

Error

1. Syntax error (Typing mistake)
2. Logical error (User itself mistake) $c = a + b$
3. Runtime error
4. $a = 10$
 $b = 0$
 $c = a/b$

Runtime - Exception handling

1. Try-catch block
2. Throw / throws.

Try - catch block.

```
try
```

```
{}
```

```
condition -
```

```
}
```

```
catch (Exception e)
```

```
// statements
```

```
finally.
```

```
    System.out.println("Error");  
    System.out.println(e.toString());
```

Array

Accept any 20 no from user and display sum 5th and 10th position.

1. Exception handling (error handling)

An exception is an abnormal condition that arises in a code sequence at runtime. An exception is runtime error.

Java exception handling is managed through five keyword; try, catch, throw, throws and finally.

Any exception that is thrown out of a method must be specified as such by a throws keyword. Any code that is absolutely must be executed before a method returns is put in a finally block.

Syntax is;

```
try  
{}
```

// block of code.

```
}
```

```
catch (exception type t ex)
```

```
{  
    // exception handler.  
}  
catch (Exception type2 ex)  
{  
    finally  
    {  
        // block of code to be executed before  
        // try blocks ends  
    }  
}
```

Program of exception handling

```
class ExceptionHandling  
{  
    public static void main (String args [ ])  
    {  
        int a, b=0, c;  
        a = 20;  
        try  
        {  
            c = a/b;  
            System.out.println ("the no. is " + c);  
        }  
        catch (Exception e)  
        {  
        }  
    }  
}
```

```
System.out.println ("cannot divided by zero");  
}
```

```
finally  
{}
```

```
System.out.println ("program is completed");  
}}}
```

Program of using try and multiple catch block .

```
class edemo  
{
```

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

```
try  
{
```

```
int a = args.length
```

```
System.out.println ("a = " + a);
```

```
int b = 42 / a
```

```
int c [ ] = { 1 } ;
```

```
c [ 42 ] = 99 ;
```

```
}
```

```
catch (ArithmaticException e)
```

```
{
```

```
System.out.println ("Divide by zero" + e);  
}
```

~~e. testing~~

classmate

Date _____

Page _____

catch (ArrayIndexOutOfBoundsException e)

}

System.out.println ("Array index out of bound
+ e);

} } }

class ss

{

public static void main (String args [])

{

try

{

int a = 2, b = 4, c = 2, x = 7, z;

int p [] = { 2 } ;

p [3] = 33;

try

{

z - x / ((b * b) - (4 * a * c));

System.out.println ("the value is " + z);

}

catch (ArithmaticException e)

{

System.out.println ("Division by zero in
Arithmatic expression");

}

```
catch (ArrayIndexOutOfBoundsException e)
{
```

```
    System.out.println("Array index is out-of-  
bounds");
```

```
} }
```

Array

An array is a group of variables which have a same name, same data types and same size. Array are categorised into two types.

1. One dimensional array

2. Multi dimensional array.

class Arraydemo

{

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

{

```
int mdays [ ];
```

```
mdays [0] = new int [6];
```

```
mdays [0] = 31;
```

```
mdays [1] = 28;
```

```
mdays [2] = 31;
```

```
mdays [3] = 30;
```

```
mdays [4] = 31;
```

```
mdays [5] = 30;
```

```
System.out.println ("Jan has " + mdays [0] + " days");
```

}

class eve
{

```
public static void main (String args[])
{
    double nums[] = {10.1, 11.2, 10.3, 13.4, 19.5};
    double result = 0;
    int i;
    for (i=0; i<5; i++)
    {
        result = result + nums[i];
    }
    System.out.println ("Average is " + result / 5);
}
```

class earn
{

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

```
# class mm
```

{

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

{

```
int i;
```

```
char m [ ] = { 'a' , 'b' , 'c' , 'd' , 'e' , 'f' , 'g' };
```

```
System.out.println ("The characters of the  
array elements are :");
```

```
for (i = 0; i < 7; i++)
```

{

```
System.out.println (m [i]);
```

{}

```
# class Testarray
```

{

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

{

```
int a [ ] = { 33, 3, 4, 5 };
```

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

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

{

{}

Passing Array to a method in java
(to get the minimum number of an array using a method)

```
class mm
{
    void min (int arr[])
    {
        int min = arr[0];
        for (int i=1; i< arr.length ; i++)
        {
            if (min > arr[i])
                min = arr[i];
        }
        System.out.println (min);
    }
}
```

```
class smain
{
```

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

```
    mm m = new mm ();
    int a [] = {33, 3, 4, 5};
    m.min (a);
}
```

Returning Array from the Method.

```
class aa
{
    static int [ ] get()
    {
        return new int [ ] {10, 30, 50, 90, 60};
    }

    public static void main (String args[ ])
    {
        int arr[ ] = get();
        for (int i=0; i<arr.length; i++)
        {
            System.out.println (arr[i]);
        }
    }
}
```

ArrayIndexOutOfBoundsException.

The Java Virtual Machine (JVM) throws an `ArrayIndexOutOfBoundsException` if length of the array is negative, equal to the array size or greater than the array size which traversing the array.

class mm
{

public static void main (String args[])

{

int arr[] = {50, 60, 70, 80};

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

{

System.out.println (arr[i]);

} } }

This program generate the error because
in this program we use ' \leq arr.length'. If
we replace the '=' sign from there then
it remove the error ' $<$ arr.length'.

