



Interface



List of Concepts Involved:

- What is interface?
- Need of Interface
- Important key points of Interface
- Abstract vs interface
- Additional features of Interface
- Functional Interface

Topics covered Yesterday's Session:

- final keyword in Java

What is an interface?

- It will allow only abstract methods.
- It cannot be instantiated
- Variables are "public static final".
- Methods are "public and abstract".
- No Constructor in Interface

Need of Interface

- Any service requirement specification (srs) is called an interface.

Important key points of Interface

- Whenever we are implementing an interface compulsory for every method of that interface we should provide implementation otherwise we have to declare class as abstract in that case child class is responsible to provide implementation for remaining methods.
- Whenever we are implementing an interface method, it should be declared as public, otherwise we will get compile time error.
- In Java applications, it is not possible to extend more than one class to a single class but it is possible to extend more than one interface to a single interface.
- In Java applications, it is possible to implement more than one interface into a single implementation class.

Abstract vs interface

Abstract class

- If we are talking about implementation but not completely (partial implementation) then we should go for abstract class.
- Every method present inside abstract class need not be public and abstract.
- There are no restrictions on abstract class method modifiers.
- Every abstract class variable need not be a public static final.
- Inside the abstract class we can take constructor.

Abstract vs interface

interface

- If we don't know anything about implementation just we have requirement specification then we should go for an interface.
- Every method present inside the interface is always public and abstract whether we are declaring or not.
- We can't declare interface methods with the modifiers private, protected, final, static, synchronized, native, strictfp.
- Every interface variable is always a public static final whether we are declaring or not following modifiers. private, protected, transient, volatile.
- Inside the interface we can't take constructor.

JAVA 8 Features over Interfaces:

1. Default Methods in Interfaces
2. Static Methods in Interfaces
3. Functional Interfaces

Functional Interface

- If any Java interface allows only one abstract method then it is called a "Functional Interface".
- To make any interface as Functional Interface then we have to use the following annotation just above of the interface. `@FunctionalInterface`

Ex: `java.lang.Runnable`
`java.lang.Comparable`

Next Lecture

- Lambda Expression



▶ THANK YOU ◀