Java Basics

Session 2

Contents

- Brief history of Java
- Language fundamentals
- Data types
- Variables
- Operators
- Conditional Operators
- Loops
- Functions
- Arrays
- Enhanced for loop

Brief History of Java

- Invented by **James Gosling** at Sun Microsystems.
- Origin name was OOK later renamed as Java.

Features of Java

- General Purpose
- Class based
- Object oriented
- Platform Independent
- Distributed
- Portable (Flavors of Java)
  - JSE
  - JME
  - JEE/J2E
- Robust
- Interpreted

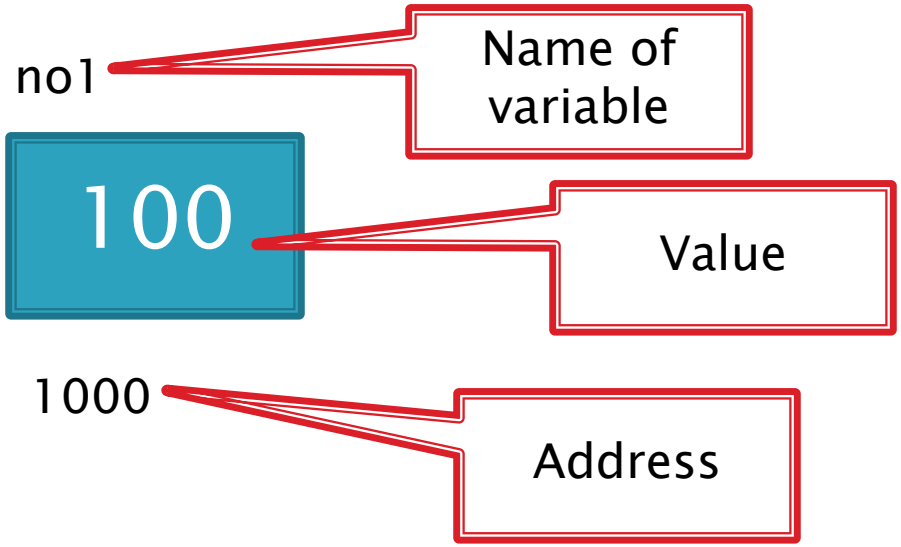
Keywords

- Keywords are certain reserved words which meaning is known to compiler.
- Compiler is capable to work on them.

- if, else, for, class, break, while, switch, case etc.

Variable

- Variable is name of memory location where certain value is stored.
- Variable is a container. Which stores some value



Rules for variable name

- Contain combination of A-Z, a-z, 0-9 & _
- Should not start with number.
- Should not contain any special character and white space.

