Java Program to Multiply given Number by 4 using Bitwise Operators

```
import java.util.Scanner;
public class Multiply_Bitwise
{
    public static void main(String[] args)
    {
      int n;
      Scanner s = new Scanner(System.in);
      System.out.print("Enter the number:");
      n = s.nextInt();
      int mul = n << 2;
      System.out.println("Answer:"+mul);
    }
}
```

```
Output:
Enter the number:2
Answer:8
```

Write a Java program to perform basic Calculator operations.

```
import
public class
public static void main(String[] args) {
Scanner reader = new Scanner(System.in);
System.out.print("Enter two numbers: ");
```

```
// nextDouble() reads the next double from the
keyboard
double first = reader.nextDouble();
double second = reader.nextDouble();
System.out.print("Enter an operator (+, -, *, /):
");
char operator = reader.next().charAt(0);
double result;
//switch case for each of the operations
switch(operator)
{
case ‘+’:
result=first+second;
break;
case ‘-':
result = first - second;
break;
case ‘*':
result = first * second;
break;
case ‘/':
result = first / second;
break;
// operator doesn't match any case constant (+, -,
//*, /)
default:
System.out.printf("Error! operator is not
correct");
return;

//printing the result of the operations
System.out.printf("%.1f %c %.1f = %.1f", first,
operator, second, result);
}
```

2. Write a Java program to calculate a Factorial of a number.

```
import
public class Factorial {
public static void main(String args[]){
//Scanner object for capturing the user input
Scanner scanner = new Scanner(System.in);
System.out.println("Enter the number:");
//Stored the entered value in variable
int num = scanner.nextInt();
//Called the user defined function fact
int factorial = fact(num);
System.out.println("Factorial of entered number
is: "+factorial);
}
static int fact(int n)
{
int output;
if(n==1){
return 1;
}
//Recursion: Function calling itself!!
output = fact(n-1)* n;
return output;
}
```

Write a Java program to calculate Fibonacci Series up to n numbers. //0 1 1 2 3 5 8 13

```
public class Fibonacci {
public static void main(String[] args) {
//initializing the constants
int n = 10, t1 = 0, t2 = 1;
System.out.print("Upto " + n + ": ");
//while loop to calculate fibonacci series upto n numbers
while (t1<= n)
{
System.out.print(t1 + " , ");
int sum = t1 + t2;
t1 = t2;
t2 = sum;
}
}
}
```

Write a Java program to find out whether the given String is Palindrome or not.

```
import java.util.Scanner;
public class Palindrome {
static void checkPalindrome(String input) {
//Assuming result to be true
boolean res = true
int length = input.length();
//dividing the length of the string by 2 and
comparing it.
for int i=0; i<= length/2; i++) {
if(input.charAt(i) != input.charAt(length-i-1)) {
res = false;
break;
}
}
System.out.println(input + " is palindrome =
"+res);
```

```
}
public static void main(String[] args) {
Scanner sc = new Scanner(System.in);
System.out.print("Enter your Statement: ");
String str = sc.nextLine();
//function call
checkPalindrome(str);
}
}
```

Write a Java program to calculate Permutation and Combination of 2 numbers.

```
import
public class nprandncr {
//calculating a factorial of a number
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 scan = new Scanner(System.in);
System.out.print("Enter Value of n : ");
n = scan.nextInt();
System.out.print("Enter Value of r : ");
r = scan.nextInt();
// NCR and NPR of a number
```

```
System.out.print("NCR = " +(fact(n)/(fact(n-
r)*fact(r))));
System.out.print("nNPR = " +(fact(n)/(fact(n-
r))));
}
}
```

Write a program in Java to find out Alphabet and Diamond Pattern.

```
import java.util.Scanner;
public class PatternA {
// Java program to print alphabet A pattern
void display(int n)
{
// Outer for loop for number of lines
for (int i = 0; i<=n; i++) {
// Inner for loop for logic execution
for (int j = 0  j<= n / 2
// prints two column lines
if ((j == 0 || j == n / 2) && i != 0
// print first line of alphabet
i == 0  && j != n / 2 ||
// prints middle line
i == n / 2
 System.out.print("*"
else
System.out.print(" ");
}


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

```
{
Scanner sc = new Scanner(System.in);
 atternA a = new PatternA();
 .display(7);
```

Java Program to Count the Number of Bits set to One

```
import java.util.Scanner;
public class Count_One
{
    public static void main(String[] args)
    {
        int n, m, count = 0;
        String x = "";
        Scanner s = new Scanner(System.in);
        System.out.print("Enter the Decimal Number:");
        n = s.nextInt();
        while(n > 0)
        {
            int a = n % 2;
            x = a + x;
            n = n / 2;
        }
        int l = x.length();
        for(int i = 0; i < l; i++)
        {
            if(x.charAt(i) == '1')
            {
                count++;
            }
        }
        System.out.println("No. of 1's are:"+count);
    }
}
```

```
Output:

Enter the Decimal Number:15
No. of 1's are:4
```

Java Program to Find Sum of Digits of a Number using Recursion

```
import java.util.Scanner;
public class Digit_Sum
{
    int sum = 0;
    public static void main(String[] args)
    {
        int n;
        Scanner s = new Scanner(System.in);
        System.out.print("Enter the number:");
        n = s.nextInt();
        Digit_Sum obj = new Digit_Sum();
        int a = obj.add(n);
        System.out.println("Sum:"+a);
    }
    int add(int n)
    {
        sum = n % 10;
        if(n == 0)
        {
            return 0;
        }
        else
        {
             return sum + add(n / 10);
        }

    }
}
```

```
Output:

Enter the number:345
Sum:12
```

Java Program to Find Sum of N Numbers using Recursion

```java
import java.util.Scanner;
public class Sum_Numbers
{
    int sum = 0, j = 0;
    public static void main(String[] args)
    {
        int n;
        Scanner s = new Scanner(System.in);
        System.out.print("Enter the no. of elements you want:");
        n = s.nextInt();
        int a[] = new int[n];
        System.out.print("Enter all the elements you want:");
        for(int i = 0; i < n; i++)
        {
            a[i] = s.nextInt();
        }
        Sum_Numbers obj = new Sum_Numbers();
        int x = obj.add(a, a.length, 0);
        System.out.println("Sum:"+x);
    }
    int add(int a[], int n, int i)
    {
        if(i < n)
        {
            return a[i] + add(a, n, ++i);
        }
        else
        {
            return 0;
        }
    }
}
```

Output:

```
Enter the no. of elements you want:6
Enter all the elements you want:2
3
5
6
7
1
Sum:24
```

Java Program to Find Reverse of a Number using Recursion

```
import static java.lang.StrictMath.pow;
import java.util.Scanner;
public class Reverse_Recursion
{
    public static void main(String[] args)
    {
        int n, count = 0, m;
        Scanner s = new Scanner(System.in);
        System.out.print("Enter the number:");
        n = s.nextInt();
        m = n;
        while(m > 0)
        {
            count++;
            m = m / 10;
        }
        Reverse_Recursion obj = new Reverse_Recursion();
        int a = obj.reverse(n, count);
        System.out.println("Reverse:"+a);
    }
    int reverse(int x, int length)
    {
        if(length == 1)
        {
            return x;
        }
        else
        {
            int b = x % 10;
            x = x / 10;
              return (int) ((b * pow(10, length - 1)) + reverse(x,
--length));
        }
    }
}
```

Output:

```
Enter the number:467
```

Reverse:764

Java Program to Find Product of 2 Numbers using Recursion

```
import java.util.Scanner;
public class Multiply_Recursion
{
    public static void main(String[] args)
    {
        int[] a = new int[2];
        Scanner s = new Scanner(System.in);
        System.out.print("Enter the first number:");
        a[0] = s.nextInt();
        System.out.print("Enter the second number:");
        a[1] = s.nextInt();
        Multiply_Recursion obj = new Multiply_Recursion();
        int mul = obj.multiply(a,0);
        System.out.println("Answer:"+mul);
    }
   int multiply(int x[], int i)
    {
        if(i < 2)
        {
            return x[i] * multiply(x, ++i);
        }
        else
        {
            return 1;
        }
    }
}
```

```
Output:
Enter the first number:6
Enter the second number:4
Answer:24
```

Java Program to Check if a Given Integer is Positive or Negative

```
import java.util.Scanner;
public class Postive_Negative
{
    public static void main(String[] args)
    {
        int n;
        Scanner s = new Scanner(System.in);
        System.out.print("Enter the number you want to check:");
        n = s.nextInt();
        if(n > 0)
        {
                    System.out.println("The given number "+n+" is
Positive");
        }
        else if(n < 0)
        {
                    System.out.println("The given number "+n+" is
Negative");
        }
        else
        {
             System.out.println("The given number "+n+" is neither
Positive nor Negative ");
        }
    }
}
```

```
Output:
Enter the number you want to check:6
The given number 6 is Positive
```

Java Program to Check if a Given Integer is Odd or Even

```
import java.util.Scanner;
public class Odd_Even
{
    public static void main(String[] args)
    {
        int n;
        Scanner s = new Scanner(System.in);
        System.out.print("Enter the number you want to check:");
```

```
        n = s.nextInt();
        if(n % 2 == 0)
        {
            System.out.println("The given number "+n+" is Even ");
        }
        else
        {
            System.out.println("The given number "+n+" is Odd ");
     }
    }
}
```

Output:

```
Enter the number you want to check:15
The given number 15 is Odd
```

Java Program to Find the Biggest of 3 Numbers

```
import java.util.Scanner;
public class Biggest_Number
{
    public static void main(String[] args)
    {
        int x, y, z;
        Scanner s = new Scanner(System.in);
        System.out.print("Enter the first number:");
        x = s.nextInt();
        System.out.print("Enter the second number:");
        y = s.nextInt();
        System.out.print("Enter the third number:");
        z = s.nextInt();
        if(x > y && x > z)
        {
            System.out.println("Largest number is:"+x);
        }
        else if(y > z)
        {
            System.out.println("Largest number is:"+y);
        }
        else
        {
```

