

## Functional Interface

An Interface which has only one abstract method is known as Functional Interface.

It can have any number of default and static method but only one abstract method is allowed.

@FunctionalInterface is the annotation used to forcefully tell the compiler that the interface is a FunctionalInterface.

Sample code:

```
package com.technicalstack.basic;
@FunctionalInterface
public interface InterfaceFun {
    public abstract void f1();

    default void f2(){

    }

    static void f3(){

    }
}
```

If we compile above code ,it compiles successfully.

now if we add another function to the above one with @FunctionalInterface annotation then the compiler will show below error in eclipse:

Invalid '@FunctionalInterface' annotation; InterfaceFun is not a functional interface InterfaceFun.java

If we try to compile the same class using command prompt,it will show below error:

```
cmd: >javac InterfaceFun.java
```

```
InterfaceFun.java:3: error: Unexpected @FunctionalInterface annotation
```

```
@FunctionalInterface
```

```
InterfaceFun is not a functional interface
```

```
multiple non-overriding abstract methods found in interface InterfaceFun
```

```
1 error
```

So here @FunctionalInterface annotation also performs work of validating the functional interface.