Using BufferedReader in Array.

import java.io.*
class arr
{

public static void main (String args[])

throws IOException

BufferedReader br = new BufferedReader (new
InputStreamReader (System.in));

```
int a[] = new int[5];
int i;
for (i=0; i<5; i++)
{
    System.out.println ("enter your no.");
    a[i] = Integer.parseInt (br.readLine ());
}
System.out.println ("the array elements is ");
for (i=0; i<5; i++)
{
    System.out.println (a[i]);
}
```

Sum of Numbers.

```
import java.io.*;
class arr
{
public static void main (String args[])
    throws IOException
{
    BufferedReader br = new BufferedReader
        (new InputStreamReader (System.in));
    int a[] = new int [5];
    int i, s=0;
    for (i=0; i<5; i++)
        s = s + a[i];
    System.out.println ("sum is " + s);
}
```

{

```
System.out.println ("Enter your numbers");
a[i] = Integer.parseInt (br.readLine ());
s = s + a[i];
}
```

```
System.out.println (a[i]); "the array elements are"
for (i=0; i<5; i++)
}
```

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

```
System.out.println ("Sum is " + s);
} }.
```

WAP to accept any five numbers and sort them in ascending order.

```
import java.io.*;
```

```
class arr
```

```
{
```

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

```
BufferedReader br = new BufferedReader (new
InputStreamReader (System.in));
int a[] = new int [5];
int i, j, k;
```

```
for (i=0; i<5; i++)
```

{

```
System.out.println("Enter your numbers!");
```

```
a[i] = Integer.parseInt(br.readLine());
```

}

```
for (i=0; i<5; i++)
```

{

```
for (j=0; j<4; j++)
```

{

```
if (a[j] > a[j+1])
```

{

```
K = a[j];
```

```
a[j] = a[j+1];
```

```
a[j+1] = K;
```

}

}

}

```
System.out.println("The stored array elements  
are in Ascending order!");
```

```
for (i=0; i<5; i++)
```

{

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

}.

WAP to accept any 10 numbers from user and display greatest number of the elements.

```
import java.io.*;
class arr
{
    public static void main (String args [])
        throws IOException
    {
```

```
    BufferedReader br = new BufferedReader (new
        InputStreamReader (System.in));
    int a[] = new int [5];
    int i, max=0;
```

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

```
        System.out.print ("Enter your no.");
        a[i] = Integer.parseInt (br.readLine ());
    }
```

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

```
        if (max < a[i])
    {
```

```
            max = a[i];
        }
```

```
    System.out.println ("the. greatest of the array
elements = :" + max); }
```

WAP to accept any five numbers from user and display the maximum and minimum of the array elements.

```
import java.*;
```

```
class arr
```

```
{
```

```
public static void main (String args [])
    throws IOException
```

```
{
```

```
BufferedReader br = new BufferedReader
    (new InputStreamReader (System.in));
```

```
int a [] = new int [5]
```

```
int i, max = 0; min = 0;
```

```
for (i=0; i<5; i++)
```

```
{
```

```
System.out.println ("Enter your no.");
```

```
a [i] = Integer.parseInt (br.readLine());
```

```
}
```

```
max = a [0];
```

```
min = a [0];
```

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

```
{
```

```
if (max < a [i])
```

```
max = a [i];
```

```
if (min > a [i])
```

```
min= a[1];
```

```
}
```

```
System.out.println ("The greatest of the  
array elements = :" + max);
```

```
System.out.println ("The greatest of the minimum  
array elements = :" + min);
```

```
}
```

```
3.
```

WAP to accept 5 numbers in array and search whether a given number is present or not and display the message.

```
import java.io.*;
```

```
class arr
```

```
{
```

```
public static void main (String args [ ])  
throws IOException
```

```
{
```

```
BufferedReader br= new BufferedReader
```

```
(new InputStreamReader (System.in)));
```

```
int a [ ] = new int [5];
```

```
int i, s, k=0;
```

```
for (i=0; i<5; i++)
```

```
{
```

```
System.out.println ("Enter your no.");
```

```
a[i] = Integer.parseInt(br.readLine());  
}
```

```
System.out.println ("Enter your no. to be searched");  
s = Integer.parseInt(br.readLine());  
for (i = 1; i < 5; i++)  
{
```

```
if (a[i] == s)
```

```
k = 1;
```

```
}
```

```
if (k == 1)
```

```
System.out.println ("The required number " + s  
+ " is present");
```

```
else
```

```
System.out.println ("The required number " + s  
+ " is not present");
```

```
}
```

```
};
```

WAP to accept any 5 names from user
and arrange the names in ascending
order.

```
import java.io.*;
```

```
class rr
```

```
{
```

```
public static void main (String args [])  
throws IOException
```

```
BufferedReader br = new BufferedReader  
(new InputStreamReader (System.in));  
String s[] = new String [5];  
int i, j;  
String sa;  
for (i=0; i<5; i++)  
{  
    System.out.println ("Enter your names");  
    s[i] = br.readLine();  
}  
for (i=0; i<4; i++)  
{  
    for (j=i+1; j<5; j++)  
    {  
        if (s[i].compareTo (s[j]) > 0)  
        {  
            sa = s[i];  
            s[i] = s[j];  
            s[j] = sa;  
        }  
    }  
}  
System.out.println ("The names are arranged in  
ascending order");  
for (i=0; i<5; i++)  
{  
    System.out.println (s[i]);  
}
```

1. One dimensional array:-

One dimensional array is essentially a list of like typed variable. The syntax is

type var-name[];

The general form of new as it applies to one-dimensional arrays appears as follows:-

array-var = new type [size];

Here, type specifies the type of data being allocated, size specifies the no. of elements in the array, and array-var is the array variable that is linked to the array. That is, to use new to allocate an array, the elements in the array allocated by new will automatically be initialized to zero.

2. Multidimensional Arrays

In java, multidimensional arrays are actually arrays of arrays. To declare a multidimensional array variable, specify each additional index using another set of square brackets. `int a[][] = new int[4][5];`

multi-dimensional array used to store the data in tabular format where row and column are defined as matrix form.