```
            System.out.println("Largest number is:"+z);
        }

    }
}
```

```
Output:
Enter the first number:10
Enter the second number:17
Enter the third number:15
Largest number is:17
```

Java Program to Calculate the Sum of Odd & Even Numbers

```
import java.util.Scanner;
public class Sum_Odd_Even
{
    public static void main(String[] args)
    {
        int n, sumE = 0, sumO = 0;
        Scanner s = new Scanner(System.in);
                System.out.print("Enter the number of elements in
array:");
        n = s.nextInt();
        int[] a = new int[n];
        System.out.println("Enter the elements of the array:");
        for(int i = 0; i < n; i++)
        {
            a[i] = s.nextInt();
        }
        for(int i = 0; i < n; i++)
        {
            if(a[i] % 2 == 0)
            {
                sumE = sumE + a[i];
            }
            else
            {
                sumO = sumO + a[i];
            }
        }
        System.out.println("Sum of Even Numbers:"+sumE);
```

```
        System.out.println("Sum of Odd Numbers:"+sumO);
    }
}
```

```
Output:
Enter the number of elements in array:6
Enter the elements of the array:
1
3
2
6
7
9
Sum of Even Numbers:8
Sum of Odd Numbers:20
```

Java Program to Read Two Integers M and N & Swap their Values

```
import java.util.Scanner;
public class Swap_Integers
{
    public static void main(String args[])
    {
        int m, n, temp;
        Scanner s = new Scanner(System.in);
        System.out.print("Enter the first number:");
        m = s.nextInt();
        System.out.print("Enter the second number:");
        n = s.nextInt();
        temp = m;
        m = n;
        n = temp;
        System.out.println("After Swapping");
        System.out.println("First number:"+m);
        System.out.println("Second number:"+n);
    }
}
```

```
Output:
Enter the first number:5
Enter the second number:7
After Swapping
```

```
First number:7
Second number:5
```

Java Program to Reverse a Given Number

```
import java.util.Scanner;
public class Reverse_Number
{
    public static void main(String args[])
    {
        int m, n, sum = 0;
        Scanner s = new Scanner(System.in);
        System.out.print("Enter the number:");
        m = s.nextInt();
        while(m > 0)
        {
            n = m % 10;
            sum = sum * 10 + n;
            m = m / 10;
        }
        System.out.println("Reverse:"+sum);
    }
}
```

```
Output:
Enter the number:567
Reverse:765
```

Java Program to Accept two Integers and Check if they are Equal

```
import java.util.Scanner;
public class Equal_Integer
{
    public static void main(String[] args)
    {
        int m, n;
        Scanner s = new Scanner(System.in);
        System.out.print("Enter the first number:");
        m = s.nextInt();
        System.out.print("Enter the second number:");
        n = s.nextInt();
        if(m == n)
```

```
        {
            System.out.println(m+" and "+n+" are equal ");
        }
        else
        {
            System.out.println(m+" and "+n+" are not equal ");
        }
    }
}
```

```
Output:
Enter the first number:5
Enter the second number:7
5 and 7 are not equal

Enter the first number:6
Enter the second number:6
6 and 6 are equal
```

Java Program to Compute the Sum of Digits in a given Integer

```
import java.util.Scanner;
public class Digit_Sum
{
    public static void main(String args[])
    {
        int m, n, sum = 0;
        Scanner s = new Scanner(System.in);
        System.out.print("Enter the number:");
        m = s.nextInt();
        while(m > 0)
        {
            n = m % 10;
            sum = sum + n;
            m = m / 10;
        }
        System.out.println("Sum of Digits:"+sum);
    }
}
```

```
Output:
Enter the number:456
Sum of Digits:15
```

Java Program to Extract Digits from A Given Integer

```
import java.util.Scanner;
public class Extract_Digits
{
    public static void main(String args[])
    {
        int n, m, a, i = 1, counter = 0;
        Scanner s=new Scanner(System.in);
        System.out.print("Enter any number:");
        n = s.nextInt();
        m = n;
        while(n > 0)
        {
            n = n / 10;
            counter++;
        }
        while(m > 0)
        {
            a = m % 10;
                                System.out.println("Digits  at  position
"+counter+":"+a);
            m = m / 10;
            counter--;
        }
    }
}
```

```
Output:
Enter any number:5678
Digits at position 4:8
Digits at position 3:7
Digits at position 2:6
Digits at position 1:5
```

Java Program to Increment by 1 All the Digits of a given Integer

```
import java.util.Scanner;
public class Increment_Digits
{
```

```
    public static void main(String[] args)
    {
        int n, m = 0, a;
        Scanner s = new Scanner(System.in);
        System.out.print("Enter any number:");
        n = s.nextInt();
        while(n > 0)
        {
            a = n % 10;
            a++;
            m = m * 10 + a;
            n = n / 10;
        }
        n = m;
        m = 0;
        while(n > 0)
        {
            a = n % 10;
            m = m * 10 + a;
            n = n / 10;
        }
        System.out.println("Result:"+m);
    }
}
```

```
Output:
Enter any number:4567
Result:5678
```

Java Program to Convert Integer Values into Binary

```
import java.util.Scanner;
public class Decimal_Binary
{
    public static void main(String[] args)
    {
        int n, m;
        String x = "";
        Scanner s = new Scanner(System.in);
        System.out.print("Enter the Decimal Number:");
        n = s.nextInt();
        while(n > 0)
        {
```

```
            int a = n % 2;
            x = a + x;
            n = n / 2;
        }
        System.out.println(x);
    }
}
```

```
Output:
Enter the Decimal Number:19
10011
```

Java Program to Convert a Given Number of Days in terms of Years, Weeks & Days

```
import java.util.Scanner;
public class Year_Week_Day
{
    public static void main(String args[])
    {
        int m, year, week, day;
        Scanner s = new Scanner(System.in);
        System.out.print("Enter the number of days:");
        m = s.nextInt(); //500
        year = m / 365; //1 year
        m = m % 365; //135
        System.out.println("No. of years:"+year);
        week = m / 7; //19 weeks
        m = m % 7; //2 days
        System.out.println("No. of weeks:"+week);
        day = m;
        System.out.println("No. of days:"+day);
    }
}
```

```
Output:
Enter the number of days:756
No. of years:2
No. of weeks:3
No. of days:5
```

Java Program to Convert Integer Values into Byte, Character, Float

```
import java.util.Scanner;
public class Integer_Conversion
{
   public static void main(String[] args)
    {
        int a;
        byte b;
        char c;
        float d;
        Scanner s = new Scanner(System.in);
        System.out.print("Enter any integer:");
        a = s.nextInt();
        b = (byte) a;
        System.out.println("Conversion into byte:"+b);
        c = (char) a;
        System.out.println("Conversion into char:"+c);
        d = a;
        System.out.println("Conversion into float:"+d);
    }
}
```

```
Output:
Enter any integer:97
Conversion into byte:97
Conversion into char:a
Conversion into float:97.0
```

Java Program to Convert Long Values into Byte

```
import java.util.Scanner;
public class Long_Byte
{
    public static void main(String[] args)
    {
        long a;
        byte b;
        Scanner s = new Scanner(System.in);
        System.out.print("Enter any long value:");
        a = s.nextLong();
        b = (byte) a;
```

```
        System.out.println("Conversion into byte:"+b);
    }
}
```

```
Output:
Enter any long value:12548
Conversion into byte:4
```

Java Program to Check if a Given Character is Vowel or Consonant

```
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;
import java.util.Scanner;
public class Vowel_Consonant
{
    public static void main(String[] args) throws Exception
    {
        char n;
                    BufferedReader bf = new BufferedReader(new
InputStreamReader(System.in));
                System.out.print("Enter the character you want to
check:");
        n = (char) bf.read();
        switch(n)
        {
            case 'a':
                  System.out.println("The given character "+n+" is
vowel");
            break;

            case 'e':
                  System.out.println("The given character "+n+" is
vowel");
            break;

            case 'i':
                  System.out.println("The given character "+n+" is
vowel");
            break;

            case 'o':
                  System.out.println("The given character "+n+" is
vowel");
```

```
            break;

            case 'u':
                  System.out.println("The given character "+n+" is
vowel");
            break;

            default:
                  System.out.println("The given character "+n+" is
consonant");
            break;
        }
    }
}
```

```
Output:
Enter the character you want to check:b
The given character b is consonant
```

Java Program to Check if given Alphabets are Uppercase or Lowercase or Digits

```
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;
public class Alphabet_Check
{
    public static void main(String args[]) throws IOException
    {
        char m;
                    BufferedReader bf = new BufferedReader(new
InputStreamReader(System.in));
        System.out.print("Enter any alphabet:");
        m = (char) bf.read();
        if(m >= 97 && m <= 123)
        {
            System.out.println("Lower Case");
        }
        else if(m >= 65 && m <= 96)
        {
            System.out.println("Upper Case");
        }
        else if(m >= 48 && m <= 57)
```

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

```
Output:
Enter any alphabet:B
Upper Case

Enter any alphabet:z
Lower Case

Enter any alphabet:9
Digit
```

Java Program to Illustrate Use of Binary Literals

```
import java.util.Scanner;
public class Binary_Literal
{
    public static void main(String[] args)
    {
        byte aB = 0b00100001;
        short aS = 0b10100010100;
        int a1 = 0b10110;
        int a2 = 0b101;
        int a3 = 0b1011;
        int aI=a2+a3;
        System.out.println("Byte value:"+aB);
        System.out.println("Short value:"+aS);
        System.out.println("Integer value:"+a1);
        System.out.println("Result:"+aI);
    }
}
```

