# Smart Programming : YouTube Channel

An investment in Knowledge pays the best interest….




# Interfaces in Java

## => Interface

-> Interfaces are similar to abstract class which can contain variables and methods but having all the variables as "public static final" and methods as "public abstract"

-> In simple way we can say that all the methods in an interface are abstract and thus methods cannot have implementation part or body part

-> Interface is a blueprint of class which specifies what must do and not how

-> Syntax :

```
access-modifier interface InterfaceName extends InterfaceName, -, -
{
    //variables (public static final)
    //abstract methods (public abstract)
}
```

```
interface Vehicle
{
    void start();
    void changeGear();
}
```

**-> Use of Interfaces :-**

1. It is used to achieve total abstraction
2. It is used to achieve multiple inheritance (as multiple inheritance is not supported in java in case of classes)
3. It is used to achieve loose coupling

## Interview Questions :-

### 1. What is difference between concrete class, abstract class & interface

-> Concrete Class : Syntax : class ClassName{}

Abstract Class : Syntax : abstract class ClassName{}

Interface : Syntax : interface InterfaceName{}

-> Concrete Class : we can declare only concrete methods

Abstract Class : we can declare concrete methods and abstract methods

Interface : we can declare only abstract methods

-> Concrete Class : we can create objects

Abstract Class : We cannot create an object but we can declare reference variable name

Interface : We cannot create an object but we can declare reference variable name

-> Concrete Class : we cannot achieve abstraction

Abstract Class : we can achieve partial abstraction

Interface : we can achieve full abstraction

-> Concrete Class : methods & variables are same as we declare

Abstract Class : methods & variables are same as we declare

Interface : methods we have declared are always "public abstract" and variables we have declared are always "public static final"

-> Concrete Class : constructors are allowed

Abstract Class : constructors are allowed

Interface : constructors are not allowed

-> Concrete Class : multiple inheritance is not supported

Abstract class : multiple inheritance is not supported

Interface : multiple inheritance is supported

---

**=> Points to remember :**

1. Interfaces cannot be private or protected but nested interface can be anything i.e. private, protected, default & public

2. Interface represents IS-A relationship

3. Interface new features :-

3.1 (JDK 8) We can create default methods in an interface which have implementation part

3.2 (JDK 8) We can create static methods in an interface

3.3 (JDK 9) We can create private methods in an interface

3.4 (JDK 9) We can create private static methods in an interface

**=> What is marker interface ?**

-> Any interface which does not contain any abstract method or any variable is known as marker interface

-> It is used to provide some extra feature or abilities to the object at runtime

-> For example :

1. Cloneable interface (java.lang)
2. Serializable interface (java.io)
3. Remote interface (java.rmi)

# Company Links & Contacts

**Company Name:** Smart Programming (+91 62838-30308)

**Address :** Chandigarh & Mohali (Punjab), India

**Websites:** https://www.smartprogramming.in/

https://courses.smartprogramming.in

**Android App:**
https://play.google.com/store/apps/details?id=com.smartprogramming

**YouTube Channel:**
https://www.youtube.com/c/SmartProgramming