- Q. WAP to accept any 5 no. from user and display greatest no.

class ss.

{

public static void main (String args[]) {

int a, b; a=10; b=2;

if (a>b)

System.out.println ("a is greatest");

else

System.out.println ("b is greatest");

}

}

```
import java.io.*;  
class arr.  
{  
    public static void main (String args[])  
        throws IOException  
{  
    BufferedReader br = new BufferedReader (new  
        InputStreamReader (System.in));  
    int a [] = new int [5];  
    int i, max = 0;  
    for (i = 0; i < 5; i++)  
    {  
        System.out.println ("Enter your no.");  
        a[i] = Integer.parseInt (br.readLine());  
    }  
    for (i = 0; i < 5; i++)  
    {  
        if (max < a[i])  
        {  
            max = a[i];  
        }  
    }  
    System.out.println ("the greatest array" + max);  
}
```

Multidimensional Arrays.

Example 1.

```
class darr
{
    public static void main (String args[])
    {
        int i,j;
        int m[ ][ ] = { { 1, 2, 3, 4 }, { 5, 6, 7, 8 }, { 9, 10, 11, 12 } };
        System.out.println ("The elements of the
                           matrix are : ");
        for (i=0 ; i<3 ; i++)
        {
            for (j=0 ; j<4 ; j++)
                System.out.print (m[i][j] + " ");
            System.out.println ();
        }
    }
}
```

1	2	3
5	6	7
9	10	11

Example 2.

```
class arr  
{  
public static void main (String args [ ] )  
{  
int i, j;  
char m [ ] [ ] , { { 'a' , 'b' , 'c' , 'd' } , { 'e' , 'f' , 'g' , 'h' } ,  
{ 'i' , 'j' , 'k' , 'l' } } ;  
System.out.println ("The elements of the  
matrix are : ");  
for (i = 0; i < 3; i++)  
{  
for (j = 0; j < 4; j++)  
{  
System.out.print (m [i] [j] + " ");  
}  
System.out.println ();  
}  
}
```

97 98 99

101 102 103

105 106 107

Example 3

```
import java.io.*;
class darr2
{
    public static void main (String args[])
        throws IOException
    {
        BufferedReader br= new BufferedReader (new
            InputStreamReader (System.in));
        int i, j;
        int m [ ] [ ] = new int [3] [3];
        System.out.println ("Enter no. of elements");
        for (i=0; i<3; i++)
        {
            for (j=0; j<3; j++)
            {
                m [i] [j] = Integer.parseInt (br.readLine ());
            }
        }
        System.out.println ("The elements of array:");
        for (i=0; i<3; i++)
        {
            for (j=0; j<3; j++)
            {
                System.out.print (m [i] [j] + " ");
            }
        }
    }
}
```

?

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

4 2 3

4 5 6

7 8 9.

Example 4

```
class darr4
```

{

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

```
{int twoD [ ][ ] = new int [4] [ ];
```

```
twoD [0] = new int [1];
```

```
twoD [1] = new int [2];
```

```
twoD [2] = new int [3];
```

```
twoD [3] = new int [4];
```

```
int i, j, k = 0;
```

```
for (i=0; i<4; i++)  
    for (j=0; j< i+1; j++)  
    {  
        twoD[i][j] = k;  
        k++;  
    }  
    for (f=0; i<4; f++) {  
        for (j=0; j< f+1; j++)  
            System.out.println (twoD[i][j] + " ");  
        System.out.println ();  
    }    }
```

Example 5

```
class darrs  
{
```

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

{

int twoD[][] = new int[4][5];

int i, j, k=0;

for (i=0; i<4 ; i++)

{

for (j=0; j<5 ; j++)

{

twoD[i][j] = k;

k++;

{

{

for (i=0; i<4 ; i++)

{

for (j=0; j<5 ; j++)

{

System.out.println (twoD[i][j] + " ");

{

System.out.println ();

{

{

Output

0 1 2 3 4

5 6 7 8 9

10 11 12 13 14

15 16 17 18 19.

Example 6.

```
class darr6
{
    public static void main (String args[])
    {
        int threeD [ ][ ][ ] = new int [3][4][5];
        int i, j, k;
        for (i=0; i<3; i++)
        {
            for (j=0; j<4; j++)
            {
                for (k=0; k<5; k++)
                {
                    threeD[i][j][k] = i*j*k;
                }
            }
        }
        for (i=0; i<3; i++)
        {
            for (j=0; j<4; j++)
            {
                for (k=0; k<5; k++)
                {
                    System.out.print (threeD[i][j][k] + " ");
                }
            }
        }
        System.out.println ();
    }
}
```

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

Example 7: Addition of 2 two matrices in Java.

class darr7

{

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

{

```
int a [ ] [ ] = { { 1, 3, 4 }, { 3, 4, 5 } } ;
```

```
int b [ ] [ ] = { { 1, 3, 4 }, { 3, 4, 5 } } ;
```

```
int c [ ] [ ] = new int [2] [3];
```

```
for (int i = 0; i < 2; i++)
```

{

```
for (int j = 0; j < 3; j++)
```

{

Example 8: Multiplication of 2 matrices in Java.

```
class darr8
{
    public static void main (String args[])
    {
        int a [][] = {{1,1,1}, {2,2,2}, {3,3,3}};
        int b [][] = {{1,1,1}, {2,2,2}, {3,3,3}};
        int c [][] = new int [3][3];
        for (int q=0; q<3; q++)
        {
            for (int p=0; p<3; p++)
            {
                for (int r=0; r<3; r++)
                {
                    c[q][p] = c[q][p] + a[q][r] * b[r][p];
                }
            }
        }
        for (int q=0; q<3; q++)
        {
            for (int p=0; p<3; p++)
            {
                System.out.print(c[q][p] + " ");
            }
            System.out.println();
        }
    }
}
```