```
Output:
Byte value:33
Short value:1300
Integer value:22
Result:16
```

Java Program to Illustrate Use of Various Boolean Operators

```
import java.util.Scanner;
public class Boolean_Operators
{
    public static void main(String args[])
    {
        Scanner s = new Scanner(System.in);
        System.out.print("Enter a:");
        boolean a = s.nextBoolean();
        System.out.print("Enter b:");
        boolean b = s.nextBoolean();
        boolean c = a | b;
        boolean d = a & b;
        boolean e = a ^ b;
        boolean f = (!a & b) | (a & !b);
        boolean g = !a;
        System.out.println("a = " + a);
        System.out.println("b = " + b);
        System.out.println("a|b = " + c);
        System.out.println("a&b = " + d);
        System.out.println("a^b = " + e);
        System.out.println("!a&b|a&!b = " + f);
        System.out.println("!a = " + g);
    }
}
Output:
Enter a:true
Enter b:false
a = true
b = false
a|b = true
a&b = false
a^b = true
!a&b|a&!b = true
!a = false
```

Java Program to Find Largest Between Three Numbers Using Ternary Operator

```
import java.util.Scanner;
public class Largest_Ternary
{
    public static void main(String[] args)
    {
        int a, b, c, d;
        Scanner s = new Scanner(System.in);
        System.out.println("Enter all three numbers:");
        a = s.nextInt();
        b = s.nextInt();
        c = s.nextInt();
        d = c > (a > b ? a : b) ? c : ((a > b) ? a : b);
        System.out.println("Largest Number:"+d);
    }
}
```

```
Output:
Enter all three numbers:
5
6
7
Largest Number:7
```

Java Program to Illustrate the Use of Arithmetic Operators

```
import java.util.Scanner;
public class Arithmetic_Operators
{
   public static void main(String args[])
   {
        Scanner s = new Scanner(System.in);
        while(true)
        {
            System.out.println("");
              System.out.println("Enter the two numbers to perform
operations ");
            System.out.print("Enter the first number : ");
            int x = s.nextInt();
            System.out.print("Enter the second number : ");
            int y = s.nextInt();
              System.out.println("Choose the operation you want to
perform ");
```

```
            System.out.println("Choose 1 for ADDITION");
            System.out.println("Choose 2 for SUBTRACTION");
            System.out.println("Choose 3 for MULTIPLICATION");
            System.out.println("Choose 4 for DIVISION");
            System.out.println("Choose 5 for MODULUS");
            System.out.println("Choose 6 for EXIT");
            int n = s.nextInt();
            switch(n)
            {
                case 1:
                int add;
                add = x + y;
                System.out.println("Result : "+add);
                break;

                case 2:
                int sub;
                sub = x - y;
                System.out.println("Result : "+sub);
                break;

                case 3:
                int mul;
                mul = x * y;
                System.out.println("Result : "+mul);
                break;

                case 4:
                float div;
                div = (float) x / y;
                System.out.print("Result : "+div);
                break;

                case 5:
                int mod;
                mod = x % y;
                System.out.println("Result : "+mod);
                break;

                case 6:
                System.exit(0);
            }
        }
    }
}
```

```
Output:
Enter the two numbers to perform operations
Enter the first number : 12
Enter the second number : 45
Choose the operation you want to perform
Choose 1 for ADDITION
Choose 2 for SUBTRACTION
Choose 3 for MULTIPLICATION
Choose 4 for DIVISION
Choose 5 for MODULUS
Choose 6 for EXIT
1
Result : 57

Enter the two numbers to perform operations
Enter the first number : 56
Enter the second number : 42
Choose the operation you want to perform
Choose 1 for ADDITION
Choose 2 for SUBTRACTION
Choose 3 for MULTIPLICATION
Choose 4 for DIVISION
Choose 5 for MODULUS
Choose 6 for EXIT
3
Result : 2352

Enter the two numbers to perform operations
Enter the first number : 45
Enter the second number : 56
Choose the operation you want to perform
Choose 1 for ADDITION
Choose 2 for SUBTRACTION
Choose 3 for MULTIPLICATION
Choose 4 for DIVISION
Choose 5 for MODULUS
Choose 6 for EXIT
6
```

Java Program to Illustrate Use of Pre and Post Increment and Decrement Operators

```
import java.util.Scanner;
public class Increment_Decrement
{
    public static void main(String[] args)
    {
        int a, b, c, d, e;
        Scanner s = new Scanner(System.in);
        System.out.print("Enter any integer a:");
        a = s.nextInt();
        b = ++a;
        System.out.println("Result after Pre Increment a:"+a);
        System.out.println("Result after Pre Increment b:"+b);
        c = a++;
        System.out.println("Result after Pre Increment a:"+a);
        System.out.println("Result after Post Increment c:"+c);
        d = --a;
        System.out.println("Result after Pre Increment a:"+a);
        System.out.println("Result after Pre Decrement d:"+d);
        e = a--;
        System.out.println("Result after Pre Increment a:"+a);
        System.out.println("Result after Post Decrement e:"+e);
    }
}
```

```
Output:
Enter any integer a:12
Result after Pre Increment a:13
Result after Pre Increment b:13
Result after Pre Increment a:14
Result after Post Increment c:13
Result after Pre Increment a:13
Result after Pre Decrement d:13
Result after Pre Increment a:12
Result after Post Decrement e:13
```

Java Program to Illustrate Use of Relational Operators

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

```
        Scanner s= new Scanner(System.in);
        System.out.print("Enter first integer:");
        int a = s.nextInt();
        System.out.print("Enter second integer:");
        int b = s.nextInt();
        System.out.println("a == b : " + (a == b) );
        System.out.println("a != b : " + (a != b) );
        System.out.println("a > b : " + (a > b) );
        System.out.println("a < b : " + (a < b) );
        System.out.println("b >= a : " + (b >= a) );
        System.out.println("b <= a : " + (b <= a) );
    }
}
```

```
Output:
Enter first integer:25
Enter second integer:30
a == b : false
a != b : true
a > b : false
a < b : true
b >= a : true
b <= a : false
```

Java Program to Read a Grade & Display the Equivalent Description

```
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;
public class Grade_Description
{
    public static void main(String[] args) throws IOException
    {
        char a;
                    BufferedReader  bf  =  new  BufferedReader(new
InputStreamReader(System.in));
        System.out.print("Enter grade of the student:");
        a = (char) bf.read();
        if(a == 'A' || a == 'a')
        {
                System.out.println("Student has got marks between
above 80 out of 100.");
        }
        else if(a == 'B' || a == 'b')
```

```
        {
            System.out.println("Student has got marks above 60 but
less than equal to 80 out of 100.");
        }
        else if(a == 'C' || a == 'c')
        {
            System.out.println("Student has got marks above 40 but
less than equal to 60 out of 100.");
        }
        else
        {
           System.out.println("Student has failed");
        }
    }
}
```

```
Output:
Enter grade of the student:B
Student has got marks above 60 but less than equal to 80 out of
100.
```

Pattern 1

```
package Patterns;
public class Star {
public static void main(String[] args) {
int rows = 5;
for (int i = 1; i <= rows; ++i) {  //Outer loop for rows
for (int j = 1; j <= i; ++j) { //Inner loop for Col
System.out.print("* "); //Print *
}
System.out.println(); //New line
}
}
}
```

Output:

```
for(i=1 ; 1<=5;++i)
for(j=1;1<=1;++j)
j=2
for(j=2;2<=1;++j)
```

```
for(i=2;2<=5;++i)
for(j=1;1<=2;++j)
for(j=2;2<=2;++j)
```

Output:

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

Pattern 2:

```
public class Star {
public static void main(String[] args) {
int rows = 5;
for (int i = rows; i >= 1; —i) {  //Outer loop for rows
for (int j = 1; j <= i; ++j) { //Inner loop for Col
System.out.print("* "); //Print *
}
System.out.println(); //New line
}
}
}
```

Output:

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

Pattern 3:

```
package Patterns;
public class Star {
    public static void main(String[] args) {
        int rows = 5;
        for(int i = 1;i<=rows;i++) {  //For Loop for Row
            for(int j = 1; j <= i; ++j) { //For Loop for Col
                for(int k=1;k<i-1;k++)
                {
                        System.out.print(" "); //Prints Space
                }
                System.out.print("* "); //Prints *

            }
            System.out.println(); //Get to newline
        }
    }
}
```

Output:

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

Pattern 4:

```
package Patterns;
import java.util.Scanner;
public class Star {
public static void main(String[] args) {
Scanner sc = new Scanner(System.in); //Input
System.out.println("Enter the number of rows: ");
int rows = sc.nextInt();
```

```
for (int i = 0; i <= rows - 1; i++) { //For Loop for Row
for (int j = 0; j <= i; j++) { //For Loop for Col
System.out.print("*" + " "); //prints * and blank space
}
System.out.println(""); // new line
}
for (int i = rows - 1; i >= 0; i--) { //For Loop for Row
for (int j = 0; j <= i - 1; j++) { //For Loop for Col
System.out.print("*" + " "); //prints * and blank space
}
System.out.println("");// new line
}
sc.close();
}
}
```

Output:

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

Pattern 5:

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

```
package Patterns;
import java.util.Scanner;
public class Star {
public static void main(String[] args) {
Scanner sc = new Scanner(System.in);
System.out.println("Enter the number of rows: ");
int rows = sc.nextInt();
for (int i = 1; i <= rows; i++) {
for (int j = i; j < rows; j++) { //Rows Loop
System.out.print(" "); // Blank Space
}
for (int k = 1; k <= i; k++) { //Cols Loop
System.out.print("* "); // Prints *
}
System.out.println("");
}
for (int i = rows; i >= 1; i--) {
for (int j = i; j <= rows; j++) { //Rows Loop
System.out.print(" ");  // Prints blank spaces
}
for (int k = 1; k < i; k++) { //Col Loop
System.out.print("* ");  // Prints *
}
System.out.println(""); // New Line1
}
sc.close();
}
}
```

Pattern 6:

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

