

Optional Class:

- Java announced a new class **Optional** in jdk8 which is used to deal with NullPointerException in Java application
- It is a public final class, so it can't be inherited.
- It is present in **java.util** package.
- Optional class offers methods which are used to check the presence of value for particular variable.

Static methods in interface

- The static methods in interface are similar to default method so we need not to override this method in implementation class.
- Since these methods are static, we cannot override them in the implementation classes.
- We can safely add them to the existing interfaces without changing the code in the implementation classes.
- Java interface static method helps us in providing security by not allowing implementation classes to override them

Java enum (Java 1.5v)

- A Java enum is nothing but a list of instance variables, methods (just like a class).
- These are a group of constants (**public**, **static** and **final** ==> **unchangeable** - **cannot be overridden**)
- An enum cannot be used to create objects, and it cannot extend other classes.
- To create an enum, use the enum keyword (instead of class or interface), and separate the constants with a comma.

NOTE:

- 1) We can declare an enum inside a class.
- 2) We can use enum in switch
- 3) We can retrieve the constants in enum through for loop and for-each
- 4) We can write methods also in enum

Inner Classes

- Inner class are defined inside the body of another class.
- These classes can have access modifier or even can be marked as abstract and final.
- Inner classes have special relationship with outer class instances, which allows them to have access to outer class members including private members too.

Types of inner classes

There are four types of inner classes:

- ✓ Nested Inner Class
- ✓ Static Inner Class.
- ✓ Method Local Inner Class
- ✓ Anonymous Inner class.

Nested Inner Class

- As the name suggests, this type of inner class involves the nesting of a class inside another class.
- The inner class can access the private variables of the outer class.
- We can modify access to the inner class by using access modifier keywords such as private, protected, and default

Static Inner Class

- A static class i.e. created inside a class is called static nested class in java.
- It cannot access non-static data members and methods. It can be accessed by outer class name.
- Inside a static Inner class we can write static block also. if we are writing two static blocks one in outer class and one in inner class, the first priority will be given to outer class.

Method Local Inner Class

- In this case the outer class method contains the inner class
- However, the inner class cannot use the variables of the outer class if they are not declared as final values.
- This rule was until JDK 1.7.
- After that, inner classes can access non-final local variables also.

Anonymous Local Inner Class

- As the name suggests these inner classes have no name at all.
- The definition of the classes are written outside the scope of the outer class