Data types

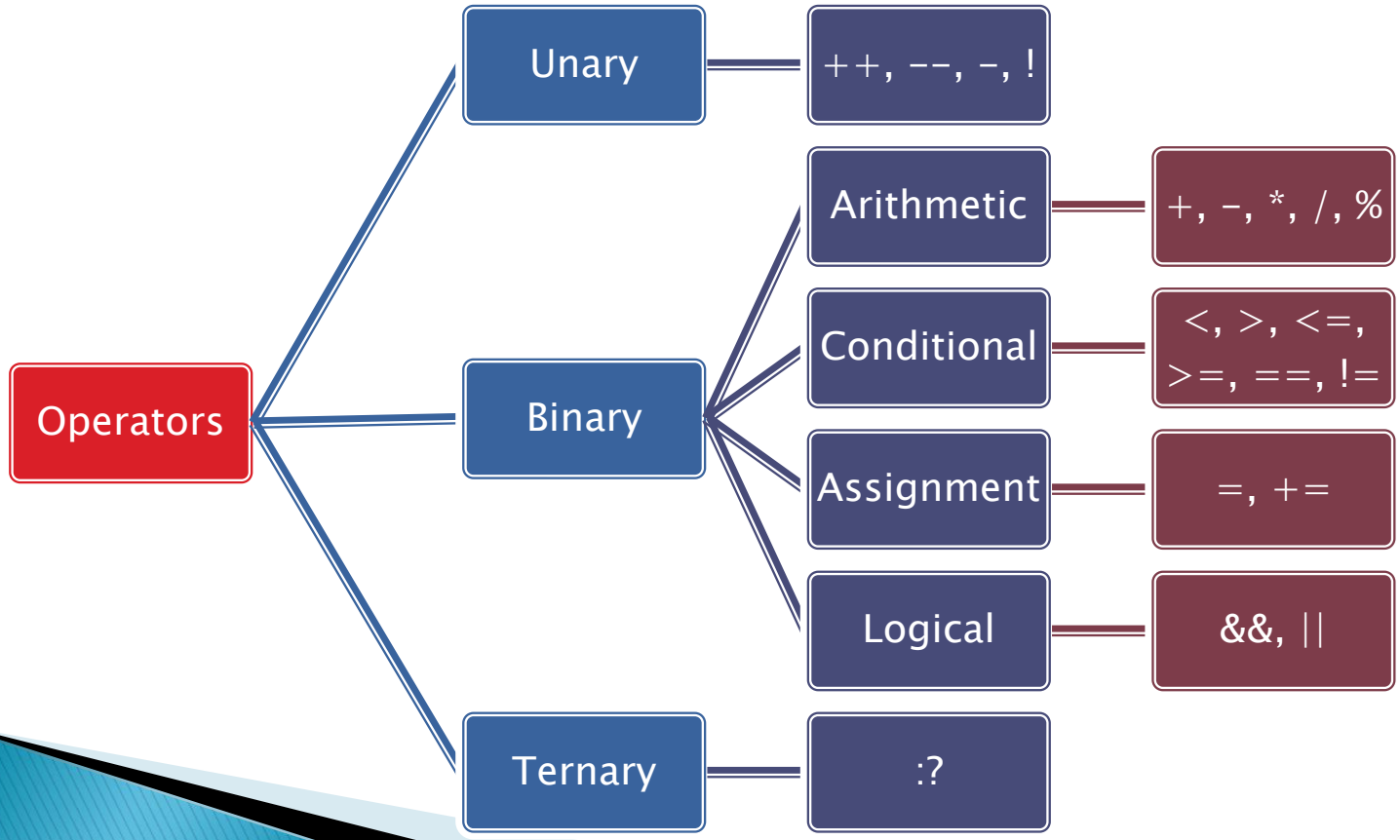
- Data type is nothing but **type of data stored in variable.**
- **Integer – whole number**
- **Decimal numbers – 3.412 – float**
- **12345679.4563217 – double**
- **Character – single character**
- **String – collection of characters.**