```
package Patterns;
public class Star {
public static void printTriagle(int n) {
for (int i = 0; i < n; i++) {
for (int j = n - i; j > 1; j--) { //Loop for blank space
System.out.print(" "); //Print Space
```

```
}
for (int j = 0; j <= i; j++) { loop for star
System.out.print("* "); //Print Star
}
System.out.println(); //Newline
}
}
public static void main(String args[]) {
int n = 5;
printTriagle(n);
}
}
```

Pattern 7:

```
/*Diagonal 11
        *
      *
    *
  *
*             */
```

```
package Patterns;
public class Star {
public static void main(String[] args) {
int i, j;
for (i = 1; i <= 5; i++) {
for (j = 0; j < 5 - i; j++) {
System.out.print("  "); //Print blank space
}
System.out.print("*\n"); //Print Star and newline
}
}
}
```

Pattern 8:

```
/* V-pattern
*       *
 *     *
  *   *
   * *
    *     */

package Patterns;
public class Star {
static void pattern(int n) {
int i, j;
for (i = n - 1; i >= 0; i--) {
for (j = n - 1; j > i; j--) {
System.out.print(" "); //Print Space
}
System.out.print("*"); //Print star
for (j = 1; j < (i * 2); j++)
System.out.print(" ");//Print space
if (i >= 1)
System.out.print("*");
System.out.print("\n");//Enter newline
}
}
public static void main(String args[]) {
int n = 5;
pattern(n); //Pattern method call
}
}
```

Pattern 9:

```
/*Box 18
**********
*        *
*        *
*        *
*        *
```

```
*         *
*         *
*         *
*         *
**********  */

package Patterns;
public class Star {
static void print_rectangle(int n, int m) {
int i, j;
for (i = 1; i <= n; i++) {
for (j = 1; j <= m; j++) {
if (i == 1 || i == n || j == 1 || j == m) //Logic to print
System.out.print("*"); //True?, print star
else
System.out.print(" "); //True?, print space
}
System.out.println();
}
}
public static void main(String args[]) {
int rows = 10, columns = 10;
print_rectangle(rows, columns); //Method call
}
}
```

Pattern 10:

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

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

```
Scanner sc = new Scanner(System.in); //Input
System.out.println("Number of rows: ");
int rows = sc.nextInt();
for (int i = 1; i <= rows; i++) {
for (int j = 1; j <= i; j++) {
System.out.print(j + " "); //Print j value and space
}
System.out.println();
}
sc.close();
}
}
```

Pattern 11:

```
/*Number Pattern 20 (Floyd's Triangle)
1
2 3
4 5 6
7 8 9 10
11 12 13 14 15  */


package Patterns;
public class Star {
public static void main(String[] args) {
int i, j, k = 1;
for (i = 1; i <= 5; i++) {
for (j = 1; j < i + 1; j++) {
System.out.print(k++ + " "); /Floyd’s triangle logic(prints K value and
increments on every iteration)
}
System.out.println();
}
}
}
```

Pattern 12:

```
/*Number Pattern 21 (Pascal's Triangle)
             1
           1   1
         1   2   1
       1   3   3   1
     1   4   6   4   1
   1   5  10  10   5   1 */
```

```
package Patterns;
public class Star {
public static void main(String[] args) {
int x = 6;
for (int i = 0; i < x; i++) {
int num = 1;
System.out.printf("%" + (x - i) * 2 + "s", ""); //Pascals triangle logic
for (int j = 0; j <= i; j++) {
System.out.printf("%4d", num);
num = num * (i - j) / (j + 1);
}
System.out.println();
}
}
}
```

Pattern 13:

```
10101
01010
10101
01010
10101
```

```
package Patterns;
import java.util.Scanner;
public class Star {
public static void main(String[] args) {
Scanner sc = new Scanner(System.in);
System.out.println("Number of rows: ");
int rows = sc.nextInt(); //Input
for (int i = 1; i <= rows; i++) {
int num;
if (i % 2 == 0) {
num = 0;
for (int j = 1; j <= rows; j++) {
System.out.print(num);
num = (num == 0) ? 1 : 0;
}
} else {
num = 1;
for (int j = 1; j <= rows; j++) {
System.out.print(num);
num = (num == 0) ? 1 : 0;
}
}
System.out.println();
}
sc.close();
}
}
```

Pattern 14:

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

```
package Patterns;
public class Alphabet {
public static void main(String[] args) {
int alphabet = 65; //ASCII value of “A”
for (int i = 0; i <= 5; i++) {
for (int j = 0; j <= i; j++) {
System.out.print((char) (alphabet + j) + " "); //Logic to print Alphabet
pattern
}
System.out.println();
}
}
}
```

Pattern 15:

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

```
package Patterns;
public class Alphabet {
public static void main(String[] args) {
for (int i = 0; i <= 4; i++) {
int alphabet = 65; //ASCII value of “A”
for (int j = 4; j > i; j--) {
System.out.print(" "); //Print Space
}
for (int k = 0; k <= i; k++) {
System.out.print((char) (alphabet + k) + " "); //Print Alphabet
}
System.out.println();
}
}
}
```

Array Programs

Java Program to Find the Largest Two Numbers in a Given Array

```
import java.util.Scanner;
public class largest_and_second
{
     public static void main (String[] args)
     {
          Scanner scn = new Scanner (System.in);
          System.out.print("Enter  no.  of  elements  you  want  in
array:");
               int n = scn.nextInt();

          int array[] = new int[n];
               System.out.println("Enter all the elements:");
          for (int i = 0; i < array.length; i++)
          {
               array[i] = scn.nextInt();
          }

          int largest1, largest2, temp;

          largest1 = array[0];
          largest2 = array[1];

          if (largest1 < largest2)
          {
               temp = largest1;
               largest1 = largest2;
               largest2 = temp;
          }

          for (int i = 2; i < array.length; i++)
          {
               if (array[i] > largest1)
               {
                    largest2 = largest1;
                    largest1 = array[i];
               }
```

```
                else if (array[i] > largest2 && array[i] !=
largest1)
                {
                     largest2 = array[i];
                }
           }

           System.out.println ("The First largest is " + largest1);
           System.out.println ("The Second largest is " +
largest2);

      }
}
```

```
Output:
Enter no. of elements you want in array:5
Enter all the elements:
1
5
4
8
7
The First largest is 8
The Second largest is 7
```

Example 2:

Java Program to Find the Second Largest & Smallest Elements in an Array

```
import java.util.Scanner;
public class SecondLargest_Smallest
{
    public static void main(String[] args)
    {
        int n, temp;
        Scanner s = new Scanner(System.in);
             System.out.print("Enter no. of elements you want in
array(Minimum 2):");
        n = s.nextInt();
        int a[] = new int[n];
        System.out.println("Enter all the elements:");
        for (int i = 0; i < n; i++)
```

```
        {
            a[i] = s.nextInt();
        }
        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.println("Second Largest:"+a[n-2]);
        System.out.println("Smallest:"+a[0]);
    }
}
```

```
Output:
Enter no. of elements you want in array(Minimum 2):8
Enter all the elements:
2
5
1
7
8
6
9
3
Second Largest:8
Smallest:1
```

Example 3:

Java Program to Find the Largest Number in an Array

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

```
        int n, max;
        Scanner s = new Scanner(System.in);
               System.out.print("Enter number of elements in the
array:");
        n = s.nextInt();
        int a[] = new int[n];
        System.out.println("Enter elements of array:");
        for(int i = 0; i < n; i++)
        {
            a[i] = s.nextInt();
        }
        max = a[0];
        for(int i = 0; i < n; i++)
        {
            if(max < a[i])
            {
                max = a[i];
            }
        }
        System.out.println("Maximum value:"+max);
    }
}
```

Output:

```
Enter number of elements in the array:5
Enter elements of array:
4
2
3
6
1
Maximum value:6
```

Example 4:

Java Program to Put Even & Odd Elements of an Array in 2 Separate Arrays

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

```
        int n, j = 0, k = 0;
        Scanner s = new Scanner(System.in);
             System.out.print("Enter no. of elements you want in
array:");
        n = s.nextInt();
        int a[] = new int[n];
        int odd[] = new int[n];
        int even[] = new int[n];
        System.out.println("Enter all the elements:");
        for(int i = 0; i < n; i++)
        {
            a[i] = s.nextInt();
        }
        for(int i = 0; i < n; i++)
        {
            if(a[i] % 2 != 0)
            {
                odd[j] = a[i];
                j++;
            }
            else
            {
                even[k] = a[i];
                k++;
            }
        }
        System.out.print("Odd:");
        if(j > 1)
        {
            for(int i = 0;i < (j-1); i++)
            {
                System.out.print(odd[i]+",");
            }
            System.out.print(odd[j-1]);
        }
        else
        {
            System.out.println("No number");
        }
        System.out.println("");
        System.out.print("Even:");
        if(k > 1)
        {
            for(int i = 0; i < (k-1); i++)
            {
```

```
                System.out.print(even[i]+",");
            }
        System.out.print(even[k-1]);
        }
        else
        {
            System.out.println("No number");
        }
    }
}
```

```
Output:
Enter no. of elements you want in array:8
Enter all the elements:
1
2
3
4
5
6
7
8
Odd:1,3,5,7
Even:2,4,6,8
```

Example 5:

Java Program to Insert an Element in a Specified Position in a Given Array

```
import java.util.Scanner;
public class Insert_Array
{
    public static void main(String[] args)
    {
        int n, pos, x;
        Scanner s = new Scanner(System.in);
             System.out.print("Enter no. of elements you want in
array:");
        n = s.nextInt();
        int a[] = new int[n+1];
        System.out.println("Enter all the elements:");
        for(int i = 0; i < n; i++)
```

