



SNS COLLEGE OF TECHNOLOGY

(AN AUTONOMOUS INSTITUTION)

COIMBATORE – 35

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING



UNIT III

INHERITANCE AND POLYMORPHISM

Constructors

Java constructors or constructors in Java is a terminology used to construct something in our programs. A constructor in Java is a special method that is used to initialize objects. The constructor is called when an object of a class is created. It can be used to set initial values for object attributes.

Constructor is a block of codes similar to the method. It is called when an instance of the class is created. At the time of calling the constructor, memory for the object is allocated in the memory. It is a special type of method that is used to initialize the object. Every time an object is created using the new() keyword, at least one constructor is called.

How Java Constructors are Different From Java Methods?

- Constructors name must have the same name as the class name
- No return type
- Cannot be abstract, final static & synchronized

Types of Constructor

1. Default Constructor
2. Parameterized constructor

```
class Demo
{
.....
// A Constructor
Demo() {
}
}
```

```
.....  
}
```

Default Constructor

A constructor that has no parameters is known as default the constructor. A default constructor is invisible. And if we write a constructor with no arguments, the compiler does not create a default constructor. It is taken out. It is being overloaded and called a parameterized constructor. The default constructor changed into the parameterized constructor. But Parameterized constructor can't change the default constructor.

```
class Box {  
    // Default Constructor  
    Box()  
{  
    System.out.println("Default constructor");  
}  
    // Driver function  
    public static void main(String[] args)  
    {  
        Box hello = new Box();  
    }  
}
```

Note: Default constructor provides the default values to the object like 0, null, etc. depending on the type.

2. Parameterized Constructor in Java

A constructor that has parameters is known as parameterized constructor. If we want to initialize fields of the class with our own values, then use a parameterized constructor.

```
class Example{  
    Example(int i, int j){  
        System.out.print("parameterized constructor");  
    }  
    Example(int i, int j, int k){  
        System.out.print("parameterized constructor");  
    }  
    public static void main(String args[]){  
        Example obj = new Example();  
    }  
}
```