

Java

Java Sparkles

(Complete Logical Programming Guide)



SRINIVAS GARAPATI

Contact for Online Classes: +91 – 911 955 6789 / 7306021113
Practice @ www.onlinejavacompiler.com

Contents

S No	Topic	Page Num
01	Formatting Output	02
02	Scanner Class – Reading input	05
03	Basic Programs	08
04	If Block Programs	13
05	If Else Block Programs	15
06	If Else If Block	19
07	Nested If Block	22
08	Loops Introduction	26
09	For Loop Programs	27
10	While Loop Programs	33
11	Break & Continue	42
12	Switch Case	43
13	Do While Loop	45
14	Nested For Loop	46
15	Range Based Programs	48
16	Pattern Programs	51
17	Methods	80
18	Recursion	82
19	Menu Driven Programs	84
20	Arrays	88
21	Two Dimensional arrays	112
22	Strings	118

Formatting Output

Introduction: It is important to format the output before display the results to end user. Proper formatting makes the user more understandable about program results.

We always display results in String format. To format the output, we concatenate the values such as int, char, double, boolean with messages (string type) as follows:

Syntax	Example
int + int = int	10 + 20 = 30
String + String = String	"Java" + "Book" = JavaBook "10" + "20" = 1020
String + int = String	"Book" + 1 = Book1 "123" + 456 = 123456
String + int + int = String	"Sum = " + 10 + 20 = Sum1020
String + (int + int) = String	"Sum = " + (10 + 20) = Sum30
String + double = String	"Value = " + 23.45 = Value = 23.45
String – int = Error	

Practice Codes

```
class Pro1
{
    public static void main(String[] args)
    {
        int a=10;
        System.out.println("a value is : " + a);
    }
}
```

Output: a value is : 10

```
class Pro2
{
    public static void main(String[] args)
    {
        int a=10, b=20 ;
        System.out.println(a + " , " + b);
    }
}
```

Output : 10, 20

```
class Pro3
{
    public static void main(String[] args)
    {
        int a=10, b=20 ;
        System.out.println("a = " + a + ", b = " + b);
    }
}
```

Output : a=10, b=20

```
class Pro4
{
    public static void main(String[] args) {
        int a=10, b=20 ;
        System.out.println("a = " + a + "\n" + "b = " + b);
    }
}
```

Output: a=10
b=20

```
class Pro5
{
    public static void main(String[] args) {
        String name = "Amar" ;
        String course = "Java" ;
        String book = "Crackles";
        System.out.println(name + " practice " + course + " from " + book);
    }
}
```

Output: Amar practice Java from Crackles

```
class Pro6
{
    public static void main(String[] args) {
        int id = 101 ;
        String name = "Amar" ;
        double salary = 38000 ;
        S.o.P("ID = " + id + "\nName = " + name + "\nSalary = " + salary);
    }
}
```

Output: ID = 101
Name = Amar
Salary = 38000

```
class Pro7
{
    public static void main(String[] args) {
        int amt = 3000;
        int num = 102 ;
        S.o.p("update account set bal = bal + " + amt + " where num = " + num);
    }
}
```

Output: update account set bal = bal + 3000 where num=102

```
class Pro8
{
    public static void main(String[] args) {
        String lang = "Java";
        System.out.println("'" + lang + "' tutorial");
    }
}
```

Output: 'Java' tutorial

```
class Pro9
{
    public static void main(String[] args) {
        int id = 101 ;
        String name = "amar" ;
        double salary = 35000 ;
        S.o.p("insert into employee values(" + id + ","+name+","+salary+")");
    }
}
```

Output: insert into employee values(101, 'amar' , 35000)

```
class Pro10
{
    public static void main(String[] args) {
        String lang = "Java";
        System.out.println("'" + lang + "' is awesome");
    }
}
```

Output: "Java" is awesome

Reading input - Scanner Class

Scanner Class:

- Using java library class Scanner, we can read input like integers, characters, strings, double values from the user.
- Different methods to read different values such as nextInt(), next(), nextDouble()...
- We need to specify the Keyboard(System.in) while creating Scanner class object.
 - Scanner scan = new Scanner(System.in);
- We access all the methods using object reference name.

Reading integer value: nextInt() method read and returns an integer value

```
import java.util.Scanner;
class ReadInt {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter integer : ");
        int n = scan.nextInt();
        System.out.println("n value is : " + n);
    }
}
```

Output: Enter integer : 10
n value is : 10

Reading Boolean value: nextBoolean() method returns Boolean value that we entered

```
import java.util.Scanner;
class ReadBoolean {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter boolean value : ");
        boolean b = scan.nextBoolean();
        System.out.println("b value is : " + b);
    }
}
```

Output: Enter Boolean value : true
b value is : true

Reading String: using next() method we can read single word string from the user

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

```
Scanner scan = new Scanner(System.in);
System.out.print("Enter your name : ");
String name = scan.next();
System.out.println("Hello : " + name);
}
}
```

Output: Enter your name: Amar
Hello : Amar

Reading character: There is no method in Scanner class to read single character, hence we read first character of String to read single character as follows

```
import java.util.Scanner;
class Code {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter character : ");
        char ch = scan.next().charAt(0);
        System.out.println("Input character is : " + ch);
    }
}
```

Output: Enter character : A
Input character is : A

Reading multi-word String: nextLine() method can read string includes spaces

```
import java.util.Scanner;
class Code {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter string with spaces : ");
        String s = scan.nextLine();
        System.out.println("Input String is : " + s);
    }
}
```

Output : Enter string with spaces : This is core concept
Input String is : This is core concept

InputMismatchException: A runtime error occurs if the user enters invalid input instead of expected. We use exception handling process(try-catch) to display the error message.

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

```
class Code {  
    public static void main(String[] args) {  
        Scanner scan = new Scanner(System.in);  
        try{  
            System.out.print("Enter integer : ");  
            int n = scan.nextInt();  
            System.out.println("Input n value : " + n);  
        }  
        catch (InputMismatchException e){  
            System.out.println("Exception : Please enter integer only");  
        }  
    }  
}
```

Output: Enter integer : abc
Exception : Please enter integer only.

Read Employee details and print:

```
import java.util.Scanner;  
class Code  
{  
    public static void main(String[] args)  
    {  
        Scanner scan = new Scanner(System.in);  
        System.out.println("Enter Emp details : ");  
        int id = scan.nextInt();  
        String name = scan.next();  
        double salary = scan.nextDouble();  
        System.out.println("Details : " + id + ", " + name + ", " + salary);  
    }  
}
```

Output : Enter Emp details :
101
Amar
53000
Details : 101, Amar, 53000.0

Basic Programs in Java

Find the average of two numbers:

```
class Code {  
    public static void main(String[] args) {  
        int num1 = 5 , num2 = 3;  
        int avg = (num1+num2)/2;  
        System.out.println("Average : " + avg);  
    }  
}
```

Output: Average : 4

Find the Sum of Square and Cube of a Number:

```
class Code{  
    public static void main(String[] args) {  
        int a = 5;  
        int sq = a*a;  
        int cu = a*a*a;  
        int sum = sq + cu;  
        System.out.println("Square of : " + a + " is : " + sq);  
        System.out.println("Cube of : " + a + " is : " + cu);  
        System.out.println("Sum is : " + sum);  
    }  
}
```

Output: Square of : 5 is : 25
Cube of : 5 is : 125

Perform Arithmetic Operations:

```
import java.util.Scanner;  
class Code {  
    public static void main(String[] args) {  
        Scanner scan = new Scanner(System.in);  
        System.out.println("Enter two integers : ");  
        int a = scan.nextInt();  
        int b = scan.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(a + " % " + b + " = " + (a%b));  
    }  
}
```

```
Output: Enter two integers :
5
2
5 + 2 = 7
5 - 2 = 3
5 * 2 = 10
5 / 2 = 2
5 % 2 = 1
```

Print Last Digit of given Number:

```
import java.util.Scanner;
class Code
{
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.println("Enter a number : ");
        int n = scan.nextInt();
        int d = n%10;
        System.out.println("Last digit of " + n + " is " + d);
    }
}
```

```
Output: Enter a number : 123
Last digit of 123 is 3
```

Remove Last Digit of Given Number

```
import java.util.Scanner;
class Code
{
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.println("Enter a number : ");
        int n = scan.nextInt();
        n = n/10;
        System.out.println("After remove last digit : " + n);
    }
}
```

```
Output: Enter a number : 123
After remove last digit : 12
```

Find the Sum of First N Natural Numbers:

Formula: $n(n+1)/2$

```
import java.util.Scanner;
class Code{
    public static void main(String[] args) {
        int n ;
        Scanner scan = new Scanner(System.in);
        System.out.println("Enter n value : ");
        n = scan.nextInt();
        int sum = n*(n+1)/2;
        System.out.println("Sum of First " + n + " numbers : " + sum);
    }
}
```

Output: Enter n value : 5
Sum of First 5 numbers : 15

Calculate Total Salary of Employee:

```
import java.util.Scanner;
class Code{
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.println("Enter basic salary : ");
        double basic = scan.nextDouble();
        double hra = 0.25*basic;
        double ta = 0.2*basic;
        double da = 0.15*basic;
        double total = basic + hra + ta + da ;
        System.out.println("Total Salary : " + total);
    }
}
```

Output: Enter basic salary : 25000
Total Salary : 40000

Find the Third Angle in Triangle:

- Sum of angles of a triangle is 180 degrees
- If two angles given, then third angle become $\rightarrow c = 180 - (a+b)$

```
import java.util.Scanner;
class Code
{
    public static void main(String[] args) {
        int a, b, c;
        Scanner scan = new Scanner(System.in);
        System.out.println("Enter two angles of triangle: ");
```

```
a = scan.nextInt();
b = scan.nextInt();
c = 180 - (a + b);
System.out.println("Third angle is : " + c);
}
}
```

Output: Enter two angles of triangle :
40
65
Third angle is : 75

Swap two Numbers:

class Code

```
{
    public static void main(String[] args) {
        int a=5, b=3;
        System.out.println("Before Swap : " + a + ", " + b);
        int temp=a;
        a=b;
        b=temp;
        System.out.println("After Swap : " + a + ", " + b);
    }
}
```

Output: Before Swap : 5, 3
After Swap : 3, 5

Swap two numbers without third variable:

class Code

```
{
    public static void main(String[] args) {
        int a=5, b=3;
        System.out.println("Before Swap : " + a + ", " + b);
        a = a+b;
        b = a-b;
        a = a-b;
        System.out.println("After Swap : " + a + ", " + b);
    }
}
```

Output: Before Swap : 5, 3
After Swap : 3, 5

Display Amount based on Quantity: Accept the rate for a dozen bananas and the quantity required to determine the cost:

```
import java.util.Scanner;
class Code
{
    public static void main(String[] args) {
        double cost, quantity, amount;
        Scanner scan = new Scanner(System.in);
        System.out.println("Enter cost for dozen bananas: ");
        cost = scan.nextDouble();
        System.out.println("Enter quantity : ");
        quantity = scan.nextDouble();
        amount = quantity/12 * cost ;
        System.out.println("Amount is : " + amount);
    }
}
```

```
Output:      Enter cost for dozen bananas : 40
              Enter quantity : 60
              Amount is : 200.0
```

Program to convert days to years weeks and days:

```
years = days / 365.
weeks = (days – (year * 365)) / 7.
days = days – ((years * 365) + (weeks * 7)).
```

```
import java.util.Scanner;
class Code{
    public static void main(String[] args) {
        int days, years, weeks;
        Scanner scan = new Scanner(System.in);
        System.out.println("Enter days : ");
        days = scan.nextInt();
        years = (days / 365);
        weeks = (days % 365) / 7;
        days = days - ((years * 365) + (weeks * 7));
        System.out.println(years+" years " + weeks + " weeks " + days + " days");
    }
}
```

```
Output:      Enter days : 500
              1 years 19 weeks 2 days
```

If Block Programs

If Block: It is used to execute a block of instructions if the condition is valid.

Syntax	Flow Chart
<pre>if(Condition) { If-block-logic; }</pre>	<pre> graph TD Start([Start]) --> Condition{Condition} Condition --> Logic[/If-block-logic/] Logic --> End([End]) Condition --> End </pre>

Program to give 15% discount on bill if the bill amount is greater than 5000

```
import java.util.Scanner;
class Code
{
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter bill amount : ");
        double bill = scan.nextDouble();
        if(bill > 5000){
            double discount = 0.15 * bill;
            bill = bill - discount;
        }
        System.out.println("Final bill to pay : " + bill);
    }
}
```

Output: Enter bill amount : 6000
Final bill to pay : 5400.0

Program to give 20% bonus on salary if the employee has more than 5 years of experience:

```
import java.util.Scanner;
class Code
{
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter salary : ");
        double salary = scan.nextDouble();
    }
}
```

```
System.out.println("Enter experience : ");
int exp = scan.nextInt();
if(exp>5){
    double bonus = 0.2*salary;
    salary = salary + bonus ;
}
System.out.println("Salary to pay : " + salary);
}
}
```

Output: Enter salary : 30000
Enter experience : 7
Salary to pay : 36000

Program to give 200 rupees cash back if the customer pay minimum 50% amount of credit bill:

```
import java.util.Scanner;
class Code
{
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter Credit card bill amount : ");
        double bill = scan.nextDouble();
        System.out.println("Enter amount to pay : ");
        double amount = scan.nextDouble();
        double min = 0.5*bill;
        int cashback = 0;
        if(amount >= min){
            cashback = 200 ;
        }
        System.out.println("Thank you for payment of : " + amount);
        System.out.println("Your cash back is : " + cashback);
    }
}
```

Output: Enter Credit car bill amount : 20000
Enter amount to pay : 12000
Thank you for payment of : 12000
Your cash back is : 200

If Else Block Programs

Else Block: It executes when the given condition of if block fails. Else block is optional for if block. Else block cannot be defined without IF block.

Syntax	Flow Chart
<pre> if(Condition) { If-stats; } else { Else-stats; } </pre>	<pre> graph TD Start([Start]) --> Condition{Condition} Condition -- True --> IfStats[If - Stats] Condition -- False --> ElseStats[Else - stats] IfStats --> End([End]) ElseStats --> End </pre>

Program to check the input number is Positive or Negative:

```

import java.util.Scanner;
class Code{
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.println("Enter an integer : ");
        int n = scan.nextInt();
        if(n>=0)
            System.out.println("Positive number");
        else
            System.out.println("Negative number");
    }
}
          
```

Output: Enter an integer: 5
Positive Number

Check the input number is Even or Odd:

```

import java.util.Scanner;
class Code
{
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.println("Enter number : ");
        int n = scan.nextInt();
        if(n%2==0)
            System.out.println("Even number");
    }
}
          
```



```
        else
            System.out.println("Not even number");
    }
}
```

Output: Enter number: 7
Not even number

Program to Check Person can Vote or Not:

```
import java.util.Scanner;
class Code
{
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.println("Enter age : ");
        int age = scan.nextInt();
        if(age >= 18)
            System.out.println("Eligible for Vote");
        else
            System.out.println("Wait " + (18-age) + " more years to vote");
    }
}
```

Output: Enter age: 13
Wait 5 more years to vote

Program to find the biggest of two numbers:

```
import java.util.Scanner;
class Code {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.println("Enter 2 numbers : ");
        int a = scan.nextInt();
        int b = scan.nextInt();
        if(a > b)
            System.out.println("a is big");
        else
            System.out.println("b is big");
    }
}
```

Output: Enter 2 numbers:
10
20
b is big

Program to check the character is Vowel or Not

```
import java.util.Scanner;
class Code{
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.println("Enter character : ");
        char ch = scan.next().charAt(0);
        if(ch=='a' || ch=='e' || ch=='i' || ch=='o' || ch=='u')
            System.out.println("Vowel");
        else
            System.out.println("Not vowel");
    }
}
```

Output: Enter character: g
Not vowel

Program to check the character is Alphabet or Not:

```
import java.util.Scanner;
class Code
{
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.println("Enter character : ");
        char ch = scan.next().charAt(0);
        if((ch>='A' && ch<='Z') || (ch>='a' && ch<='z'))
            System.out.println("Alphabet");
        else
            System.out.println("Not an Alphabet");
    }
}
```

Output: Enter character : 3
Not an Alphabet

Program to Check Login details are Correct or Not:

```
import java.util.Scanner;
class Code
{
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.println("Enter user name : ");
        String user = scan.next();
        System.out.println("Enter password : ");
        int pwd = scan.nextInt();
    }
}
```

```
        if(user.equals("logic") && pwd==1234)
            System.out.println("Login Success");
        else
            System.out.println("Error : Invalid login");
    }
}
```

Output: Enter user name : logic
Enter password : 1111
Error : Invalid login

Program to Check the Number between 30 and 50 or not:

```
import java.util.Scanner;
class Code{
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.println("Enter number between 30 and 50 : ");
        int num = scan.nextInt();
        if(num>=30 && num<=50)
            System.out.println("Valid number entered");
        else
            System.out.println("Invalid number entered");
    }
}
```

Output: Enter number between 30 and 50 : 54
Invalid number entered

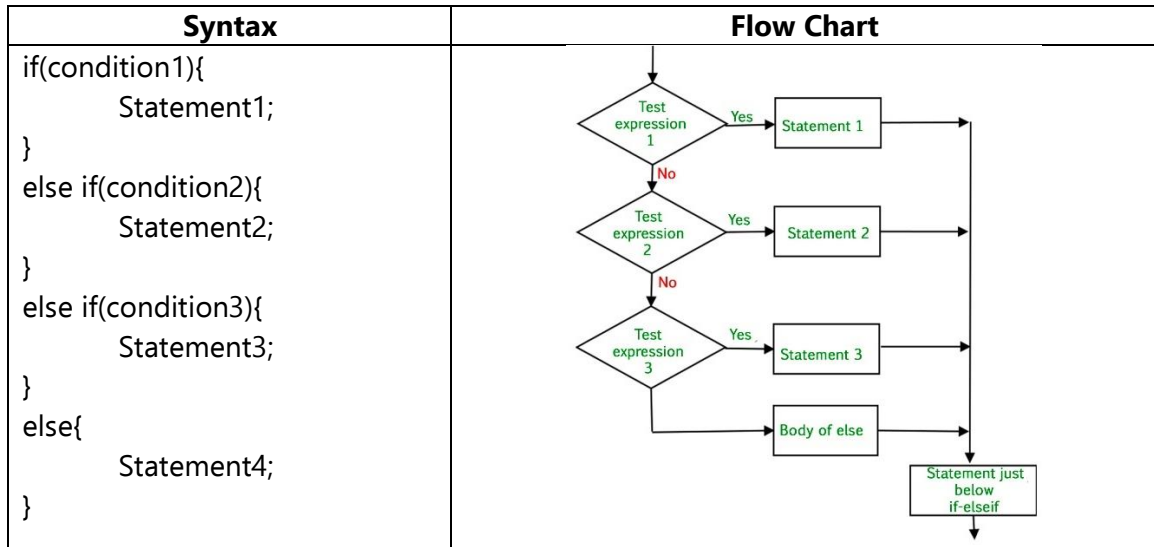
Check the input year is Leap year or Not:

```
import java.util.Scanner;
class Code{
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.println("Enter year : ");
        int n = scan.nextInt();
        if((n%400==0) || (n%4==0 && n%100!=0))
            System.out.println("Leap year");
        else
            System.out.println("Not a leap year");
    }
}
```

Output: Enter year : 1996
Leap year

If Else If block

If else If: The if-else-if ladder statement executes only block among multiple we defined based on valid condition.



