramidPattern1
{
	public static void main(String[] args)
	{
		int n=4;
		for(int i=0;i<n;i++)
		{
			System.out.println("\n");
			for(int j=0;j<=i;j++)
			{
				System.out.print(" * ");
			}
		}

	}

}
```

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

```
public class PyramidPattern2
{
	public static void main(String args[])
   {
      int i, j, k=8;
      for(i=0; i<5; i++)
      {
         for(j=0; j<k; j++)
         {
            System.out.print(" ");
         }
         k = k - 2;
         for(j=0; j<=i; j++)
         {
            System.out.print("* ");
         }
         System.out.println();
      }
   }
}
```

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

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

```
public class DifferentPatternPrograms1
 {
        public static void main(String args[])
        {
                int n,i,j,k,l,m,p,q,r,s;
                Scanner sc=new Scanner(System.in);
                System.out.println("Enter the n values");
                n=sc.nextInt();
                p=n;
                q=n;
                for(i=n;i>=1;i--)
                {
                        for(j=1;j<=i;j++)
                        {
                                System.out.print("*");
                        }
                        for(k=p*2;k<n*2-1;k++)
                        {
                                System.out.print(" ");
                        }
                        for(l=i;l!=0;l--)
                        {
                                if(l==n)
                                {
                                        continue;
                                }
                                System.out.print("*");
                        }
                        p--;
                        System.out.println();
                }
                for(i=1;i<=n;i++)
                {
                        for(j=1;j<=i;j++)
                        {
                                System.out.print("*");
                        }
                        for(k=q*2-2;k>1;k--)
```

```
                        {
                                System.out.print(" ");
                        }
                        for(m=i;m!=0;m--)
                        {
                                if(m==n)
                                {
                                        continue;
                                }
                                System.out.print("*");
                        }
                        System.out.println();
                        q--;
                }
        }
}
```

Print Floyds triangle

```
import java.util.Scanner;

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

      Scanner scan = new Scanner(System.in);
      System.out.println("Enter the number of rows\n");

      int rows = scan.nextInt();
      System.out.println("Floyd's Triangle Generated\n");
      int count = 1;
      for ( int i = 1 ; i <= rows ; i++ )
      {
         for ( int j = 1 ; j <= i ; j++ )
         {
             System.out.print(count+" ");

             count++;
         }

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

```
1
2 3
4 5 6
7 8 9 10
11 12 13 14 15
16 17 18 19 20 21
```

Print numbers in sequence way

```
public class PatternNumberSequence
{
	public static void main(String[] args)
	{
		int a = 3;
		int b = 4;
		int n = 8;

		for (int i = 1; i <= n; i++)
		{
			int c = a + b;
			System.out.print(a + " " + b + " " + c);
			System.out.println(" ");
			a = c;
			b = b + 1;

		}
	}
 }
```

```
3 4 7
7 5 12
12 6 18
18 7 25
25 8 33
33 9 42
42 10 52
52 11 63
```

Print numbers in triangle and pyramid vice

```
1
121
12321
1234321
123454321
```

```
 import java.util.Scanner;
   public class PatternNuberPyramidPrevRev
  {
	public static void main(String args[])
	{
		int s = 1;
		int n;
		Scanner sc = new Scanner(System.in);
		System.out.println("Enter the N values");
		n = sc.nextInt();
		for (int i = 1; i <= n; i++)
		{
			while (s <= i)
			{
				System.out.print(s);
				s++;
```

```
                        }
                        s--;
                        while (s > 1)
                        {
                                System.out.print(--s);

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

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

public class PatternNumberPyramidUptoN
{

        public static void main(String args[])
        {
                {
                        int i, j, n = 1;
                        for (i = 0; i < 5; i++)
                        {
                                for (j = 0; j <= i; j++)
                                {
                                        System.out.print(n + " ");
                                        n++;
                                }
                                System.out.println();
                        }
                }

        }
}

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

import java.util.Scanner;

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

```
        {

                int i, j, n;

                Scanner sc = new Scanner(System.in);
                System.out.println("Enter the Row value  n");
                n = sc.nextInt();
                for (i = 1; i <= n; i++)
                {
                        for (j = 1; j <= i; j++)
                                System.out.print(" " + j);
                        System.out.print("\n");

