

Function.identity()

0.348 2018.11.04 12:46:44 字数 390 阅读 7780

Function.identity()是什么？

```
1 // 将Stream转换成容器或Map
2 Stream<String> stream = Stream.of("I", "love", "you", "too");
3 Map<String, Integer> map = stream.collect(Collectors.toMap(Function.identity(), String::length));
```

Function是一个接口，那么Function.identity()是什么意思呢？解释如下：

Java 8允许在接口中加入具体方法。接口中的具体方法有两种，default方法和static方法，identity()就是Function接口中的一个静态方法。
Function.identity()返回一个输出跟输入一样的Lambda表达式对象，等价于形如 `t -> t` 形式的Lambda表达式。

identity() 方法JDK源码如下：

```
1 static Function identity() {
2     return t -> t;
3 }
```

写下你的评论...

评论0 赞5

下面的代码中，`Task::getTitle` 需要一个task并产生一个仅有一个标题的key。`task -> task` 是一个用来返回自己的lambda表达式，上例中返回一个task。

```
1 private static Map<String, Task> taskMap(List<Task> tasks) {
2     return tasks.stream().collect(toMap(Task::getTitle, task -> task));
3 }
```

可以使用 `Function` 接口中的默认方法 `identity` 来让上面的代码代码变得更简洁明了、传递开发者意图时更加直接，下面是采用 `identity` 函数的代码。

```
1 import static java.util.function.Function.identity;
2
3 private static Map<String, Task> taskMap(List<Task> tasks) {
4     return tasks.stream().collect(toMap(Task::getTitle, identity()));
5 }
```

Function.identity() or t->t?

```
1 Arrays.asList("a", "b", "c")
2     .stream()
3     .map(Function.identity()) // <- This,
4     .map(str -> str)         // <- is the same as this.
5     .collect(Collectors.toMap(
6         Function.identity(), // <- And this,
7         str -> str));        // <- is the same as this.
```

上面的代码中，为什么要使用 `Function.identity()` 代替 `str->str` 呢？它们有什么区别呢？

在上面的代码中 `str -> str` 和 `Function.identity()` 是没什么区别的因为它们都是 `t->t`。但是我们有时候不能使用 `Function.identity`，看下面的例子：

```
1 List list = new ArrayList<>();
2 list.add(1);
3 list.add(2);
```

下面这段代码可以运行成功：

```
1 int[] arrayOK = list.stream().mapToInt(i -> i).toArray();
```

但是如果你像下面这样写：

```
1 int[] arrayProblem = list.stream().mapToInt(Function.identity()).toArray();
```

运行的时候就会错误，因为 `mapToInt` 要求的参数是 `ToIntFunction` 类型，但是 `ToIntFunction` 类型和 `Function` 没有关系

5人点赞

Java8新特性

不知名的蛋挞
拥有8帖 (约0.90元)

关注

“小礼物走一走，来简书关注我”

赞赏

写下你的评论...



不知名的蛋挞
拥有8帖 (约0.90元)

关注



resultMap使用总结
阅读 19



final修饰的变量
阅读 0



spring中过滤器 (filter)、拦截器 (interceptor) 和切面 (aop) 的执...
阅读 14

精彩继续

2019年，同性恋终于
起异性恋

阅读 6743

深圳纪事 (8)：离职华为
阅读 38844

推荐阅读

更多精彩内容>

Java面试宝典Beta5.0

pdf下载地址: Java面试宝典 第一章内容介绍 20 第二章JavaSE基础 21 一、Java面向对象 21 ...



百战程序员V1.2——尚学堂旗下高端培训_ Java1573题

百战程序员_ Java1573题 QQ群: 561832648489034603 掌握80%年薪20万掌握50%年薪...



拆书作业0105+赞小碧+学以致用，重读读书笔记是关键！

本文是雷梅读书群拆书活动，第一期第五次作业。拆解的是日本作家奥野宣之《如何有效阅读—本书+超实用笔记读书法》，第四...



甲醛检测治理热线18905340112

甲醛



2018-01-12

如何找寻内心的平静 自从放弃机会之后，我并未收获内心的平静，反而变得更加思绪复杂。复杂在于几点：第一、放弃机会意...

