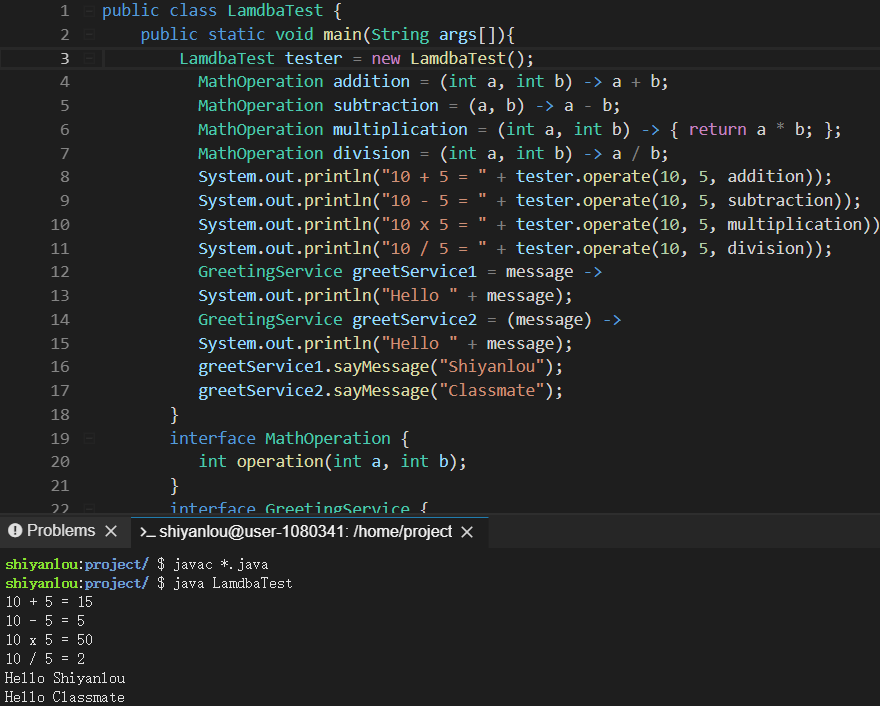
2019-05-28学习心得

1. 函数式编程

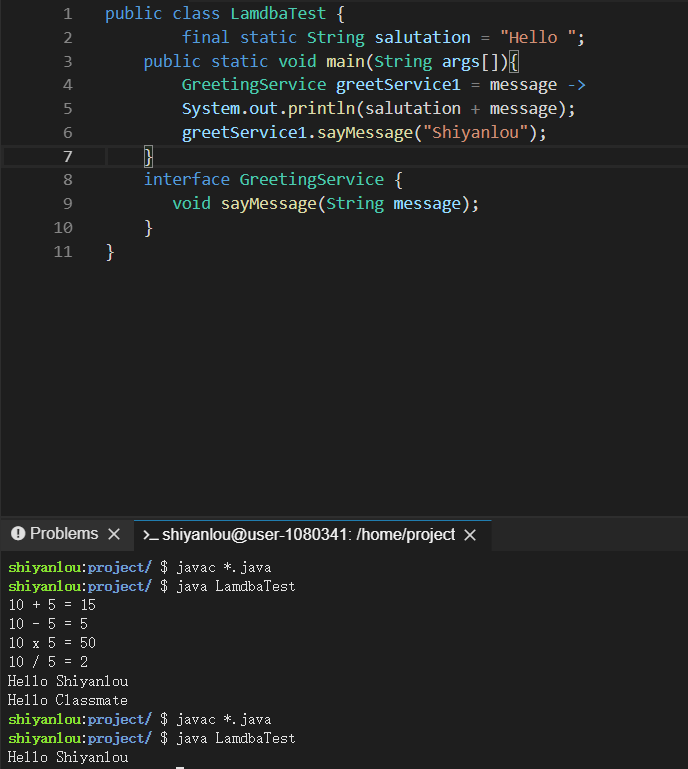
它将计算机运算视为数学上的函数计算，并且避免使用程序状态以及易变对象。函数编程语言最重要的基础是λ演算。而且λ演算的函数可以接受函数当作输入（引数）和输出（传出值）

1. Lambda表达式

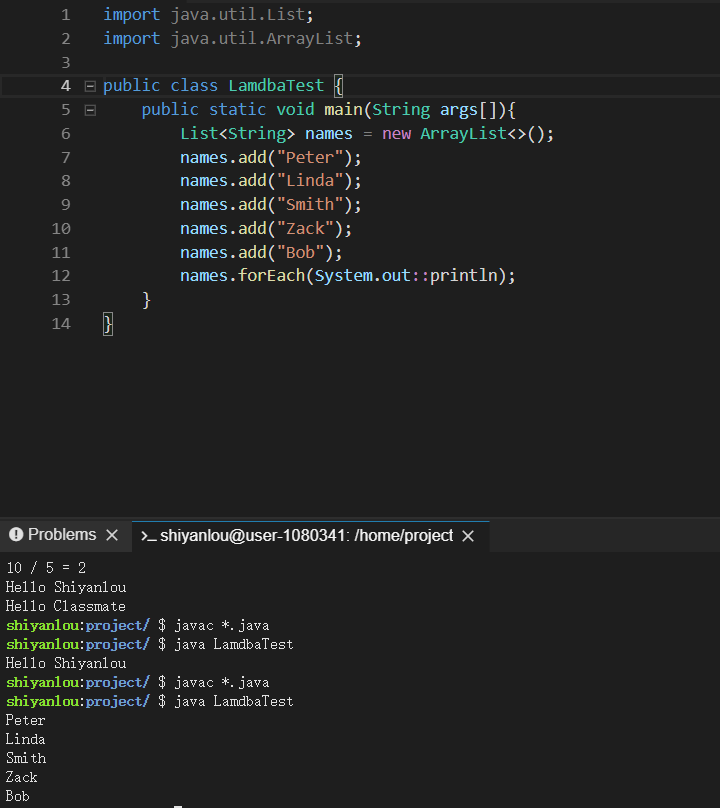
Parameter->expression body



1. 作用域



1. 方法引用



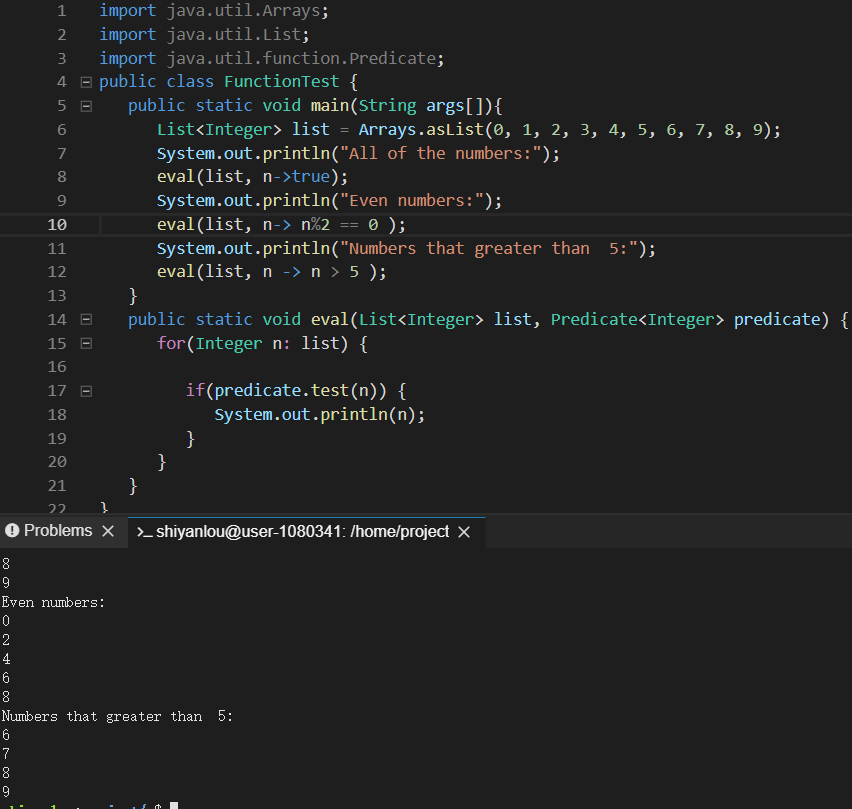
1. 函数式接口

函数式接口是只包含一个方法的接口

java.util.function 包中包含了大量的函数式接口，基本可以满足我们的日常开发需求



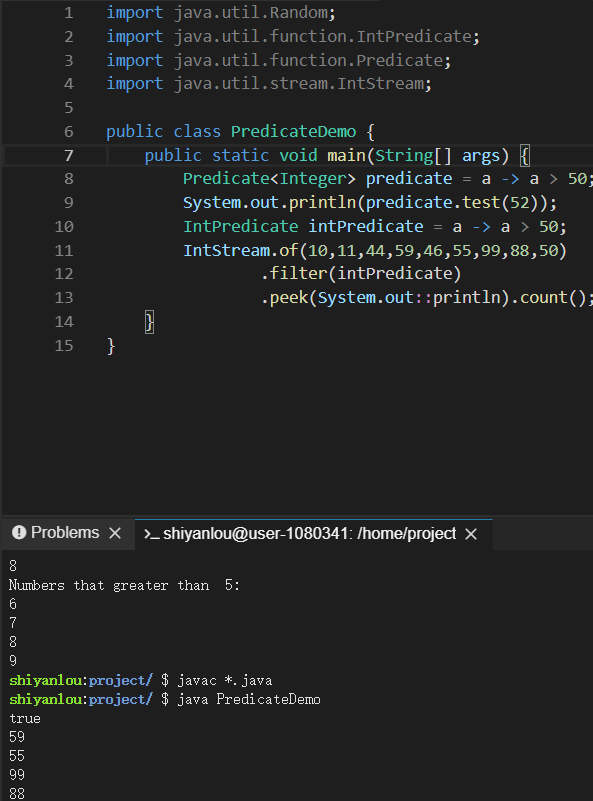




1. Predicate

Predicate 是 Java 提供的一个函数式接口，他接受一个参数 t，并执行断言操作返回一个 Boolean

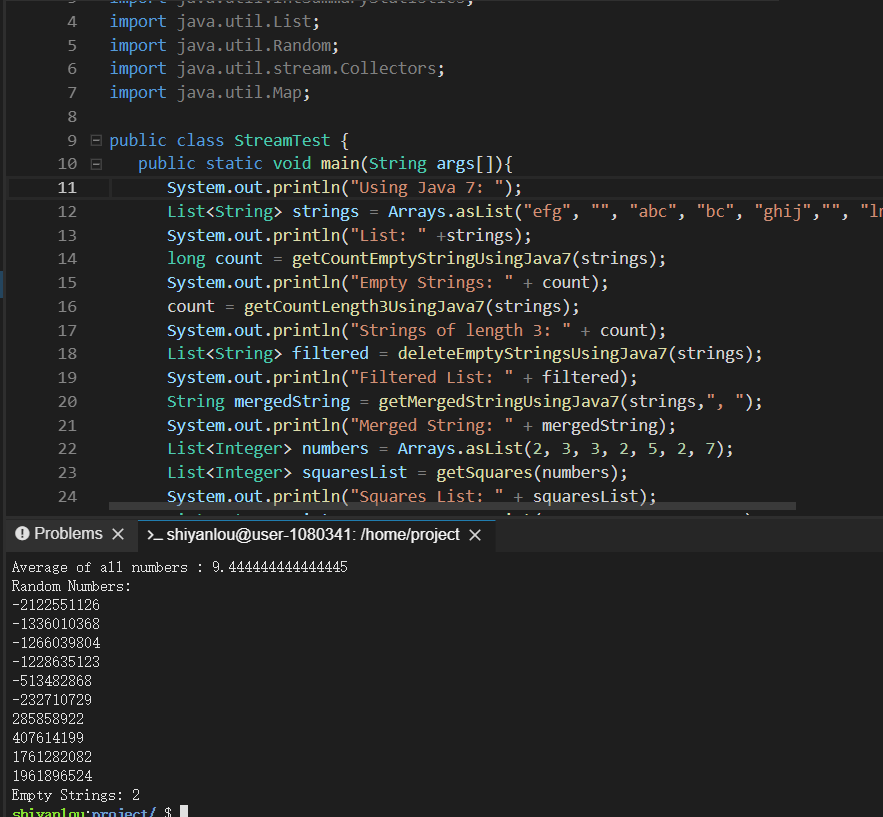




1. Stream流

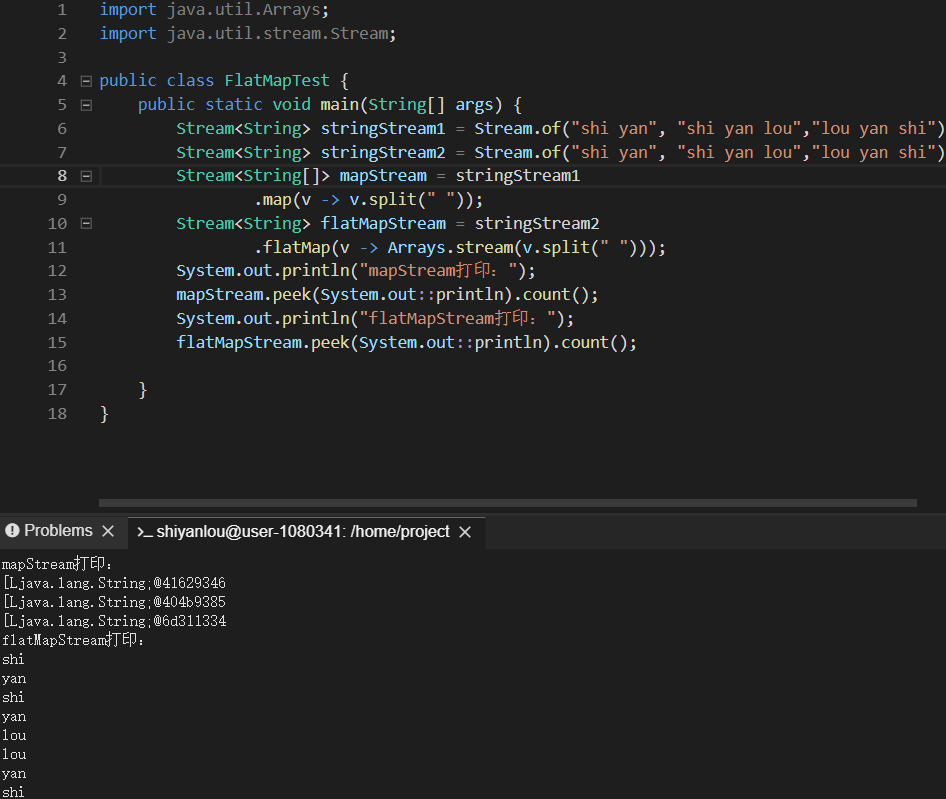
Stream 是 Java 8 开始的一个新的抽象层。通过使用 Stream，你能以类似于 SQL 语句的声明式方式处理数据

1. stream()：该方法返回一个将集合视为源的连续流
2. parallelStream()：该方法返回一个将集合视为源的并行流
3. forEach：该方法用于对 Stream 中的每个元素进行迭代操作
4. map：该方法用于将每个元素映射到对应的结果上
5. filter：该方法用于过滤满足条件的元素
6. limit：该方法用于减少 Stream 的大小
7. sorted：该方法用于对 Stream 排序



1. FlatMap

FlatMap 用于将多个流合并为一个流，使用 FlatMap 时，表达式的返回值必须是 Stream 类型。而 Map 用于将一种流转化为另外一个流



明日计划IO操作