                }
        }
}
```

Print numbers in pyramid vice

```
import java.util.Scanner;
 public class PatternNumberPyramidArrow
{
        public static void main(String args[])
        {
                int i, j, n;
                Scanner sc = new Scanner(System.in);
                System.out.println("Enter the values ");
                n = sc.nextInt();
                for (i = 1; i <= n; i++)
                {
                        for (j = 1; j <= i; j++)
                                System.out.print(" " + j);
                        System.out.print("\n");
                }
                for (i = n - 1; i >= 1; i--)
                {
                        for (j = 1; j <= i; j++)
                                System.out.print(" " + j);
                        System.out.print("\n");
                }
        }
}
```

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

```
import java.util.Scanner;
public class PatternNumberPyramidRev
{
        public static void main(String args[])
        {
                int i, j, k, n, a;

                Scanner sc = new Scanner(System.in);
                System.out.println("Enter the n values");
                n = sc.nextInt();
                a = n;
                for (i = 1; i <= n; i++)
                {
                        for (j = a; j > 1; j--)
                        {
                                System.out.print(" ");
                        }
                        for (k = i; k != 0; k--)
                        {
                                System.out.print(k);

                        }
                        a--;

                        for (int l = 2; l <= i; l++)
                        {
                                System.out.print(l);

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

```
     1
    212
   32123
  4321234
543212345
```

Print different patterns using stars

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

```
import java.util.Scanner;
public class Star1
{
        public static void main(String args[])
        {
```

```
                int i, j, t;
                System.out.println("How many row you want ");
                Scanner sc = new Scanner(System.in);
                t = sc.nextInt();
                for (j = 0; j < t; j++)
                {
                        for (i = t - 1; i >= j; i--)
                        {
                                System.out.print("*");

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

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

```
public class Star3
{
        public static void main(String[] x)
        {
                int i, j, k, n = 3;
                for (i = 0; i < n; i++)
                {
                        for (j = 0; j <= i; j++)
                        {
                                System.out.print("*");
                        }
                        for (j = (n - i); j >= 2; j--)
                        {
                                System.out.print("  ");
                        }
                        for (k = i; k >= 0; k--)
                        {
                                System.out.print("*");
                        }
                        System.out.println();
                }
        }
}
```

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

Print pyramid triangle with star and numbers

```
public class Star10
{

	public static void main(String args[])
	{
		int i, j, k;
		for (i = 1; i <= 5; i++)
		{
			for (j = i; j < 5; j++)
			{
				System.out.print(" ");
			}
			for (k = 1; k < (i * 2); k++)
			{
				System.out.print("*");
			}
			System.out.println("");
		}
		for (i = 4; i >= 1; i--)
		{
			for (j = 5; j > i; j--)
			{
				System.out.print(" ");
			}
			for (k = 1; k < (i * 2); k++)
			{
				System.out.print("*");
			}
			System.out.println("");
		}
	}
}
```

Program to find largest number in an array

```
  class LargestNumber
  {
	public static void main(String args[])
	{
		int[] a = new int[] { 20, 30, 50, 4, 71, 100};
		int max = a[0];
		for(int i = 1; i < a.length;i++)
		{
			if(a[i] > max)
			{
				max = a[i];
			}
		}

		System.out.println("The Given Array Element is:");
		for(int i = 0; i < a.length;i++)
```

```
                {
                        System.out.println(a[i]);
                }

                System.out.println("From The Array Element Largest Number is:" + max);
        }
   }
```

Program to find second largest number in an array

```
public class SecondLargest {

        public static void main(String[] args) {

                int arr[] = { 14, 46, 47, 86, 92, 52, 48, 36, 66, 85 };
                int largest = arr[0];
                int secondLargest = arr[0];

                System.out.println("The given array is:" );
                for (int i = 0; i < arr.length; i++) {
                        System.out.print(arr[i]+"\t");
                }
                for (int i = 0; i < arr.length; i++) {

                        if (arr[i] > largest) {
                                secondLargest = largest;
                                largest = arr[i];