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

Example 9: WAP subtract 100 matrices.

```
class subarr
{
    public static void main (String args[])
    {
        int rows , cols;
        int a[][] = {
            { 4, 5, 6 },
            { 3, 4, 1 },
            { 1, 2, 3 }
        };
    }
}
```

```
int b[][] = {
    { 2, 0, 3 },
    { 0, 3, 1 },
    { 1, 1, 1 }
};
```

```
rows = a.length;
cols = a[0].length;
int diff[][] = new int [rows][cols];
for (int i=0; i<rows; i++)
{
    for (int j=0; j<cols; j++)
}
```

System.out.println (diff)

```
diff[i][j] = a[i][j] - b[i][j];
```

{

{

```
System.out.println ("Subtraction of two matrix");
```

```
for (int i=0; i<rows; i++)
```

{

```
for (int j=0; j<cols; j++)
```

{

```
System.out.print (diff[i][j] + " ");
```

}

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

{

{

Output

2	5	3
1	1.	0
0.	1	2

Example #10

```
class darrg
{
    public static void main (String args[])
    {
        int rows, cols;
        int a[][] = {
            {1, 2, 3},
            {4, 5, 6},
            {7, 8, 9}
        };
        rows = a.length;
        cols = a[0].length;
        int t[][] = new int [cols] [rows];
        for (int i=0; i<cols; i++)
        {
            for (int j=0; j<rows; j++)
            {
                t[i][j] = a[j][i];
            }
        }
        System.out.println ("Transpose of given matrix:");
        for (int i=0; i<cols; i++)
        {
            for (int j=0; j<rows; j++)
            {
                System.out.print (t[i][j] + " ");
            }
            System.out.println ();
        }
    }
}
```

{

```
System.out.print(t[i][j] + " ")
```

}

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

?

} }

1	2	3	<u>Output</u>
4	5	6	1 4 7
7	8	9	2 5 8
			3 6 9.

Applet

```
import java.awt.*;
import java.applet.*;
public class aa extends Applet
{
    public void paint (Graphics g)
    {
        g.drawString ("welcome to Applet", 10, 50);
    }
}
```

```
<html>
<applet code = "aa.class" width = 200 height = 200>
</applet>
</html>
```

1. Program of sum using applet.

```
import java.awt.*;
import java.applet.*;
public class aa extends Applet
{
    int a, b, c;
    public void init ()
    {
```

```
a = 5;
```

```
b = 10;
```

```
c = a + b;  
}
```

```
public void paint (Graphics g)  
{
```

```
g.drawString ("the sum is " + String.valueOf(c),  
10, 50);  
}
```

2. Program of factorial using applet.

```
import java.awt.*;  
import java.applet.*;  
public class fact extends Applet  
{
```

```
int m = 1; int i; int m, n, i;
```

```
public void int ()
```

```
{
```

```
for ( i = 1
```

Applet in java program. that executes on web page.

Program of factorial using applet.

```
import java.awt.*;  
import java.applet.*;  
public class fact extends Applet  
{  
    int m, n, p;  
    public void init()  
    {  
        m = 1;  
        n = 5;  
        for (i = 1; i <= n; i++)  
        {  
            m = m * i;  
        }  
        p = m;  
    }  
    public void paint (Graphics g)  
    {  
        g.drawString ("the fact is " + String.valueOf(m),  
                     10, 50);  
    }  
}
```

```
<html>  
<applet code = "fact.class" width=200 height=200>  
</applet>  
</html>
```

Frame

```
import java.awt.*;  
class mm extends Frame  
{
```

```
Label l1, l2;  
public mm (String )  
{
```

```
l1 = new Label ("java program");  
l2 = new Label ("Awt component &");  
add setSize (300,300); setVisible (true);
```

(Using Frame)

```
# import java.awt.*;  
public class Fra extends frame  
{
```

```
Label l1, l2;  
public Fra()  
{
```

```
setLayout (new FlowLayout());
```

```
l1 = new Label ("Using GUI components");  
l2 = new Label ("using frame in program");  
add (l1);  
add (l2);  
}
```

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

```
fra f = new fra();
f.setVisible(true);
f.setSize(200, 200);
}
}
```

#

```
import java.awt.*;
import java.applet.*;
public class lab extends Applet
{
    Label l1, l2;
    public void init()
    {
        l1 = new Label("Welcome to GUI");
        l2 = new Label("Using Label component");
        add(l1);
        add(l2);
    }
}
```

```
<html>
<applet code="Lab.class" width=200 height=200>
</applet>
</html>
```

```
# import java.awt.*;
import java.applet.*;
public class Tex extends Applet
{
```

```
Label l1, l2;
TextField t1, t2;
Button b1;
```

```
public void init()
{
    l1 = new Label ("Name:");
    l2 = new Label ("Address:");
    t1 = new TextField (20);
    t2 = new TextField (20);
    b1 = new Button ("Ok");
    add (l1);
    add (t1);
    add (l2);
    add (t2);
    add (b1);
}
```

```
}
```

```
<html>
<applet code = "Tex.class" width=200 height=200>
</applet>
</html>
```

```
#import java.awt.*;  
public class Ftex extends Frame  
{  
    Label l1, l2;  
    TextField t1, t2;  
    Button b1;  
    public Ftex()  
    {  
        setLayout(new FlowLayout());  
        l1 = new Label ("Enter your 1st no.");  
        t1 = new TextField (20);  
        l2 = new Label ("Enter your 2nd no.");  
        t2 = new TextField (20);  
        b1 = new Button ("OK");  
        add (l1);  
        add (t1);  
        add (l2);  
        add (t2);  
        add (b1);  
    }  
    public static void main (String args [ ])  
    {  
        Ftex f = new Ftex();  
        f.setVisible (true);  
        f.setSize (200, 200);  
    }  
}
```

```
# import java.awt.*;
import java.applet.*;
public class cow extends Applet
{
    Label l1, l2, l3;
    TextField t1, t2, t3;
    Button b1;
```

```
public void init()
{
    l1 = new Label ("Principal:");
    l2 = new Label ("Rate:");
    l3 = new Label ("Pme:");
    t1 = new TextField (20);
    t2 = new TextField (20);
    t3 = new TextField (20);
    b1 = new Button ("Ok");
    add (l1);
    add (t1);
    add (l2);
    add (t2);
    add (l3);
    add (t3);
    add (b1);
}
```

```
<html>
<applet code = "ww.java" width = 200 height = 200>
</applet>
</html>

# import java.awt.*;
public class mm extends Frame
{
    Label l1, l2, l3, l4;
    TextField t1, t2, t3, t4;
    Button b1;
    public mm()
    {
        setLayout(new FlowLayout());
        l1 = new Label("Length:");
        t1 = new TextField(20);
        l2 = new Label("breadth:");
        t2 = new TextField(20);
        l3 = new Label("height");
        t3 = new TextField(20);
        l4 = new Label("Total");
        t4 = new TextField(20);
        b1 = new Button("Ok");
        add(l1);
        add(t1);
        add(l2);
```

```
add(t2);  
add(ls);  
add(ts);  
add(l4);  
add(t1);  
add(b1);  
}
```

```
public static void main (String args [ ])  
{  
    mm f = new mm();  
    f.setVisible (true);  
    f.setSize (200, 200);  
}
```