Check the Number is Positive or Negative or Zero

```

import java.util.Scanner;
class Code{
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.println("Enter n value : ");
        int n = scan.nextInt();
        if(n>0)
            System.out.println("Positive num");
        else if(n<0)
            System.out.println("Negative num");
        else
            System.out.println("Zero");
    }
}
          
```

Output: Enter n value : -13
Negative num

Check the Character is Alphabet or Digit or Symbol

```

import java.util.Scanner;
class Code{
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.println("Enter character : ");
          
```

```
char ch = scan.nextLine().charAt(0);
if((ch>='A' && ch<='Z') || (ch>='a' && ch<='z'))
    System.out.println("Alphabet");
else if(ch>='0' && ch<='9')
    System.out.println("Digit");
else
    System.out.println("Special Symbol");
}
```

Output: Enter character: 0
Digit

Program to check the input year is Leap or Not:

Leap Year rules:

- 400 multiples are leap years : 400, 800, 1200, 1600....
- 4 multiples are leap years: 4, 8, 12, ... 92, 96...
- 100 multiples are not leap years: 100, 200, 300, 500, 600...

```
import java.util.Scanner;
class Code{
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.println("Enter year : ");
        int n = scan.nextInt();
        if((n%400==0))
            System.out.println("Leap year");
        else if(n%4==0 && n%100!=0)
            System.out.println("Leap Year");
        else
            System.out.println("Not leap year");
    }
}
```

Output: Enter year: 1996
Leap year

Program to find Biggest of Three Numbers:

```
import java.util.Scanner;
class Code{
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.println("Enter a, b, c values : ");
        int a = scan.nextInt();
        int b = scan.nextInt();
        int c = scan.nextInt();
    }
}
```

```
        if(a>b && a>c)
            System.out.println("a is big");
        else if(b>c)
            System.out.println("b is big");
        else
            System.out.println("c is big");
    }
}
```

Output: Enter a, b, c values :
10
30
20
b is big

Program to Calculate current bill based on units consumed:

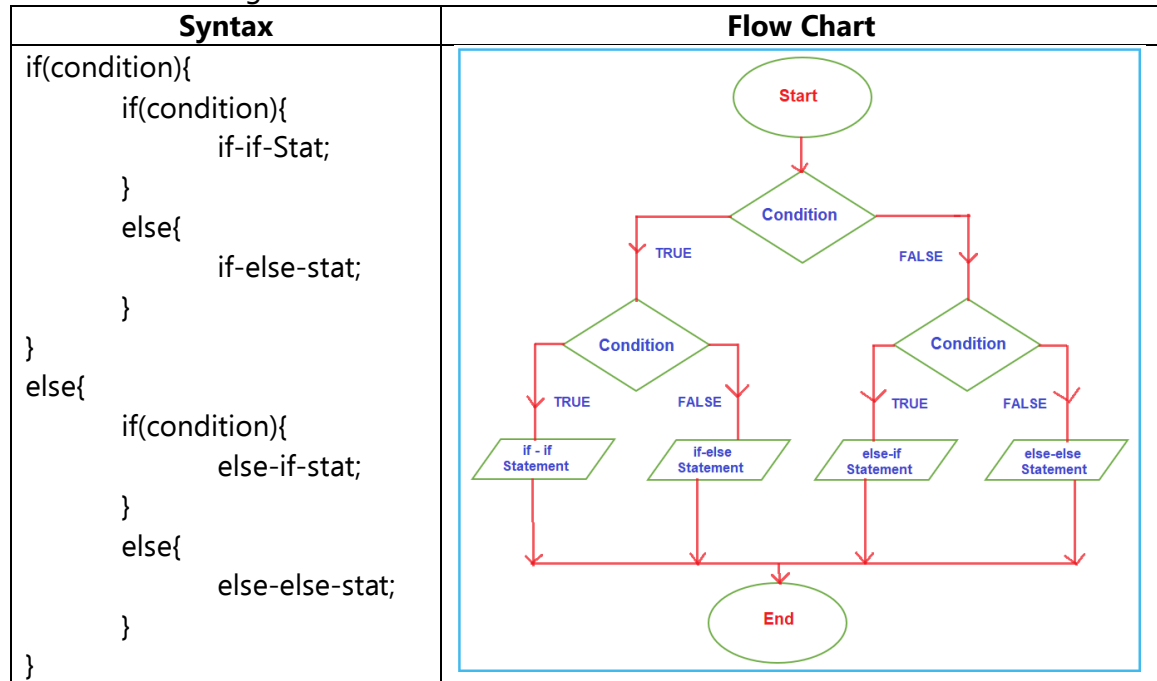
0-100 -> 0.80 per unit
101-200 -> 1.2 per unit
201-300 -> 1.5 per unit
above 300 -> 1.8 per unit

```
import java.util.Scanner;
class Code
{
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter number of units : ");
        int units = sc.nextInt();
        double bill=0.0;
        if(units>=0 && units<=100)
            bill = units*0.8;
        else if(units>100 && units<=200)
            bill = 80 + (units-100)*1.2;
        else if(units>200 && units<=300)
            bill = 200 + (units-200)*1.5;
        else
            bill = 350 + (units-300)*1.8;
        System.out.println("Total bill amount : " + bill);
    }
}
```

Output: Enter number of units: 150
Total bill amount : 140.0

Nested If Block

Nested If: Defining if block inside another if block.



Nested If Programs

Check Even Number or Not only if it is Positive

```

import java.util.Scanner;
class Code{
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.println("Enter number : ");
        int n = scan.nextInt();
        if(n>0){
            if(n%2==0)
                System.out.println("Even number");
            else
                System.out.println("Not even number");
        }
        else{
            System.out.println("Negative number given");
        }
    }
}

```

Output: Enter number: 5
Not even number

Biggest of two only if the two numbers or not equal

```
import java.util.Scanner;
class Code{
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.println("Enter a, b values : ");
        int a = scan.nextInt();
        int b = scan.nextInt();
        if(a!=b){
            if(a>b)
                System.out.println("a is big");
            else
                System.out.println("b is big");
        }
        else{
            System.out.println("a and b are equal");
        }
    }
}
```

Output: Enter a, b values : 10
20
b is big

Program to check the person can donate blood or not

Requirements: Age between 18 to 60 and Weight must be greater than 50

```
import java.util.Scanner;
class Code{
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.println("Enter age : ");
        int age = scan.nextInt();
        System.out.println("Enter weight : ");
        int weight = scan.nextInt();
        if(age>=18 && age<=60){
            if(weight>=50)
                System.out.println("Can donate blood");
            else
                System.out.println("Under weight");
        }
        else
            System.out.println("Not suitable age");
    }
}
```



```
Output: Enter age : 25
        Enter weight : 49
        Under weight
```

Display Student Grade only if the student passed in all Subjects

```
import java.util.Scanner;
class Code{
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.println("Enter marks of 3 subjects : ");
        int m1 = scan.nextInt();
        int m2 = scan.nextInt();
        int m3 = scan.nextInt();
        if(m1>=40 && m2>=40 && m3>=40){
            int avg = (m1+m2+m3)/3;
            if(avg>=60)
                System.out.println("Grade-A");
            else if(avg>=50)
                System.out.println("Grade-B");
            else
                System.out.println("Grade-C");
        }
        else{
            System.out.println("Student failed");
        }
    }
}
```

```
Output: Enter marks of 3 subjects:
        45
        56
        67
        Grade-B
```

Check the Triangle is Equilateral or Isosceles or Scalene

Equilateral: All sides are equal $a=b=c$

Isosceles: Any two sides are equal $a=b$ or $a=c$ or $b=c$

Scalene: No sides are equal $a \neq b \neq c$

```
import java.util.Scanner;
class Code{
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.println("Enter 3 sides of triangle : ");
        int a = scan.nextInt();
```

```
int b = scan.nextInt();
int c = scan.nextInt();
if(a==b && b==c)
    System.out.println("Equilateral");
else if(a==b || a==c || b==c)
    System.out.println("Isosceles");
else
    System.out.println("Scalene");
}
}
```

Output: Enter 3 sides of triangle :
70
70
40
Isosceles

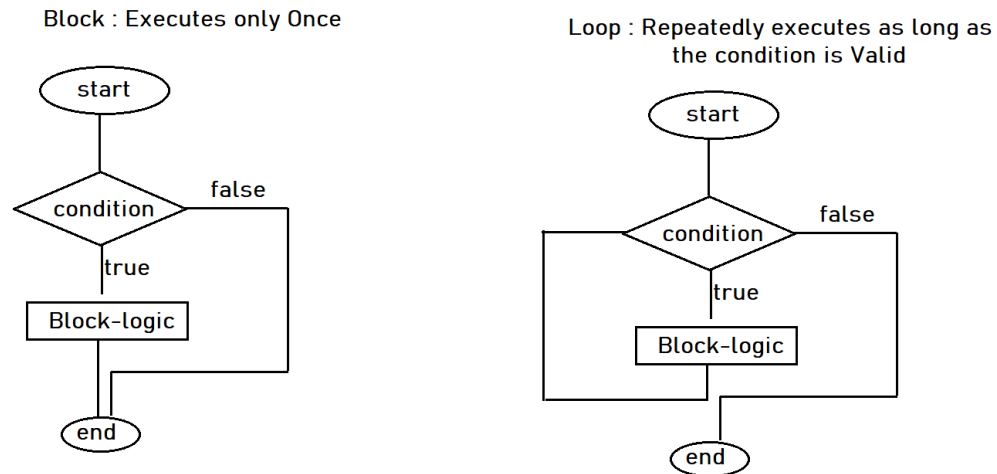
Display days for the specified month

```
import java.util.Scanner;
class Code
{
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.println("Enter Month number : ");
        int n = scan.nextInt();
        if(n>=1 && n<=12){
            if(n==2)
                System.out.println("28 days / 29 days");
            else if(n==4 || n==6 || n==9 || n==11)
                System.out.println("30 days");
            else
                System.out.println("31 days");
        }
        else{
            System.out.println("Invalid month number");
        }
    }
}
```

Output: Enter Month number : 2
28 days / 29 days

Introduction to Loops

Loop: A Block of instructions execute repeatedly as long the condition is valid.



Note: Block executes only once whereas Loop executes until condition become False

Java Supports 3 types of Loops:

1. For Loop
2. While Loop
3. Do-While Loop

For Loop: We use for loop only when we know the number of repetitions. For example,

- Print 1 to 10 numbers
- Print Array elements
- Print Multiplication table
- Print String character by character in reverse order

While loop: We use while loop when we don't know the number of repetitions.

- Display contents of File
- Display records of Database table

Do while Loop: Execute a block at least once and repeat based on the condition.

- ATM transaction: When we swipe the ATM card, it starts the first transaction. Once the first transaction has been completed, it asks the customer to continue with another transaction and quit.

For Loop

for loop: Execute a block of instructions repeatedly as long as the condition is valid. We use for loop only when we know the number of iterations to do.

Syntax	Flow Chart
<pre>for(init ; condition ; modify) { statements; } for(start ; stop ; step) { statements; }</pre>	<pre> graph TD Start([Start]) --> Condition{Condition} Condition -- TRUE --> Statements[Statements] Statements --> Condition Condition -- FALSE --> End([End]) </pre>

Program to Print 1 to 10 Numbers

```
class Code {
    public static void main(String[] args) {
        for (int i=1 ; i<=10 ; i++){
            System.out.println("i val : " + i);
        }
    }
}
```

Program to Print 10 to 1 numbers

```
class Code{
    public static void main(String[] args) {
        for (int i=10 ; i>=1 ; i--){
            System.out.println("i val : " + i);
        }
    }
}
```

Program to display A-Z alphabets

```
class Code {
    public static void main(String[] args) {
        for (char ch='A' ; ch<='Z' ; ch++){
            System.out.print(ch + " ");
        }
    }
}
```

Program to Print 1 to N numbers

```
import java.util.Scanner;
class Code {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter n val : ");
        int n = scan.nextInt();
        for (int i=1 ; i<=n ; i++){
            System.out.println("i val : " + i);
        }
    }
}
```

Program to display A-Z alphabets with ASCII values

```
class Code {
    public static void main(String[] args) {
        for (char ch='A' ; ch<='Z' ; ch++){
            System.out.println(ch + " : " + (int)ch);
        }
    }
}
```

Program to display ASCII character set

```
class Code {
    public static void main(String[] args) {
        for (int i=0 ; i<256 ; i++)
            System.out.println(i + " : " + (char)i);
    }
}
```

Find Sum of First N numbers

```
import java.util.Scanner;
class Code {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.println("Enter n value : ");
        int n = scan.nextInt();
        int sum=0;
        for (int i=1 ; i<=n ; ++i)
            sum = sum+i;
        System.out.println("Sum of first " + n + " numbers is : " + sum);
    }
}
```

Program to Find Factorial of Number

```
class Code {  
    public static void main(String[] args) {  
        Scanner scan = new Scanner(System.in);  
        System.out.println("Enter n value : ");  
        int n = scan.nextInt();  
        int fact=1;  
        for (int i=1 ; i<=n ; ++i){  
            fact = fact*i;  
        }  
        System.out.println("Factorial of " + n + " is : " + fact);  
    }  
}
```

Program to display Multiplication table

```
import java.util.Scanner;  
class Code {  
    public static void main(String[] args) {  
        Scanner scan = new Scanner(System.in);  
        System.out.println("Enter table number : ");  
        int n = scan.nextInt();  
        for (int i=1 ; i<=10 ; ++i){  
            System.out.println(n + " x " + i + " = " + (n*i));  
        }  
    }  
}
```

Print Even Numbers from 1 to 10

```
class Code {  
    public static void main(String[] args) {  
        for (int i=1 ; i<=10 ; ++i){  
            if(i%2==0){  
                System.out.println(i);  
            }  
        }  
    }  
}
```

Find Sum of Even Numbers in Java

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

```
        System.out.print("Enter range : ");
        int n = scan.nextInt();
        int sum=0;
        for (int i=1 ; i<=n ; i++){
            if(i%2==0)
                sum=sum+i;
        }
        System.out.println("Even numbers sum : " + sum);
    }
}
```

Program to print factors of given number

```
import java.util.Scanner;
class Code
{
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter n value : ");
        int n = sc.nextInt();
        for (int i=1 ; i<=n ; i++){
            if(n%i==0)
                System.out.println(i + " is a factor");
        }
    }
}
```

Program to count factors of given number

```
import java.util.Scanner;
class Code
{
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter n value : ");
        int n = sc.nextInt();
        int count=0;
        for (int i=1 ; i<=n ; i++){
            if(n%i==0)
                count++;
        }
        System.out.println("Factors count is : " + count);
    }
}
```

Program to check the input number is Prime or Not

```
import java.util.Scanner;
class Code
{
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter n value : ");
        int n = sc.nextInt();
        int count=0;
        for (int i=1 ; i<=n ; i++){
            if(n%i==0)
                count++;
        }
        if(count==2)
            System.out.println("Prime Number");
        else
            System.out.println("Not a Prime Number");
    }
}
```

Program to find the sum of factors of given number

```
import java.util.Scanner;
class Code
{
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter n value : ");
        int n = sc.nextInt();
        int sum=0;
        for (int i=1 ; i<=n ; i++){
            if(n%i==0)
                sum=sum+i;
        }
        System.out.println("Sum of Factors : " + sum);
    }
}
```

Program to check the input number is Perfect or Not

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



```
System.out.print("Enter n value : ");
int n = sc.nextInt();
int sum=0;
for (int i=1 ; i<n ; i++){
    if(n%i==0)
        sum=sum+i;
}
if(n==sum)
    System.out.println("Perfect Number");
else
    System.out.println("Not a Perfect Number");
}
}
```

Program to print Fibonacci series

```
import java.util.Scanner;
class Code
{
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter number of factors in series : ");
        int n = sc.nextInt();
        int a=0, b=1, c;
        for (int i=1 ; i<=n ; i++){
            System.out.print(a + " ");
            c=a+b;
            a=b;
            b=c;
        }
    }
}
```

While Loop

While loop: Execute a block of instructions repeatedly until the condition is false. We use while loop only when don't know the number of iterations to do.