Types of Data Type

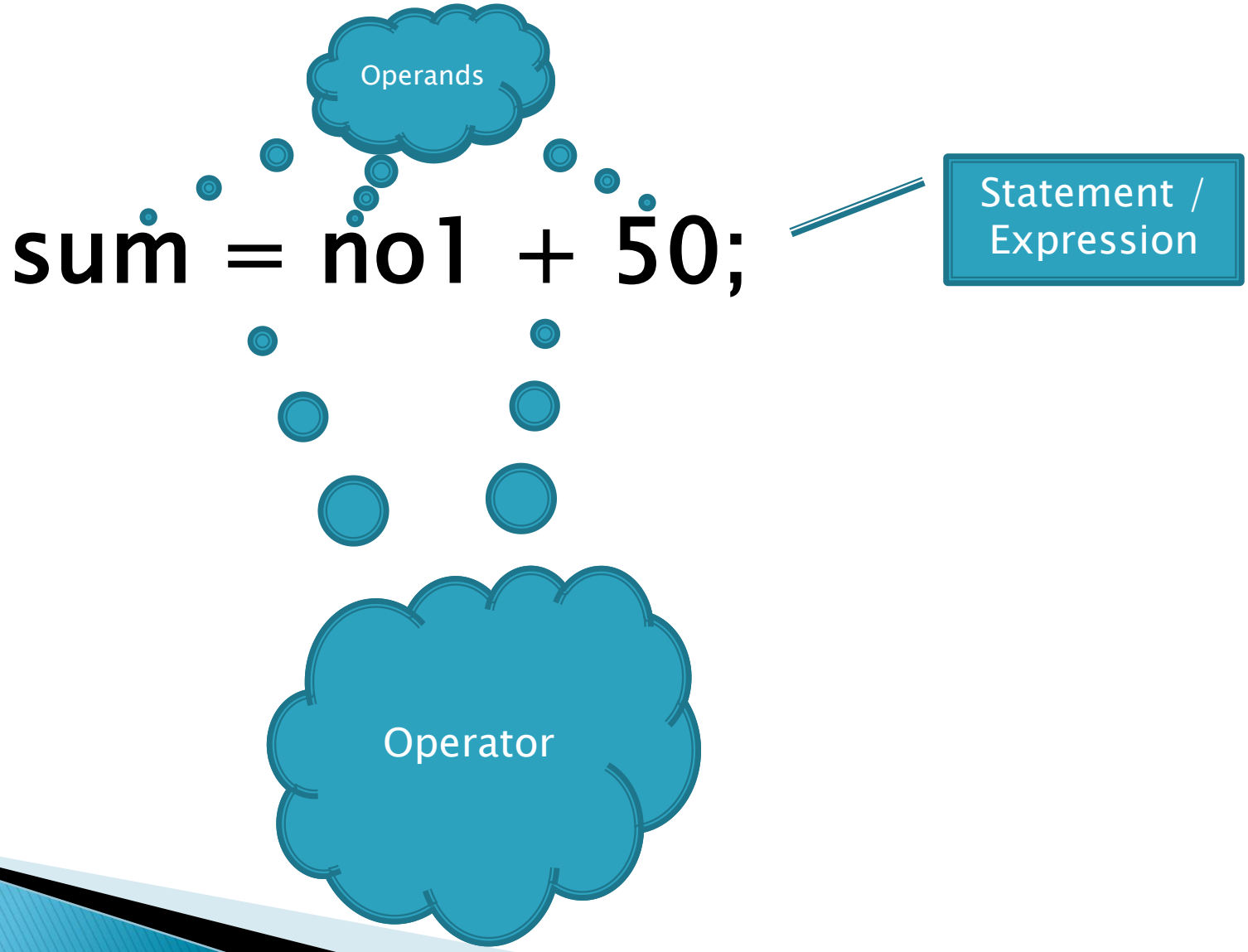
- Primitive
  - int, char, float, double, boolean
- Non Primitive
  - String

Operators

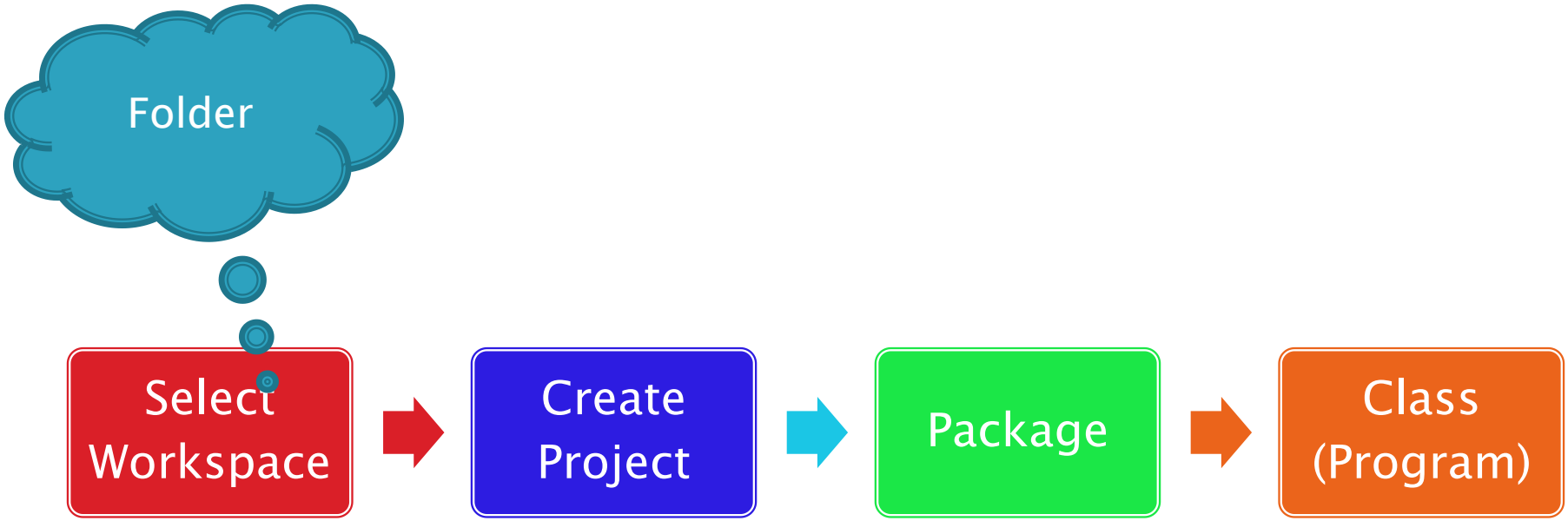
- Operators are mathematical symbols which are used to perform some operations.