Check box:-

The check box is used to create two types of components. They are; check boxes and radio buttons. Java.awt.checkbox package is used.

Program of check box:

```
import java.awt.*;
import java.applet.*;
public class ee extends Applet
{
    Checkbox cb1, cb2;
    Button b1;
    public void init()
    {
        setBackground (Color.cyan);
        setBackground (Color.black);
        b1 = new Button ("java");
        cb1 = new Checkbox ("bold");
        cb2 = new Checkbox ("italic");
        add (b1);
        add (cb1);
        add (cb2);
    }
}
```

Program to create radio button.

```
import java.awt.*;
import java.applet.*;
public class vr extends Applet
{
    Checkbox cb1, cb2, cb3;
    CheckboxGroup cb;
    public void init()
    {
        cb=new CheckboxGroup();
        cb1=new Checkbox ("Bold ", cb, false);
        cb2=new Checkbox ("Italic ", cb, false);
        cb3=new Checkbox ("Plain ", cb, true);
        add(cb1);
        add(cb2);
        add(cb3);
    }
}
```

Choice :-

The choice object provides a list of items from which an item can be selected by the user. It is also called dropdown list.

Example of choice.

```
import java.awt.*;  
import java.applet.*;  
public class ss extends Applet  
{  
    Label ll;  
    Choice cb;  
    public void init()  
    {  
        ll = new Label ("oring choice");  
        cb = new Choice();  
        cb.add ("Button");  
        cb.add ("Label");  
        cb.add ("List");  
        cb.add ("TextField");  
        add (ll);  
        add (cb);  
    } }
```

Lst :-

Lst is a collection of items from which the user may select one item or some items more than one items.
Java.awt.List.*; package is used.

Program of LPst.

```
import java.awt.*;
import java.applet.*;
public class pp extends Applet
{
    Label l1;
    LPst co;
    String str[] = {"Java", "C++", "VB", "VB.Net",
                    "Java Beans"};
    public void init()
    {
        co = new LPst(2, false);
        for (int i = 0; i < str.length; i++)
        {
            co.add(str[i]);
        }
        add(co);
    }
}
```

Text area:-

The text area class provides an area where multiple text lines are visible and can be manipulated. Java.awt.TextArea package is used.

Example of text area is -

```
import java.awt.*;
import java.awt.event.*;
import java.applet.*;
public class pp extends Applet
{
```

Button b1;

TextArea T1, T2;

String msg = "This class can also implements text area class in a program.";

```
public void init()
```

```
B1 = new Button ("copy");
```

```
T1 = new TextArea (msg, 5, 14);
```

```
T2 = new TextArea (5, 14);
```

```
T2.setEditable (false);
```

```
B1.addActionListener (this);
```

```
add (T1);
```

```
add (B1);
```

```
add (T2);
```

```
}
```

```
public void actionPerformed (ActionEvent e)
```

```
str = T1.getSelectedText() + "\n";
T2.setText(str);
}
}
```

```
# import java.awt.*;
import java.awt.event.*;
import java.applet.*;
public class wo extends Applet implements
ActionListener
{
Label L1, L2, L3, L4;
TextField T1, T2, T3, T4;
Button b1;
public void init()
{
L1 = new Label ("Enter Principal:");
L2 = new Label ("Enter TPme:");
L3 = new Label ("Enter Rate:");
L4 = new Label ("Enter Result:");
T1 = new TextField (20);
T2 = new TextField (20);
T3 = new TextField (20);
T4 = new TextField (20);
b1 = new Button ("OK");
b1.addActionListener(this);
}
```

```
b1.addActionListener(this);  
add(l1);  
add(T1);  
add(l2);  
add(T2);  
add(l3);  
add(T3);  
add(l4);  
add(T4);  
add(b1);  
}
```

```
public void actionPerformed(ActionEvent e)
```

```
{ if (e.getSource() == b1)
```

```
    int P = Integer.parseInt(T1.getText());  
    int T = Integer.parseInt(T2.getText());  
    int R = Integer.parseInt(T3.getText());  
    int SI = (P * T * R) / 100;  
    T4.setText(String.valueOf(SI));  
}
```

```
# import java.awt.*;  
import java.awt.event.*;  
import java.applet.*;  
public class mm extends Applet implements  
ActionListener.
```

