Class

A class is a model to create objects. It means that we write properties and actions of objects in the class.

let us consider two objects Samsung Galaxy S4 and iPhone. Suppose Samsung Galaxy S4 have some properties like width = “6.98 cm”, height = “13.6 cm”, OS = “Android”, brand = “Samsung”, price = “1000$” and actions are call(), sendMessage(), browser(), share().

Now, suppose iPhone has some properties such as width = “5.86 cm”, height = “12.3 cms”, OS = “iOS”, brand = “Apple”, price = “1200$” and actions are call(), sendMessage(), browse(), share().

Both objects have the similar properties and actions, but the type is the same “Phone”. This is the class. i.e the name of the class is “Phone”.

A class can be declared using the keyword class followed by a class name. It has also a body within braces. The general syntax to declare a class in Java is shown below:

Syntax:

modifierName class classNAme

{

//class body.

}

# Method

A **method** is a block of code which only runs when it is called. You can pass data, known as parameters, into a method. Methods are used to perform certain actions, and they are also known as **functions.**

Example

Create a method inside Main:

public class main {

static void myMethod ( ) {

//code to be executed

}

}

Object

An object is a basic unit of an object-oriented programming language. It is any real-world thing that has properties and actions. In other words, an entity that has state and behavior is known as**object in Java.**

 Example: - we can take “Pencil”. A pencil is an object. Its name is Natraj.

State: color is black.

Behavior: It is used to write. So, writing is behavior.

Thus, you can take any object around you and think about what property it has? and What action it does?

# Constructor

A **constructor in java**is a block of code, similar to a method that is used to initialize the state of an object in a class through a new operator.

In other words, a constructor is a special type of method that is used to initialize instance in a class.

There are two types of constructors in Java:

Default constructor (no-arg constructor)

Parameterized constructor

Default constructor (no-arg constructor)

A constructor is called "Default Constructor" when it doesn't have any parameter.

Syntax of default constructor:

<class\_name>(){}

Parameterized constructor

A constructor which has a specific number of parameters is called a parameterized constructor.

# Operator

Operators are symbols that perform operations on variables and values. For example, + is an operator used for addition, while \* is also an operator used for multiplication.

|  |  |  |
| --- | --- | --- |
| **Operator Type** | **Category** | **Precedence** |
| Unary | postfix | *expr*++ *expr*-- |
| prefix | ++*expr* --*expr* +*expr* -*expr* ~ ! |
| Arithmetic | multiplicative | \* / % |
| additive | + - |
| Shift | shift | << >> >>> |
| Relational | comparison | < > <= >= instanceof |
| equality | == != |
| Bitwise | bitwise AND | & |
| bitwise exclusive OR | ^ |
| bitwise inclusive OR | | |
| Logical | logical AND | && |
| logical OR | || |
| Ternary | ternary | ? : |
| Assignment | assignment | = += -= \*= /= %= &= ^= |= <<= >>= >>>= |

# Variables

A variable is a container which holds the value while the [Java program](https://www.javatpoint.com/simple-program-of-java) is executed. A variable is assigned with a data type.

There are three types of [variables in java](https://www.scientecheasy.com/2020/05/variables-in-java.html/), depending on their scope:

local variables

instance variables

class variables (static variables).

Data types

**Data types**

Data types specify the different sizes and values that can be stored in the variable. There are two types of data types in Java:

**Primitive data types** - includes byte, short, int, long, float, double, boolean and char

**Non-primitive data types** - such as String, Arrays and Classes

**Primitive Data Types**

In Java language, primitive data types are the building blocks of data manipulation. These are the most basic data types available in [Java language](https://www.javatpoint.com/java-tutorial).

|  |  |  |
| --- | --- | --- |
| **Data Type** | **Default Value** | **Default size** |
| boolean | false | 1 bit |
| char | '\u0000' | 2 bytes |
| byte | 0 | 1 byte |
| short | 0 | 2 bytes |
| int | 0 | 4 bytes |
| long | 0L | 8 bytes |
| float | 0.0f | 4 bytes |
| double | 0.0d | 8 bytes |

**Non-primitive data types** –

They are created by programmers. They are not predefined in java like primitive data types. These data types are used to store a group of values or several values.

For example, we take an array. It can store a group of values. Similarly, another example is a class that can store different values. Therefore, these data types are also known as **advanced**

# Parameters

In Java, a parameter is a variable name with type that is declared within the [method signature](https://www.scientecheasy.com/2020/06/java-methods.html/). The list of parameters is enclosed in parenthesis. Each parameter consists of two parts: type name, and variable name.

Types of parameters:

**Formal Parameter:** A variable and its type as they appear in the prototype of the function or method.   
**Syntax:**

function\_name(datatype variable\_name)

**Actual Parameter :** The variable or expression corresponding to a formal parameter that appears in the function or method call in the calling environment.   
**Syntax:** a

func\_name(variable name(s);

# Main method

**Java main() method**

The main() is the starting point for JVM to start execution of a Java program. Without the main() method, JVM will not execute the program. The syntax of the main() method is:

