

参考材料：一入Java深似海 B站 第1季第4期

**核心内容：**

理解@FunctionalInterface

函数式接口设计

函数式在框架中使用

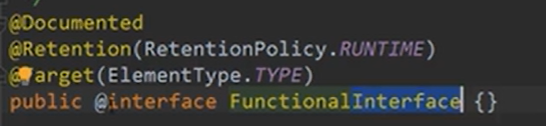
Stream API设计

# FunctionalInterface

**Java Doc说明：**用于函数式接口类型声明的注解类型，接口实例被Lambda表示式、方法应用或构造器应用创建。

只能有一个抽象方法，排除接口默认方法及覆盖Object的公开方法。同时不能被标注在注解、类及枚举上。

**如果任一接口满足以上函数式接口的要求，无论声明是否标注该注解，均被编译视作函数式接口。**

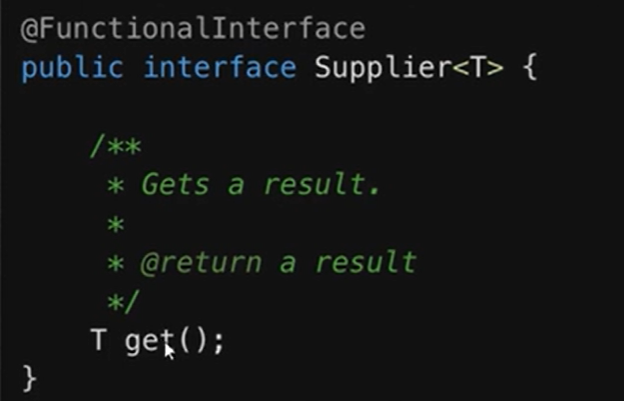


## 函数式接口类型: **java.util.function.xxxx**

1. 提供类型： Supplier<T>
2. 消费类型： Consumer<T>
3. 转换类型: Function<T,R>
4. 断定类型: Predicate<T>
5. 隐藏类型: Action：Runnable等类型，无入参

使用时参考各接口定义，调用accept(入参)或apply等方法

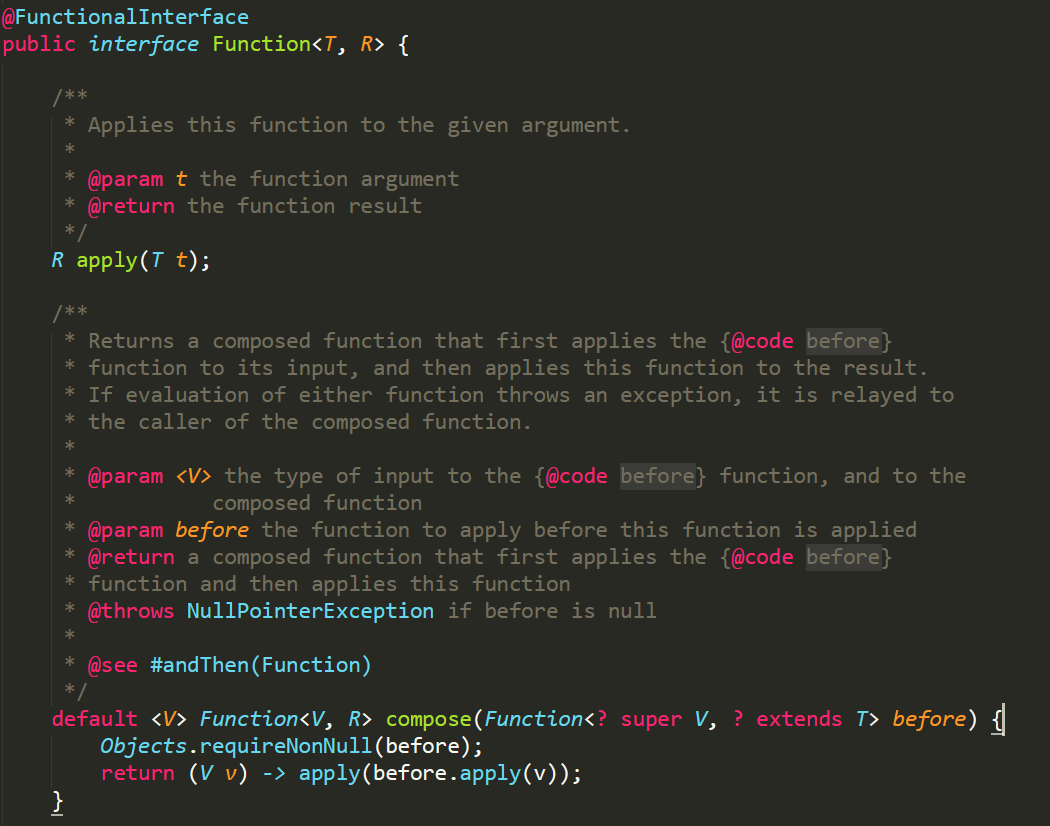
**Supplier类型**

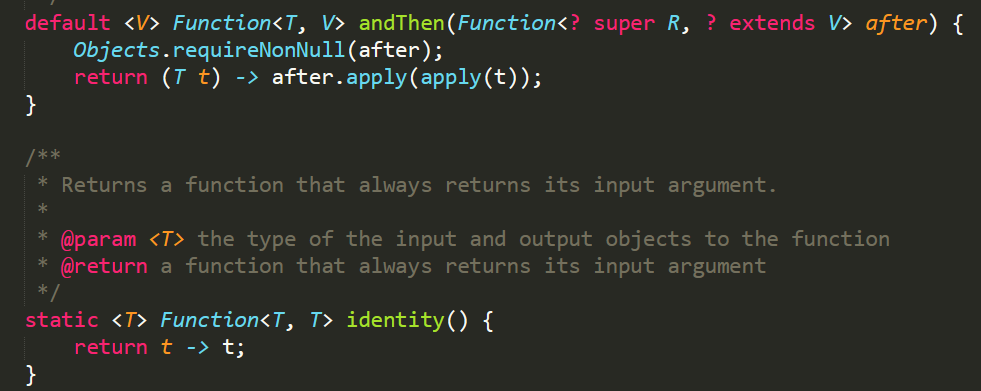


Consumer<T>



Function





可以借助andThen、compose等方法实现链式操作

样例：



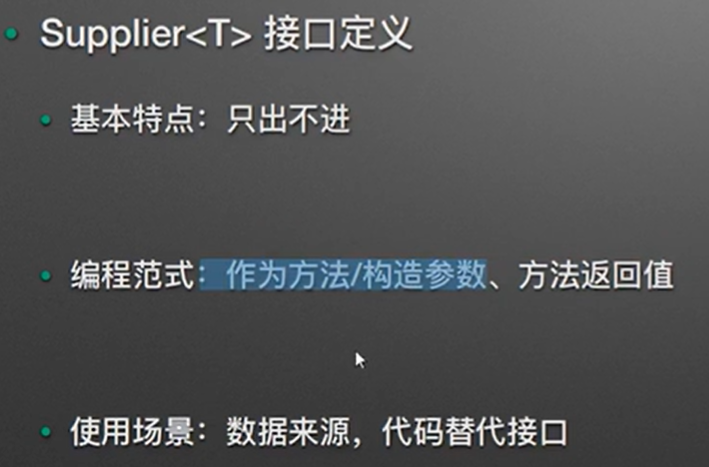
