

ORACLE®

Lesson 1-3: Functional Interfaces And Their Definition

Lambda Expression Types

- A Lambda expression is an anonymous function
 - It is not associated with a class
- But Java is a strongly typed language
 - So what is the type of a Lambda expression?
- A Lambda expression can be used wherever the type is a functional interface
 - This is a single abstract method type
 - The Lambda expression provides the implementation of the abstract method

Functional Interface Definition

- An interface
- Has only one abstract method
- Before JDK 8 this was obvious
 - Only one method
- JDK 8 introduces default methods
 - Multiple inheritance of behaviour for Java
- JDK 8 also now allows static methods in interfaces
- `@FunctionalInterface` annotation

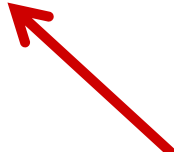
Functional Interfaces

Examples

```
interface FileFilter      { boolean accept(File x); }  
interface ActionListener { void actionPerformed(...); }  
interface Callable<T>    { T call(); }
```

Is This A Functional Interface?

```
@FunctionalInterface  
public interface Runnable {  
    public abstract void run();  
}
```



Yes. There is only
one abstract
method

Is This A Functional Interface?

@FunctionalInterface

```
public interface Predicate<T> {  
    default Predicate<T> and(Predicate<? super T> p) {...};  
    default Predicate<T> negate() {...};  
    default Predicate<T> or(Predicate<? super T> p) {...};  
    static <T> Predicate<T> isEqual(Object target) {...};  
    boolean test(T t);  
}
```

Yes. There is still only one abstract method

Is This A Functional Interface?

`@FunctionalInterface`

```
public interface Comparator {  
    // Static and default methods elided  
    int compare(T o1, T o2);  
    boolean equals(Object obj);  
}
```

Therefore only one
abstract method

The equals(Object) method is
implicit from the Object class

Example Uses Of Lambda Expressions

- Variable assignment

```
Callable c = () -> process();
```

- Method parameter

```
new Thread(() -> process()).start();
```


Section 3

Summary

- Lambda expressions can be used anywhere the type is a functional interface
 - A functional interface has only one abstract method
- The Lambda expression provides the implementation of the single abstract method of the functional interface

ORACLE®