{

Label L1, L2;

TextField T1, T2;

Button b1;

public void init()

{

L1 = new Label ("Enter a no.");

L2 = new Label ("Enter Result");

T1 = new TextField (20);

T2 = new TextField (20);

b1 = new Button ("OK");

b1.addActionListener (this);

add (L1);

add (T1);

add (L2);

add (T2);

add (b1);

}

public void actionPerformed (ActionEvent e)

{

```
if (e.getSource() == b1)
```

```
    int m = 1, i;
```

```
    for (i =
```

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

```
}
```

```
    int n = Integer.parseInt (T1.getSource())
```

```
    if (e.getSource() == b1)
```

```
{
```

```
    int m = 1, i;
```

```
    int n = Integer.parseInt (T1.getText());
```

```
    for (i = 1; i <= n; i++) {
```

```
        m = m * i;
```

```
}
```

```
    T2.setText (String.valueOf (m));
```

```
}
```

```
}
```

```
}
```

```
# Import java.awt.*;
import java.awt.event.*;
import java.applet.*;
public class pp extends Applet implements
    ActionListener
{
    Label l1, l2, l3;
    TextField t1, t2, t3;
    Button b1;
    public void init()
    {
        l1 = new Label ("Enter 1st no.");
        l2 = new Label ("Enter 2nd no.");
        l3 = new Label ("Result");
        t1 = new TextField (20);
        t2 = new TextField (20);
        t3 = new TextField (20);
        b1 = new Button ("OK");
        b1.addActionListener (this);
        add (l1);
        add (t1);
        add (l2);
        add (t2);
        add (l3);
        add (t3);
        add (b1);
    }
```

```
public void actionPerformed (ActionEvent e)
{
    if (e.getSource () == b1)
    {
        int n = Integer.parseInt (T1.getText ());
        int m = Integer.parseInt (T2.getText ());
        int r = m / n;
        T3.setText (String.valueOf (r));
    }
}
```

Java foundation class.

The Java foundation classes enable the original AWT by adding a set of GUI class libraries. JFC provides us with additional visual component classes and a unique way of designing the screen.

Swing

Swing is a set of classes under the JFC that provide lightweight visual components and enable creation of an attractive GUI.

Swing not only contains replacement components for AWT visual components but also complex components like trees and tables that do not have AWT.

In Swing, the main window, also called a top-level container, is the root of a hierarchy, which contains all of the Swing components that appears inside the window.

All Swing components names start with J. For instance, the swing button class is named JButton, whereas the AWT button class is named Button. The swing components are in the javax.swing package.

Top level containers in Swing are:-

- Frames
- Dialogs
- Applets

import javax.swing.*;

```
public class Sweety  
{
```

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

```
    JFrame f = new JFrame ( );
```

```
    JButton b = new JButton ("Click me");
```

```
    b.setBounds (130, 100, 100, 40);
```

```
    f.add (b);
```

```
    f.setSize (400, 500);
```

```
    f.setLayout (null);
```

```
    f.setVisible (true);
```

```
}
```

```
}
```

(130, 100, 100, 40)



(x-args, y-args, width, height)

JFrame

The JFrame is a top-level container or window, which provides a place for placing other Swing components.

A JFrame component is used to create windows in a swing program.

```
import java.awt.*;
import javax.swing.*;
public class ss extends JFrame
{
    public class (String title)
    {
        super (title);
    }
    public static void main (String args[])
    {
        ss j=new ss ("Using Swing");
        j.setVisible (true);
        j.setSize (200, 200);
    }
}
```

JLabel:-

```
import java.awt.*;
import javax.swing.*;
public class ss extends Applet
{
    public void init()
    {
        getContentPane().setLayout(new FlowLayout());
        ImageIcon ic = new ImageIcon("a.gif");
        JLabel ll = new JLabel(ic);
        getContentPane().add(ll);
    }
}
```

JTextField.

The object of JLabel class is a component for placing text in a container. It is used to display a single line of read only text.

```
import java.awt.*;
import javax.swing.*;
public class ss extends JFrame
{
    public ss()
    {
```

{

```
Container con = getContentPane();
con.setLayout(new FlowLayout());
JLabel l1 = new JLabel ("Enter your name");
JTextField t1 = new JTextField (20);
con.add(l1);
con.add(t1);
setVisible (true);
setSize (200, 200);
}
```

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

```
Jtext f = new JTextField();
}
```

```
.
```

JButton :-

The JButton class is used to create a labeled button that has platform independent implementation.

[We can use setBounds (x, y, width, height) to specify the position and size of a GUI component if you set the layout to null. Then (x, y) is the coordinate of the upper-left corner of that component. setBounds is used to define the bounding

rectangle of a component. Ths, Pncludes it;
position and size.]

```
import javax.swing.*;  
public class ss.  
{
```

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

```
JFrame f = new JFrame ("Button");  
JButton b = new JButton ("Click Here");  
b.setBounds (50, 100, 95, 30);  
f.add(b);
```

```
f.setSize (400, 400);  
f.setLayout (null);  
f.setVisible (true);  
}
```

{

(Using no Layout)

```
import javax.swing.*;  
public class ss.  
{
```

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

```
JFrame f = new JFrame ("Button");
```

```
JLabel l1 = new JLabel ("Enter your name");
l1.setBounds (10, 10, 100, 30);
JTextField t1 = new JTextField (20);
t1.setBounds (120, 10, 100, 30);
JButton b = new JButton ("Click Here");
b.setBounds (20, 50, 95, 30);
f.add (l1);
f.add (t1);
f.add (b);
f.setSize (400, 400);
f.setLayout (null);
f.setVisible (true);
}
}.
```