Predicate接口



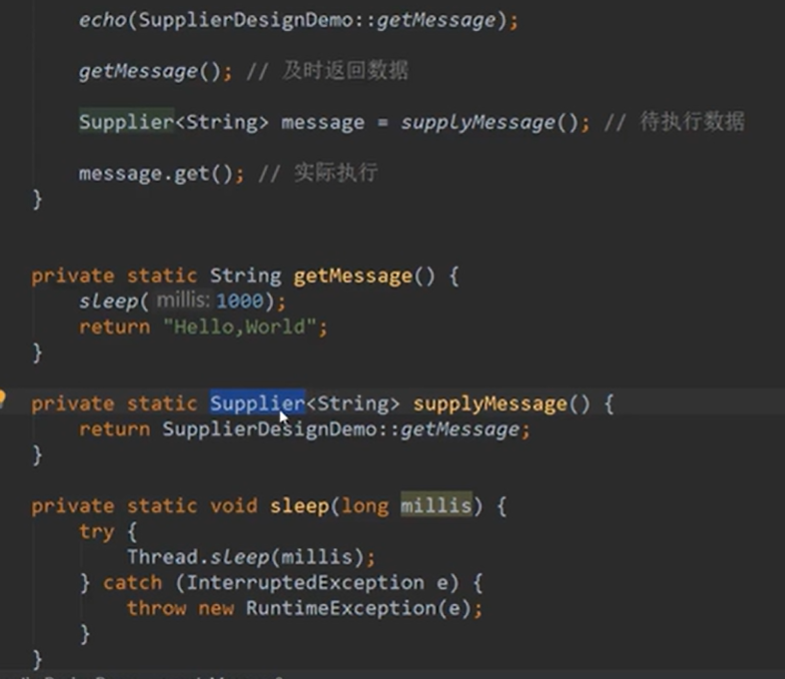


# 函数式接口设计

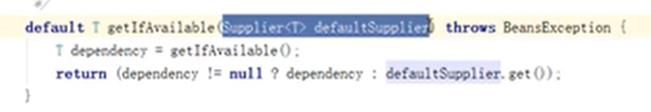
## Supplier理念



只有执行get的时候才触发。也就是说下面代码，如果没有get调用，则只打印一次Hello world



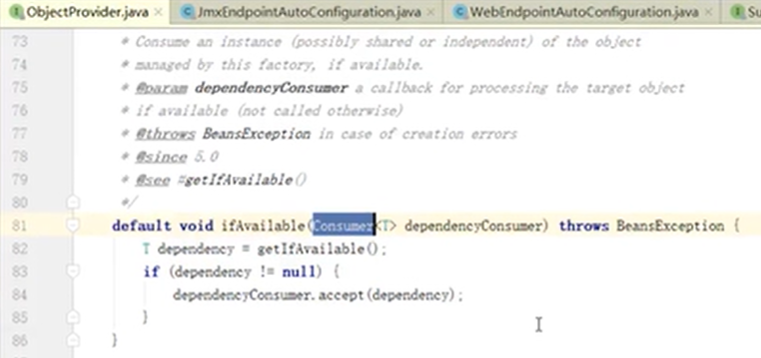