Operator, Operand & Expression



Writing 1st Java Program



Basics of Java

- Displaying something on console

- System.out.println()
- This is the function which is used to display some message on screen.
- Eg.
  System.out.println("Hello Friends!!!");
- Workspace (Folder) → Project → Package → Class

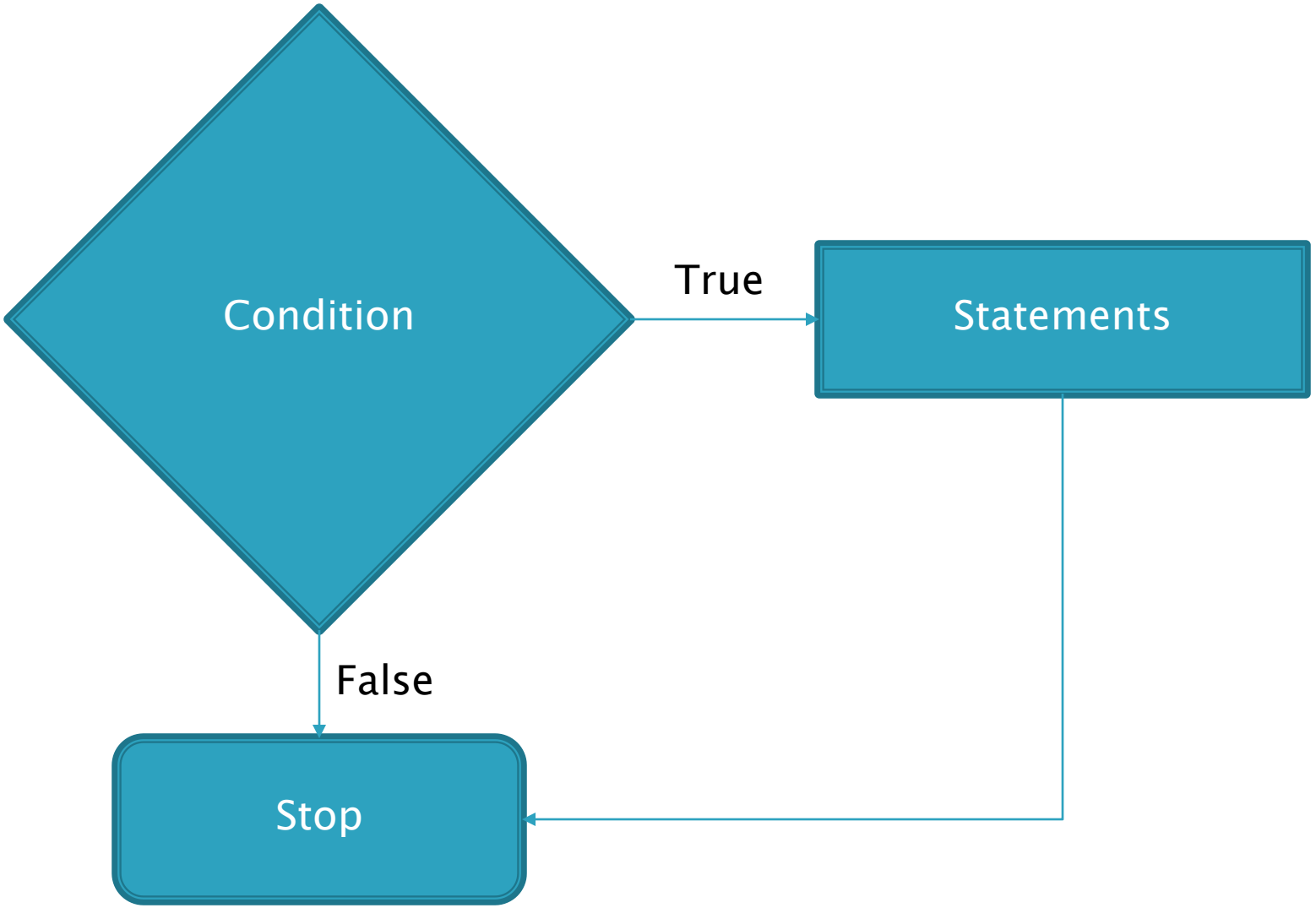
Checking conditions

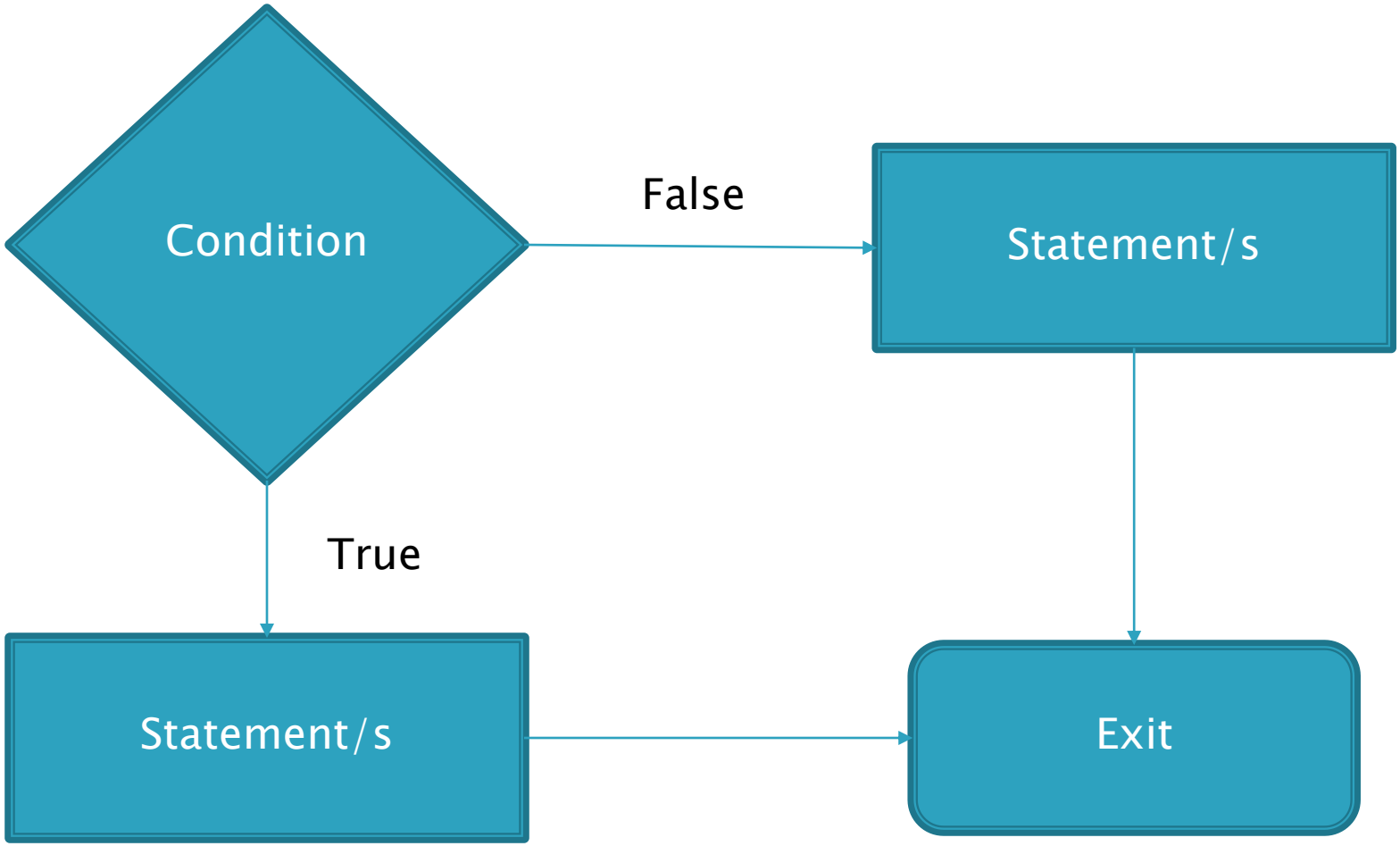
- For checking any condition in Java, one can use several statements like
  - if
  - if – else
  - nested if
  - switch-case

if condition

- if condition is one of the conditional statement in Java.
- It is used to check a condition and executing or skipping some statements depending on condition.
- Syntax

```
if(condition / expression)
{
    Statement(s);
}
```