                        } else if (arr[i] > secondLargest) {
                                secondLargest = arr[i];

                        }
                }

                System.out.println("\nSecond largest number is:" + secondLargest);

        }
}
```

Find largest and smallest number in an array in java

```
public class LargestSmallest
{
        public static void main(String[] args)
        {
                int a[] = new int[] { 23, 34, 13, 64, 72, 90, 10, 15, 9, 27 };

                int min = a[0]; //  assume first elements as smallest number
                int max = a[0]; //  assume first elements as largest number

                for (int i = 1; i < a.length; i++)  // iterate for loop from arrays 1st index (second
element)
                {
                        if (a[i] > max)
                        {
```

```
                        max = a[i];
                    }
                    if (a[i] < min)
                    {
                        min = a[i];
                    }
                }

                System.out.println("Largest Number in a given array is : " + max);
                System.out.println("Smallest Number in a given array is : " + min);
        }

}
```

Program to find largest and second largest in an array

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

                int nums[] = { 5, 34, 78, 2, 45, 1, 99, 23 };
                int maxOne = 0;
                int maxTwo = 0;
                for (int i=0;i<nums.length; i++)
                {
                        if (maxOne < nums[i])
                        {
                                maxTwo = maxOne;
                                 maxOne = nums[i];
                        }
                        else if (maxTwo < nums[i])
                        {
                                maxTwo = nums[i];
                        }
                }
                System.out.println("Largest Number: " + maxOne);
                System.out.println("Second Largest Number: " + maxTwo);
        }
}
```

Find the index of the largest number in an array

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

                int a[] = new int[] { 12, 44, 23, 56, 23, 78, 13 };

                int max = a[0];
                int index = 0;

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

```
                    if (max < a[i])
                    {
                        max = a[i];
                        index = i;
                    }
                }

                System.out.println("Index position of Maximum value in an array is : " +
index);
} }
```

Find the index of the smallest number in an array

```
public class SmallestNumberIndex
{
    public static void main(String[] args) {
        int a[] = new int[]{12,44,23,56,9,23,78,13};

        int min = a[0];
        int index=0;

        for(int i = 0; i < a.length; i++)
        {
            if(min > a[i])
            {
                min = a[i];
                index=i;
            }
        }

        System.out.println("Index position of Smallest value in a given array is :
"+index);
    }

}
```

Program to remove duplicate element in an array

```
public class RemoveDuplicateElements
{
    public static int[] removeDuplicates(int[] input)
    {
        int j = 0;
        int i = 1;
        // return if the array length is less than 2
        if (input.length < 2)
        {
            return input;
        }
        while (i < input.length)
        {
            if (input[i] == input[j])
            {
                i++;
            }
```

```
                else
                {
                        input[++j] = input[i++];
                }
        }
        int[] output = new int[j + 1];
        for (int k = 0; k < output.length; k++)
        {
                output[k] = input[k];
        }
        return output;
    }

    public static void main(String a[])
    {
        int[] input1 = { 2, 3, 6, 6, 8, 9, 10, 10, 10, 12, 12 };
        int[] output = removeDuplicates(input1);

        System.out.print("Input Elements: \n");
        for (int i : input1)
        {
                System.out.print(i + " ");
        }
        System.out.print("\nOutput Elements: \n");
        for (int i : output)
        {
                System.out.print(i + " ");
        }
    }
}
```

Program to print odd and even numbers from an array

```
public class OddEvenArray
{
    public static void main(String args[])
    {
        int s, i;

        int[] a = { 33, 2, 4, 71, 88, 92, 9, 1 };

        for (i = 0; i < a.length; i++)
        {
                for (int j = i + 1; j < a.length; j++)

                {
                        if (a[i] > a[j])
                        {
                                s = a[i];
                                a[i] = a[j];
                                a[j] = s;

                        }
```

```
                }

        }

        System.out.print("Input numbers :");

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

        {

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

        }

        System.out.print("\nOdd numbers :");

        for (i = 0; i <= a.length - 1; i++)
        {

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

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

        }

        System.out.print("\nEven numbers :");

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

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

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

                }

        }

    }

}
```

