

CONSTRUCTOR

CONSTRUCTOR :

Constructor is a special type of non-static method whose name is the same as the class name but it does not have a return type.

Syntax to create the constructor :

A programmer can define a constructor by using the following syntax :

```
[access_modifier] className([Formal_Arguments])  
    {  
        // initialization ;  
    }
```

CONSTRUCTOR BODY :

A constructor body will have the following things :

- Load instructions added by the compiler during compile time.
- Non static initializers of the class.
- Programmer written instructions.

PURPOSE OF THE CONSTRUCTOR

PURPOSE OF THE CONSTRUCTOR :

During the execution of the constructor,

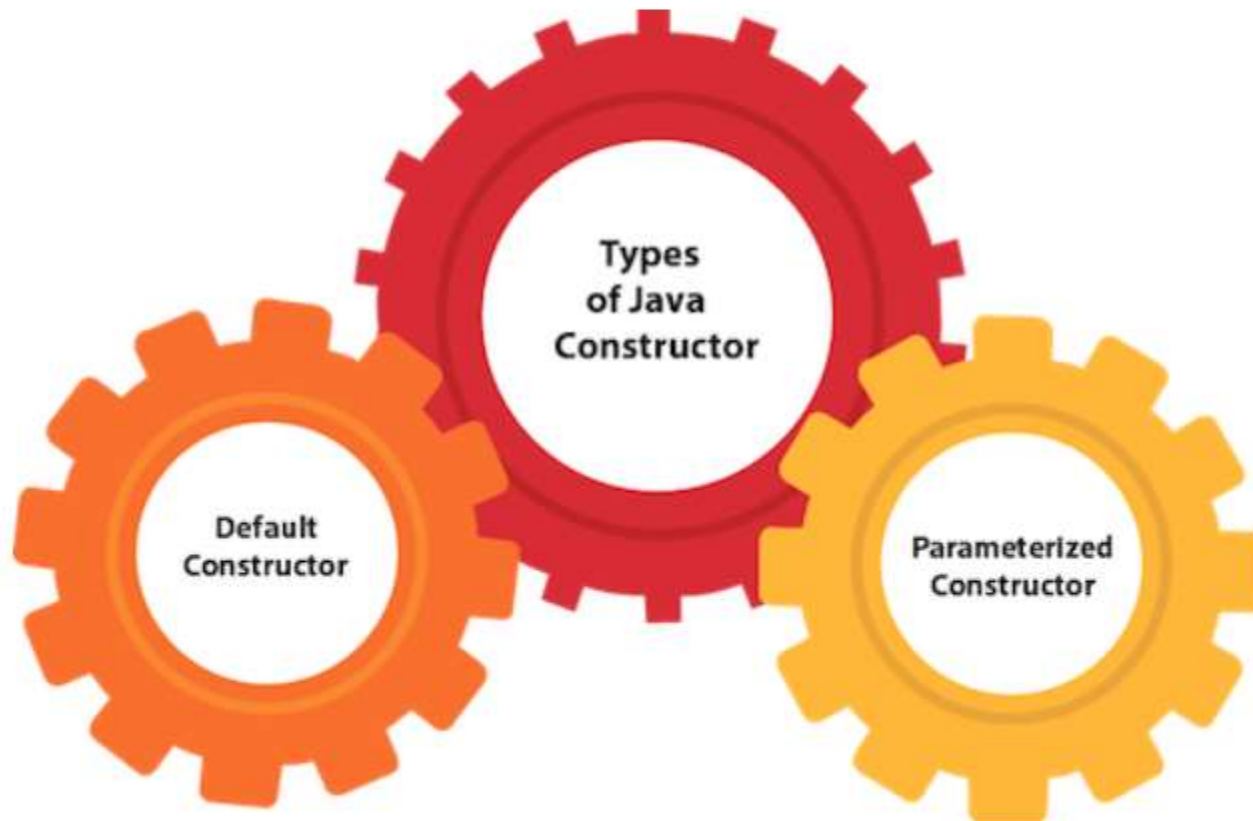
- Non-static members of the class will be loaded into the object.
- If there is a non-static initializer in the class they start executing from top to bottom order.
- Programmer written instruction of the constructor gets executed.

NOTE :

- **If the programmer fails to create a constructor then the compiler will add a default constructor.**

There are two types of constructors in Java:

1. Default constructor (no-arg constructor)
2. Parameterized constructor



CLASSIFICATION OF CONSTRUCTOR :

Constructors can be classified into two types based on the formal argument,

- 1. No argument constructor**
- 2. Parameterized constructor**

NO ARGUMENT CONSTRUCTOR :

A constructor which doesn't have a formal argument is known as a no-argument constructor.

PARAMETERIZED CONSTRUCTOR :

A constructor which has a formal argument is known as parameterized constructor.

CONSTRUCTOR OVERLOADING :

If a class is having more than one constructor it is known as constructor overloading.

RULE :

The signature of the constructor must be different.

NO ARGUMENT CONSTRUCTOR :

A constructor which doesn't have a formal argument is known as a no-argument constructor.

Syntax to create no argument constructor :

```
[access modifier] className()  
{  
    //code ;  
}
```

NOTE :

If the programmer fails to create a constructor then the compiler implicitly adds a no-argument constructor only

PARAMETERIZED CONSTRUCTOR :

The constructor which has a formal argument is known as parameterized constructor.

PURPOSE OF THE PARAMETERIZED CONSTRUCTOR :

Parameterized constructors are used to initialize the variables (non-static) by accepting the data from the constructor in the object creation statement.

CONSTRUCTOR CHAINING :

- A constructor calling another constructor is known as constructor chaining.
- In java, we can achieve constructor chaining by using two ways
 1. this() (this call statement)
 2. super() (super call statement)

this() :

It is used to call the constructor of the same class from another constructor.

RULE :

- this() can be used only inside the constructor.
- It should always be the first statement in the constructor.
- The recursive call to the constructor is not allowed (Calling by itself).
- If a class has n constructors we can use this statement in n-1 constructors only(at least a constructor should be without this())

Difference between constructor and method in Java

There are many differences between constructors and methods. They are given below.

Java Constructor	Java Method
A constructor is used to initialize the state of an object.	A method is used to expose the behavior of an object.
A constructor must not have a return type.	A method must have a return type.
The constructor is invoked implicitly.	The method is invoked explicitly.
The Java compiler provides a default constructor if you don't have any constructor in a class.	The method is not provided by the compiler in any case.
The constructor name must be same as the class name.	The method name may or may not be same as the class name.