```
        {
            a[i] = s.nextInt();
        }
           System.out.print("Enter the position where you want to
insert element:");
        pos = s.nextInt();
        System.out.print("Enter the element you want to insert:");
        x = s.nextInt();
        for(int i = (n-1); i >= (pos-1); i--)
        {
            a[i+1] = a[i];
        }
        a[pos-1] = x;
        System.out.print("After inserting:");
        for(int i = 0; i < n; i++)
        {
            System.out.print(a[i]+",");
        }
        System.out.print(a[n]);
    }
}
```

```
Output:
Enter no. of elements you want in array:6
Enter all the elements:
2
4
6
9
4
5
Enter the position where you want to insert element:3
Enter the element you want to insert:7
After inserting:2,4,7,6,9,4,5
Example 6:
```

Java Program to Delete the Specified Integer from an Array

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

```
    {
        int n, x, flag = 1, loc = 0;
        Scanner s = new Scanner(System.in);
             System.out.print("Enter no. of elements you want in
array:");
        n = s.nextInt();
        int a[] = new int[n];
        System.out.println("Enter all the elements:");
        for (int i = 0; i < n; i++)
        {
            a[i] = s.nextInt();
        }
        System.out.print("Enter the element you want to delete:");
        x = s.nextInt();
        for (int i = 0; i < n; i++)
        {
            if(a[i] == x)
            {
                flag =1;
                loc = i;
                break;
            }
            else
            {
                flag = 0;
            }
        }
        if(flag == 1)
        {
            for(int i = loc+1; i < n; i++)
            {
                a[i-1] = a[i];
            }
            System.out.print("After Deleting:");
            for (int i = 0; i < n-2; i++)
            {
                System.out.print(a[i]+",");
            }
            System.out.print(a[n-2]);
        }
        else
        {
            System.out.println("Element not found");
        }
    } }
```

Output:

```
Enter no. of elements you want in array:5
Enter all the elements:
3
5
8
1
4
Enter the element you want to delete:5
After Deleting:3,8,1,4
```

Example 7:

Program to remove duplicate elements in java

```
import java.util.Scanner;

public class RemoveDuplicateElementFromArray{
        public static void main(String[] args){
                int[] arr_elements = new int[20];
                int i,j;
                Scanner sc = new Scanner(System.in);
                System.out.print("Enter array size: ");
                int arr_size = sc.nextInt();
                System.out.println("Read Array Elements From User :");

                for(i=0;i<arr_size;++i)
                {
                        System.out.print("Enter array elements of index " +i +": ");
                        arr_elements[i] = sc.nextInt();
                }

                /* Display array before removing duplicate element */
                System.out.println("Before removing duplicate element array are :");

                /* Loop for displaying array elements */
                for(i=0;i<arr_size;++i)
                {
                        System.out.println(arr_elements[i]);
                }

                /* Get new line
```

```
            System.out.println();
            System.out.println("After removing duplicate element array are :");
            for(i=0;i<arr_size;++i)
            {
                  for(j=i+1;j<arr_size;){
                        /* if i matches to j
                        then take next _element and matches till end */
                        if(arr_elements[i] == arr_elements[j]){
                              for(int temp = j; temp<arr_size; ++temp){
                                    arr_elements[temp] = arr_elements[temp+1];
                              }
                              arr_size = arr_size-1;
                        }
                        else
                              j++;
                  }
            }

            /* Loop to display array after removing duplicate element */
            for(i=0;i<arr_size;++i)
                  System.out.println(arr_elements[i]);
      }
}
```

Output:

```
Enter array size: 7

Read Array Elements From User :
Enter array elements of index 0: 1
Enter array elements of index 1: 2
Enter array elements of index 2: 3
Enter array elements of index 3: 1
Enter array elements of index 4: 2
Enter array elements of index 5: 3
Enter array elements of index 6: 4

Before removing duplicate element array are :
1
2
3
1
2
3
```

```
4

After removing duplicate element array are :
1
2
3
4
```

Example 8:

Java Program to Sort the Array in an Ascending Order

```
import java.util.Scanner;
public class Ascending _Order
{
    public static void main(String[] args)
    {
        int n, temp;
        Scanner s = new Scanner(System.in);
         System.out.print("Enter no. of elements you want
in array:");
        n = s.nextInt();
        int a[] = new int[n];
        System.out.println("Enter all the elements:");
        for (int i = 0; i < n; i++)
        {
            a[i] = s.nextInt();
        }
        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("Ascending Order:");
        for (int i = 0; i < n - 1; i++)
        {
            System.out.print(a[i] + ",");
        }
        System.out.print(a[n - 1]);
    }
}
```

Output:

```
Enter no. of elements you want in array:5
Enter all the elements:
4
3
2
6
1
Ascending Order:1,2,3,4,6
```

Example 9:

Java Program to Sort the Array in Descending Order

```
import java.util.Scanner;
public class Descending_Order
{
    public static void main(String[] args)
    {
        int n, temp;
        Scanner s = new Scanner(System.in);
             System.out.print("Enter no. of elements you want in
array:");
        n = s.nextInt();
        int a[] = new int[n];
```

```
        System.out.println("Enter all the elements:");
        for (int i = 0; i < n; i++)
        {
            a[i] = s.nextInt();
        }
        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("Descending Order:");
        for (int i = 0; i < n - 1; i++)
        {
            System.out.print(a[i] + ",");
        }
        System.out.print(a[n - 1]);
    }
}
```

Output:

```
Enter no. of elements you want in array:5
Enter all the elements:
2
3
5
1
4
Descending Order:5,4,3,2,1
```

Example 10:

Java Program to Sort Names in an Alphabetical Order

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

```
public class Alphabetical_Order
{
    public static void main(String[] args)
    {
        int n;
        String temp;
        Scanner s = new Scanner(System.in);
             System.out.print("Enter number of names you want to
enter:");
        n = s.nextInt();
        String names[] = new String[n];
        Scanner s1 = new Scanner(System.in);
        System.out.println("Enter all the names:");
        for(int i = 0; i < n; i++)
        {
            names[i] = s1.nextLine();
        }
        for (int i = 0; i < n; i++)
        {
            for (int j = i + 1; j < n; j++)
            {
                if (names[i].compareTo(names[j])>0)
                {
                    temp = names[i];
                    names[i] = names[j];
                    names[j] = temp;
                }
            }
        }
        System.out.print("Names in Sorted Order:");
        for (int i = 0; i < n - 1; i++)
        {
            System.out.print(names[i] + ",");
        }
        System.out.print(names[n - 1]);
    }
}
```

```
Output:
Enter number of names you want to enter:5
Enter all the names:
bryan
adam
rock
chris
```

```
scott
Names in Sorted Order:adam,bryan,chris,rock,scott

Example 11:
```

Java Program to Split an Array from Specified Position

```
import java.util.Scanner;
public class Split
{
    public static void main(String[] args)
    {
        int n, x, flag = 1, loc = 0, k = 0,j = 0;
        Scanner s = new Scanner(System.in);
             System.out.print("Enter no. of elements you want in
array:");
        n = s.nextInt();
        int a[] = new int[n];
        int b[] = new int[n];
        int c[] = new int[n];
        System.out.println("Enter all the elements:");
        for (int i = 0; i < n; i++)
        {
            a[i] = s.nextInt();
        }
          System.out.print("Enter the position from where you want
to split:");
        loc = s.nextInt();
        for(int i = 0; i < loc; i++)
        {
            b[k] = a[i];
            k++;
        }
        for(int i = loc; i < n; i++)
        {
            c[j] = a[i];
            j++;
        }
        System.out.print("First array:");
        for(int i = 0;i < k; i++)
        {
            System.out.print(b[i]+" ");
```

```
        }
        System.out.println("");
        System.out.print("Second array:");
        for(int i = 0; i < j; i++)
        {
            System.out.print(c[i]+" ");
        }
    }
}
```

```
Output:
Enter no. of elements you want in array:8
Enter all the elements:
2
3
4
7
1
9
11
6
Enter the position from where you want to split:4
First array:2 3 4 7
Second array:1 9 11 6
```

Example 12:

Java Program to Calculate Sum & Average of an Array

```
import java.util.Scanner;
public class Sum_Average
{
    public static void main(String[] args)
    {
        int n, sum = 0;
        float average;
        Scanner s = new Scanner(System.in);
             System.out.print("Enter no. of elements you want in
array:");
        n = s.nextInt();
        int a[] = new int[n];
        System.out.println("Enter all the elements:");
```

```
        for(int i = 0; i < n ; i++)
        {
            a[i] = s.nextInt();
            sum = sum + a[i];
        }
        System.out.println("Sum:"+sum);
        average = (float)sum / n;
        System.out.println("Average:"+average);
    }
}
```

```
Output:
Enter no. of elements you want in array:5
Enter all the elements:
4
7
6
9
3
Sum:29
Average:5.8
```

Example 13:

Java Program to Increment Every Element of the Array by One & Print Incremented Array

```
import java.util.Scanner;
public class Increment_Array
{
    public static void main(String[] args)
    {
        int n, i = 0;
        Scanner s = new Scanner(System.in);
             System.out.print("Enter no. of elements you want in
array:");
        n = s.nextInt();
        int a[] = new int[n];
        System.out.println("Enter all the elements:");
        for(i = 0; i < n; i++)
        {
```

```
            a[i] = s.nextInt();
            a[i]++;
        }
            System.out.print("Elements of array after increment by
1:");
        for(i = 0; i < n - 1; i++)
        {
            System.out.print(a[i]+",");
        }
        System.out.print(a[n-1]);
    }
}
```