Program to add two matrix

```
class MatrixAddition
{
        public static void main(String args[])
        {
                int[][] a = new int[][] { { 1, 2, 3},{ 4, 5, 6},{ 7, 8, 9} };
```

```
        int[][] b = new int[][] { { 10, 11, 12},{ 13, 14, 15},{ 16, 17, 18} };
        int[][] c = new int[3][3];

        if(a.length == b.length && a[0].length == b[0].length)
        {
            for(int i = 0;i < a.length;i++)
            {
                for(int j = 0;j < a[i].length;j++)
                {
                    c[i][j] = a[i][j] + b[i][j];
                }
            }
        }
        else
        {
            System.out.println("'A' and 'B' Matrix are not SAME");
            return;
        }

        System.out.println("The Matrix 'A' Value:");
        for(int i = 0;i < a.length;i++)
        {
            for(int j = 0;j < a[i].length;j++)
            {
                System.out.print(a[i][j] + " ");
            }
            System.out.println();
        }

        System.out.println("The Matrix 'B' Value:");
        for(int i = 0;i < a.length;i++)
        {
            for(int j = 0;j < a[i].length;j++)
            {
                System.out.print(b[i][j]+ " ");
           }
            System.out.println();
        }

    System.out.println("The Addition Matrix of 'A' and 'B' Value:");
        for(int i = 0;i < a.length;i++)
        {
            for(int j = 0;j < a[i].length;j++)
            {
                System.out.print(c[i][j] + " ");
            }
            System.out.println();
        }
    }
}
```

Program to check given matrix is null matrix

```
class NullMatrix
{
	public static void main(String args[])
	{
		int[][] a = new int[][] { { 0, 0, 0},{ 0, 0, 1},{ 0, 0, 0} };
		boolean setValue = true;

		abc: for(int i = 0;i < a.length;i++)
		{
			for(int j = 0;j < a[i].length;j++)
			{
				if(a[i][j] != 0)
				{
					setValue = false;
					break abc;
				}
			}
		}

		System.out.println("The Given Matrix Value:");
		for(int i = 0;i < a.length;i++)
		{
			for(int j = 0;j < a[i].length;j++)
			{
				System.out.print(a[i][j] + " ");
			}
			System.out.println();
		}

		if(setValue == true)
		{
			System.out.println("The Given Matrix is a Null Matrix");
		}
		else
		{
			System.out.println("The Given Matrix is not a Null Matrix");
		}
	}
}
```

Program to check given matrix is diagonal matrix

```
  class DiagonalMatrix
  {
	public static void main(String args[])
	{
		int[][] a = new int[][] { { 1, 0, 1},{ 0, 3, 0},{ 0, 0, 3} };
		boolean setValue = true;
		abc: for(int i = 0;i < a.length;i++)
		{
			for(int j = 0;j < a[i].length;j++)
			{
```

```
                        if(i == j)
                        {
                                if(a[i][j] == 0)
                                {
                                        setValue = false;
                                        break abc;
                                }
                        }
                        else if(a[i][j] != 0)
                        {
                                setValue = false;
                                break abc;
                        }
                }
        }

        System.out.println("The Given Matrix Value:");
        for(int i = 0;i < a.length;i++)
        {
                for(int j = 0;j < a[i].length;j++)
                {
                        System.out.print(a[i][j] + " ");
                }
                System.out.println();
        }

        if(setValue == true)
        {
                System.out.println("The Given Matrix is a Diagonal Matrix");
        }
        else
        {
                System.out.println("The Given Matrix is not a Diagonal Matrix");
        }
    }
  }
```

Program for Linear search

```
import java.util.Scanner;

class LinearSearch
{
        public static void main(String args[])
        {
                int i, num, searchval, array[];

                Scanner in = new Scanner(System.in);
                System.out.println("Enter number of elements");
                num = in.nextInt();

                array = new int[num];
                System.out.println("Enter " + num + " integers");
```

```
		for (i = 0; i < num; i++)
			array[i] = in.nextInt();

		System.out.println("Enter the search value:");
		searchval = in.nextInt();

		in.close();
		for (i = 0; i < num; i++)
		{
			if (array[i] == searchval)
			{
				System.out.println(searchval + " is present at location " + (i + 1));
				break;
			}
		}
		if (i == num)
			System.out.println(searchval + " is not exist in array.");
	}
}
```