Condition
True
Statements
False
Stop


Condition
False
Statement/s
True
Statement/s
Exit

if – else statement

- It is extension of simple if statement.
- Syntax

```
if(Condition / Expression)
{
    Statement(s);
}
else
{
    statement(s);
}
```

- < 1000 2%
- >= 1000 & < 3000 5%
- >= 3000 & < 5000 10%
- >= 5000 15%

Nested if – else

- Used when more then one condition is needs to be checked.
- Syntax

```
if(condition)
{
   statement(s);
}
else
{
   if(condition)
   {
        statement(s);
   }
   else
   {
        statement(s);
   }
}
```

Logical Operators

Condition 1	Condition 2	&& Operator	|| Operator
True	True	True	True
True	False	False	True
False	True	False	True
False	False	False	False

switch-case

- Switch statement is used to choose one of several statements based on value.
- Syntax

```
switch(variable)
{
  case 1:
      statement(s);
      break;
  case 2:
      statement(s);
      break;
  default
      statement(s);
}
```

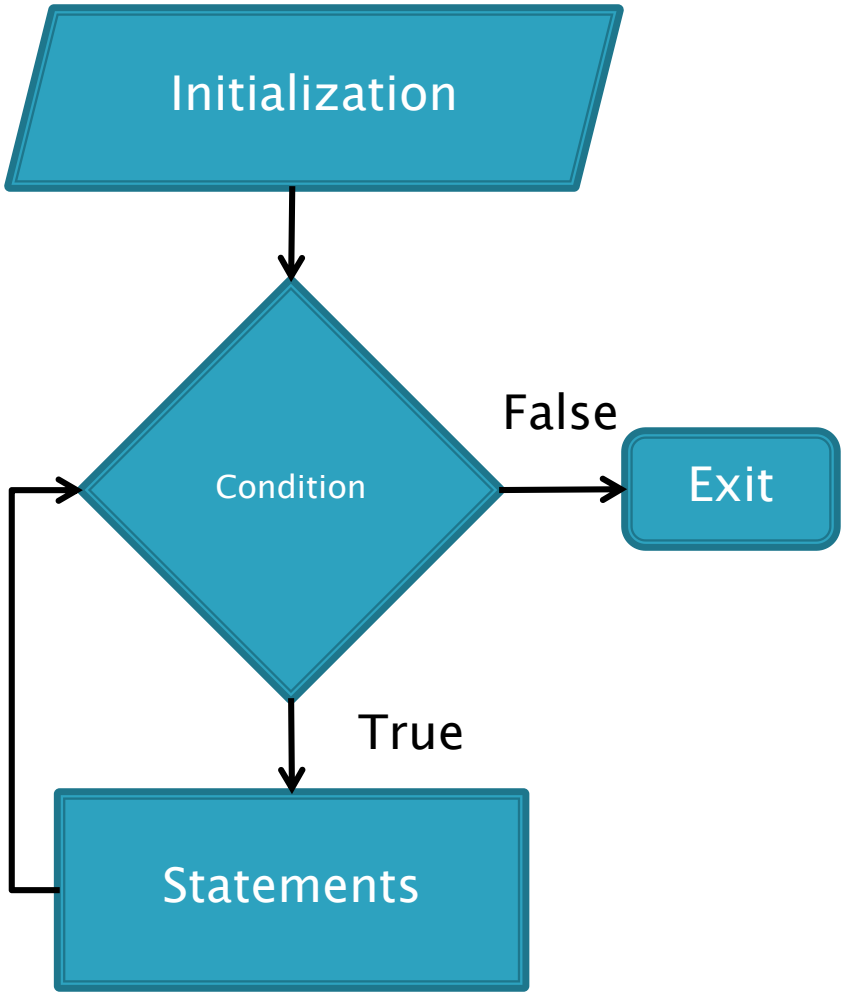
Loops

- When some of statements needs to repeatedly executed, one need to use loops.
- Java supports 2 types of loops
  - Pre-Tested loops
    - for loop
    - while loop
  - Post-Tested loop
    - do-while loop