Using FlowLayout:

```
import java.awt.*;
import javax.swing.*;
public class ss {
    public static void main (String args[])
    JFrame f = new JFrame ("Button");
    JLabel l1 = new JLabel ("Enter your name");
    JTextField t1 = new JTextField (20);
    JButton b = new JButton ("Click Here");
```

```
f.add(l1);
f.add(t1);
f.add(b);
f.setSize(400, 900);
f.setLayout(new FlowLayout(FlowLayout.LEFT));
f.setVisible(true);
}
}
```

```
# import java.awt.*;
import javax.swing.*;
public class ss {
    public static void main (String args[]) {
        JFrame f = new JFrame ("Enter your 1st No.");
        JLabel l1 = new JLabel ("Enter your name");
        JTextField t1 = new JTextField (20);
        JLabel l2 = new JLabel ("Enter your 2nd no.");
        JTextField t2 = new JTextField (20);
        JLabel l3 = new JLabel ("Result");
        JTextField t3 = new JTextField (20);
        JButton b = new JButton ("OK");
        f.add(l1);
        f.add(t1);
        f.add(l2);
        f.add(t2);
```

```
f.add(l3);
f.add(t3);
f.add(b);
f.setSize(400, 400);
f.setLayout(new FlowLayout.LEFT));
f.setVisible(true);
}
}.
```

Find factorial value in swing
(ActionEvent is used.)

```
import java.awt.*;
import java.awt.event.*;
import java.swing.*;
public class ss implements ActionListener
{
JLabel l1, l2;
 JTextField t1, t2;
 JButton b1;
 public ss()
 {
 JFrame f = new JFrame("Swing frame");
 l1 = new JLabel("Enter your no.");
 t1 = new JTextField(10);
```

```
t2 = new JLabel ("Result");  
t2 = new JTextField (10);  
b1 = new JButton ("OK");  
b1.addActionListener (this);  
f.add (t1);  
f.add (t2);  
f.add (b1);  
f.add (l1);  
f.add (l2);  
f.setLayout (new FlowLayout (FlowLayout.LEFT));  
f.setVisible (true);  
}
```

```
public void actionPerformed (ActionEvent e)
```

```
{  
if (e.getSource () == b1)
```

```
int n = Integer.parseInt (t1.getText ());
```

```
int m = 1, i;
```

```
for (i = 1; i <= n; i++)  
{
```

```
m = m * i;
```

```
}
```

```
t2.setText (String.valueOf (m));  
}
```

}

```
public static void main (String args [ ] )  
{  
    ss j = new ss ();  
}  
}
```

JTextArea :-

The object of a JTextArea class is a multiline region that displays text. It allows the editing of multiple line text.

```
import javax.swing.*;  
public class jtextarea  
{  
    jtextarea ( )
```

```
    JFrame f = new JFrame ( );  
    JTextArea ww = new JTextArea ("Welcome to  
        java swing programming");  
    ww.setBounds (10, 30, 200, 200);  
    f.add (ww);  
    f.setSize (300, 300);  
    f.setLayout (null);  
    f.setVisible (true);  
}
```

```
public static void main (String args[])
{
    new JPasswordField ();
}
```

JPasswordField.

The object of a JPasswordField class is a text component specialized for password entry. It allows the editing of a single line of text.

```
# import javax.swing.*;
public class SS
{
```

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

```
JFrame f = new JFrame ("Password field e.g.");
JPasswordField jp = new JPasswordField ();
JLabel l1 = new JLabel ("enter Password");
l1.setBounds (10, 100, 120, 30);
jp.setBounds (130, 100, 100, 30);
f.add (jp);
f.add (l1);
f.setSize (300, 300);
f.setLayout (null);
f.setVisible (true);    }    }
```

```
# import javax.swing.*;  
import java.awt.event.*;  
public class ss implements ActionListener  
{
```

```
JLabel lb, l1, l2;
```

```
JPasswordField fp;
```

```
JTextField tL;
```

```
JButton b;
```

```
public ss()
```

```
{
```

```
JFrame f = new JFrame ("Password Field");
```

```
lb = new Label ();
```

```
lb.setBounds (20, 150, 900, 50);
```

```
fp = new JPasswordField ();
```

```
fp.setBounds (100, 45, 100, 30);
```

```
l1 = new JLabel ("Username:");
```

```
l1.setBounds (20, 20, 80, 30);
```

```
l2 = new JLabel ("Password:");
```

```
l2.setBounds (20, 75, 80, 30);
```

```
b = new JButton ("Login");
```

```
b.setBounds (100, 120, 80, 30);
```

```
t1 = new JTextField();  
t1.setBounds(100, 20, 100, 30);
```

```
f.add(t1);  
f.add(t1);  
f.add(l2);  
f.add(t2);  
f.add(jp);  
f.add(b);  
f.add(lb);  
f.setSize(300, 300);  
f.setLayout(null);  
f.setVisible(true);  
b.addActionListener(lets);  
}
```

```
public void actionPerformed(ActionEvent e)  
{
```

```
if (e.getSource() == b)  
{
```

```
String data = "Username" + t1.getText();  
data += ", Password:" + new String(jp.getPassword());  
lb.setText(data);  
}  
}
```

```
public static void main(String args[])  
{  
new ss(); } }
```

JCheckBox:- (multiple tick)

The Jcheckbox class is used to create a checkbox. It is used to turn an option on (true) or off (false). Clicking on a checkbox changes its state from "on" to "off" or from "off" to "on".

```
import javax.swing.*;  
public class ss  
{  
    public ss()  
    {  
        JFrame f = new JFrame ("CheckBox E.g.");  
        JCheckBox c1 = new JCheckBox ("C");  
        c1.setBounds (100, 100, 50, 50);  
        JCheckBox c2 = new JCheckBox ("Java");  
        c2.setBounds (100, 150, 100, 50);  
        f.add (c1);  
        f.add (c2);  
        f.setSize (400, 400);  
        f.setLayout (null);  
        f.setVisible (true);  
    }  
  
    public static void main (String args [ ])  
    {  
        new ss();  
    }  
}
```