Syntax	Flow Chart
<pre>while(condition) { statements; }</pre>	<pre> graph TD Start([Start]) --> Condition{Condition?} Condition -- True --> DoTask[Do Task] DoTask --> Condition Condition -- False --> End([End]) </pre>

Program to Display Last Digit of given number

- When we divide any number with 10 then last digit will be the remainder.
- $1234\%10 \rightarrow 4$
- $1005\%10 \rightarrow 5$
- $200\%10 \rightarrow 0$

```
import java.util.Scanner;
class Code {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter Num : ");
        int n = scan.nextInt();
        int d = n%10;
        System.out.println("Last digit is : " + d);
    }
}
```

Output: Enter Num : 1234
Last digit is : 4

Program to remove the last digit of number:

- When we divide any number with 10 then it gives quotient by removing last digit
- $1234/10 \rightarrow 123$
- $1000/10 \rightarrow 100$

```
import java.util.Scanner;
class Code {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter Num : ");
```

```
        int n = scan.nextInt();
        n = n/10;
        System.out.println("After removing last digit : " + n);
    }
}
```

Output: Enter Num : 1234
After removing last digit : 123

Display digits of a number in reverse order:

```
import java.util.Scanner;
class Code {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter Num : ");
        int n = scan.nextInt();
        while(n!=0){
            System.out.println(n%10);
            n=n/10;
        }
    }
}
```

Output: Enter Num : 234
4
3
2

Program to Count the digits in given number

```
import java.util.Scanner;
class Code {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter Num : ");
        int n = scan.nextInt();
        int count=0;
        while(n!=0){
            n=n/10;
            count++;
        }
        System.out.println("Digits count : " + count);
    }
}
```

Output: Enter Num : 1234
Digits count : 4

Display sum of the digits in the given number

```
import java.util.Scanner;
class Code {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter Num : ");
        int n = scan.nextInt();
        int sum=0;
        while(n!=0){
            sum = sum + n%10;
            n=n/10;
        }
        System.out.println("Sum of digits : " + sum);
    }
}
```

Output: Enter Num : 1234
Sum of digits : 10

Program to display the sum of even digits:

```
import java.util.Scanner;
class Code{
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter Num : ");
        int n = scan.nextInt();
        int sum=0, r;
        while(n>0){
            r = n%10;
            if(r%2==0){
                sum = sum+r;
            }
            n = n/10;
        }
        System.out.println("Sum of Even digits are : " + sum);
    }
}
```

Output: Enter Num : 12345
Sum of Even digits are : 6

Program to count the number of 0 digits in the given number:

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

```
Scanner scan = new Scanner(System.in);
System.out.print("Enter Num : ");
int n = scan.nextInt();
int count=0, r;
while(n>0){
    r = n%10;
    if(r==0){
        count++;
    }
    n = n/10;
}
System.out.println("Zero's count : " + count);
}
```

Output: Enter Num : 10005
Zero's count : 3

Program to display the first digit of given number:

```
import java.util.Scanner;
class Code {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter Num : ");
        int n = scan.nextInt();
        while(n>=10){
            n = n/10;
        }
        System.out.println("First Digit : " + n%10);
    }
}
```

Output: Enter Num : 2345
First Digit : 2

Program to Find the Sum of First and Last digits of given number

```
import java.util.Scanner;
class Code
{
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter Num : ");
        int n = scan.nextInt();
        int first = n%10;
        n=n/10;
```

```
        while(n>=10){
            n = n/10;
        }
        int last = n%10;
        System.out.println("Sum of First & Last Digits : " + (first+last));
    }
}
```

Output: Enter Num : 1234
Sum of First & Last Digits : 5

Program to reverse the given number

```
import java.util.Scanner;
class Code
{
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter Num : ");
        int n = scan.nextInt();
        int rev=0, r;
        while(n>0){
            r = n%10;
            rev = rev*10 + r;
            n = n/10;
        }
        System.out.println("Reverse Number is : " + rev);
    }
}
```

Output: Enter Num: 5342
Reverse Number is : 2435

Palindrome Number: The number become same when we reverse is called Palindrome number

Examples: 121, 1001, 123321

```
import java.util.Scanner;
class Code
{
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter Num : ");
        int n = scan.nextInt();
        int rev=0, r, temp=n;
        while(n>0){
```

```
        r = n%10;
        rev = rev*10 + r;
        n = n/10;
    }
    if(temp==rev)
        System.out.println("Palindrome Number");
    else
        System.out.println("Not Palindrome Number");
}
}
```

Output: Enter Num : 1221
Palindrome Number

Program to display highest digit in the number

```
import java.util.Scanner;
class Code
{
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter Num : ");
        int n = scan.nextInt();
        int r, large=0;
        while(n>0){
            r = n%10;
            if(r>large){
                large = r;
            }
            n = n/10;
        }
        System.out.println("Larger Digit is : " + large);
    }
}
```

Output: Enter Num: 38953
Larger Digit is : 9

Check the given 3 digit number is Armstrong Number or not :

Sum of cubes of individual digits equals to the same number

Example: 153 -> $1^3 + 5^3 + 3^3 = 153$

```
import java.util.Scanner;
class Code {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter Num : ");
```

```
int n = scan.nextInt();
int temp, sum=0, r;
temp=n;
while(n>0){
    r = n%10;
    sum = sum + r*r*r;
    n = n/10;
}
if(temp==sum)
    System.out.println("ArmStrong Number");
else
    System.out.println("Not an ArmStrong Number");
}
```

Output: Enter 3 digit num : 145
Not an Armstrong Number

Program to check the given number is Armstrong number or not:

Armstrong Number: Sum of its own digits raised to power number of digits

Examples : $153 = 1^3 + 5^3 + 3^3 = 153$
 $1634 = 1^4 + 6^4 + 3^4 + 4^4 = 1634$

```
import java.util.Scanner;
class Code {
    public static void main(String[] args) {
        int n, r, sum=0, temp, c=0, s, i;
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter Num : ");
        n = scan.nextInt();
        temp=n;
        while(n>0){
            n=n/10;
            c++;
        }
        n=temp;
        while(n>0){
            r = n%10;
            s=1;
            for(i=1 ; i<=c ; i++){
                s = s*r;
            }
            sum = sum + s;
        }
    }
}
```



```
        n = n/10;
    }
    if(temp==sum)
        System.out.println("ArmStrong Number");
    else
        System.out.println("Not an ArmStrong Number");
    }
}
```

Output: Enter Number : 153
Armstrong Number

Program to find Sum of Digits till Single Digit:

9657 -> 9+6+5+7 -> 27 -> 2+7 -> 9

```
import java.util.Scanner;
class Code
{
    public static void main(String[] args) {
        int num, sum, dig;
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter Num : ");
        num = scan.nextInt();
        System.out.print(num + "->");

        while(num/10!=0){
            sum = 0;
            while(num!=0){
                dig=num%10;
                sum+=dig;
                num/=10;
            }
            System.out.print(sum + "->");
            num=sum;
        }
    }
}
```

Output: Enter Num: 9657
9657 -> 27 -> 9 ->

ADAM Number: Take a number then square it then reverse it then find its square root then reverse. If the given number equals to the final number then it is called ADAM.

- Take the number (12)
- Square the number (144)
- Reverse the number(441)
- Square root of number (21)
- Reverse the number(12)

```
import java.util.Scanner;
class Code
{
    public static void main(String[] args) {
        int num, temp, r1, r2, sq, rev1 = 0, rev2 = 0;
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter Num : ");
        num = scan.nextInt();
        temp = num * num;
        System.out.println("Square of the num : " + temp);
        while (temp != 0){
            r1 = temp % 10;
            rev1 = rev1 * 10 + r1;
            temp = temp / 10;
        }
        System.out.println("Reverse Num : " + rev1);
        sq = (int)Math.sqrt(rev1);
        System.out.println("Sqrt num : " + sq);
        while (sq != 0){
            r2 = sq % 10;
            rev2 = rev2 * 10 + r2;
            sq = sq / 10;
        }
        System.out.println("Reverse Num : " + rev2);
        if (rev2 == num)
            System.out.println(num + " is an Adam number");
        else
            System.out.println(num + " is not an Adam number");
    }
}
```

```
Output: Enter Num: 12
        12 is an Adam number
```

Break and Continue

break: A branching statement that terminates the execution flow of a Loop or Switch case.

```
class Code {  
    public static void main(String[] args) {  
        for (int i=1 ; i<=10 ; i++){  
            if(i==5){  
                break;  
            }  
            System.out.print(i + " ");  
        }  
    }  
}
```

Output: 1 2 3 4

Break statement terminates the flow of infinite loop also:

```
class Code {  
    public static void main(String[] args) {  
        while(true){  
            System.out.print("Loop");  
            break;  
        }  
    }  
}
```

Output: Loop prints infinite times

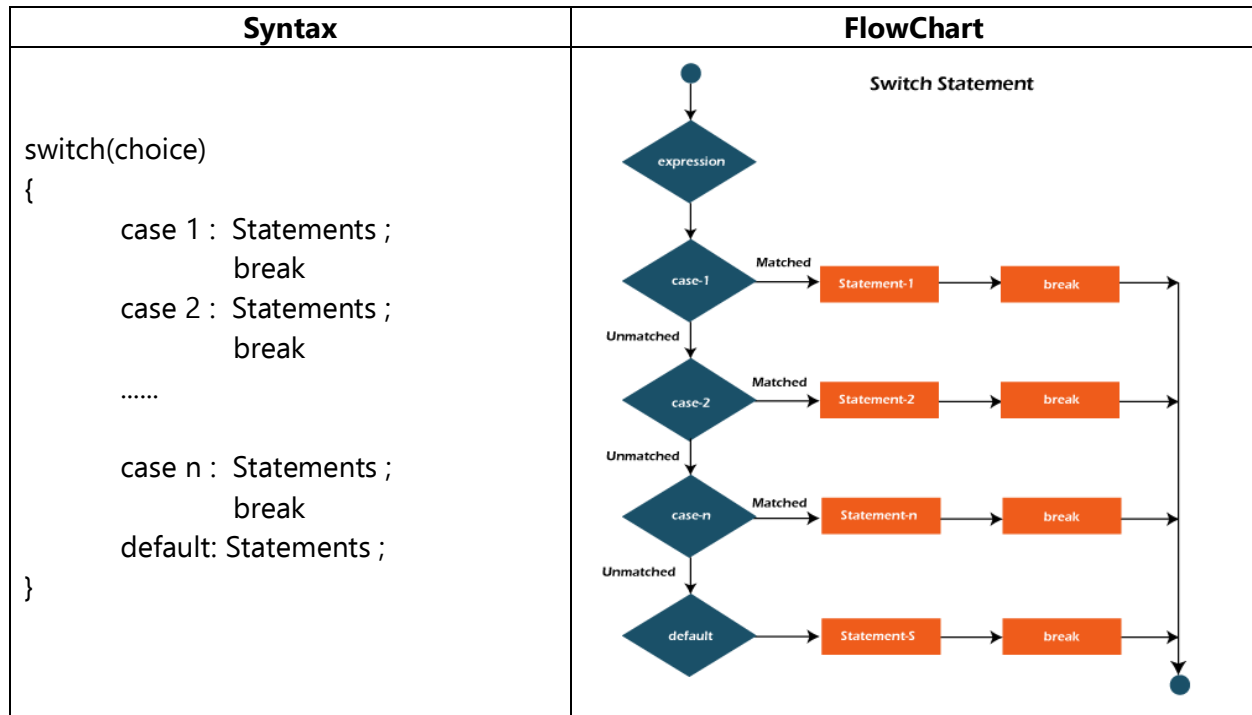
Continue: A branching statement that terminates the current iteration of loop execution.

```
class Code {  
    public static void main(String[] args) {  
        for (int i=1 ; i<=10 ; i++){  
            if(i==5){  
                continue;  
            }  
            System.out.print(i + " ");  
        }  
    }  
}
```

Output: 1 2 3 4 5 6 7 8 9 10

Switch case

Switch: It is a conditional statement that executes specific set of statements(case) based on given choice. Default case executes if the user entered invalid choice. Case should terminate with break statement.



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

```
class Code {
```

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

```
        Scanner sc = new Scanner(System.in);
```

```
        System.out.print("Enter character(r, g, b) : ");
```

```
        char ch = sc.next().charAt(0);
```

```
        switch(ch){
```

```
            case 'r' :    System.out.println("Red");
                        break;
```

```
            case 'g' :    System.out.println("Green");
                        break;
```

```
            case 'b' :    System.out.println("Blue");
                        break;
```

```
            default :    System.out.println("Weird");
```

```
        }
```

```
    }
```

```
}
```

Output: Enter character(r, g, b): g
Green

Arithmetic Operations using Menu options:

```
import java.util.Scanner;
class Code
{
    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        System.out.println("1.Add");
        System.out.println("2.Subtract");
        System.out.println("3.Multiply");

        System.out.print("Enter your choice : ");
        int ch = sc.nextInt();

        int a=0, b=0;
        if(ch>=1 && ch<=3)
        {
            System.out.println("Enter 2 nums : ");
            a = sc.nextInt();
            b = sc.nextInt();
        }
        switch(ch)
        {
            case 1 :      System.out.println("Sum : " + (a+b));
                          break;
            case 2 :      System.out.println("Difference : " + (a-b));
                          break;
            case 3 :      System.out.println("Product : " + (a*b));
                          break;
            default:      System.out.println("Invalid choice");
        }
    }
}
```

```
Output : 1. Add
         2. Subtract
         3. Multiply
         Enter your choice : 2
         30
         10
         Difference : 20
```

Do-While Loop

do-while: Executes a block at least once and continue iteration until condition is false.

Syntax	Flow Chart
<pre>do { statements; } while(condition);</pre>	<pre> graph TD Start([Start]) --> DoTask[Do Task] DoTask --> Condition{Condition?} Condition -- True --> DoTask Condition -- False --> End([End]) </pre>

Check Even numbers until user exits:

```
import java.util.Scanner;
class Code
{
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        char res='n';
        do{
            System.out.print("Enter num : ");
            int n = sc.nextInt();
            if(n%2==0)
                System.out.println(n + " is even");
            else
                System.out.println(n + " is not even");

            System.out.print("Do you want to check another num (y/n) : ");
            res = sc.next().charAt(0);
        }while (res == 'y');
    }
}
```

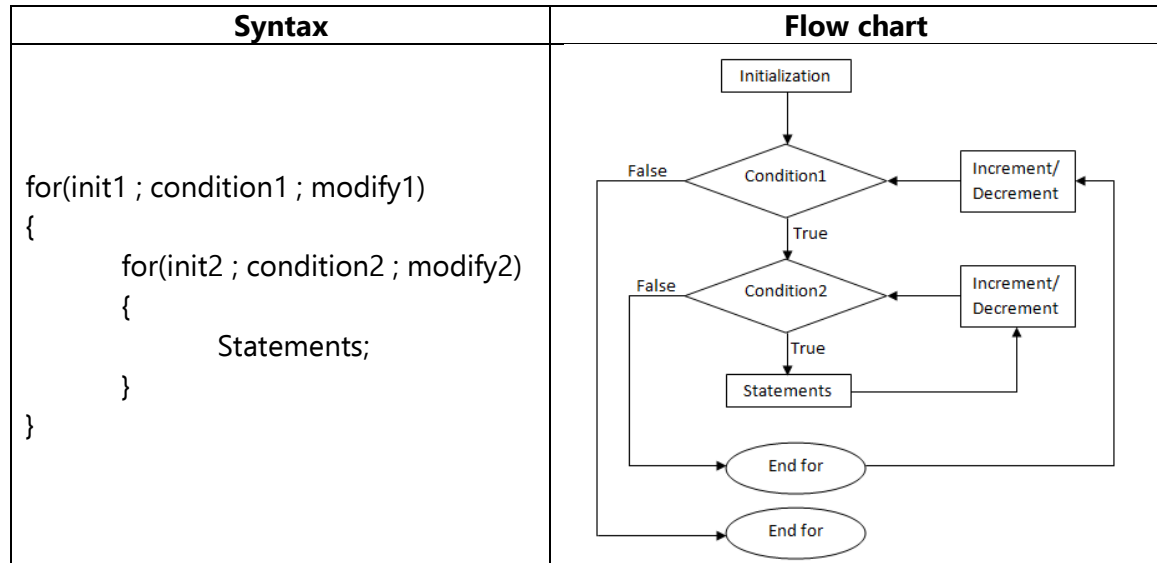
Output:

```

Enter num : 5
5 is not even
Do you want to check another num (y/n) : y
Enter num : 6
6 is even
Do you want to check another num (y/n) : n
```

Nested For Loop

Nested for loop: Defining for loop inside another for loop. We used nested loops to process two-dimensional data such as patterns, matrix type arrays, range based programs, sorting, remove duplicates in arrays, display duplicate count of each element in array and so on.



Number of iterations in Nested loop is equals to “**outer loop iterations multiplied by inner loop iterations**”.

```
class Code {
    public static void main(String[] args) {
        int count=0;
        for (int i=1 ; i<=5 ; i++){
            for (int j=1 ; j<=5 ; j++){
                count++;
            }
        }
        System.out.println("Number of Iterations : " + count);
    }
}
```

Output : Number of Iterations : 25

Nested while loop: Defining while loop inside another while loop

```
while(cond1){
    while(cond2){
        statements;
    }
}
```

Example:

```
class Code{
    public static void main(String[] args) {
        int count=0;
        int i=1;
        while(i<=5){
            int j=1;
            while(j<=5){
                count++;
                j++;
            }
            i++;
        }
        System.out.println("Number of Iterations : " + count);
    }
}
```

Output : Number of Iterations : 25

If break statement executes inside the inner loop, it terminates the flow of Inner loop only

```
class Code
{
    public static void main(String[] args) {
        for (int i=1 ; i<=4 ; i++){
            for (int j=1 ; j<=4 ; j++){
                if(i==j)
                    break;
                System.out.println(i+","+j);
            }
        }
    }
}
```

Output:

```
2, 1
3, 1
3, 2
4, 1
4, 2
4, 3
```


Range Based Programs

Program to display even numbers in the given range

```
import java.util.*;
class Code {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter range : ");
        int range = sc.nextInt();
        System.out.print("Even Numbers in the given range :");
        for(int n=1 ; n<=range ; n++){
            if(n%2==0)
                System.out.print(n + " ");
        }
    }
}
```

Output: Enter range : 10
Even Numbers in the given range : 2 4 6 8 10

Program to print factorial for numbers 2 to 7

```
class Code {
    public static void main(String[] args) {
        for(int n=2 ; n<=7 ; n++){
            int fact=1;
            for (int i=1 ; i<=n ; i++){
                fact=fact*i;
            }
            System.out.println("Factorial of " + n + " is : " + fact);
        }
    }
}
```

Output: Factorial of 2 is : 2
Factorial of 3 is : 6
.....

Program to display multiplication tables in the given range

```
import java.util.*;
class Code {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter Lower Limit : ");
        int lower = sc.nextInt();
        System.out.print("Enter Upper Limit : ");
```

```
int upper = sc.nextInt();
for(int n=lower ; n<=upper ; n++){
    System.out.println("Table : " + n);
    for (int i=1 ; i<=10 ; i++){
        System.out.println(n + " x " + i + " = " + (n*i));
    }
    System.out.println();
}
}
```

Program to display Prime numbers in the given range

```
import java.util.*;
class Code {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter Lower Limit : ");
        int lower = sc.nextInt();
        System.out.print("Enter Upper Limit : ");
        int upper = sc.nextInt();

        System.out.println("Prime Numbers from " + lower + " to " + upper);
        for(int n=lower ; n<=upper ; n++){
            int count=0;
            for (int i=1 ; i<=n ; i++){
                if(n%i==0)
                    count++;
            }
            if(count==2)
                System.out.println(n);
        }
    }
}
```

Program to display Perfect numbers in the given range

```
import java.util.*;
class Code
{
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter Lower Limit : ");
        int lower = sc.nextInt();
        System.out.print("Enter Upper Limit : ");
```

```
int upper = sc.nextInt();

System.out.println("Perfect Numbers from " + lower + " to " + upper);
for(int n=lower ; n<=upper ; n++){
    int sum=0;
    for (int i=1 ; i<n ; i++){
        if(n%i==0)
            sum=sum+i;
    }
    if(n==sum)
        System.out.println(n);
}
}
```

Program to display Palindrome numbers in the given range

```
import java.util.*;
class Code
{
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter Lower Limit : ");
        int lower = sc.nextInt();
        System.out.print("Enter Upper Limit : ");
        int upper = sc.nextInt();

        System.out.println("Palindrome Numbers from " + lower + " to " + upper);
        for(int n=lower ; n<=upper ; n++){
            int temp=n;
            int rev=0, r=0;
            while(temp!=0){
                r=temp%10;
                rev=rev*10+r;
                temp=temp/10;
            }
            if(rev==n)
                System.out.println(n);
        }
    }
}
```

Pattern Programs

Pattern:

- Representation of data in two-dimensional format (rows and columns)
- We use nested loops to print patterns.
- We can print patterns with numbers, characters, stars, symbols
- Patterns can be in different shapes like triangle, rectangle, half triangle, pyramid and so on.