具体实现：**ObjectProvider**类--Spring源码



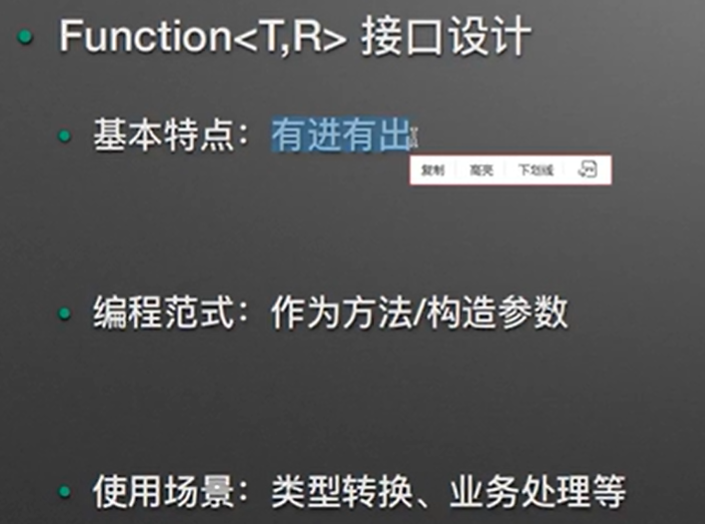
## Consumer理念



Spring framework 5.0源码

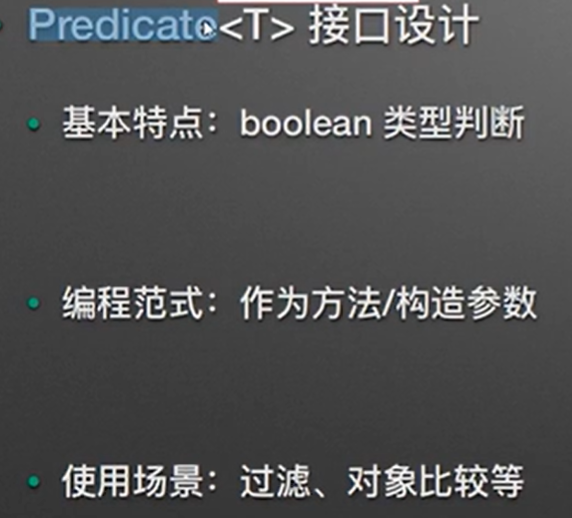


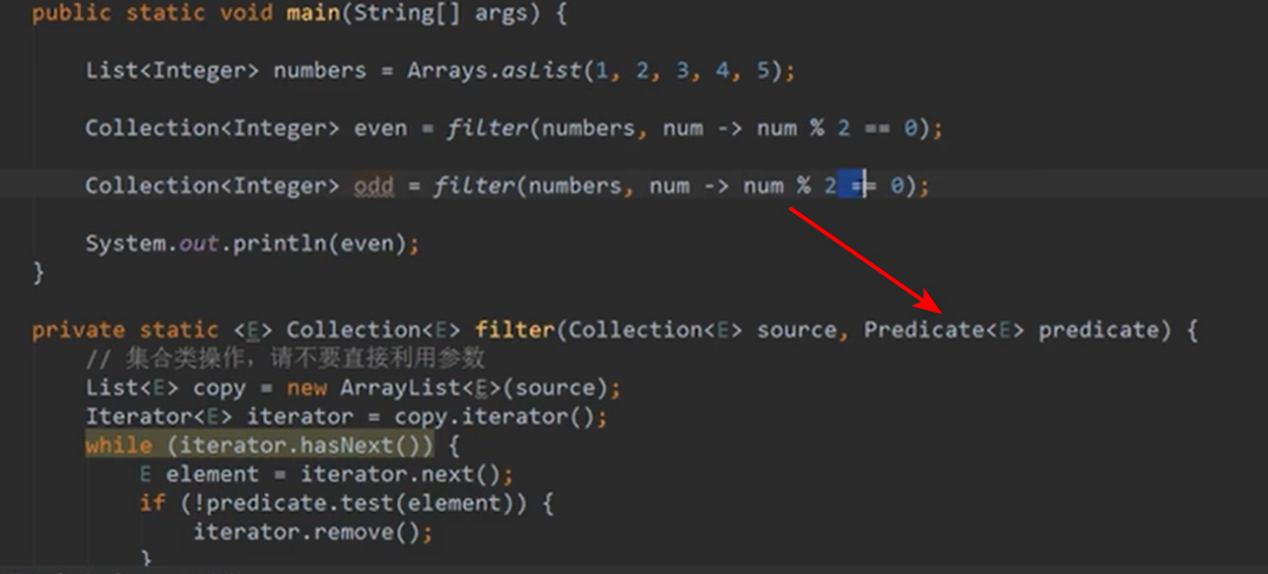