```
Output:
Enter no. of elements you want in array:5
Enter all the elements:
2
5
8
6
9
Elements of array after increment by 1:3,6,9,7,10
```

Example 14:

Java Program to Find the Number of Non-Repeated Elements in an Array

```
import java.io.BufferedReader;
import java.io.InputStreamReader;

public class NoDuplicates {
    // Function to print elements with no duplicates
    static void printElementsWithNoDuplicates(int[] array){
        int i,j;
        int count;
        int x = 0;
        boolean[] flag = new boolean[array.length];
        for(i=0; i<array.length; i++){
            if(!flag[i]){
                count  = 1;
                for(j=i+1; j<array.length;j++){
                    if(array[j] == array[i])
```

```
                    {
                        count++;
                        flag[j] = true;
                    }
                }
                if(count == 1){
                    System.out.println(array[i]);
                    x++;
                }
            }
        }
        if(x==0){
            System.out.println("All elements are repeated");
        }
    }
    // Function to read input
    public static void main(String[] args) {
                    BufferedReader br = new BufferedReader(new
InputStreamReader(System.in));
        int size;
        System.out.println("Enter the size of the array");
        try {
            size = Integer.parseInt(br.readLine());
        } catch (Exception e) {
            System.out.println("Invalid Input");
            return;
        }
        int[] array = new int[size];
        System.out.println("Enter array elements");
        int i;
        for (i = 0; i < array.length; i++) {
            try {
                array[i] = Integer.parseInt(br.readLine());
            } catch (Exception e) {
                System.out.println("An error Occurred");
            }
        }
        System.out.println("The elements are ");
        printElementsWithNoDuplicates(array);
    }
}
```

Output:

Case 1 (Simple Test Case - some elements are unique):

```
Enter the size of the array
6
Enter array elements
1
2
3
4
5
5
The elements are
1
2
3
4
```

```
Case 2 (Simple Test Case - all elements have a duplicate):

Enter the size of the array
6
Enter array elements
1
2
3
3
2
1
The elements are
All elements are repeated
```

```
Example 15:
```

Java Program to Identify Missing Numbers in a Given Array

```
public class Missing
{
    static int getMissingNo (int a[], int n)
    {
        int i, total;
          total  = (n + 1) * (n + 2) / 2;
          for ( i = 0; i < n; i++)
          total -= a[i];
          return total;
    }
```

```
    public static void main(String... s)
    {
          int a[ ] = {1, 2, 4, 5, 6};
          int miss = getMissingNo(a, 5);
          System.out.println("The number missing is :"+miss);
    }
}
```

Output:

```
The number missing is :3
```

Example 16:

Java Program to Find 2 Elements in the Array such that Difference between them is Largest

```
import java.util.Scanner;
public class Largest_Difference
{
    public static void main(String[] args)
    {
        int n, x, count = 0, i = 0, temp = 0;
        Scanner s = new Scanner(System.in);
             System.out.print("Enter no. of elements you want in
array:");
        n = s.nextInt();
        int a[] = new int[n];
        System.out.println("Enter all the elements:");
        for(i = 0; i < n; i++)
        {
            a[i] = s.nextInt();
        }
        int diff, greatest_diff;
        greatest_diff = 0;
        int a1 = 0, a2 = 0;
        for(i = 0; i < n; i++)
        {
            for(int j = i + 1; j < n; j++)
            {
                diff = Math.abs(a[i] - a[j]);
```

```
                if(diff > greatest_diff)
                {
                    greatest_diff = diff;
                    a1 = i;
                    a2 = j;
                }
            }
        }
        System.out.println("Greatest Difference:"+greatest_diff);
                    System.out.println("Two elements with largest
difference:"+a[a1]+" and "+a[a2]);
    }
}
```

```
Output:

Enter no. of elements you want in array:7
Enter all the elements:
-2
4
5
6
2
7
-3
Greatest Difference:10
Two elements with largest difference:7 and -3
```

```
Example 17:
```

Java Program to Count the Number of Occurrence of an Element in an Array

```
import java.util.Scanner;
public class Count_Occurrence
{
    public static void main(String[] args)
    {
        int n, x, count = 0, i = 0;
        Scanner s = new Scanner(System.in);
              System.out.print("Enter no. of elements you want in
array:");
        n = s.nextInt();
```

```
        int a[] = new int[n];
        System.out.println("Enter all the elements:");
        for(i = 0; i < n; i++)
        {
            a[i] = s.nextInt();
        }
          System.out.print("Enter the element of which you want to
count number of occurrences:");
        x = s.nextInt();
        for(i = 0; i < n; i++)
        {
            if(a[i] == x)
            {
                count++;
            }
        }
                 System.out.println("Number of Occurrence of the
Element:"+count);
    }
}
```

```
Output:
Enter no. of elements you want in array:5
Enter all the elements:
2
3
3
4
3
Enter the element of which you want to count number of
occurrences:3
Number of Occurrence of the Element:3
```

Example 18:

Java Program to Accept Array Elements and Calculate Sum

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

```
        int n, sum = 0;
        Scanner s = new Scanner(System.in);
             System.out.print("Enter no. of elements you want in
array:");
        n = s.nextInt();
        int a[] = new int[n];
        System.out.println("Enter all the elements:");
        for(int i = 0; i < n; i++)
        {
            a[i] = s.nextInt();
            sum = sum + a[i];
        }
        System.out.println("Sum:"+sum);
    }
}
```

```
Output:
Enter no. of elements you want in array:5
Enter all the elements:
1
2
3
4
5
Sum:15
```

Example 19:

Java Program to Segregate 0s on Left Side & 1s on Right Side of the Array

```
public class ArraySegregate
{
    public static void main(String... a)
    {
        int array[] = { 0, 1, 0, 1, 1, 0 };
        segregate0and1(array, 6);
        for(int i = 0 ; i < 6; i++)
        {
            System.out.print(array[i]+"\t");
        }
    }
    static void segregate0and1(int array[], int size)
```

```
    {
        int left = 0, right = size-1;
     while (left < right)
     {
            /* Increment left index while we see 0 at left */
            while (array[left] == 0 && left < right)
            left++;
            /* Decrement right index while we see 1 at right */
            while (array[right] == 1 && left < right)
            right--;
             /* If left is smaller than right then there is a 1 at
left and a 0 at right.  Exchange it */
            if (left < right)
            {
                array[left] = 0;
                array[right] = 1;
                left++;
                right--;
            }
        }
    }
}
```

```
Output:

0        0        0        1        1        1

Example 20:
```

Java Program to Find the Number of Elements in an Array

```
public class Length
{
    public static void main(String[] args)
    {
        int a[] = {1,2,3,4,5};
        int count = 0, i = 0, n;
        try
        {
            while(a[i] != 'a')
            {
                count++;
```

```
                i++;
            }
        }
        catch(Exception e)
        {
                        System.out.println("Number of elements in
array:"+count);
        }
        n = a.length;
             System.out.println("Number of elements(Using inbuilt
method named length):"+n);
    }
}
```

Output:

```
Number of elements in array:5
Number of elements(Using inbuilt method named length):5
```

Example 21:

Java Program to Display Transpose Matrix

```
import java.util.Scanner;
public class Transpose
{
    public static void main(String args[])
    {
     int i, j;
     System.out.println("Enter total rows and columns: ");
     Scanner s = new Scanner(System.in);
     int row = s.nextInt();
     int column = s.nextInt();
     int array[][] = new int[row][column];
     System.out.println("Enter matrix:");
     for(i = 0; i < row; i++)
     {
         for(j = 0; j < column; j++)
              {
          array[i][j] = s.nextInt();
          System.out.print(" ");
            }
```

```
     }
     System.out.println("The above matrix before Transpose is ");
     for(i = 0; i < row; i++)
     {
               for(j = 0; j < column; j++)
              {
                 System.out.print(array[i][j]+" ");
              }
              System.out.println(" ");
         }
     System.out.println("The above matrix after Transpose is ");
     for(i = 0; i < column; i++)
     {
               for(j = 0; j < row; j++)
              {
                  System.out.print(array[j][i]+" ");
              }
              System.out.println(" ");
         }
    }
}
```

Output:

```
Enter total rows and columns:
3 3
Enter matrix:
1
2
3
4
5
6
7
8
9
The above matrix before Transpose is
1 2 3
4 5 6
7 8 9
The above matrix after Transpose is
1 4 7
2 5 8
3 6 9
```

Example 22:

Java Program to Find the Trace & Normal of a given Matrix

```
import java.util.*;
public class Trace
{
    public static void main(String args[])
    {
        int array[][]=new int[10][10];
        int i, j;
        double sum = 0, square = 0, result = 0;
     System.out.println("Enter total rows and columns: ");
     Scanner s = new Scanner(System.in);
     int row = s.nextInt();
     int column = s.nextInt();
     System.out.println("Enter matrix:");
     for(i = 0; i < row; i++)
     {
         for(j = 0; j < column; j++)
              {
                  array[i][j] = s.nextInt();
                System.out.print(" ");
              }
        }
     System.out.println("The entered Matrix is :");
     for(i = 0; i < row; i++)
        {
              for(j = 0; j < column; j++)
             {
          System.out.print(array[i][j]+" ");
            }
            System.out.println(" ");
     }
        System.out.println("The Trace of the above matrix is ");
     for(i = 0; i < row; i++)
     {
         for(j = 0; j < column; j++)
              {
                if(i == j)
                {
                          sum = sum + (array[i][j]);
```

```
                    }
                }
            }
            System.out.println(sum);
            System.out.println("The Normal of the above matrix is ");
        for(i = 0; i < row; i++)
        {
            for(j = 0; j < column; j++)
                {
                    square = square + (array[i][j])*(array[i][j]);
                }
        }
            result = Math.sqrt(square);
            System.out.println(result);
    }
}
```

Output:

