### ****https://www.codingninjas.com/studio/library/functional-interface-in-java****

### ****https://www.digitalocean.com/community/tutorials/java-8-functional-interfaces****

### ****Types of Interface****

There are two types of interface in Java:  
**1. Marker Interface:**In this type of interface in Java has no methods in it. The main purpose of this type of interface is to mark a class as having a certain capability.

**2. Functional Interface:** In this type of interface in Java has only one abstract method. It is used in functional programming. It also helps us to use [**lambda expressions**](https://www.codingninjas.com/studio/library/java-lambda-expression).

In this blog, we will be discussing about the **functional interface in Java**. Let us understand more about it.

**What is a functional interface?**

A functional interface is an interface that contains only a single abstract method (a method that doesn’t have a body).

**Why do we need a functional interface?**

The main reason is that without a functional interface, we would have to use an Anonymous class. Functional interfaces make the code more straightforward to read. For example, an anonymous class is created like this:

**Types of Functional Interface**

**1. Consumer**

**2. Predicate**

**3. Function**

**4. Supplier**

C:\Users\Elanchezhian>javap java.util.function.Function

Compiled from "Function.java"

public interface java.util.function.Function<T, R> {

public abstract R apply(T);

public <V> java.util.function.Function<V, R> compose(java.util.function.Function<? super V, ? extends T>);

public <V> java.util.function.Function<T, V> andThen(java.util.function.Function<? super R, ? extends V>);

public static <T> java.util.function.Function<T, T> identity();

}