## Function 理念

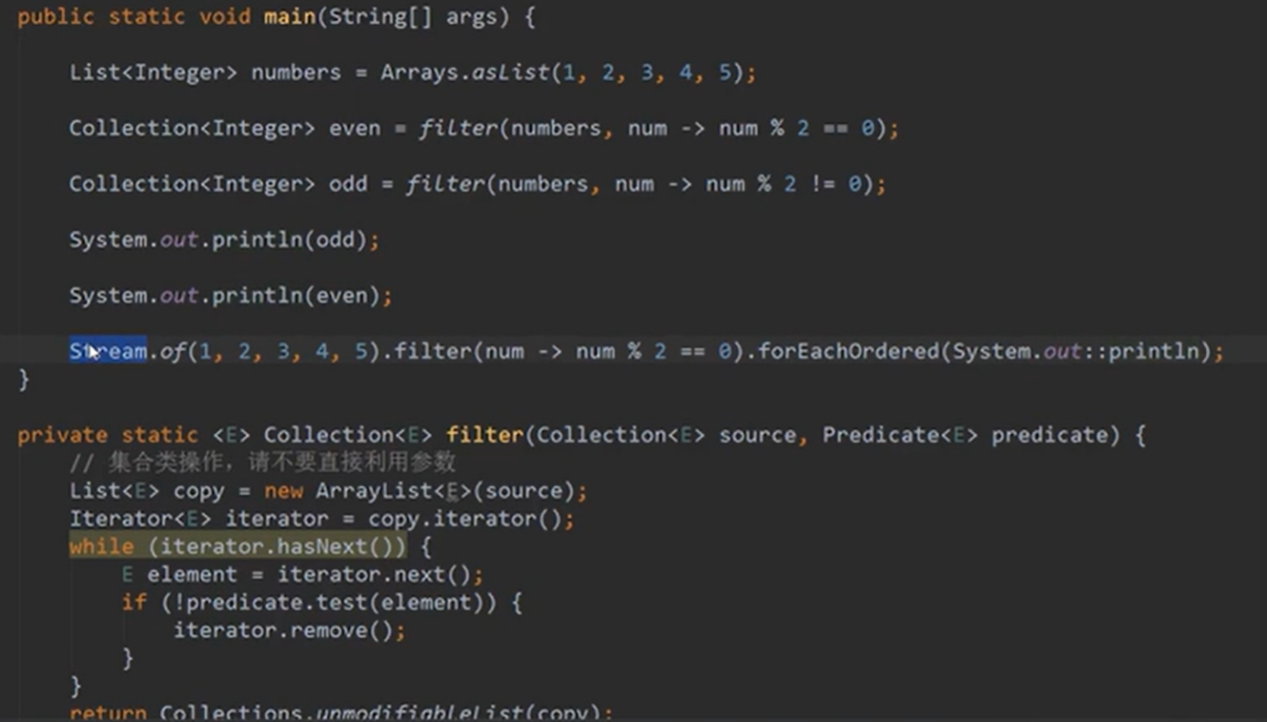




## Predict 理念

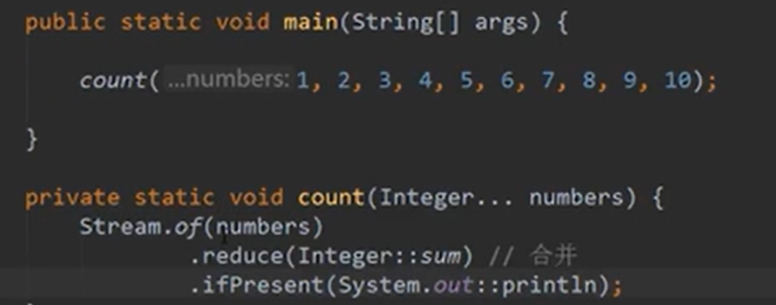






# Stream API

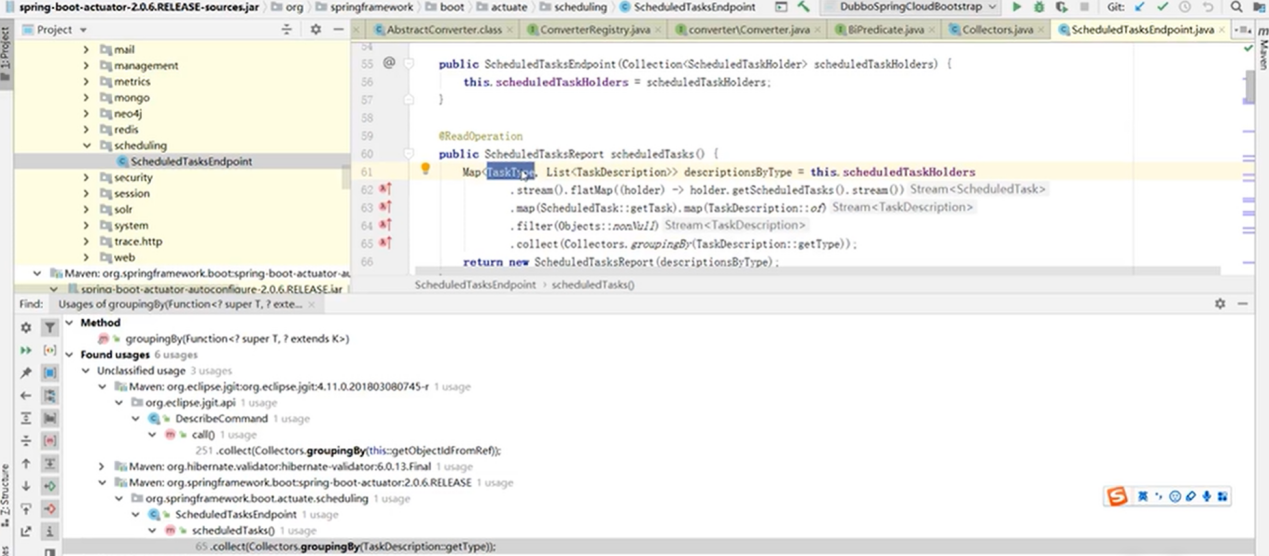








**分组操作**



**聚合操作**

Max

Having

扩展知识：

注解