```
Enter total rows and columns:
3
3
Enter matrix:
1
2
3
4
5
6
7
8
9
The entered Matrix is :
1 2 3
4 5 6
7 8 9
The Trace of the above matrix is
15.0
The Normal of the above matrix is
16.881943016134134
```

Example 23:

Java Program to Display Upper/Lower Triangle of a Matrix

Given a square matrix, print it's upper and lower triangle.
Example:
Matrix:
0 0 1 1
0 1 1 1
0 0 0 1
0 0 0 0
Output:
Upper Triangle
0 0 1 1
0 1 1
0 0
0
Lower Triangle
1
0 1
0 0 0

```
import java.io.BufferedReader;
import java.io.InputStreamReader;

public class UpperAndLowerTriangle {
    // Function to display upper and lower triangle
    static void displayUpperAndLowerTriangle(int[][] matrix){
        int order = matrix.length;
        int i,j;
        for(i=0; i<order; i++){
            for(j=0; j<order;j++){
                if((i+j) <order)
                    System.out.print(matrix[i][j] + "\t");
            }
            System.out.println();
        }
        for(i=0; i<order; i++){
            for(j=0; j<order;j++){
                if((i+j) >=order)
                    System.out.print(matrix[i][j] + "\t");
```

```
            }
            System.out.println();
        }
    }
    // Function to read user input
    public static void main(String[] args) {
                    BufferedReader  br  =  new  BufferedReader(new
InputStreamReader(System.in));
        int order;
        System.out.println("Enter the order of the matrix");
        try{
            order = Integer.parseInt(br.readLine());
        }
        catch(Exception e){
            System.out.println("An error occurred");
            return;
        }
        int[][] matrix = new int[order][order];
        System.out.println("Enter matrix elements");
        int i,j;
        for(i=0; i<order; i++){
            for(j=0; j<order; j++){
                try{
                                                        matrix[i][j]  =
Integer.parseInt(br.readLine());
                }
                catch(Exception e){
                    System.out.println("An error occurred");
                    return;
                }
            }
        }
        System.out.println("Tha matrix is");
        for(i=0; i<order; i++){
            for(j=0; j<order; j++){
                System.out.print(matrix[i][j] + "\t");
            }
            System.out.println();
        }
        System.out.println("The upper and lower triangle is");
        displayUpperAndLowerTriangle(matrix);
    }
}
```

Example 24:

Java Program to Display Lower Triangular Matrix

```
import java.util.Scanner;
public class Lower_Matrix
{
public static void main(String args[])
    {
        int a[][] = new int[5][5];
        System.out.println("Enter the order of your Matrics ");
        System.out.println("Enter the rows:");
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();
        System.out.println("Enter the columns:");
        Scanner s = new Scanner(System.in);
        int m = s.nextInt();
        if(n == m)
         {
             System.out.println("Enter your elements:");
              for(int i = 0; i < n; i++)
            {
                for(int j = 0; j < n; j++)
                {

                    Scanner ss = new Scanner(System.in);
                    a[i][j] = ss.nextInt();
                    System.out.print(" ");
                 }
            }
              System.out.println("You have entered:");
              for(int i=0; i<n; i++)
               {
                for(int j=0;j<n;j++)
                {
                     System.out.print(a[i][j] + " ");
                 }
                System.out.println("");
               }
                   System.out.println("The Lower Triangular Matrix
is:");
              for(int i=0;i<n;i++)
               {
```

```
                for(int j=0;j<n;j++)
                 {
                  if(i>=j)
                   {
                     System.out.print(a[i][j] +" ");
                   }
                else
                {
                    System.out.print("0"+" ");
                }
              }
             System.out.println("");
            }
         }
          else
         {
             System.out.println("you have entered improper order");
         }
    }
}
```

Output:

```
Enter the order of your Matrics
Enter the rows:
3
Enter the columns:
3
Enter your elements:
1
2
3
4
5
6
7
8
9
You have entered:
1 2 3
4 5 6
7 8 9
The Lower Triangular Matrix is:
1 0 0
4 5 0
```

```
7 8 9
```

Addition of two matrix in Java

Matrix Addition:

```
import java.util.Scanner;

class AddTwoMatrix
{
  public static void main(String args[])
  {
    int m, n, c, d;
    Scanner in = new Scanner(System.in);
    System.out.println("Enter the number of rows and
columns of matrix");
    m = in.nextInt();
    n = in.nextInt();
    int first[][] = new int[m][n];
    int second[][] = new int[m][n];
    int sum[][] = new int[m][n];
    System.out.println("Enter the elements of first
matrix");
    for (c = 0; c < m; c++)
      for (d = 0; d < n; d++)
        first[c][d] = in.nextInt();
    System.out.println("Enter the elements of second
matrix");
    for (c = 0 ; c < m; c++)
      for (d = 0 ; d < n; d++)
        second[c][d] = in.nextInt();
    for (c = 0; c < m; c++)
      for (d = 0; d < n; d++)
        sum[c][d] = first[c][d] + second[c][d];  //
replace '+' with '-' to subtract matrices
    System.out.println("Sum of the matrices:");
    for (c = 0; c < m; c++)
```

```
    {
      for (d = 0; d < n; d++)
        System.out.print(sum[c][d] + "\t");
      System.out.println();
    }
  }
}
```

Example 25:

Java Program to Interchange any two Rows & Columns in the given Matrix

```
import java.util.Scanner;
public class Interchange
{
    public static void main(String[] args)
    {
        int p, q, n, x , y, temp = 0, k = 0;
        Scanner s = new Scanner(System.in);
        System.out.print("Enter number of rows in matrix:");
        p = s.nextInt();
        System.out.print("Enter number of columns in matrix:");
        q = s.nextInt();
        int a[][] = new int[p][q];
        System.out.println("Enter all the elements of matrix:");
        for (int i = 0; i < p; i++)
        {
            for (int j = 0; j < q; j++)
            {
                a[i][j] = s.nextInt();
            }
        }
        System.out.println("Given Matrix:");
        for (int i = 0; i < p; i++)
        {
            for (int j = 0; j < q; j++)
            {
                System.out.print(a[i][j] + " ");
            }
            System.out.println("");
        }
        while (true)
```

```
        {
            System.out.println("Enter 1 to interchange rows");
            System.out.println("Enter 2 to interchange columns");
            System.out.println("Enter 3 to Exit");
            n=s.nextInt();
            switch (n)
            {
                case 1:
                System.out.println("Enter the row numbers:");
                x = s.nextInt();
                y = s.nextInt();
                for(int i = 0; i < p; i++)
                {
                    temp = a[(x-1)][i];
                    a[x-1][i] = a[y-1][i];
                    a[y-1][i] = temp;
                }
                     System.out.println("Matrix after interchanging
rows:"+x +" and "+y);
                for (int i = 0; i < p; i++)
                {
                    for (int j = 0; j < q; j++)
                    {
                        System.out.print(a[i][j] + " ");
                    }
                System.out.println("");
                }
                break;
                case 2:
                System.out.println("Enter the column numbers:");
                x = s.nextInt();
                y = s.nextInt();
                for(int i = 0; i < p; i++)
                {
                    temp = a[i][(x-1)];
                    a[i][x-1] = a[i][(y-1)];
                    a[i][y-1] = temp;
                }
                     System.out.println("Matrix after interchanging
columns:"+x +" and "+y);
                for (int i = 0; i < p; i++)
                {
                    for (int j = 0; j < q; j++)
                    {
                        System.out.print(a[i][j] + " ");
```

```
                    }
                System.out.println("");
                }
                break;
               case 3:
                System.exit(0);
            }
        }
    }
}
```

Output:

```
Enter number of rows in matrix:3
Enter number of columns in matrix:3
Enter all the elements of matrix:
1
2
3
4
5
6
7
8
9
Given Matrix:
1 2 3
4 5 6
7 8 9
Enter 1 to interchange rows
Enter 2 to interchange columns
Enter 3 to Exit
1
Enter the row numbers:
2
3
Matrix after interchanging rows:2 and 3
1 2 3
7 8 9
4 5 6
Enter 1 to interchange rows
Enter 2 to interchange columns
Enter 3 to Exit
2
Enter the column numbers:
```

```
1
2
Matrix after interchanging columns:1 and 2
2 1 3
8 7 9
5 4 6
Enter 1 to interchange rows
Enter 2 to interchange columns
Enter 3 to Exit
3
```

Example 26:

Java Program to Find the Frequency of Odd & Even Numbers in the given Matrix

```
import java.util.Scanner;
public class Frequency
{
    public static void main(String[] args)
    {
        int p, q, count1 = 0, count2 = 0;
        Scanner s = new Scanner(System.in);
        System.out.print("Enter number of rows in matrix:");
        p = s.nextInt();
        System.out.print("Enter number of columns in matrix:");
        q = s.nextInt();
        int a[][] = new int[p][q];
        System.out.println("Enter all the elements of matrix:");
        for (int i = 0; i < p; i++)
        {
            for (int j = 0; j < q; j++)
            {
                a[i][j] = s.nextInt();
            }
        }
        System.out.println("Given Matrix:");
        for (int i = 0; i < p; i++)
        {
            for (int j = 0; j < q; j++)
            {
                System.out.print(a[i][j] + " ");
            }
```

```
            System.out.println("");
        }
        for (int i = 0; i < p; i++)
        {
            for (int j = 0; j < q; j++)
            {
                if((a[i][j] % 2) == 0)
                {
                    count1++;
                }
                else
                {
                    count2++;
                }
            }
        }
        System.out.println("Even number frequency:"+count1);
        System.out.println("Odd number frequency:"+count2);
    }
}
```

Output:

```
Enter number of rows in matrix:3
Enter number of columns in matrix:3
Enter all the elements of matrix:
1
2
3
4
5
6
7
8
9
Given Matrix:
1 2 3
4 5 6
7 8 9
Even number frequency:4
Odd number frequency:5
```

Example 27:

Java Program to Accept a Matrix of Order MxN & Interchange the Diagonals