Program for Binary Search

```
import java.util.Scanner;
public class BinarySearch
{
	public static void main(String args[])
	{
		int counter, num, item, array[], first, last, middle;
		Scanner input = new Scanner(System.in);
		System.out.println("Enter number of elements:");
		num = input.nextInt();

		array = new int[num];

		System.out.println("Enter " + num + " integers");
		for (counter = 0; counter < num; counter++)
			array[counter] = input.nextInt();

		System.out.println("Enter the search value:");
		item = input.nextInt();
		first = 0;
		last = num - 1;
		middle = (first + last) / 2;

		while (first <= last)
		{
			if (array[middle] < item)
				first = middle + 1;
			else if (array[middle] == item)
			{
			System.out.println(item + " found at location " + (middle + 1) + ".");
				break;
```

```
                        }
                        else
                        {
                                last = middle - 1;
                        }
                        middle = (first + last) / 2;
                }
                if (first > last)
                        System.out.println(item + " is not found.\n");
        }
}
```

Program to calculate HCF and LCM

```
public class FindHCFAndLCM
{
        public static void main(String args[])
        {
                int a, b, x, y, t, hcf, lcm;
                x = 6;
                y = 10;
                a = x;
                b = y;
                while (b != 0)
                {
                        t = b;
                        b = a % b;
                        a = t;
                }

                hcf = a;
                lcm = (x * y) / hcf;

                System.out.print("HCF and LCM of : " + x + " and " + y + " is :\n");
                System.out.print("HCF = " + hcf);
                System.out.print("\nLCM = " + lcm);
        }
}
```

Program to find volume of cube

```
public class Cube {

        public static void main(String arg[]) {
                int side=5;
                float volume=side * side * side;
                System.out.println("Volume of Cube :"+ volume);
        }
}
```

program to print the reverse of a given number

```
public class ReverseNum
{

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

```
		int rev = 0;
		int num = 1234;
		int no=num;
		while (num > 0)
		{
			int rem = num % 10;
			rev = rem + (rev * 10);
			num = num / 10;

		}

		System.out.println("Number = "+no);
		System.out.println("Reverse = "+rev);
	}
}
```

Program to convert integer to roman letters

```
import java.util.HashMap;
import java.util.Scanner;

public class IntegertoRoman
{
	private static int[] bases = { 1000, 900, 500, 400, 100, 90, 50, 40, 10, 9, 5, 4, 1 };
	private static HashMap<Integer, String> map = new HashMap<Integer, String>();

	private static void setup()
	{
		map.put(1, "I");
		map.put(4, "IV");
		map.put(5, "V");
		map.put(9, "IX");
		map.put(10, "X");
		map.put(40, "XL");
		map.put(50, "L");
		map.put(90, "XC");
		map.put(100, "C");
		map.put(400, "CD");
		map.put(500, "D");
		map.put(900, "CM");
		map.put(1000, "M");
	}

	public String intToRoman(int num)
	{
		setup();
		String result = new String();
		for (int i : bases)
		{
			while (num >= i)
			{
```

```
                        result += map.get(i);
                        num -= i;
                    }
                }
                return result;
        }

        public static void main(String arg[])
        {
                System.out.println("Enter the number : ");
                Scanner sc = new Scanner(System.in);
                int no = sc.nextInt();
                IntegertoRoman in = new IntegertoRoman();
                int value=no;
                String sd = in.intToRoman(value);
                System.out.println(value+" ---> " + sd);
        }
}
```

Program to count number of words in given string

```
public class WordCount
{
        public static void main(String args[])
        {
                String s = "welcome to CSI2008 java tutorial";

                int count = 1;

                for (int i = 0; i < s.length() - 1; i++)
                {
                    if ((s.charAt(i) == ' ') && (s.charAt(i + 1) != ' '))
                    {
                        count++;

                    }
                }
                System.out.println("Number of words in a string = " + count);
        }
}
```

Program to count number of duplicate words in given string