Basic Number Patterns

Pattern	Logic
12345 12345 12345 12345	<pre>for (int i=1 ; i<=5 ; i++){ for (int j=1 ; j<=5 ; j++){ System.out.print(j); } System.out.println(); }</pre>

Pattern	Logic
11111 22222 33333 44444 55555	<pre>for (int i=1 ; i<=5 ; i++){ for (int j=1 ; j<=5 ; j++){ System.out.print(i); } System.out.println(); }</pre>

Pattern	Logic
***** ***** ***** ***** *****	<pre>for (int i=1 ; i<=5 ; i++){ for (int j=1 ; j<=5 ; j++){ System.out.print("*"); } System.out.println(); }</pre>

Pattern	Logic
54321 54321 54321 54321 54321	<pre>for (int i=5 ; i>=1 ; i--){ for (int j=5 ; j>=1 ; j--){ System.out.print(j); } System.out.println(); }</pre>

Pattern	Logic
<pre> 55555 44444 33333 22222 11111 </pre>	<pre> for (int i=5 ; i>=1 ; i--){ for (int j=5 ; j>=1 ; j--){ System.out.print(i); } System.out.println(); } </pre>

Pattern	Logic
<pre> 10101 10101 10101 10101 10101 </pre>	<pre> for (int i=1 ; i<=5 ; i++){ for (int j=1 ; j<=5 ; j++){ System.out.print(j%2); } System.out.println(); } </pre>

Pattern	Logic
<pre> 11111 00000 11111 00000 11111 </pre>	<pre> for (int i=1 ; i<=5 ; i++){ for (int j=1 ; j<=5 ; j++){ System.out.print(i%2); } System.out.println(); } </pre>

Basic Character Patterns

Pattern	Logic
<pre> AAAAA BBBBB CCCCC DDDDD EEEEE </pre>	<pre> for(char x='A' ; x<='E'; x++){ for(char y='A' ; y<='E' ; y++){ System.out.print(x); } System.out.println(); } </pre>

Pattern	Logic
<pre> ABCDE ABCDE ABCDE ABCDE ABCDE </pre>	<pre> for(char x='A' ; x<='E'; x++){ for(char y='A' ; y<='E' ; y++){ System.out.print(y); } System.out.println(); } </pre>

Pattern	Logic
<pre> EEEE DDDD CCCC BBBB AAAA </pre>	<pre> for(char x='E' ; x>='A'; x--){ for(char y='E' ; y>='A' ; y--){ System.out.print(x); } System.out.println(); } </pre>

Pattern	Logic
<pre> EDCBA EDCBA EDCBA EDCBA EDCBA </pre>	<pre> for(char x='E' ; x>='A'; x--){ for(char y='E' ; y>='A' ; y--){ System.out.print(y); } System.out.println(); } </pre>

Basic Patterns with Conditions

Pattern	Logic
<pre> \$\$\$\$ ##### \$\$\$\$ ##### \$\$\$\$ </pre>	<pre> for (int i=1 ; i<=5 ; i++) { for (int j=1 ; j<=5 ; j++){ if(i%2==0) System.out.print("#"); else System.out.print("\$"); } System.out.println(); } </pre>

Pattern	Logic
<pre> \$#\$#\$ \$#\$#\$ \$#\$#\$ \$#\$#\$ \$#\$#\$ </pre>	<pre> for (int i=1 ; i<=5 ; i++) { for (int j=1 ; j<=5 ; j++){ if(j%2==0) System.out.print("#"); else System.out.print("\$"); } System.out.println(); } </pre>

Pattern	Logic
<pre> 12345 67891 23456 78912 34567 </pre>	<pre> int k=1; for (int i=1 ; i<=5 ; i++){ for (int j=1 ; j<=5 ; j++){ System.out.print(k++%10); } System.out.println(); } </pre>
Pattern	Logic
<pre> 11111 12345 33333 12345 55555 </pre>	<pre> for (int i=1 ; i<=5 ; i++) { for (int j=1 ; j<=5 ; j++){ if(i%2==0) System.out.print(j); else System.out.print(i); } System.out.println(); } </pre>

Pattern	Logic
<pre> 55555 54321 33333 54321 11111 </pre>	<pre> for (int i=1 ; i<=5 ; i++) { for (int j=1 ; j<=5 ; j++){ if(i%2==0) System.out.print(j); else System.out.print(i); } System.out.println(); } </pre>

Patterns Plus Cross Swastika

Pattern	Logic
<pre> +++++++ +++++++ +++++++ +++++++ +++++++ +++++++ +++++++ </pre>	<pre> for (int i=1 ; i<=7 ; i++) { for (int j=1 ; j<=7 ; j++) { System.out.print("+"); } System.out.println(); } </pre>

Pattern	Logic
<pre> +++++++ + + + + + + + + + + +++++++ </pre>	<pre> for (int i=1 ; i<=7 ; i++){ for (int j=1 ; j<=7 ; j++){ if(i==1 i==7 j==1 j==7) System.out.print("+"); else System.out.print(" "); } System.out.println(); } </pre>
Pattern	Logic
<pre> + + + +++++++ + + + </pre>	<pre> for (int i=1 ; i<=7 ; i++){ for (int j=1 ; j<=7 ; j++){ if(i==4 j==4) System.out.print("+"); else System.out.print(" "); } System.out.println(); } </pre>

Pattern	Logic
<pre> +++++++ + + + + + + +++++++ + + + + + + +++++++ </pre>	<pre> for (int i=1 ; i<=7 ; i++){ for (int j=1 ; j<=7 ; j++){ if(i==1 i==4 i==7 j==1 j==4 j==7) System.out.print("+"); else System.out.print(" "); } System.out.println(); } </pre>
Pattern	Logic
<pre> + + + + + + + </pre>	<pre> for (int i=1 ; i<=7 ; i++){ for (int j=1 ; j<=7 ; j++){ if(i==j) System.out.print("+"); else System.out.print(" "); } System.out.println(); } </pre>

Pattern	Logic
<pre> + + + + + + + </pre>	<pre> for (int i=1 ; i<=7 ; i++){ for (int j=1 ; j<=7 ; j++){ if(j==8-i) System.out.print("+"); else System.out.print(" "); } System.out.println(); } </pre>
Pattern	Logic
<pre> + + + + + + + + + + + + + </pre>	<pre> for (int i=1 ; i<=7 ; i++){ for (int j=1 ; j<=7 ; j++){ if(i==j j==8-i) System.out.print("+"); else System.out.print(" "); } System.out.println(); } </pre>

Pattern	Logic
<pre> + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + </pre>	<pre> for (int i=1 ; i<=7 ; i++){ for (int j=1 ; j<=7 ; j++){ if(i==1 i==7 j==1 j==7 i==j j==8-i) System.out.print("+"); else System.out.print(" "); } System.out.println(); } </pre>

Pattern	Logic
<pre> +++++ + + + + + +++++ </pre>	<pre> for (int i=1 ; i<=7 ; i++){ for (int j=1 ; j<=7 ; j++){ if((i==1 && j<=4) j==4 (i==7 && j>=4)) System.out.print("+"); else System.out.print(" "); } System.out.println(); } </pre>

Pattern	Logic
<pre> + + + +++++++ + + + </pre>	<pre> for (int i=1 ; i<=7 ; i++){ for (int j=1 ; j<=7 ; j++){ if(i==4 (j==1 && i>=4) (j==7 && i<=4)) System.out.print("+"); else System.out.print(" "); } System.out.println(); } </pre>
Pattern	Logic
<pre> +++++ + + + + + +++++++ + + + + + + </pre>	<pre> for (int i=1 ; i<=7 ; i++){ for (int j=1 ; j<=7 ; j++){ if(i==4 (j==1 && i>=4) (j==7 && i<=4) (i==1 && j<=4) j==4 (i==7 && j>=4)) System.out.print("+"); else System.out.print(" "); } System.out.println(); } </pre>

Patterns – All Digits

Pattern	Logic
<pre> + + + + + + + + + + + + + + + + + + </pre>	<pre> for (int i=1 ; i<=7 ; i++){ for (int j=1 ; j<=5 ; j++){ if(i==1 i==7 j==1 j==5) System.out.print("+"); else System.out.print(" "); } System.out.println(); } </pre>
Pattern	Logic
<pre> + + + + + + + + + + + + + + + </pre>	<pre> for (int i=1 ; i<=7 ; i++){ for (int j=1 ; j<=5 ; j++){ if(j==4-i j==3 i==7) System.out.print("+"); else System.out.print(" "); } System.out.println(); } </pre>

Pattern	Logic
<pre> + + + + + + + + + + + + + + + + + + + </pre>	<pre> for (int i=1 ; i<=7 ; i++){ for (int j=1 ; j<=5 ; j++){ if(i==1 i==4 i==7 j==5&& i<=4) j==1&&i>=4)) System.out.print("+"); else System.out.print(" "); } System.out.println(); } </pre>

Pattern	Logic
<pre> + + + + + + + + + + + + + + + + + + + </pre>	<pre> for (int i=1 ; i<=7 ; i++){ for (int j=1 ; j<=5 ; j++){ if(i==1 i==4 i==7 j==5) System.out.print("+"); else System.out.print(" "); } System.out.println(); } </pre>

Pattern	Logic
<pre> + + + + + + + + + + + + + + + + + </pre>	<pre> for (int i=1 ; i<=7 ; i++){ for (int j=1 ; j<=7 ; j++){ if(j==5 i==5 j==6-i) System.out.print("+"); else System.out.print(" "); } System.out.println(); } </pre>
Pattern	Logic
<pre> + + + + + + + + + + + + + + + + + + + </pre>	<pre> for (int i=1 ; i<=7 ; i++){ for (int j=1 ; j<=5 ; j++){ if(i==1 i==4 i==7 j==1&&i<=4) j==5&&i>=4)) System.out.print(""); else System.out.print(" "); } System.out.println(); } </pre>

Pattern	Logic
<pre> + + + + + + + + + + + + + + + + + + + + + </pre>	<pre> for (int i=1 ; i<=7 ; i++){ for (int j=1 ; j<=5 ; j++){ if(i==1 j==1 i==4 i==7 (j==5&&i>=4)) System.out.print("*"); else System.out.print(" "); } System.out.println(); } </pre>

Pattern	Logic
<pre> + + + + + + + + + + + + </pre>	<pre> for (int i=1 ; i<=7 ; i++){ for (int j=1 ; j<=5 ; j++){ if(i==1 j==7-i) System.out.print("*"); else System.out.print(" "); } System.out.println(); } </pre>

Pattern	Logic
<pre> + + + + + + + + + + + + + + + + + + + + + + + </pre>	<pre> for (int i=1 ; i<=7 ; i++){ for (int j=1 ; j<=5 ; j++){ if(i==1 i==4 i==7 j==1 j==5) System.out.print("*"); else System.out.print(" "); } System.out.println(); } </pre>

Pattern	Logic
<pre> + + + + + + + + + + + + + + + + + + + + + </pre>	<pre> for (int i=1 ; i<=7 ; i++){ for (int j=1 ; j<=5 ; j++){ if(i==1 i==4 i==7 (j==1&&i<=4) j==5) System.out.print("*"); else System.out.print(" "); } System.out.println(); } </pre>

Patterns – All Alphabets

Pattern	Logic
<pre> + + + + + + + + + + + + + </pre>	<pre> for (int i=1 ; i<=5 ; i++){ for (int j=1 ; j<=9 ; j++){ if((j==6-i) ((i==3&&j>=3&&j<=7) j==4+i)) System.out.print(" "); else System.out.print(" "); } System.out.println(); } </pre>
Pattern	Logic
<pre> + + + + + + + + + + + + + + + + + + + + </pre>	<pre> for (int i=1 ; i<=7 ; i++){ for (int j=1 ; j<=7 ; j++){ if((j==1) ((i==1&&j<=6) ((i==4&&j<=6) ((i==7&&j<=6))) j==7&&i>1&&i<4) j==7&&i>4&&i<7)) System.out.print(" "); else System.out.print(" "); } System.out.println(); } </pre>
Pattern	Logic
<pre> + + + + + + + + + + + + + + </pre>	<pre> for (int i=1 ; i<=7 ; i++){ for (int j=1 ; j<=7 ; j++){ if((i==1&&j>1) j==1&&i>1&&i<7) ((i==7&&j>1)) System.out.print(" "); else System.out.print(" "); } System.out.println(); } </pre>
Pattern	Logic
<pre> + + + + + + + + + + + + + + + + + + </pre>	<pre> for (int i=1 ; i<=7 ; i++){ for (int j=1 ; j<=7 ; j++){ if(j==1 ((i==1&&j<7) ((i==7&&j<7) j==7&&i>1&&i< 7)) System.out.print(" "); else System.out.print(" "); } System.out.println(); } </pre>

Pattern	Logic
<pre> + + + + + + + + + + + + + + + + + + + </pre>	<pre> for (int i=1 ; i<=7 ; i++){ for (int j=1 ; j<=7 ; j++){ if(j==1 i==1 i==4 i==7) System.out.print("+"); else System.out.print(" "); } System.out.println(); } </pre>

Pattern	Logic
<pre> + + + + + + + + + + + + + + + </pre>	<pre> for (int i=1 ; i<=7 ; i++){ for (int j=1 ; j<=7 ; j++){ if(j==1 i==1 i==4) System.out.print("+"); else System.out.print(" "); } System.out.println(); } </pre>

Pattern	Logic
<pre> + + + + + + + + + + + + + + + + + + + </pre>	<pre> for (int i=1 ; i<=7 ; i++){ for (int j=1 ; j<=7 ; j++){ if(j==1 i==1 i==7 (j==7&& i>=4) i==4&& j>=4)) System.out.print("+"); else System.out.print(" "); } System.out.println(); } </pre>

Pattern	Logic
<pre> + + + + + + + + + + + + + + + + + </pre>	<pre> for (int i=1 ; i<=7 ; i++){ for (int j=1 ; j<=7 ; j++){ if(j==1 i==4 j==7) System.out.print("+"); else System.out.print(" "); } System.out.println(); } </pre>

Pattern	Logic
<pre> + + + + + + + + + + + + + + + </pre>	<pre> for (int i=1 ; i<=7 ; i++){ for (int j=1 ; j<=7 ; j++){ if(i==1 j==4 i==7) System.out.print("+"); else System.out.print(" "); } System.out.println(); } </pre>

Pattern	Logic
<pre> + + + + + + + + + + + </pre>	<pre> for (int i=1 ; i<=7 ; i++){ for (int j=1 ; j<=7 ; j++){ if(i==1 j==4 i==j+3) System.out.print("+"); else System.out.print(" "); } System.out.println(); } </pre>

Pattern	Logic
<pre> + + + + + + + + + + + + + </pre>	<pre> for (int i=1 ; i<=7 ; i++){ for (int j=1 ; j<=7 ; j++){ if(j==1 j==6-i (i==2+j)) System.out.print("+"); else System.out.print(" "); } System.out.println(); } </pre>

Pattern	Logic
<pre> + + + + + + + + + + </pre>	<pre> for (int i=1 ; i<=7 ; i++){ for (int j=1 ; j<=7 ; j++){ if(j==1 i==7) System.out.print("+"); else System.out.print(" "); } System.out.println(); } </pre>

Pattern	Logic
<pre> + + + + + + + + + + + + + + + + + + + </pre>	<pre> for (int i=1 ; i<=7 ; i++){ for (int j=1 ; j<=7 ; j++){ if(j==1 j==7 (i==j&&j<=4) j==8-i&&j>4)) System.out.print(" "); else System.out.print(" "); } System.out.println(); } </pre>

Pattern	Logic
<pre> + + + + + + + + + + + + + + + + + </pre>	<pre> for (int i=1 ; i<=7 ; i++){ for (int j=1 ; j<=7 ; j++){ if(j==1 j==7 (i==j)) System.out.print(" "); else System.out.print(" "); } System.out.println(); } </pre>

Numbers Triangle Patterns

Pattern	Logic
<pre> 1 12 123 1234 12345 </pre>	<pre> for (int i=1 ; i<=5 ; i++) { for (int j=1 ; j<=i ; j++){ System.out.print(j); } System.out.println(); } </pre>

Pattern	Logic
<pre> 1 21 321 4321 54321 </pre>	<pre> for (int i=1 ; i<=5 ; i++) { for (int j=i ; j>=1 ; j--){ System.out.print(j); } System.out.println(); } </pre>

Pattern	Logic
12345 1234 123 12 1	<pre> for (int i=5 ; i>=1 ; i--) { for (int j=1 ; j<=i ; j++){ System.out.print(j); } System.out.println(); } </pre>

Pattern	Logic
12345 2345 345 45 5	<pre> for (int i=1 ; i<=5 ; i++) { for (int j=i ; j<=5 ; j++){ System.out.print(j); } System.out.println(); } </pre>

Pattern	Logic
5 54 543 5432 54321	<pre> for (int i=5 ; i>=1 ; i--){ for (int j=5 ; j>=i ; j--){ System.out.print(j); } System.out.println(); } </pre>

Pattern	Logic
5 45 345 2345 12345	<pre> for (int i=5 ; i>=1 ; i--){ for (int j=i ; j<=5 ; j++){ System.out.print(j); } System.out.println(); } </pre>

Pattern	Logic
54321 5432 543 54 5	<pre> for (int i=1 ; i<=5 ; i++) { for (int j=5 ; j>=i ; j--){ System.out.print(j); } System.out.println(); } </pre>

Pattern	Logic
54321 4321 321 21 1	<pre> for (int i=5 ; i>=1 ; i--) { for (int j=i ; j>=1 ; j--){ System.out.print(j); } System.out.println(); } </pre>

Pattern	Logic
1 22 333 4444 55555	<pre> for (int i=1 ; i<=5 ; i++) { for (int j=1 ; j<=i ; j++){ System.out.print(i); } System.out.println(); } </pre>

Pattern	Logic
11111 2222 333 44 5	<pre> for (int i=1 ; i<=5 ; i++) { for (int j=i ; j<=5 ; j++){ System.out.print(i); } System.out.println(); } </pre>

Pattern	Logic
5 44 333 2222 11111	<pre> for (int i=5 ; i>=1 ; i--) { for (int j=i ; j<=5 ; j++){ System.out.print(i); } System.out.println(); } </pre>

Pattern	Logic
55555 4444 333 22 1	<pre> for (int i=5 ; i>=1 ; i--){ for (int j=1 ; j<=i ; j++){ System.out.print(i); } System.out.println(); } </pre>

Pattern	Logic
<pre> 1 23 456 7891 23456 </pre>	<pre> int k=1; for (int i=1 ; i<=5 ; i++){ for (int j=1; j<=i ; j++){ System.out.print(k++); if(k>9) k=1; } System.out.println(); } </pre>
Pattern	Logic
<pre> 12345 6789 123 45 6 </pre>	<pre> int k=1; for (int i=5 ; i>=1 ; i--){ for (int j=1; j<=i ; j++){ System.out.print(k++); if(k>9) k=1; } System.out.println(); } </pre>