```
import java.util.Scanner;
public class Interchange_Diagonals
{
    public static void main(String[] args)
    {
        int p, q, temp = 0;
        Scanner s = new Scanner(System.in);
        System.out.print("Enter number of rows in matrix:");
        p = s.nextInt();
        System.out.print("Enter number of columns in matrix:");
        q = s.nextInt();
        if (p == q)
        {
            int a[][] = new int[p][q];
                    System.out.println("Enter all the elements of
matrix:");
            for (int i = 0; i < p; i++)
            {
                for (int j = 0; j < q; j++)
                {
                    a[i][j] = s.nextInt();
                }
            }
            System.out.println("Given Matrix:");
            for (int i = 0; i < p; i++)
            {
                for (int j = 0; j < q; j++)
                {
                    System.out.print(a[i][j] + " ");
                }
                System.out.println("");
             }
            for(int j = 0; j < q; j++)
            {
                temp = a[j][j];
                a[j][j] = a[j][q-1-j];
                a[j][q-1-j] = temp;
            }
                    System.out.println("Matrix after interchanging
diagonals");
```

```
            for (int i = 0; i < p; i++)
            {
                for (int j = 0; j < q; j++)
                {
                    System.out.print(a[i][j] + " ");
                }
                System.out.println("");
            }
        }
        else
        {
            System.out.println("Rows not equal to column");
        }
    }
}
```

Output:

```
Enter number of rows in matrix:3
Enter number of columns in matrix:3
Enter all the elements of matrix:
1
2
3
4
5
6
7
8
9
Given Matrix:
1 2 3
4 5 6
7 8 9
Matrix after interchanging diagonals
3 2 1
4 5 6
9 8 7
```

Example 28:

Java Program to Determine if a given Matrix is a Sparse Matrix

This is a Java Program to Determine if a given Matrix is a Sparse Matrix. If the number of zero elements are more than the non-zero elements of the matrix then it is known as Sparse Matrix

```
import java.util.Scanner;
public class Sparse
{
    public static void main(String args[])
    {
     int i, j, zero = 0, count = 0;
     int array[][] = new int[10][10];
     System.out.println("Enter total rows and columns: ");
     Scanner s = new Scanner(System.in);
     int row = s.nextInt();
     int column = s.nextInt();
     System.out.println("Enter matrix:");
        for(i = 0; i < row; i++)
        {
            for(j = 0; j < column; j++)
         {
                array[i][j] = s.nextInt();
                System.out.print(" ");
          }
        }
        for(i = 0; i < row; i++)
        {
         for(j = 0; j < column; j++)
         {
               if(array[i][j] == 0)
          {
              zero++;
          }
          else
               {
                     count++;
          }
         }
       }
       if(zero>count)
       {
           System.out.println("the matrix is sparse matrix");
       }
       else
```

```
        {
                    System.out.println("the matrix is not a sparse
matrix");
        }
    }
}
```

```
Output:

Enter total rows and columns:
3
3
Enter matrix:
1
0
5
0
0
8
0
0
0
the matrix is sparse matrix
```

Program to convert char to String

```
class CharToStringDemo
{
   public static void main(String args[])
   {
      // Method 1: Using toString() method
      char ch = 'a';
      String str = Character.toString(ch);
      System.out.println("String is: "+str);

      // Method 2: Using valueOf() method
      String str2 = String.valueOf(ch);
      System.out.println("String is: "+str2);
   }
}
```

```
Output:
String is: a
String is: a
```

Converting String to Char

```
class StringToCharDemo
{
   public static void main(String args[])
   {
      // Using charAt() method
      String str = "Hello";
      for(int i=0; i<str.length();i++){
        char ch = str.charAt(i);
        System.out.println("Character at "+i+" Position:
"+ch);
      }
   }
}

Output:

Character at 0 Position: H
Character at 1 Position: e
Character at 2 Position: l
Character at 3 Position: l
Character at 4 Position: o
```

Java Program to find duplicate Characters in a String

```
import java.util.HashMap;
import java.util.Map;
import java.util.Set;

public class Details {

  public void countDupChars(String str){

    //Create a HashMap
    Map<Character, Integer> map = new HashMap<Character,
Integer>();

    //Convert the String to char array
    char[] chars = str.toCharArray();
```

```
    /* logic: char are inserted as keys and their count
     * as values. If map contains the char already then
     * increase the value by 1
     */
    for(Character ch:chars){
      if(map.containsKey(ch)){
         map.put(ch, map.get(ch)+1);
      } else {
         map.put(ch, 1);
        }
    }

    //Obtaining set of keys
    Set<Character> keys = map.keySet();

    /* Display count of chars if it is
     * greater than 1. All duplicate chars would be
     * having value greater than 1.
     */
    for(Character ch:keys){
      if(map.get(ch) > 1){
        System.out.println("Char "+ch+" "+map.get(ch));
      }
    }
  }

  public static void main(String a[]){
    Details obj = new Details();
    System.out.println("String: BeginnersBook.com");
    System.out.println("------------------------");
    obj.countDupChars("BeginnersBook.com");

    System.out.println("\nString: ChaitanyaSingh");
    System.out.println("------------------------");
    obj.countDupChars("ChaitanyaSingh");

    System.out.println("\nString: #@$@!#$%!!%@");
    System.out.println("------------------------");
    obj.countDupChars("#@$@!#$%!!%@");
  }
}

Output:
String: BeginnersBook.com
------------------------
Char e 2
Char B 2
```

```
Char n 2
Char o 3

String: ChaitanyaSingh
-----------------------
Char a 3
Char n 2
Char h 2
Char i 2

String: #@$@!#$%!!%@
-----------------------
Char # 2
Char ! 3
Char @ 3
Char $ 2
Char % 2
```

java program to check palindrome string using Stack, Queue, for or while loop

Program 1: Palindrome check Using Stack

```
import java.util.Stack;
import java.util.Scanner;
class PalindromeTest {

    public static void main(String[] args) {

     System.out.print("Enter any string:");
        Scanner in=new Scanner(System.in);
        String inputString = in.nextLine();
        Stack stack = new Stack();

        for (int i = 0; i < inputString.length(); i++) {
            stack.push(inputString.charAt(i));
        }

        String reverseString = "";
```

```
        while (!stack.isEmpty()) {
            reverseString = reverseString+stack.pop();
        }

        if (inputString.equals(reverseString))
            System.out.println("The input String is a
palindrome.");
        else
            System.out.println("The input String is not a
palindrome.");

    }
}
```

```
Output:

Enter any string:abccba
The input String is a palindrome.
```

Program 2: Palindrome check Using Queue

```
import java.util.Queue;
import java.util.Scanner;
import java.util.LinkedList;
class PalindromeTest {

    public static void main(String[] args) {

     System.out.print("Enter any string:");
        Scanner in=new Scanner(System.in);
        String inputString = in.nextLine();
        Queue queue = new LinkedList();

        for (int i = inputString.length()-1; i >=0; i--) {
            queue.add(inputString.charAt(i));
        }

        String reverseString = "";

        while (!queue.isEmpty()) {
            reverseString = reverseString+queue.remove();
        }
        if (inputString.equals(reverseString))
            System.out.println("The input String is a
palindrome.");
        else
```

```
            System.out.println("The input String is not a
palindrome.");

    }
}
```

```
Output:

Enter any string:xyzzyx
xyzzyx
The input String is a palindrome.
```

Program 3: Using for loop/While loop and String function charAt

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

      System.out.println("Enter a string to check if it is a
palindrome:");
      String inputString = scanner.nextLine();

      int length = inputString.length();

      for ( int i = length - 1 ; i >= 0 ; i-- )
         reverseString = reverseString +
inputString.charAt(i);

      if (inputString.equals(reverseString))
         System.out.println("Input string is a palindrome.");
      else
         System.out.println("Input string is not a
palindrome.");

   }
}
```

```
Output:

Enter a string to check if it is a palindrome:
aabbaa
Input string is a palindrome.
```

Java Program to reverse words in a String

Example: Program to reverse every word in a String using methods

```
public class Example
{
   public void reverseWordInMyString(String str)
   {
     /* The split() method of String class splits
      * a string in several strings based on the
      * delimiter passed as an argument to it
      */
     String[] words = str.split(" ");
     String reversedString = "";
     for (int i = 0; i < words.length; i++)
         {
            String word = words[i];
            String reverseWord = "";
            for (int j = word.length()-1; j >= 0; j--)
        {
          /* The charAt() function returns the character
           * at the given position in a string
           */
          reverseWord = reverseWord + word.charAt(j);
        }
        reversedString = reversedString + reverseWord + " ";
     }
     System.out.println(str);
     System.out.println(reversedString);
   }
   public static void main(String[] args)
   {
     Example obj = new Example();
     obj.reverseWordInMyString("Welcome to BeginnersBook");
     obj.reverseWordInMyString("This is an easy Java
Program");
   }
}
```

```
Output:
```

```
Welcome to BeginnersBook
emocleW ot kooBsrennigeB
This is an easy Java Program
sihT si na ysae avaJ margorP
```

Java program to perform Bubble Sort on Strings

```
public class JavaExample {
   public static void main(String []args) {
     String str[] = { "Ajeet", "Steve", "Rick", "Becky",
"Mohan"};
     String temp;
     System.out.println("Strings in sorted order:");
     for (int j = 0; j < str.length; j++) {
        for (int i = j + 1; i < str.length; i++) {
          // comparing adjacent strings
          if (str[i].compareTo(str[j]) < 0) {
               temp = str[j];
               str[j] = str[i];
               str[i] = temp;
          }
        }
        System.out.println(str[j]);
     }
   }
}
```