Classes and Objects in Java

In this we will learn:

- ✓ What is class in Java?
- ✓ What is an object in Java?
- ✓ Real life examples of classes & objects in Java
- ✓ Declaration of class in Java
- ✓ Components of class in Java
- ✓ A simple programe structure of Java classes and objects.
- ✓ Difference between class and object

Classes and Objects in Java

- Java is a true object-oriented programming language.
- In the object-oriented programming language, we design and develop an application program using classes and objects.
- All Java programs is made up of classes and program must be encapsulated in a class.
- A class is a model to create objects.
- It means that we write properties and actions of objects in the class.
- Properties are represented by variables, and actions are represented by methods.
- Therefore, a class consists of variables and methods.

What is Object in Java

- 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.
- The state represents properties and behavior represents actions or functionality.
- For example, book, pen, pencil, mobile phone, etc are an Objects.
- Objects in Java consist of states or attributes (called data members) and behaviors (called methods).
- An object is an instance of a class. Each instance of an object holds its own relevant data.

Characteristics of Object

An object in Java has three characteristics:

1. **State:** State represents properties or attributes of an object. It is represented by instance variable.
2. **Behavior**: Behavior represents functionality or actions. It is represented by methods in Java.
3. **Identity:** Identity represents the unique name of an object. It differentiates one object from the other. The unique name of an object is used to identify the object.

What is Class?

- A class in Java is a fundamental building block of object-oriented programming (OOP) language.
- In other words, a class is the basic unit of OOP.
- According to OOPs concept in Java, a class is the blueprint/template of an object.
- It contains the similar types of objects having the same states (properties) and behavior.
- In other words, a class can also be defined as “a class is a group of objects which are common to all objects of one type”.
- A class is basically user-defined data types that act as a template for creating objects of the identical type.
- Every Java class contains attributes and methods.
- Attributes represent the state of a class. Methods (also known as functions) represent the behavior of a class.

Class Declaration in Java

- 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:

```
modifierName class className
        {
            // class body.
        }
```

Example:

```
public class Employee {
        // class code
}
```

Points to remember about Objects:

1. You can create any number of objects of a class.
2. The process of creating an object of a particular class is called instantiating of an object.
3. The object is called an instance of class.
4. Objects are the basic runtime entities in object-oriented systems.
5. All objects in a system take a separate memory space independently of each other.

Components of Class in Java

A class can have the following components to act as a template.

1. Field declarations;
2. Constructor declarations;
3. Method declarations;
4. Instance block declarations;
5. Static block declarations;

Difference between Classes and Objects in Java

- A class is a user-defined data type, whereas an object is an instance of class data type.
- A class generates objects, whereas an object gives life to a class.
- Classes do not occupy memory location, but objects occupy memory location.
- Classes cannot be manipulated due to not available in the memory location, but objects can be manipulated.