Pattern	Logic
<pre> 1 21 321 4321 54321 </pre>	<pre> for (int i=1 ; i<=5 ; i++) { for (int j=i ; j<5 ; j++){ System.out.print(" "); } for (int k=i ; k>=1 ; k--){ System.out.print(k); } System.out.println(); } </pre>
Pattern	Logic
<pre> 12345 1234 123 12 1 </pre>	<pre> for (int i=5 ; i>=1 ; i--){ for (int j=i ; j<5 ; j++){ System.out.print(" "); } for (int k=1 ; k<=i ; k++){ System.out.print(k); } System.out.println(); } </pre>

Pattern	Logic
<pre> 12345 2345 345 45 5 </pre>	<pre> for (int i=1 ; i<=5 ; i++){ for (int j=1 ; j<i ; j++){ System.out.print(" "); } for (int k=i ; k<=5 ; k++){ System.out.print(k); } System.out.println(); } </pre>

Pattern	Logic
<pre> 5 45 345 2345 12345 </pre>	<pre> for (int i=5 ; i>=1 ; i--){ for (int j=1 ; j<i ; j++){ System.out.print(" "); } for (int k=i ; k<=5 ; k++){ System.out.print(k); } System.out.println(); } </pre>

Pattern	Logic
<pre> 5 54 543 5432 54321 </pre>	<pre> for (int i=5 ; i>=1 ; i--){ for (int j=1 ; j<i ; j++){ System.out.print(" "); } for (int k=5 ; k>=i ; k--){ System.out.print(k); } System.out.println(); } </pre>

Pattern	Logic
<pre> 54321 4321 321 21 1 </pre>	<pre> for (int i=5 ; i>=1 ; i--){ for (int j=i ; j<5 ; j++){ System.out.print(" "); } for (int k=i ; k>=1 ; k--){ System.out.print(k); } System.out.println(); } </pre>

Pattern	Logic
<pre> 54321 5432 543 54 5 </pre>	<pre> for (int i=1 ; i<=5 ; i++){ for (int j=1 ; j<i ; j++){ System.out.print(" "); } for (int k=5 ; k>=i ; k--){ System.out.print(k); } System.out.println(); } </pre>

Characters Triangle Patterns

Pattern	Logic
<pre> A AB ABC ABCD ABCDE </pre>	<pre> for (char i='A' ; i<='E' ; i++){ for (char j='A' ; j<=i ; j++){ System.out.print(j); } System.out.println(); } </pre>

Pattern	Logic
<pre> A BA CBA DCBA EDCBA </pre>	<pre> for (char i='A' ; i<='E' ; i++){ for (char j=i ; j>='A' ; j--){ System.out.print(j); } System.out.println(); } </pre>

Pattern	Logic
<pre> ABCDE ABCD ABC AB A </pre>	<pre> for (char i='E' ; i>='A' ; i--){ for (char j='A' ; j<=i ; j++){ System.out.print(j); } System.out.println(); } </pre>

Pattern	Logic
<pre> ABCDE BCDE CDE DE E </pre>	<pre> for (char i='A' ; i<='E' ; i++){ for (char j=i ; j<='E' ; j++){ System.out.print(j); } System.out.println(); } </pre>

Pattern	Logic
E ED EDC EDCB EDCBA	<pre> for (char i='E' ; i>='A' ; i--){ { for (char j='E' ; j>=i ; j--){ System.out.print(j); } System.out.println(); } </pre>

Pattern	Logic
E DE CDE BCDE ABCDE	<pre> for (char i='E' ; i>='A' ; i--){ for (char j=i ; j<='E' ; j++){ System.out.print(j); } System.out.println(); } </pre>

attern	Logic
EDCBA EDCB EDC ED E	<pre> for (char i='A' ; i<='E' ; i++){ for (char j='E' ; j>=i ; j--){ System.out.print(j); } System.out.println(); } </pre>

Pattern	Logic
EDCBA DCBA CBA BA A	<pre> for (char i='E' ; i>='A' ; i--){ for (char j=i ; j>='A' ; j--){ System.out.print(j); } System.out.println(); } </pre>

Star Triangle Patterns

Pattern	Logic
* ** *** **** *****	<pre> for (int i=1 ; i<=5 ; i++){ for (int j=1 ; j<=i ; j++){ System.out.print("*"); } System.out.println(); } </pre>

Pattern	Logic
<pre> ***** **** *** ** *</pre>	<pre> for (int i=5 ; i>=1 ; i--){ for (int j=1 ; j<=i ; j++){ System.out.print("*"); } System.out.println(); }</pre>
Pattern	Logic
<pre> * ** *** **** *****</pre>	<pre> for (int i=1 ; i<=5 ; i++){ for (int j=i ; j<5 ; j++){ System.out.print(" "); } for (int k=1 ; k<=i ; k++){ System.out.print("*"); } System.out.println(); }</pre>
Pattern	Logic
<pre> ***** **** *** ** *</pre>	<pre> for (int i=1 ; i<=5 ; i++){ for (int j=1 ; j<i ; j++){ System.out.print(" "); } for (int k=i ; k<=5 ; k++){ System.out.print("*"); } System.out.println(); }</pre>
Pattern	Logic
<pre> * ** *** **** ***** **** *** ** *</pre>	<pre> for(int i=1 ; i<10 ; i++){ if(i<=5){ for(int j=1 ; j<=i ; j++){ System.out.print("*"); } System.out.println(); } else{ for(int k=i ; k<10 ; k++){ System.out.print("*"); } System.out.println(); } }</pre>

Pattern	Logic
<pre> * ** *** **** ***** ***** **** *** ** *</pre>	<pre> for(int i=1 ; i<10 ; i++) { if(i<=5){ for(int x=i ; x<=5 ; x++){ System.out.print(" "); } for(int j=1 ; j<=i ; j++){ System.out.print("*"); } System.out.println(); } else{ for(int x=i ; x>=5 ; x--){ System.out.print(" "); } for(int k=i ; k<10 ; k++){ System.out.print("*"); } System.out.println(); } } </pre>

Pattern	Logic
<pre> ***** **** *** ** * ** *** **** *****</pre>	<pre> for(int i=1 ; i<10 ; i++) { if(i<=5){ for(int j=i ; j<=5 ; j++){ System.out.print("*"); } System.out.println(); } else{ for(int k=5 ; k<=i ; k++){ System.out.print("*"); } System.out.println(); } } </pre>

Pattern	Logic
<pre> ***** **** *** ** * ** *** **** ***** </pre>	<pre> for(int i=1 ; i<10 ; i++){ if(i<=5){ for(int x=1 ; x<=i ; x++){ System.out.print(" "); } for(int j=i ; j<=5 ; j++){ System.out.print("*"); } System.out.println(); } else{ for(int x=i ; x<10 ; x++){ System.out.print(" "); } for(int k=5 ; k<=i ; k++){ System.out.print("*"); } System.out.println(); } } </pre>

Binary Number Patterns

Pattern	Logic
<pre> 1 01 010 1010 10101 </pre>	<pre> int k=1; for (int i=1 ; i<=5 ; i++) { for (int j=1; j<=i ; j++){ System.out.print(k++%2); } System.out.println(); } </pre>

Pattern	Logic
<pre> 10101 0101 010 10 1 </pre>	<pre> int k=1; for (int i=5 ; i>=1 ; i--) { for (int j=1; j<=i ; j++){ System.out.print(k++%2); } System.out.println(); } </pre>

Pattern	Logic
<pre> 1 10 101 1010 10101 </pre>	<pre> for (int i=1 ; i<=5 ; i++) { for (int j=1; j<=i ; j++) { System.out.print(j%2); } System.out.println(); } </pre>
Pattern	Logic
<pre> 1 00 111 0000 11111 </pre>	<pre> for (int i=1 ; i<=5 ; i++){ for (int j=1; j<=i ; j++) { System.out.print(i%2); } System.out.println(); } </pre>
Pattern	Logic
<pre> 11111 0000 111 00 1 </pre>	<pre> for (int i=5 ; i>=1 ; i--) { for (int j=1; j<=i ; j++) { System.out.print(i%2); } System.out.println(); } </pre>

Border Only Patterns

Pattern	Logic
<pre> * ** * * * * * * * * * * * * ***** </pre>	<pre> int n=7; for(int i=1 ; i<=n ; i++) { for(int j=1 ; j<=i ; j++) { if(i==1 i==n j==1 j==i) System.out.print("*"); else System.out.print(" "); } System.out.println(); } </pre>

Pattern	Logic
<pre> ***** * * * * * * * * * * ** *</pre>	<pre> int n=7; for(int i=n ; i>=1 ; i--) { for(int j=1 ; j<=i ; j++){ if(i==1 i==n j==1 j==i) System.out.print("*"); else System.out.print(" "); } System.out.println(); }</pre>

Pattern	Logic
<pre> ***** * * * * * * * * * * ** *</pre>	<pre> for(int i=1 ; i<=7 ; i++) { for(int j=1 ; j<i ; j++){ System.out.print(" "); } for(int k=i ; k<=7 ; k++){ if(i==1 i==7 k==i k==7) System.out.print("*"); else System.out.print(" "); } System.out.println(); }</pre>

Pattern	Logic
<pre> * ** * * * * * * * * * * *****</pre>	<pre> for(int i=1 ; i<=7 ; i++) { for(int j=i ; j<7 ; j++){ System.out.print(" "); } for(int k=1 ; k<=i ; k++){ if(i==1 i==7 k==1 k==i) System.out.print("*"); else System.out.print(" "); } System.out.println(); }</pre>

Pyramid Patterns

Pattern	Logic
<pre> * *** ***** * * * </pre>	<pre> int n=7; for(int i=1 ; i<=n ; i++) { for(int j=i ; j<n ; j++){ System.out.print(" "); } for(int k=1 ; k<=2*i-1 ; k++){ System.out.print("*"); } System.out.println(); } </pre>

Pattern	Logic
<pre> * * * * * * * * * * * </pre>	<pre> int n=7; for(int i=1 ; i<=n ; i++) { for(int j=i ; j<n ; j++){ System.out.print(" "); } for(int k=1 ; k<=2*i-1 ; k++){ if(i==1 i==n k==1 k==2*i-1) System.out.print("*"); else System.out.print(" "); } System.out.println(); } </pre>

Pattern	Logic
<pre> 1 222 33333 4444444 55555555 </pre>	<pre> int n=7; for(int i=1 ; i<=n ; i++) { for(int j=i ; j<n ; j++){ System.out.print(" "); } for(int k=1 ; k<=2*i-1 ; k++){ System.out.print(i); } System.out.println(); } </pre>

Reverse Pyramid Patterns

Pattern	Logic
<pre> ***** ***** ***** *** *</pre>	<pre> int n=7; for(int i=n ; i>=1 ; i--) { for(int j=i ; j<n ; j++){ System.out.print(" "); } for(int k=1 ; k<=2*i-1 ; k++){ System.out.print("*"); } System.out.println(); }</pre>

Pattern	Logic
<pre> ***** * * * * * * *</pre>	<pre> int n=7; for(int i=n ; i>=1 ; i--) { for(int j=i ; j<n ; j++){ System.out.print(" "); } for(int k=1 ; k<=2*i-1 ; k++){ if(i==1 i==n k==1 k==2*i-1) System.out.print("*"); else System.out.print(" "); } System.out.println(); }</pre>

Pattern	Logic
<pre> 55555555 4444444 33333 222 1</pre>	<pre> int n=7; for(int i=n ; i>=1 ; i--) { for(int j=i ; j<n ; j++){ System.out.print(" "); } for(int k=1 ; k<=2*i-1 ; k++){ System.out.print(i); } System.out.println(); }</pre>

Pattern	Logic
<pre> 1 1 1 1 2 1 1 3 3 1 1 4 6 4 1 1 5 10 10 5 1 </pre>	<pre> int r=6, c=1; for(int i=0; i<r ; i++) { for(int s=1; s<r-i; s++) { System.out.print(" "); } for(int j=0; j<=i; j++) { if (j==0 i==0) c=1; else c=c*(i-j + 1)/j; System.out.printf("%4d", c); } System.out.println(); } </pre>
<pre> 123454321 1234321 12321 121 1 </pre>	<pre> for (int i = 5; i >= 1; i--){ for (int j = 5 - i; j >= 1; j--){ System.out.print(" "); } for (int j = 1; j <= i; j++){ System.out.print(j); } for (int j = i - 1; j >= 1; j--){ System.out.print(j); } System.out.println(); } </pre>

Complex Pattern programs

Pattern	Logic
<pre> 1 121 12321 1234321 123454321 </pre>	<pre> int n=7; for(int i=1 ; i<=n ; i++){ for(int j=i ; j<n ; j++){ System.out.print(" "); } for(int k=1 ; k<=i ; k++){ System.out.print(k); } for(int l=i-1 ; l>=1 ; l--){ System.out.print(l); } System.out.println(); } </pre>

Pattern	Logic
<pre> 123454321 1234321 12321 121 1 </pre>	<pre> int n=7; for(int i=n ; i>=1 ; i--){ for(int j=i ; j<n ; j++){ System.out.print(" "); } for(int k=1 ; k<=i ; k++){ System.out.print(k); } for(int l=i-1 ; l>=1 ; l--){ System.out.print(l); } System.out.println(); } </pre>
Pattern	Logic
<pre> 1 212 32123 4321234 543212345 </pre>	<pre> int n=7; for(int i=1 ; i<=n ; i++){ for(int j=i ; j<n ; j++){ System.out.print(" "); } for(int k=i ; k>=1 ; k--){ System.out.print(k); } for(int l=1+1 ; l<=i ; l++){ System.out.print(l); } System.out.println(); } </pre>
Pattern	Logic
<pre> 543212345 4321234 32123 212 1 </pre>	<pre> int n=7; for(int i=n ; i>=1 ; i--){ for(int j=i ; j<n ; j++){ System.out.print(" "); } for(int k=i ; k>=1 ; k--){ System.out.print(k); } for(int l=1+1 ; l<=i ; l++){ System.out.print(l); } System.out.println(); } </pre>

Pattern	Logic
<pre> * *** ***** ********* *********** * </pre>	<pre> int n=7; for(int i=1 ; i<=n ; i++){ for(int j=i ; j<n ; j++){ System.out.print(" "); } for(int k=1 ; k<=2*i-1 ; k++){ System.out.print("* "); } System.out.println(); } for(int i=n-1 ; i>=1 ; i--){ for(int j=i ; j<n ; j++){ System.out.print(" "); } for(int k=1 ; k<=2*i-1 ; k++){ System.out.print("* "); } System.out.println(); } </pre>

Methods

Variable: is used to stores data

Method: is used to perform operations on data

Syntax	Example
<pre>returntype identity(arguments) { statements; }</pre>	<pre>int add(int a, int b) { int c=a+b; return c; }</pre>

Classification of Methods: Based on taking input and returning output, methods are classified into 4 types.

1. No arguments and No return values
2. With arguments and No return values
3. With arguments and with return values
4. No arguments and with return values

No input - No output	with input - no output	With input - With output	No input - With output
<pre>void m1() { logic ; return ; }</pre>	<pre>void m2(int a, int b) { logic ; return ; }</pre>	<pre>char m3(String x, int y) { logic ; return 'a'; }</pre>	<pre>double m4() { logic ; return 2.34; }</pre>
<u>Invoke:</u> m1();	<u>Invoke :</u> m2(10, 20); <u>Invoke:</u> int x=10, y=20; m2(x, y);	<u>Invoke :</u> char x = m3("abcd" , 5);	<u>Invoke :</u> double d = m4() ;

No arguments and No return values method:

Static method	Instance method
<pre>class Code { public static void main(String[] args) { Code.fun(); } static void fun(){ System.out.println("fun"); } }</pre>	<pre>class Code { public static void main(String[] args) { Code obj = new Code(); obj.fun(); } void fun(){ System.out.println("fun"); } }</pre>

With arguments and No return values:

Static Method	Instance Method
<pre> class Code { static void main(String[] args) { Code.isEven(4); Code.isEven(13); } static void isEven(int n){ if(n%2==0) S.o.p(n+" is Even"); else S.o.p(n+" is Not even"); } } </pre>	<pre> class Code { static void main(String[] args) { Code obj = new Code(); obj.isEven(4); obj.isEven(13); } static void isEven(int n){ if(n%2==0) S.o.p(n + " is Even"); else S.o.p(n + " is Not even"); } } </pre>

With arguments with return values:

Static Method	Instance Method
<pre> class Code { static void main(String[] args) { int r1 = Code.add(10, 20); S.o.p(r1); } static int add(int a, int b){ return a+b; } } </pre>	<pre> class Code { static void main(String[] args) { Code obj = new Code(); int r1 = obj.add(10, 20); S.o.p(r1); } int add(int a, int b){ return a+b; } } </pre>

No arguments and with return values:

Static Method	Instance Method
<pre> class Code { static void main(String[] args) { double PI = Code.getPI(); S.o.p("PI val : " + PI); } static double getPI(){ double PI = 3.142; return PI; } } </pre>	<pre> class Code { static void main(String[] args) { Code obj = new Code(); double PI = obj.getPI(); S.o.p("PI val : " + PI); } double getPI(){ double PI = 3.142; return PI; } } </pre>

Recursion

Recursion:

- Calling method itself is called Recursion.
- Invoking the method from the body of same method.