```
public class CountWords
{

        public static void main(String[] args)
        {
                String input="Welcome to Java Session Session Session";
                String[] words=input.split(" ");
                int wrc=1;

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

```
                        for(int j=i+1;j<words.length;j++)
                        {

                        if(words[i].equals(words[j]))
                                {
                                        wrc=wrc+1;
                                        words[j]="0";
                                }
                        }
                        if(words[i]!="0")
                        System.out.println(words[i]+"--"+wrc);
                        wrc=1;

            }

        }

}
```

Program to remove duplicate words in given string

```
public class RemoveDuplicate
{

        public static void main(String[] args)
        {
                String input="Welcome to Java Session Java Session Session Java";
                String[] words=input.split(" ");
                for(int i=0;i<words.length;i++)
                {
                        if(words[i]!=null)
                        {

                        for(int j=i+1;j<words.length;j++)
                        {

                        if(words[i].equals(words[j]))
                                {
                                        words[j]=null;
                                }
                        }
                        }
                }
                for(int k=0;k<words.length;k++)
                {
                        if(words[k]!=null)
                        {
                                System.out.println(words[k]);
                        }

            }
        }
}
```

Program to count each words and total number of words in given string

```
import java.io.IOException;

public class FindTtalCountWords
{

	public static void main(String args[]) throws IOException
	{
		countWords("apple banna apple fruit fruit apple hello hi hi hello hi");

	}

	static void countWords(String st)
	{
		String[] words = st.split("\\s");
		int[] fr = new int[words.length];
		for (int i = 0; i < fr.length; i++)
			fr[i] = 0;
		for (int i = 0; i < words.length; i++)
		{
			for (int j = 0; j < words.length; j++)
			{
				if (words[i].equals(words[j]))
				{

					fr[i]++;

				}
			}
		}

		for (int i = 0; i < words.length; i++)
		{
			for (int j = 0; j < words.length; j++)
			{
				if (words[i].equals(words[j]))
				{
					if (i != j)
					{
						words[i] = "";
					}
				}
			}
		}

		int total = 0;
		System.out.println("Words and words count:");
		for (int i = 0; i < words.length; i++)
		{
			if (words[i] != "")
			{
```

```
                        System.out.println(words[i] + "=" + fr[i]);

                        total += fr[i];

                    }
                }
                System.out.println("Total words counted: " + total);
            }
        }
```

Program to reverse the string and check whether it is palindrome or not

```
public class PalindromeChecking
{
        public static void main(String[] args)
        {
                String inpstr ="AMMA";
                char[] inpArray = inpstr.toCharArray();
                char[] revArray = new char[inpArray.length];
                int j=0;
                for (int i = inpArray.length - 1; i >= 0; i--)                          {
                        revArray[j]=inpArray[i];
                        j++;
                }
                String revstr=String.valueOf(revArray);
                if(inpstr.equals(revstr))
                {
                        System.out.println("The given string is a Palindrome");
                }
                else
                {
                        System.out.println("The given string is not a Palindrome");
                }

        }
}
```

Program to delete vowels in a given string

```
    public class RemoveAllVovels {

        public static void main(String[] args) {
                String string = "Welcome to CSI2008 Java Programming";
                System.out.println("Input String : "+string);
                string = string.replaceAll("[AaEeIiOoUu]", "");
                System.out.println(string);
        }
    }
```

Program to capitalize first letter of each word in string

```
    public class StringCapital
    {
        public static void main(String[] args)
        {
                String str = "welcome to CSI2008 java program";
                StringBuilder result = new StringBuilder(str.length());
```

```
                String words[] = str.split("\\ ");
                for (int i = 0; i < words.length; i++)
                {

result.append(Character.toUpperCase(words[i].charAt(0))).append(words[i].substring(1)).append
(" ");

                }

                System.out.println(result);

        }

   }
```

Program to split a comma-separated string

```
public class CommaSeparated
{
        public static void main(String[] args)
        {
                String input="Welcome,to,Java Session Session Session";
                String[] words=input.split(",");
                for(int k=0;k<words.length;k++)

                {

                                System.out.println(words[k]);

            }
        }

}
```