while loop

- Syntax

```
while(condition)
{
  statement(s);
}
```



for loop

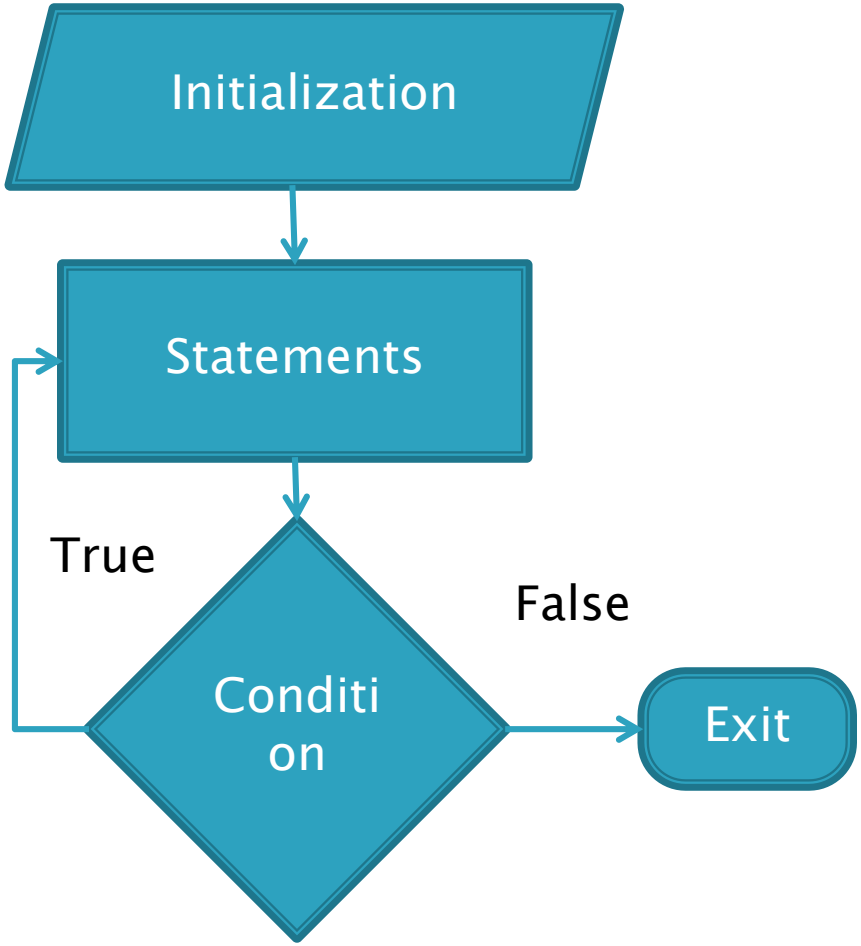
- Syntax

```
for(initilization;condition;increment)
{
  statement(s);
}
```

do-while loop

- Syntax

```
do
{
  statement(s);
}while(condition);
```



Statements in do-while loop executes at least once irrespective of it's condition

Functions

- Function is pre-defined block of statements which can be used / Executed n number of times.
- Types of function
  - Pre-Defined / In-built / Library Function
  - User defined function
- Parts of function
  - Function definition
  - Function call

Function Overloading

These are functions with same name and different signature.

- calAdd(int, int)
- calAdd(int, int, int)
- calAdd(double, double)
- calAdd(double, int)
- calAdd(int, double)

- Function Signature
  - No of arguments
  - Data type of arguments
  - Sequence of arguments

Array

- Array is finite set of homogenous elements stored in continuous memory locations.
- Arrays are known as **First Class Objects** in java.

Syntax:

```
<datatype> <arrayname>[]=new
  <datatype>[size];
```

Array Element is having +ve, integer & Zero based Index No

Enhanced for loop

```
for(<datatype> <variable> : <collection>)
{
     Statement(s);
}
```

2 Dimensional array

[0][0]	0 1	0 2
1 0	1 1	1 2
2 0	2 1	2 2
3 0	3 1	3 2


Thank
You!