Program to display 1 to 10 numbers using recursion:

```
class Code {  
    public static void main(String[] args) {  
        Code.print(1);  
    }  
    static void print(int n){  
        System.out.println(n);  
        if(n<10)  
            print(n+1);  
    }  
}
```

Program to display 10 to 1 number using recursion:

```
class Code {  
    public static void main(String[] args) {  
        Code.print(1);  
    }  
    static void print(int n){  
        if(n<10)  
            print(n+1);  
        System.out.println(n);  
    }  
}
```

Program to display Sum of First N numbers using recursion:

```
class Code {  
    public static void main(String[] args) {  
        int n=5;  
        int sum = Code.calc(n);  
        System.out.println(sum);  
    }  
    static int calc(int n){  
        if(n>0)  
            return n + calc(n-1);  
        else  
            return n;  
    }  
}
```

Program to display Factorial of given number using recursion:

```
class Code {  
    public static void main(String[] args) {  
        int n=5;  
        int f = Code.fact(n);  
        System.out.println(f);  
    }  
    static int fact(int n){  
        if(n==0)  
            return 1;  
        else  
            return n*fact(n-1);  
    }  
}
```

Program to display Fibonacci series using recursion:

```
import java.util.Scanner;  
class Code{  
    public static void main(String[] args) {  
        Scanner sc = new Scanner(System.in);  
        System.out.print("Enter limit : ");  
        int n = sc.nextInt();  
        for(int i=1; i<=n; i++){  
            System.out.print(fib(i)+" ");  
        }  
    }  
    static int fib(int n){  
        if(n==1)  
            return 0;  
        else if(n == 2)  
            return 1;  
        else  
            return fib(n-2)+fib(n-1);  
    }  
}
```

Menu Driven Programs

Arithmetic Operations Menu Driven Program using if block:

```
import java.util.Scanner;
class Code{
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        while(true){
            System.out.println("1. Add");
            System.out.println("2. Subtract");
            System.out.println("3. Multiply");
            System.out.println("4. Divide");
            System.out.println("5. Quit");
            System.out.print("Enter your choice : ");
            int ch = sc.nextInt();
            if(ch==1){
                System.out.println("Enter 2 numbers : ");
                int a=sc.nextInt();
                int b=sc.nextInt();
                int c=a+b;
                System.out.println(a + " + " + b + " = " + c);
            }
            else if(ch==2){
                System.out.println("Enter 2 numbers : ");
                int a=sc.nextInt();
                int b=sc.nextInt();
                int c=a-b;
                System.out.println(a + " - " + b + " = " + c);
            }
            else if(ch==3){
                System.out.println("Enter 2 numbers : ");
                int a=sc.nextInt();
                int b=sc.nextInt();
                int c=a*b;
                System.out.println(a + " * " + b + " = " + c);
            }
            else if(ch==4){
                System.out.println("Enter 2 numbers : ");
                int a=sc.nextInt();
                int b=sc.nextInt();
                int c=a/b;
                System.out.println(a + " / " + b + " = " + c);
            }
        }
    }
}
```

```
    }
    else if(ch==5){
        System.out.println("End of Program");
        System.exit(1);
    }
    else{
        System.out.println("Invalid Choice");
    }
}
}
```

Output:

```
1. Add
2. Subtract
3. Multiply
4. Divide
5. Quit
Enter your choice : 4
Enter 2 numbers :
5
2
5 / 2 = 2
1. Add
2. Subtract
3. Multiply
4. Divide
5. Quit
Enter your choice : 5
End of Program
```

Arithmetic Operations Menu Driven Approach – Switch Case:

```
import java.util.Scanner;
class Code
{
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        while(true){
            System.out.println("1. Add");
            System.out.println("2. Subtract");
            System.out.println("3. Multiply");
            System.out.println("4. Divide");
            System.out.println("5. Quit");
```

```
System.out.println("Enter your choice : ");
int ch = sc.nextInt();
int a=0, b=0;
if(ch>=1 && ch<=4){
    System.out.println("Enter 2 numbers : ");
    a=sc.nextInt();
    b=sc.nextInt();
}
switch(ch){
    case 1 : System.out.println(a+" + "+b+" = "+(a+b));
              break;
    case 2 : System.out.println(a+" - "+b+" = "+(a-b));
              break;
    case 3 : System.out.println(a+" * "+b+" = "+(a*b));
              break;
    case 4 : System.out.println(a+" / "+b+" = "+(a/b));
              break;
    case 5 : System.out.println("End of Program");
              System.exit(1);
    default: System.out.println("Invalid choice");
}
}
}
```

Arithmetic Operations Menu Driver Approach – using Methods:

```
import java.util.Scanner;
class Code
{
    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        while(true)
        {
            System.out.println("1. Add");
            System.out.println("2. Subtract");
            System.out.println("3. Multiply");
            System.out.println("4. Divide");
            System.out.println("5. Quit");

            System.out.println("Enter your choice : ");
            int ch = sc.nextInt();
```

```
int a=0, b=0;
if(ch>=1 && ch<=4)
{
    System.out.println("Enter 2 numbers : ");
    a=sc.nextInt();
    b=sc.nextInt();
}

switch(ch){
    case 1 : add(a,b);
              break;
    case 2 : subtract(a,b);
              break;
    case 3 : multiply(a,b);
              break;
    case 4 : divide(a,b);
              break;
    case 5 : System.out.println("End of Program");
              System.exit(1);
    default: System.out.println("Invalid choice");
}
}

static void add(int a, int b)
{
    System.out.println(a+" + "+b+" = "+(a+b));
}
static void subtract(int a, int b)
{
    System.out.println(a+" - "+b+" = "+(a-b));
}
static void multiply(int a, int b)
{
    System.out.println(a+" * "+b+" = "+(a*b));
}
static void divide(int a, int b)
{
    System.out.println(a+" / "+b+" = "+(a/b));
}
}
```


Arrays

Program to Find the length of given array

- 'length' is a pre-defined instance variable that returns length of Array.

class Code

```
{  
    public static void main(String[] args) {  
        int[] arr = {4, 2, 8, 9, 1, 6, 7, 4};  
        System.out.println("Length is : " + arr.length);  
    }  
}
```

Display first and last element of array:

class Code

```
{  
    public static void main(String[] args) {  
        int[] arr = {4, 2, 8, 9, 1, 6, 7, 4};  
        int n = arr.length;  
        int first = arr[0];  
        int last = arr[n-1];  
        System.out.println("First element is : " + first);  
        System.out.println("Last element is : " + last);  
    }  
}
```

Program to display array elements

class Code

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

For Each Loop:

- It is called Enhanced for loop in Java
- For-each loop is used to process Array or Collection elements easily.
- For-each loop process elements from start to end without using index.
- For-each loop process elements only in forward direction.

Program to display element using for-each loop:

class Code

```
{
    public static void main(String[] args) {
        int[] arr = {4, 2, 8, 9, 1, 6, 7, 5};
        System.out.println("Array elements using for-each loop : ");
        for (int x : arr){
            System.out.println(x);
        }
    }
}
```

Check the first element of Array is Even or not

class Code

```
{
    public static void main(String[] args) {
        int[] arr = {4, 2, 8, 9, 1, 6, 7, 5};
        int first = arr[0];
        if(first%2==0)
            System.out.println("First element is Even");
        else
            System.out.println("First element is not Even");
    }
}
```

Check the sum of first and last elements in Array equals to 10 or not

class Code

```
{
    public static void main(String[] args) {
        int[] arr = {4, 2, 8, 9, 1, 6, 7, 5};
        int first = arr[0];
        int last = arr[arr.length-1];
        if(first+last==10)
            System.out.println("Equals");
        else
            System.out.println("Not Equals");
    }
}
```

Display the median value of array:

- If the length is ODD, Print middle element
- If the length is EVEN, Print average of middle 2 elements

```
class Code
{
    public static void main(String[] args) {
        int[] arr = {6, 2, 8, 9, 1, 6, 7, 5};
        int n = arr.length;
        if(n%2!=0){
            System.out.println("Mean : " + arr[n/2]);
        }
        else{
            int x = arr[n/2-1];
            int y = arr[n/2];
            System.out.println("Mean : " + ((x+y)/2));
        }
    }
}
```

Swap array elements of specified locations:

```
import java.util.Scanner;
class Code {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        int[] arr = {6, 3, 9, 1, 2, 8, 4, 5};
        int n = arr.length;
        System.out.println("Enter two locations from 0 to " + (n-1));
        int a = scan.nextInt();
        int b = scan.nextInt();
        System.out.print("Before swap, the Array is: ");
        for (int x : arr){
            System.out.print(x + " ");
        }
        System.out.println();
        int temp = arr[a];
        arr[a] = arr[b];
        arr[b] = temp;
        System.out.print("After swap, the Array is: ");
        for (int x : arr){
            System.out.print(x + " ");
        }
        System.out.println();
    }
}
```

Program to find the sum of array elements:

```
class Code{
    public static void main(String[] args){
        int[] arr = {6, 3, 9, 1, 2, 8, 4, 5};
        int sum=0;
        for (int i=0 ; i<arr.length ; i++){
            sum = sum + arr[i];
        }
        System.out.println("Sum of elements : " + sum);
    }
}
```

Program to find the average of array elements

```
class Code {
    public static void main(String[] args) {
        int[] arr = {6, 3, 9, 1, 2, 8, 4, 5};
        double sum=0;
        for (int x : arr){
            sum = sum + x;
        }
        double avg = sum/arr.length;
        System.out.println("Average of array elements : " + avg);
    }
}
```

Display Array element which are greater than average of all elements:

```
class Code {
    public static void main(String[] args) {
        int[] arr = {6, 3, 9, 1, 2, 8, 4, 5};
        double sum=0;
        for (int x : arr){
            sum = sum + x;
        }
        double avg = sum/arr.length;
        for(int x : arr)
        {
            if(x>avg)
                System.out.println(x);
        }
    }
}
```

Program to increase each value by one in array

class Code

```
{
    public static void main(String[] args) {
        int[] arr = {6, 3, 9, 1, 2, 8, 4, 5};
        for (int i=0 ; i<arr.length ; i++){
            arr[i]++;
        }
        System.out.println("Array elements : ");
        for (int x : arr){
            System.out.print(x + " ");
        }
    }
}
```

Program to print even numbers of array using for each loop:

class Code

```
{
    public static void main(String[] args){
        int[] arr = {6, 3, 9, 1, 2, 8, 4, 5};
        System.out.println("Even numbers of array : ");
        for (int x : arr){
            if(x%2==0)
                System.out.println(x);
        }
    }
}
```

Program to display odd numbers in the array using for loop

class Code

```
{
    public static void main(String[] args) {
        int[] arr = {6, 3, 9, 1, 2, 8, 4, 5};
        System.out.println("Odd numbers of array using for loop : ");
        for (int i=0 ; i<arr.length ; i++){
            if(arr[i]%2!=0)
                System.out.println(arr[i]);
        }
    }
}
```

Program to find the sum of even numbers in the array:

```
class Code{
    public static void main(String[] args) {
        int[] arr = {6, 3, 9, 1, 2, 8, 4, 5};
        int sum=0;
        for (int i=0 ; i<arr.length ; i++){
            if(arr[i]%2==0)
                sum=sum+arr[i];
        }
        System.out.println("Sum of even numbers : " + sum);
    }
}
```

Program Read array values from user:

```
import java.util.Scanner;
class Code {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter array size : ");
        int n = scan.nextInt();
        int[] arr = new int[n];
        System.out.println("Enter " + n + " elements : ");
        for (int i=0 ; i<n ; i++)
            arr[i] = scan.nextInt();
        System.out.println("Array elements are : ");
        for (int x : arr)
            System.out.print(x + " ");
    }
}
```

Program to find the smallest element in the array:

```
class Code {
    public static void main(String[] args) {
        int[] arr = {4, 2, 8, 1, 9, 6, 3, 7};
        int small = arr[0];
        for (int i=1 ; i<arr.length ; i++){
            if(arr[i] < small)
                small = arr[i];
        }
        System.out.println("Smallest element is : " + small);
    }
}
```

Check specified element is duplicated or not in Array:

```
import java.util.Scanner;
class Code
{
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        int[] arr = {4, 2, 8, 1, 9, 6, 2, 7};
        System.out.print("Enter element : ");
        int ele = scan.nextInt();
        int count=0;
        for (int i=0 ; i<arr.length ; i++){
            if(arr[i] == ele)
                count++;
        }
        if(count==0)
            System.out.println("Element Not found");
        else if(count==1)
            System.out.println("Element not duplicated");
        else
            System.out.println("Duplicated element");
    }
}
```

Program to display array elements in reverse order

```
class Code
{
    public static void main(String[] args) {
        int[] arr = {5, 9, 2, 1, 8, 4, 6};
        System.out.println("Reverse Array : ");
        for (int i=arr.length-1 ; i>=0 ; i--){
            System.out.println(arr[i]);
        }
    }
}
```

Print only perfect numbers in the given array

```
class Code
{
    public static void main(String[] args) {
        int[] arr = {6, 3, 9, 1, 2, 8, 4, 5};
        for (int i=0 ; i<arr.length ; i++){
            int n = arr[i];
            int sum=0;
```

```
        for (int j=1 ; j<n ; j++){
            if(n%j==0)
                sum=sum+j;
        }
        if(n==sum)
            System.out.println(n + " is perfect");
    }
}
```

Program to Display the First even number in the Array:

```
class Code
{
    public static void main(String[] args) {
        int[] arr = {5, 9, 2, 1, 8, 4, 6};
        for (int i=0 ; i<arr.length ; i++){
            if(arr[i]%2==0){
                System.out.println("First Even Number : " + arr[i]);
                break;
            }
        }
    }
}
```

Program to display the Last Odd Number in the array:

```
class Code
{
    public static void main(String[] args) {
        int[] arr = {5, 9, 2, 1, 8, 4, 6};
        for (int i=arr.length-1 ; i>=0 ; i--){
            if(arr[i]%2!=0){
                System.out.println("Last Odd Number : " + arr[i]);
                break;
            }
        }
    }
}
```

Program to Swap the First even number and Last odd number in the array:

```
class Code
{
    public static void main(String[] args) {
        int[] arr = {5, 2, 3, 8, 1, 4, 6};
```



```
int i, j;
for (i=0 ; i<arr.length ; i++){
    if(arr[i]%2==0)
        break;
}
for (j=arr.length-1 ; j>=0 ; j--){
    if(arr[j]%2!=0)
        break;
}
int temp=arr[i];
arr[i]=arr[j];
arr[j]=temp;
System.out.println("Array after swap : ");
for (i=0 ; i<arr.length ; i++){
    System.out.println(arr[i]);
}
}
```

Program to Merge 2 arrays:

```
import java.util.Arrays;
class Code
{
    public static void main(String[] args)
    {
        int[] a1 = {10,20,30,40};
        int[] a2 = {50,60,70};
        int[] merge = new int[a1.length+a2.length];
        System.out.println("a1 array : " + Arrays.toString(a1));
        System.out.println("a2 array : " + Arrays.toString(a2));
        for (int i=0 ; i<a1.length ; i++){
            merge[i] = a1[i];
        }
        for (int i=0 ; i<a2.length ; i++){
            merge[a1.length+i] = a2[i];
        }
        System.out.println("Merged array : " + Arrays.toString(merge));
    }
}
```

Program to reverse all elements in the given array without second array.

```
class Code {  
    public static void main(String[] args) {  
        int[] arr = {5, 9, 2, 1, 8, 4, 6};  
        int i=0;  
        int j=arr.length-1;  
        while(i<j){  
            int temp = arr[i];  
            arr[i] = arr[j];  
            arr[j] = temp;  
            i++;  
            j--;  
        }  
        System.out.println("Reverse Array : ");  
        for (i=0 ; i<arr.length ; i++){  
            System.out.println(arr[i]);  
        }  
    }  
}
```

Program to search for an element in the array:

Linear Search:

- Linear search algorithm is used to search an element in the array.
- It is index based searching.
- It starts from index 0 and continues searching upto last index.
- If element found, it display "Element found @ location" else display error message that "Element not found"

```
import java.util.Scanner ;  
class Demo {  
    public static void main(String[] args) {  
        Scanner scan = new Scanner(System.in);  
        System.out.print("Enter array size : ");  
        int n = scan.nextInt();  
        int arr[ ] = new int[n];  
        System.out.println("Enter " + n + " values : ");  
        for (int i=0 ; i<n ; i++){  
            arr[i] = scan.nextInt();  
        }  
        System.out.println("Enter element to search : ");  
        int ele = scan.nextInt();  
        boolean found=false;  
        for (int i=0 ; i<n ; i++){
```

```
        if(arr[i] == ele){
            System.out.println("Element found at location : " + i);
            found = true;
            break;
        }
    }
    if(!found)
        System.out.println("Element not found");
}
}
```

Program to pass array as parameter to method.

```
class Code
{
    public static void main(String[] args) {
        int[] arr = {10, 20, 30, 40, 50};
        Code.display(arr);
    }
    static void display(int[] x){
        System.out.println("Array elements : ");
        for (int i=0 ; i<x.length ; i++){
            System.out.println(x[i]);
        }
    }
}
```

Program to return an array from the method

```
class Code
{
    public static void main(String[] args) {
        int[] x = Code.getArray();
        System.out.println("Array elements : ");
        for (int i=0 ; i<x.length ; i++){
            System.out.println(x[i]);
        }
    }
    static int[] getArray()
    {
        int[] arr = {10, 20, 30, 40, 50};
        return arr;
    }
}
```

Program to copy elements of one array into another array.

```
import java.util.Arrays;
class Code
{
    public static void main(String[] args) {
        int src[ ] = {10,20,30,40,50};
        int dest[ ] = new int[src.length];
        int i=0;
        for( int ele : src ){
            dest[i] = ele ;
            i++;
        }
        System.out.println("Copied array : " + Arrays.toString(dest));
    }
}
```

Program to merge two arrays

```
class Code
{
    public static void main(String[] args)
    {
        int[] x = {10, 20, 30, 40};
        int[] y = {50, 60, 70};
        int[] z = new int[x.length + y.length];
        for (int i=0 ; i<x.length ; i++){
            z[i] = x[i];
        }
        for (int j=0 ; j<y.length ; j++){
            z[x.length+j] = y[j];
        }
        System.out.println("Merged array : ");
        for (int i=0 ; i<z.length ; i++){
            System.out.println(z[i]);
        }
    }
}
```

Program to create Array with only even numbers from the given array:

```
class Code
{
    public static void main(String[] args) {
        int[] arr = {5, 2, 3, 8, 1, 4, 6, 7, 9};
        int count=0;
```

```
        for (int i=0 ; i<arr.length ; i++){
            if(arr[i]%2==0)
                count++;
        }
        int[] evens = new int[count];
        int j=0;
        for (int i=0 ; i<arr.length ; i++){
            if(arr[i]%2==0){
                evens[j]=arr[i];
                j++;
            }
        }
        System.out.println("Even numbers array : ");
        for (int i=0 ; i<evens.length ; i++){
            System.out.println(evens[i]);
        }
    }
}
```

Program to divide the given array into 2 arrays with even numbers and odd numbers:

class Code

```
{
    public static void main(String[] args) {
        int[] arr = {5, 2, 3, 8, 1, 4, 6, 7, 9};
        int count=0;
        for (int i=0 ; i<arr.length ; i++){
            if(arr[i]%2==0)
                count++;
        }
        int[] evens = new int[count];
        int[] odds = new int[arr.length-count];
        int x=0, y=0;
        for (int i=0 ; i<arr.length ; i++){
            if(arr[i]%2==0){
                evens[x]=arr[i];
                x++;
            }
            else{
                odds[y]=arr[i];
                y++;
            }
        }
        System.out.println("Even numbers array : ");
    }
}
```

```
        for (int i=0 ; i<evens.length ; i++){
            System.out.println(evens[i]);
        }
        System.out.println("Odd numbers array : ");
        for (int i=0 ; i<odds.length ; i++){
            System.out.println(odds[i]);
        }
    }
}
```

Sort array elements and display as String:

```
import java.util.Random;
import java.util.Arrays;
class SortArray
{
    public static void main(String[] args) {
        Random rand = new Random();
        int arr[] = new int[5];
        for(int i=0 ; i<5 ; i++){
            arr[i] = rand.nextInt(100);
        }
        System.out.println("Before sort : " + Arrays.toString(arr));
        Arrays.sort(arr);
        System.out.println("After sort : " + Arrays.toString(arr));
    }
}
```