JCheckBox with Event.

```
import javax.swing.*;
import java.awt.event.*;
public class ss extends JFrame implements
ActionListener
{
JLabel l;
JCheckBox cb1, cb2, cb3;
JButton b;
public ss();
{
l = new JLabel("Food Ordering System");
l.setBounds(50, 50, 300, 20);
cb1 = new JCheckBox("cake @ 200");
cb1.setBounds(100, 100, 150, 20);
cb2 = new JCheckBox("fruitcake @ 300");
cb2.setBounds(100, 150, 150, 20);
cb3 = new JCheckBox("Tea @ 20");
cb3.setBounds(100, 200, 150, 20);
b = new JButton("Order");
b.setBounds(100, 250, 80, 30);
b.addActionListener(this);
add(l); add(cb1); add(cb2); add(cb3);
add(b);
setSize(400, 400);
```

```
setLayout(null);  
setVisible(true);  
setDefaultCloseOperation(EXIT_ON_CLOSE);  
}
```

```
public void actionPerformed(ActionEvent e)
```

```
{  
    float amount = 0;
```

```
    String msg = " ";
```

```
    if (cb1.isSelected())  
    {
```

```
        amount += 200;
```

```
        msg += " Cake: 200\n";
```

```
}
```

```
    if (cb2.isSelected())  
    {
```

```
        amount += 300;
```

```
        msg += " Fruit Cake: 300\n";
```

```
}
```

```
    if (cb3.isSelected())  
    {
```

```
        amount += 20;
```

```
        msg += " Tea: 20\n";
```

```
}
```

```
        msg += " - - - \n";
```

JOptionPane.showMessageDialog(this, msg +
"Total : " + amount);

{

public static void main (String args [])

{

new ss();

{

{.

JRadioButton :- The JRadioButton is used to create a radio button. If is used to choose one option from multiple options. It is widely used in exam systems or quiz.

import javax.swing.*;

public class jradio.

{

JFrame f;

JRadioButton r1, r2;

public jradio()

{

f = new JFrame();

r1 = new JRadioButton ("Male");

r2 = new JRadioButton ("Female");

r1.setBounds (75, 50, 100, 30);

r2.setBounds (75, 100, 100, 30);

```
ButtonGroup bg = new ButtonGroup();
f.add(r1); bg.add(r2);
f.add(r1); f.add(r2);
f.setSize(300, 300);
f.setLayout(null);
f.setVisible(true);
}

public static void main(String args[])
{
    new jradP();
}
}
```

```
# import javax.swing.*;
import java.awt.event.*;
class jradP extends JFrame implements
ActionListener
{
    JRadioButton rb1, rb2;
    JButton b;
    public ss()
    {
        rb1 = new JRadioButton("Male");
        rb1.setBounds(100, 50, 100, 30);
    }
}
```

```
rb1.setBounds(100, 50, 100, 30);
```

```
rb2 = new JRadioButton("Female");  
rb2.setBounds(100, 100, 100, 30);
```

Button Group

```
ButtonGroup bg = new ButtonGroup();
```

```
bg.add(rb1);                   bg.add(rb2);
```

```
b = new JButton("click");
```

```
b.setBounds(100, 150, 80, 30);
```

```
b.addActionListener(this);
```

```
add(rb1); add(rb2); add(b);
```

```
setSize(300, 300)
```

```
setLayout(null);
```

```
setVisible(true);
```

```
}
```

```
public void actionPerformed(ActionEvent e)
```

```
{
```

```
if (rb1.isSelected())
```

```
{
```

```
JOptionPane.showMessageDialog(this, "You are  
Male.");
```

```
}
```

```
if (rb2.isSelected())
```

```
{
```

```
JOptionPane.showMessageDialog(this, "You are  
Female.");
```

```
}
```

```
}
```

```
public static void main (String args[])
{
    new ss();
}
```

JComboBox :-

The object of Choice class is used to show popup menu of choices. Choice selected by user is shown on the top of a menu.

```
import javax.swing.*;
public class ss
{
```

```
JFrame f;
```

```
public ss():
{
```

```
f = new JFrame ("ComboBox Example");
```

```
String country [] = {"Nepal", "India", "Aus", "USA"};
```

```
cb JComboBox cb = new JComboBox (country);
```

```
cb.setBounds (50, 50, 90, 20);
```

```
f.add (cb);
```

```
f.setLayout (null);
```

```
f.setSize (400, 500);
```

```
f.setVisible (true);
```

```
}
```

```
public static void main (String args[])
{
    new ss();
}
```

#

```
import javax.swing.*;
import java.awt.event.*;
public class ss
{
    JFrame f;
    public ss()
    {
        f = new JFrame ("comboBox Example");
        final JLabel label = new JLabel ();
        label.setHorizontalAlignment (JLabel.CENTER);
        label.setSize (400, 100);
        JButton b = new JButton ("show");
        b.setBounds (200, 100, 75, 20);
        String languages [] = {"C", "C++", "Java"};
        final JComboBox cb = new JComboBox (languages);
        cb.setBounds (50, 100, 90, 20);
        f.add (cb); f.add (label); f.add (b);
        f.setLayout (null);
        f.setSize (350, 350);
        f.setVisible (true);
    }
}
```

b. add ActionListener (new ActionListener()) ;
{

public void actionPerformed (ActionEvent e)
{

String data = "Programming language Selected:";
+ cb.getSelectedItem (cb.getSelectedIndex());

label.setText (data);

}

}

public static void main (String args [])

{

new ss ();

}

}