Program to convert ASCI value to String

```
public class AsciiToCharacter
{

        public static void main(String[] args)
        {
                char c;
                for(int i=65;i<=90;i++)
                {
                        c =(char)i;
                        System.out.println(i+"  =  "+c);
                }
        }

}
```

Program to replace vowels with star

```
public class VowelswithStar
{
```

```
        public static void main(String[] args)
        {
                String string = "Welcome to CSI2008 Java Programming"; //Input String
                System.out.println("Input String : "+string);  //Displaying Input String
                string = string.replaceAll("[AaEeIiOoUu]", "*"); //Replace vowels with star
                System.out.println(string);      //Display the word after replacement

        }

}
```

Program to print character position count in a given string

```
  public class LetterPositionCount
  {

        public static void main(String args[])
        {

                String s = "CSI2008JAVA";
                char[] a = s.toCharArray();
                int i = 1;
                {
                        for (char output : a)
                        {

                                System.out.print(output + " " + i + " ");
                                i++;

                        }
                }

        }
  }
```

Program to print reversed string by word in given line

```
public class ReverseWord
{

        public static void main(String[] args)
        {
                String input="Welcome to Java Session";
                String[] words=input.split(" ");
                String[] revwords=new String[words.length];
                int j=0;
                for(int i=words.length-1;i>=0;i--)
                {
                        revwords[j]=words[i];
                        System.out.print(revwords[j]+" ");
                        j++;
                }
        }
```

```
}
```

Program to returning a string as reverse text

```
public class StringReverse {

public static void main(String args[])
{
        String string = "Welcome to Java Programming and Dotnet Programming";
        String[] wordsCount = string.split(" ");

        System.out.println("The Given String is:\n" + string + "\n");
        System.out.println("After Reverse String is:");

        for(int i = wordsCount.length;i > 0;i--)
        {
                System.out.print(wordsCount[i - 1] + " ");
                }
        }
}
```

Program to find difference of minimum and maximum numbers of array in java

```
import java.util.Scanner;
class MinMaxInArray
{
        int getMax(int[]inputArray)
        {
                int maxValue=inputArray[0];

                for(int i=1;i<inputArray.length;i++)
                {
                        if(inputArray[i]>maxValue)
                        {
                                maxValue=inputArray[i];
                        }
                }
                return maxValue;
        }

        int getMin(int[]inputArray)
        {
                int minValue=inputArray[0];

                for(int i=1;i<inputArray.length;i++)
                {
                        if(inputArray[i]<minValue)
                        {
                                minValue=inputArray[i];
                        }
                }
```

```
                return minValue;
        }
}

public class ExArrayDifference
{
        public static void main(String[] args)
        {
                int n;
                Scanner sc = new Scanner(System.in);
                System.out.print("Enter number of elements you wants to enter :" );
                n=sc.nextInt();
                int arr[]=new int[n];
                for(int i=0;i<arr.length;i++)
                {
                        System.out.print("Enter ["+(i+1)+"] element :" );
                        arr[i]=sc.nextInt();
                }

                MinMaxInArray mm=new MinMaxInArray();
                System.out.println("Maximum value is :" +mm.getMax(arr));
                System.out.println("Minimum value is :" +mm.getMin(arr));
                int Difference=mm.getMax(arr)-mm.getMin(arr);
                System.out.print("Difference between Minnimum and Maximum in array is : "
+Difference );
        }
}
```

Program to count the occurrences of each character

```
class NoOfOccurenceOfCharacters
 {
   static final int MAX_CHAR = 256;
    static void getOccuringChar(String str)
   {
      int count[] = new int[MAX_CHAR];
       int len = str.length();
       for (int i = 0; i < len; i++)
          count[str.charAt(i)]++;
       char ch[] = new char[str.length()];
      for (int i = 0; i < len; i++) {
          ch[i] = str.charAt(i);
          int find = 0;
          for (int j = 0; j <= i; j++) {
              if (str.charAt(i) == ch[j])
                 find++;
          }

          if (find == 1)
             System.out.println("Number of Occurrence of " +
```

```
                str.charAt(i) + " is:" + count[str.charAt(i)]);
        }
    }
    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        String str = "geeksforgeeks";
        getOccuringChar(str);
    }
}
```