Program to search for an element using Array.binarySearch() method.

```
import java.util.Scanner ;
import java.util.Arrays ;
class Search
{
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter array size : ");
        int n = scan.nextInt();
        int arr[ ] = new int[n];
        System.out.println("Enter " + n + " values : ");
        for (int i=0 ; i<n ; i++){
            arr[i] = scan.nextInt();
        }
        System.out.print("Enter element to be searched : ");
        int key = scan.nextInt();
```

```
int result = Arrays.binarySearch(arr,key);
if (result < 0)
    System.out.println("Element is not found!");
else
    System.out.println("Element is found at index : " + result);
}
}
```

Bubble Sort Program in Java:

```
import java.util.Scanner ;
class Demo
{
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter array size : ");
        int n = scan.nextInt();
        int arr[ ] = new int[n];
        System.out.println("Enter " + n + " values : ");
        for (int i=0 ; i<n ; i++)
            arr[i] = scan.nextInt();
        System.out.println("Before sort : ");
        for(int ele : arr){
            System.out.print(ele + "\t");
        }
        System.out.println();
        for (int i=0 ; i<n-1 ; i++){
            for (int j=0 ; j<n-1-i ; j++ ){
                if(arr[j] > arr[j+1]){
                    int temp = arr[j];
                    arr[j] = arr[j+1];
                    arr[j+1] = temp;
                }
            }
        }

        System.out.println("After sort : ");
        for(int ele : arr){
            System.out.print(ele + "\t");
        }
        System.out.println();
    }
}
```

Binary Search Algorithm and Program in Java:

```
import java.util.Scanner ;
class Demo {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter array size : ");
        int n = scan.nextInt();
        int arr[ ] = new int[n];
        System.out.println("Enter " + n + " values : ");
        for (int i=0 ; i<n ; i++){
            arr[i] = scan.nextInt();
        }
        System.out.println("Enter element to search : ");
        int ele = scan.nextInt();
        boolean found=false;
        int low=0;
        int high=n-1;
        while(low <= high){
            int mid = (low+high)/2;
            if(ele < arr[mid]){
                high = mid-1;
            }
            else if(ele > arr[mid]){
                low = mid+1;
            }
            else if(ele == arr[mid]){
                System.out.println("Found at location : " + mid);
                found = true;
                break;
            }
        }
        if(!found)
            System.out.println("Element not found");
    }
}
```

Program to display pair of elements in array whose sum equals to 10:

```
class Code {
    public static void main(String[] args){
        int[] arr = {6, 3, 9, 1, 2, 8, 4, 5};
        int n=8;
        for (int i=0 ; i<n-1 ; i++){
            for (int j=i+1 ; j<n ; j++){
```



```
                if(arr[i]+arr[j]==10)
                    System.out.println("(" + arr[i] + "," + arr[j] + ")");
            }
        }
    }
}
```

Program to replace all duplicates with 0:

```
class Code{
    public static void main(String[] args){
        int[] arr = {6, 2, 3, 1, 6, 7, 6, 3, 6, 2, 3, 2, 1};
        for (int i=0 ; i<arr.length-1 ; i++){
            for (int j=i+1 ; j<arr.length ; j++){
                if(arr[i]==arr[j])
                    arr[j]=0;
            }
        }
        System.out.println("Final array is : ");
        for (int x : arr)
            System.out.println(x);
    }
}
```

Program to display pair of elements whose sum is 10 (contains duplicates in array):

```
class Code {
    public static void main(String[] args) {
        int[] arr = {6, 3, 9, 1, 2, 8, 4, 5, 7, 6, 2, 9, 1, 4};
        int n=14;
        for (int i=0 ; i<arr.length-1 ; i++){
            for (int j=i+1 ; j<arr.length ; j++){
                if(arr[i]==arr[j])
                    arr[j]=0;
            }
            for (int k=i+1 ; k<n ; k++){
                if(arr[i]+arr[k]==10){
                    System.out.println("(" + arr[i] + "," + arr[k] + ")");
                    break;
                }
            }
        }
    }
}
```

Program to find the common elements from 2 arrays.

```
import java.util.Arrays;
class Code
{
    public static void main(String[] args)
    {
        int[] a1 = {1, 2, 5, 5, 8, 9, 5, 7, 10};
        int[] a2 = {3, 6, 15, 6, 5, 4, 7, 2, 1};
        System.out.println("Array1 : " + Arrays.toString(a1));
        System.out.println("Array2 : " + Arrays.toString(a2));
        for (int i=0 ; i<a1.length ; i++){
            for (int j=0 ; j<a2.length ; j++){
                if(a1[i] == (a2[j])){
                    System.out.println("Common element is : " + (a1[i]));
                    a2[j]=0;
                }
            }
        }
    }
}
```

Program to find the duplicates in the array

```
import java.util.Arrays;
class Code
{
    public static void main(String[] args)
    {
        int[] arr = {1, 2, 5, 5, 8, 9, 2, 5, 7, 1, 10, 1, 2};
        System.out.println("Array : " + Arrays.toString(arr));
        for (int i=0 ; i<arr.length ; i++){
            for (int j=i+1 ; j<arr.length ; j++){
                if(arr[i]!=0 && (arr[i] == arr[j])){
                    System.out.println(arr[i] + " is duplicated");
                    arr[j]=0;
                    break;
                }
            }
        }
    }
}
```

Program to find the second largest element in the given array:

```
import java.util.Arrays;
class Code{
    public static void main(String[] args) {
        int[] arr = {4, 7, 2, 9, 3, 8, 6, 1, 5};
        System.out.println("Array : " + Arrays.toString(arr));
        int f=arr[0];
        int s=arr[1];
        if(f<s){
            int t=f;
            f=s;
            s=t;
        }
        for (int i=2 ; i<arr.length ; i++){
            if(arr[i]>f){
                s=f;
                f=arr[i];
            }
            else if(arr[i]<f && arr[i]>s){
                s=arr[i];
            }
        }
        System.out.println("First Largest : " + f);
        System.out.println("Second Largest : " + s);
    }
}
```

Program to arrange even numbers to left side and odd numbers to right side of Array:

```
import java.util.Arrays;
class Code {
    public static void main(String[] args) {
        int[] arr = {7, 2, 9, 8, 4, 1, 2, 6, 5};
        System.out.println("Given array : " + Arrays.toString(arr));
        int i=0;
        int j=arr.length-1;
        while(i<j){
            while(true){
                if(arr[i]%2!=0)
                    break;
                else
                    i++;
            }
            while(true){
                if(arr[j]%2==0)
                    break;
                else
                    j--;
            }
            if(i<j)
                swap(arr, i, j);
        }
        System.out.println("Array after rearranging : " + Arrays.toString(arr));
    }
}
```

```
        if(arr[j]%2==0)
            break;
        else
            j--;
    }
    if(i<j){
        int t=arr[i];
        arr[i]=arr[j];
        arr[j]=t;
    }
    i++;
    j--;
}
System.out.println("Result array : " + Arrays.toString(arr));
}
}
```

Program to check 2 arrays are equal or not:

```
import java.util.Arrays;
class Code {
    public static void main(String[] args) {
        int[] a1 = {3, 4, 5, 6, 7};
        int[] a2 = {3, 4, 5, 6, 7};
        String s1 = Arrays.toString(a1);
        String s2 = Arrays.toString(a2);
        if(s1.equals(s2))
            System.out.println("Arrays are equal");
        else
            System.out.println("Arrays are not equal");
    }
}
```

Program to test the equality of 2 arrays without using pre-defined method.

```
import java.util.Arrays;
class Code {
    public static void main(String[] args){
        int[] a1 = {3, 4, 5, 6, 7};
        int[] a2 = {3, 4, 5, 6, 7};
        if(a1.length == a2.length){
            boolean equal=true;
            for (int i=0 ; i<a1.length ; i++){
                if(a1[i] != a2[i])
                    equal=false;
            }
        }
    }
}
```

```
    }
    if(equal)
        System.out.println("Arrays are equal");
    else
        System.out.println("Arrays are not equal");
}
else
    System.out.println("Arrays are not equal");
}
}
```

Program to find the missing number in the given array.

```
class Code {
    public static void main(String[] args) {
        int[] arr = {2, 3, 4, 5, 7, 8, 9};
        int start=arr[0];
        for (int i=1 ; i<arr.length ; i++){
            if(++start != arr[i]){
                System.out.println("Missing number is : " + start);
                break;
            }
        }
    }
}
```

Program to move all zero to the end of given array

```
import java.util.Arrays;
class Code {
    public static void main(String[] args) {
        int[] arr = {9, 0, 0, 4, 0, 3, 0, 2, 0};
        System.out.println("Given array : " + Arrays.toString(arr));
        int i=0;
        int j=arr.length-1;
        while(i<j){
            while(true){
                if(arr[i]==0)
                    break;
                else
                    i++;
            }
            while(true){
                if(arr[j]!=0)
                    break;
            }
        }
    }
}
```

```
        else
            j--;
    }
    if(i<j){
        int t = arr[i];
        arr[i] = arr[j];
        arr[j] = t;
    }
    i++;
    j--;
}
System.out.println("Resultant array: " + Arrays.toString(arr));
}
}
```

Program to find the largest difference of Array elements:

```
import java.util.Arrays;
class Code {
    public static void main(String[] args) {
        int[] arr = {5, 7, 3, 8, 6, 9, 4};
        System.out.println("Given array : " + Arrays.toString(arr));
        int small=arr[0];
        int big=arr[0];
        for (int i=1 ; i<arr.length ; i++){
            if(small>arr[i])
                small=arr[i];
            if(big<arr[i])
                big=arr[i];
        }
        System.out.println("Largest Difference between elements : " + (big-small));
    }
}
```

Program to find the index difference between smallest and largest elements:

```
import java.util.Arrays;
class Code {
    public static void main(String[] args) {
        int[] arr = {5, 7, 3, 8, 6, 9, 4};
        System.out.println("Given array : " + Arrays.toString(arr));
        int small=arr[0];
        int big=arr[0];
        int x=0, y=0;
        for (int i=1 ; i<arr.length ; i++){
```

```
        if(small>arr[i]){
            small=arr[i];
            x=i;
        }
        if(big<arr[i]){
            big=arr[i];
            y=i;
        }
    }
    System.out.println("Index difference between Small and Large Elements : " +
Math.abs(x-y));
    }
}
```

Program to check array contains only positive numbers or not

```
import java.util.Arrays;
class Code {
    public static void main(String[] args) {
        int[] arr = {5, 7, 3, 8, 6, -4, 4};
        System.out.println("Given array : " + Arrays.toString(arr));
        boolean found=false;
        for (int i=0 ; i<arr.length ; i++){
            if(arr[i]<0){
                found=true;
                break;
            }
        }
        if(found)
            System.out.println("Array has negative elements");
        else
            System.out.println("Array has only positive elements");
    }
}
```

Program to print leader elements in the array : All elements to its right must be smaller to Leader element

```
import java.util.Arrays;
class Code {
    public static void main(String[] args) {
        int arr[] = {10, 9, 14, 23, 15, 0, 9};
        int size = arr.length;
        for (int i=0 ; i<size ; i++) {
            int j;
```

```
        for (j=i+1 ; j<size ; j++) {  
            if (arr[i] <= arr[j])  
                break;  
        }  
        if (j==size)  
            System.out.print(arr[i] + " ");  
    }  
}
```


Two Dimensional Arrays

Two-dimensional Array: 2D arrays are also called Matrix in java. 2D arrays are used to store and process information of 2-dimensional format (rows and columns). We can create 2D array directly using any one of the following syntax

```
int arr[][] = new int[][] { { 1, 2, 3 }, { 4, 5, 6 } };
or
int[][] arr = new int[][] { { 1, 2, 3 }, { 4, 5, 6 } };
or
int[] arr[] = new int[][] { { 1, 2, 3 }, { 4, 5, 6 } };
or
int arr[][] = { { 1, 2, 3 }, { 4, 5, 6 } };
```

Processing 2D array: We can perform all operations like reading, modifying, displaying using nested loops.

```
public class Logic
{
    public static void main(String args[])
    {
        int arr[][] = new int[3][3];
        System.out.println("Elements are : ");
        for (int i=0 ; i<3 ; i++)
        {
            for(int j=0 ; j<3 ; j++)
            {
                System.out.print(arr[i][j] + " ");
            }
            System.out.println();
        }
    }
}
```

Find the row size and column size of array:

```
public class Logic {
    public static void main(String args[]){
        int[][] test = new int[5][10];
        int row = test.length;
        int col = test[0].length;
        System.out.println("Row size is : " + row);
        System.out.println("Col size is : " + col);
    }
}
```

Find the length of Array:

```
public class Logic{
    public static void main(String args[]) {
        int A[][] = {{10,20,30},{40,50},{70,80}};
        System.out.println("Length of array : " + (A[0].length + A[1].length + A[2].length));
    }
}
```

We can initialize the 2D array directly using assignment operator as follows:

```
public class Logic{
    public static void main(String args[]) {
        int arr[][] = { {1,2,3}, {3,4,5}, {5,6,7}};
        System.out.println("Elements are : ");
        for (int i=0 ; i<3 ; i++){
            for(int j=0 ; j<3 ; j++){
                System.out.print(arr[i][j] + " ");
            }
            System.out.println();
        }
    }
}
```

Construct 2D matrix using end user input data:

```
import java.util.Scanner;
public class Logic{
    public static void main(String args[]) {
        Scanner scan = new Scanner(System.in);
        System.out.print("Row size : ");
        int m = scan.nextInt();
        System.out.print("Column size : ");
        int n = scan.nextInt();
        int arr[][] = new int[m][n];
        System.out.println("Enter " + (m*n) + " elements : ");
        for (int i=0 ; i<m ; i++){
            for(int j=0 ; j<n ; j++){
                arr[i][j] = scan.nextInt();
            }
        }
        System.out.println(m + "X" + n + " matrix is : ");
        for (int i=0 ; i<m ; i++){
            for(int j=0 ; j<n ; j++){
```

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

Find the maximum row sum:

```
import java.util.Scanner;
public class Logic
{
    public static void main(String args[]) {
        Scanner scan = new Scanner(System.in);
        System.out.print("Row size : ");
        int m = scan.nextInt();
        System.out.print("Column size : ");
        int n = scan.nextInt();
        int arr[][] = new int[m][n];
        System.out.println("Enter " + (m*n) + " elements : ");
        for (int i=0 ; i<m ; i++){
            for(int j=0 ; j<n ; j++){
                arr[i][j] = scan.nextInt();
            }
        }
        int row[] = new int[m];
        for (int i=0 ; i<m ; i++){
            int sum=0;
            for(int j=0 ; j<n ; j++){
                sum = sum + arr[i][j] ;
            }
            row[i] = sum;
        }
        int max = row[0];
        for(int i=1 ; i<row.length ; i++){
            if(max < row[i])
                max = row[i];
        }
        System.out.println("Maximum row sum is : " + max);
    }
}
```

Addition of 2 matrixes:

```
import java.util.Scanner;
public class Logic{
    public static void main(String args[]) {
        Scanner scan = new Scanner(System.in);
        int A[][] = new int[2][2];
        int B[][] = new int[2][2];
        int C[][] = new int[2][2];
        System.out.println("Enter 4 elements into A : ");
        for (int i=0 ; i<2 ; i++)
            for(int j=0 ; j<2 ; j++)
                A[i][j] = scan.nextInt();

        System.out.println("Enter 4 elements into B : ");
        for (int i=0 ; i<2 ; i++)
            for(int j=0 ; j<2 ; j++)
                B[i][j] = scan.nextInt();

        for (int i=0 ; i<2 ; i++)
            for(int j=0 ; j<2 ; j++)
                C[i][j] = A[i][j] +B[i][j] ;

        System.out.println("Added matrix is : ");
        for (int i=0 ; i<2 ; i++){
            for(int j=0 ; j<2 ; j++){
                System.out.print(C[i][j] + " ");
            }
            System.out.println();
        }
    }
}
```

Matrix multiplication:

```
import java.util.Scanner;
public class Logic{
    public static void main(String args[]) {
        Scanner scan = new Scanner(System.in);
        int A[][] = new int[2][2];
        int B[][] = new int[2][2];
        int C[][] = new int[2][2];
        System.out.println("Enter 4 elements into A : ");
        for (int i=0 ; i<2 ; i++)
            for(int j=0 ; j<2 ; j++)
```

```

        A[i][j] = scan.nextInt();
    System.out.println("Enter 4 elements into B : ");
    for (int i=0 ; i<2 ; i++)
        for(int j=0 ; j<2 ; j++)
            B[i][j] = scan.nextInt();
    for (int i=0 ; i<2 ; i++){
        for(int j=0 ; j<2 ; j++){
            C[i][j] = 0;
            for (int k=0 ; k<2 ; k++ ){
                C[i][j] = C[i][j] + A[i][k] * B[k][j];
            }
        }
    }
    System.out.println("Mutliplied matrix is : ");
    for (int i=0 ; i<2 ; i++){
        for(int j=0 ; j<2 ; j++){
            System.out.print(C[i][j] + " ");
        }
        System.out.println();
    }
}
}

```

Transpose of a Matrix:

```

public class Logic{
    public static void main(String args[]) {
        int A[][] = {{10,20,30},{40,50,60},{70,80,90}};
        System.out.println("Input Matrix is : ");
        for (int i=0 ; i<3 ; i++){
            for(int j=0 ; j<3 ; j++){
                System.out.print(A[i][j] + " ");
            }
            System.out.println();
        }
        for (int i=0 ; i<3 ; i++){
            for(int j=0 ; j<3 ; j++){
                if(i<j){
                    int temp = A[i][j];
                    A[i][j] = A[j][i];
                    A[j][i] = temp;
                }
            }
        }
    }
}

```

```

        System.out.println("Transposed Matrix is : ");
        for (int i=0 ; i<3 ; i++){
            for(int j=0 ; j<3 ; j++){
                System.out.print(A[i][j] + " ");
            }
            System.out.println();
        }
    }
}

```

Program to sort Each row in given Matrix:

```

class Code
{
    public static void main(String[] args) {
        int[][] a = { {5, 3, 7, 2, 8}, {9, 2, 8, 4, 6}, {6, 1, 9, 7, 3} };
        System.out.println("Given Matris is : ");
        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();
        }
        for (int i=0 ; i<a.length ; i++){
            for (int j=0 ; j<a[i].length-1 ; j++){
                for (int k=0 ; k<a[i].length-1 ; k++){
                    if(a[i][k] > a[i][k+1]){
                        int temp = a[i][k];
                        a[i][k] = a[i][k+1];
                        a[i][k+1] = temp;
                    }
                }
            }
        }
        System.out.println("Sorted Matris is : ");
        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();
        }
    }
}

```

Strings

Can we assign character to integer variable?

We can directly assign character to integer called "Implicit Cast". The ASCII value will store into integer variable.

```
char ch = 'A';  
int x = ch;
```

How to convert integer into character?

We cannot assign integer value directly to character variable. We must explicitly cast(convert)

```
int x = 100 ;  
char ch = (char)x;
```

Program to Check the character is vowel or consonant

```
import java.util.Scanner;  
class Code  
{  
    public static void main(String[] args) {  
        Scanner scan = new Scanner(System.in);  
        System.out.print("Enter character : ");  
        char ch = scan.next().charAt(0);  
        if(ch=='a' || ch=='e' || ch=='i' || ch=='o' || ch=='u')  
            System.out.println("It is vowel");  
        else  
            System.out.println("It is not a vowel");  
    }  
}
```

program to check the given character is digit or not

```
import java.util.Scanner;  
class Code  
{  
    public static void main(String[] args) {  
        Scanner scan = new Scanner(System.in);  
        System.out.print("Enter character : ");  
        char ch = scan.next().charAt(0);  
        if(ch>='0' && ch<='9')  
            System.out.println("It is Digit");  
        else  
            System.out.println("It is not a Digit");  
    }  
}
```

Program to check the character is Alphabet or not

```
import java.util.Scanner;
class Code
{
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter character : ");
        char ch = scan.next().charAt(0);
        if((ch>='A' && ch<='Z') || (ch>='a' && ch<='z'))
            System.out.println("It is an Alphabet");
        else
            System.out.println("It is not an Alphabet");
    }
}
```

Program to check the character is special symbol or not

```
import java.util.Scanner;
class Code
{
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter character : ");
        char ch = scan.next().charAt(0);
        if(!((ch>='A' && ch<='Z') || (ch>='a' && ch<='z') || (ch>='0' && ch<='9'))))
            System.out.println("It is a symbol");
        else
            System.out.println("It is not a symbol");
    }
}
```

Program to convert upper case character to lower case character**ASCII Character Set:**

A-65	a-97	0-48	*-34
B-66	b-98	1-49	#-35
...	\$-36
...
...
Z-90	z-122	9-57	...

From ASCII table:

Upper Case Character + 32 = Lower Case Character

Lower Case Character – 32 = Upper Case Character


```
import java.util.Scanner;
class Code {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter Upper case character : ");
        char ch = scan.next().charAt(0);
        if (ch >= 'A' && ch <= 'Z'){
            ch = (char)(ch+32);
            System.out.println("Lower case character is : " + ch);
        }
        else
            System.out.println("Invalid Character Entered");
    }
}
```

Display the length of given character array

```
class Code {
    public static void main(String[] args) {
        char[] arr = {'a', 'x', '3', 'm', 'x', '@', 'p', '7', 'm', '4', '$'};
        int n = arr.length;
        System.out.println("Length of Array is : " + n);
    }
}
```

Convert String object into character array: String class providing toCharArray() method for this conversion.

```
class Code {
    public static void main(String[] args) {
        String str = "Coding";
        char[] arr = str.toCharArray();
        System.out.println("Elements :");
        for(char ele : arr)
            System.out.println(ele);
    }
}
```

Program to find the length of character array:

```
class Code {
    public static void main(String[] args) {
        char[] arr = {'a', 'e', 'i', 'o', 'u'};
        System.out.println("Length of array : " + arr.length);
    }
}
```

Display First and Last characters of Array:

```
class Code
{
    public static void main(String[] args) {
        char[] arr = {'a', 'e', 'i', 'o', 'u'};
        System.out.println("First character : " + arr[0]);
        System.out.println("Last character : " + arr[arr.length-1]);
    }
}
```

Display Character array using for-each loop:

```
class Code {
    public static void main(String[] args) {
        char[] arr = {'a', 'e', 'i', 'o', 'u'};
        System.out.println("Array elements : ");
        for(char ch : arr)
            System.out.println(ch);
    }
}
```

Display Character array in reverse order:

```
class Code
{
    public static void main(String[] args) {
        char[] arr = {'a', 'e', 'i', 'o', 'u'};
        System.out.println("Reverse array : ");
        for (int i=arr.length-1 ; i>=0 ; i--)
            System.out.println(arr[i]);
    }
}
```

Display only symbols digits in character array:

```
class Code {
    public static void main(String[] args) {
        char[] arr = {'a', 'b', 'c', '@', '1', '2', '3'};
        System.out.println("Digits in array : ");
        for (int i=0 ; i<arr.length ; i++){
            if(arr[i]>='0' && arr[i]<='9')
                System.out.println(arr[i]);
        }
    }
}
```

Program to count Alphabets, Digits and Symbols in the character array:

```
class Code
{
    public static void main(String[] args)
    {
        char[] arr = {'a', 'b', 'c', '@', '1', '2', '3'};
        int alpha=0, digit=0, sym=0;
        for (int i=0 ; i<arr.length ; i++){
            char ch=arr[i];
            if((ch>='A'&&ch<='Z') || (ch>='a'&&ch<='z'))
                alpha++;
            else if(ch>='0' && ch<='9')
                digit++;
            else
                sym++;
        }
        System.out.println("Alphabets : " + alpha);
        System.out.println("Digits : " + digit);
        System.out.println("Symbols : " + sym);
    }
}
```

Program display length of String:

```
class Code
{
    public static void main(String[] args)
    {
        String s = "Coding";
        System.out.println("Length is : " + s.length());
    }
}
```

Program to display First and Last characters of String:

```
class Code
{
    public static void main(String[] args)
    {
        String s = "Coding";
        System.out.println("First Character : " + s.charAt(0));
        System.out.println("Last Character : " + s.charAt(s.length()-1));
    }
}
```

Program to print String character by character:

```
class Code {  
    public static void main(String[] args) {  
        String s = "Coding";  
        for (int i=0 ; i<=s.length()-1 ; i++){  
            System.out.println(s.charAt(i));  
        }  
    }  
}
```

Program to print the String in reverse order:

```
class Code {  
    public static void main(String[] args) {  
        String s = "Coding";  
        for (int i=s.length()-1 ; i>=0 ; i--){  
            System.out.println(s.charAt(i));  
        }  
    }  
}
```

Program to check the 2 strings equal or not:

```
class Code {  
    public static void main(String[] args) {  
        String s1="Coding";  
        String s2="Code";  
        if(s1.equals(s2))  
            System.out.println("Equal");  
        else  
            System.out.println("Not equal");  
    }  
}
```

Program to create reverse string from given string:

```
class Code {  
    public static void main(String[] args) {  
        String s = "Coding";  
        String rev="";  
        for (int i=s.length()-1 ; i>=0 ; i--){  
            rev=rev+s.charAt(i);  
        }  
        System.out.println("Reverse string is : " + rev);  
    }  
}
```

Program to count Alphabets, Digits and Symbols in the given String

```
class Code {  
    public static void main(String[] args) {  
        String s = "Coding@365";  
        int c1=0, c2=0, c3=0;  
        for (int i=0 ; i<=s.length()-1 ; i++){  
            char ch=s.charAt(i);  
            if((ch>='A'&&ch<='Z')||(ch>='a'&&ch<='z'))  
                c1++;  
            else if(ch>='0'&&ch<='9')  
                c2++;  
            else  
                c3++;  
        }  
        System.out.println("Alphabets : " + c1);  
        System.out.println("Digits : " + c2);  
        System.out.println("Symbols : " + c3);  
    }  
}
```

How to check if a string is a palindrome?

```
class Code {  
    public static void main(String[] args) {  
        String line = "madam";  
        String rev = reverseString(line);  
        if (line.equals(rev))  
            System.out.println("Palindrome String");  
        else  
            System.out.println("Not a palindrome String.");  
    }  
    public static String reverseString(String str) {  
        String rev = "";  
        for (int i = str.length() - 1; i >= 0; i--)  
        {  
            rev += str.charAt(i);  
        }  
        return rev;  
    }  
}
```

Program to find the sum of digits in the given String:

```
class Code {  
    public static void main(String[] args) {  
        String s = "Coding@365";  
        int sum=0;  
        for (int i=0 ; i<=s.length()-1 ; i++){  
            char ch=s.charAt(i);  
            if(ch>='0'&&ch<='9')  
                sum = sum + (int)(ch-48);  
        }  
        System.out.println("String is : " + s);  
        System.out.println("Sum of Digits : " + sum);  
    }  
}
```

Display the ASCII values of characters in the given String:

```
class Code {  
    public static void main(String[] args) {  
        String s = "Coding";  
        System.out.println(s + " ASCII values : ");  
        for (int i=0 ; i<s.length() ; i++){  
            char ch=s.charAt(i);  
            System.out.println(ch + " : " + (int)ch);  
        }  
    }  
}
```

Program to convert upper case string to lower case string:

```
class Code {  
    public static void main(String[] args) {  
        String s = "CoDiNg@365";  
        String res="";  
        for (int i=0 ; i<s.length() ; i++){  
            char ch=s.charAt(i);  
            if(ch>='A' && ch<='Z')  
                res=res+(char)(ch+32);  
            else  
                res=res+ch;  
        }  
        System.out.println("Given string is : " + s);  
        System.out.println("Lower case String : " + res);  
    }  
}
```

Program to display highest digit in the given String:

class Code

```
{
    public static void main(String[] args) {
        String s = "coding@365";
        int n=0;
        for (int i=0 ; i<s.length() ; i++){
            char ch=s.charAt(i);
            if(ch>='0'&&ch<='9'){
                int x=(int)(ch-48);
                if(x>n)
                    n=x;
            }
        }
        System.out.println("Higher digit in String : " + n);
    }
}
```

Program to find the length of String array

class Code

```
{
    public static void main(String[] args) {
        String[] arr = {"java", "jsp", "servlets", "hibernate", "springBoot"};
        System.out.println("Length of array : " + arr.length);
    }
}
```

Program to display Strings from Array:

class Code

```
{
    public static void main(String[] args) {
        String[] arr = {"java", "jsp", "servlets", "hibernate", "springBoot"};
        for (int i=0 ; i<arr.length ; i++){
            System.out.println(arr[i]);
        }
    }
}
```

Program to display the length of each String in the array:

class Code

```
{
    public static void main(String[] args)
    {
        String[] arr = {"java", "jsp", "servlets", "hibernate", "springBoot"};
        for (int i=0 ; i<arr.length ; i++){
            String s = arr[i];
            System.out.println(s + " : " + s.length());
        }
    }
}
```

Program to display First and Last characters of each string in the array:

class Code

```
{
    public static void main(String[] args)
    {
        String[] arr = {"java", "jsp", "servlets", "hibernate", "springBoot"};
        for (int i=0 ; i<arr.length ; i++){
            String s = arr[i];
            System.out.println(s+" : "+s.charAt(0)+"-"+s.charAt(s.length()-1));
        }
    }
}
```

Program to display String array in reverse order:

class Code

```
{
    public static void main(String[] args) {
        String[] arr = {"java", "jsp", "servlets", "hibernate", "springBoot"};
        for (int i=arr.length-1 ; i>=0 ; i--){
            System.out.println(arr[i]);
        }
    }
}
```


Program to display each String in reverse order from Array:

```
class Code {  
    public static void main(String[] args) {  
        String[] arr = {"java", "jsp", "servlets", "hibernate", "springBoot"};  
        for (int i=0 ; i<=arr.length-1 ; i++){  
            String s = arr[i];  
            for (int j=s.length()-1 ; j>=0 ; j--){  
                System.out.print(s.charAt(j));  
            }  
            System.out.println();  
        }  
    }  
}
```

Program to Split the String into Words:

```
class Code {  
    public static void main(String[] args){  
        String s = "This is core java test";  
        String[] arr = s.split(" ");  
        System.out.println("String is : " + s);  
        System.out.println("Words count : " + arr.length);  
    }  
}
```

Program to count number of words in the given string without split() method.

```
class Code {  
    public static void main(String[] args) {  
        String s = "This is core java test";  
        int spaces=0;  
        for (int i=0 ; i<s.length() ; i++){  
            if(s.charAt(i)==' '){  
                spaces++;  
            }  
        }  
        System.out.println("String is : " + s);  
        System.out.println("Words count : " + (spaces+1));  
    }  
}
```

Program to check two strings are Anagrams or not:

We can say if two strings are an anagram of each other if they contain the same characters but at different orders. For example, army & mary

```
import java.util.Arrays;
class Code
{
    public static void main(String[] args) {
        String s1 = "army";
        String s2 = "mary";
        System.out.println("Check Anagram or not : "+ isAnagram(s1, s2));
    }
    public static boolean isAnagram(String s1, String s2){
        char[] a1 = s1.toLowerCase().toCharArray();
        char[] a2 = s2.toLowerCase().toCharArray();
        Arrays.sort(a1);
        Arrays.sort(a2);
        return Arrays.equals(a1, a2);
    }
}
```

Program to display longest word in the given String:

```
class Code
{
    public static void main(String[] args) {
        String s = "This is the Longest Sentense in Java";
        String[] arr = s.split(" ");
        int loc=0;
        int h=arr[0].length();
        for (int i=1 ; i<arr.length ; i++){
            if(arr[i].length()>h){
                h=arr[i].length();
                loc=i;
            }
        }
        System.out.println("Longest Word is : " + arr[loc]);
    }
}
```

Program to remove duplicates in the String:

class Code

```
{
    public static void main(String[] args) {
        String str = "aaabbababaacccbabdcccbbdddbac";
        String res = "";
        for (int i=0 ; i<str.length()-1 ; i++){
            int j=0;
            for (j=0 ; j<i ; j++){
                if(str.charAt(i)==str.charAt(j))
                    break;
            }
            if(j==i)
                res = res + str.charAt(i);
        }
        System.out.println("Given String : " + str);
        System.out.println("Resultant string : " + res);
    }
}
```

Program to print String permutations using Iterator(loop):

class Code

```
{
    public static void main(String[] args) {
        String str = "abc";
        for (char x='a' ; x<='c' ; x++){
            for (char y='a' ; y<='c' ; y++){
                for (char z='a' ; z<='c' ; z++){
                    if(x!=y && y!=z && z!=x)
                        System.out.println(x+""+y+""+z);
                }
            }
        }
    }
}
```

Program to print String permutations using Recursion:

```
class Code {  
    public static void main(String[] args) {  
        String str = "ABC";  
        permutations("", str);  
    }  
    static void permutations(String s, String rem){  
        if (rem == "")  
            return;  
        if (rem.length() == 0)  
            System.out.println(s);  
        for (int i=0 ; i<rem.length(); i++)  
        {  
            String n = s + rem.charAt(i);  
            String nrem = rem.substring(0,i) + rem.substring(i + 1);  
            permutations(n, nrem);  
        }  
    }  
}
```

Program to remove spaces in String:

```
class Code {  
    public static void main(String[] args) {  
        String s = "This is a String";  
        s = s.replaceAll(" ", "");  
        System.out.println("Result : " + s);  
    }  
}
```

Program to remove spaces in String using regular expression:

```
class Code {  
    public static void main(String[] args) {  
        String s = "This is a String";  
        s = s.replaceAll("\\s", "");  
        System.out.println("Result : " + s);  
    }  
}
```

Program to remove multiple spaces in the given string:

```
class Code {  
    public static void main(String[] args) {  
        String s = "This   is   a   String";  
        s = s.replaceAll("\\s+", " ");  
        System.out.println("Result : " + s);  
    }  
}
```

Program to remove multiple spaces in string without using library method:

```
class Code {  
    public static void main(String[] args) {  
        String s1 = "This   is   a   String";  
        String s2 = "";  
        for (int i=0 ; i<s1.length() ; i++){  
            char ch = s1.charAt(i);  
            if(ch != ' '){  
                s2=s2+ch;  
            }  
            else{  
                if(s1.charAt(i+1) != ' '){  
                    s2=s2+ch;  
                }  
            }  
        }  
        System.out.println("Result : " + s2);  
    }  
}
```

Program to print characters specified by the number of times:

Input: abc3e2f4

Expected output: abcccceeffff

```
class Code {  
    public static void main(String[] args) {  
        String s = "abc3e2f4";  
        System.out.println("Input : " + s);  
        System.out.print("Output : ");  
        for (int i=0 ; i<s.length() ; i++){  
            char ch = s.charAt(i);
```

```
        char ch1 = s.charAt(i+1);
        if(ch1>='0' && ch1<='9'){
            for (int j=1 ; j<=(ch1-48) ; j++){
                System.out.print(ch);
            }
            i++;
        }
        else
            System.out.print(ch);
    }
    System.out.println();
}
}
```

Program to swap the side-by-side characters in the given string:

```
class Code {
    public static void main(String[] args) {
        String s1 = "abcde";
        String s2 = "";
        for (int i=0 ; i<s1.length() ; i+=2){
            if(i+2 < s1.length())
                s2 = s2 + s1.charAt(i+1) + s1.charAt(i);
            else
                s2 = s2 + s1.charAt(i);
        }
        System.out.println("Input String is : " + s1);
        System.out.println("Output String is : " + s2);
    }
}
```

Program to remove duplicates in the given String:

```
class Code
{
    public static void main(String[] args)
    {
        String s1 = "aaaabbaababaaabccccaaab";
        String s2 = "";
```

```
        for (int i=0 ; i<s1.length() ; i+=2){
            if(!s2.contains(""+s1.charAt(i)))
                s2 = s2 + s1.charAt(i);
        }
        System.out.println("Input String is : " + s1);
        System.out.println("Output String is : " + s2);
    }
}
```

Program to check how many times the given sub string is present:

class Code

```
{
    public static void main(String[] args) {
        String s = "abcaabcaaabcabacabcaabcaabc";
        String[] arr = s.split("abc");
        System.out.println("abc repeated : " + (arr.length) + " times");
    }
}
```

Program to display the character count in the given string:

Input : aaabbbacccddacdd

Output : a-5, b-3, c-5, d-4

class Code {

```
    public static void main(String[] args) {
        String s = "aaabbbacccddacdd";
        int[] arr = new int[256];
        for (int i=0 ; i<s.length() ; i++){
            arr[(int)s.charAt(i)]++;
        }
        for (int i=0 ; i<256 ; i++){
            if(arr[i]!=0){
                System.out.println((char)i + " : " + arr[i]);
            }
        }
    }
}
```

Program to remove the sub string in the given String:

```
class Code {  
    public static void main(String[] args) {  
        String s = "This is Core Java String";  
        System.out.println("Given String : " + s);  
        s = s.replace("Core", "");  
        System.out.println("After removing Sub string : " + s);  
    }  
}
```

Program to remove sub string in the given string without library method:

```
class Code {  
    public static void main(String[] args) {  
        String s1 = "This is Core Java String";  
        String s2 = "";  
        String sub = "Core";  
        int n=sub.length();  
        for (int i=0 ; i<s1.length() ; i++){  
            char ch = s1.charAt(i);  
            if(ch != sub.charAt(0)){  
                s2 = s2 + ch;  
            }  
            else {  
                String match = s1.substring(i, i+n);  
                i=i+n;  
            }  
        }  
        System.out.println("Given String : " + s1);  
        System.out.println("After removing Sub string : " + s2);  
    }  
}
```