

22 方法引用

- 定义：

把已有的方法拿过来用，当做函数式接口中抽象方法的方法体

- 要求：

- 1. 引用处必须是函数式接口。
- 2. 被引用的方法必须已经存在。
- 3. 被引用方法的形参和返回值需要跟抽象方法保持一致。
- 4. 被引用方法的功能需要满足当前的要求。

```
//表示引用FunctionDemo1类里面的subtraction方法
//把这个方法当做抽象方法的方法体
Arrays.sort(arr, FunctionDemo1::subtraction);

//可以是Java已经写好的，也可以是一些第三方的工具类
public static int subtraction(int num1, int num2) {
    return num2 - num1;
}
```

- 分类：

① 引用静态方法

I. 引用其他类的成员方法

② 引用成员方法

II. 引用本类的成员方法

③ 引用构造方法

III. 引用父类的成员方法

④ 其他调用方式

I. 使用类名引用成员方法

II. 引用数组的构造方法

- 1. 引用静态方法

格式：类名::静态方法

范例：Integer.parseInt

- 2. 引用成员方法

格式：对象::成员方法

① **其他类：**其他类对象::方法名

② **本类：**this::方法名

③ **父类：**super::方法名

引用处不能是静态方法

- 如果本类是静态的，则必须创建本类的对象，才能引用。因为静态方法(main方法)中是没有this和super关键字的。

• 3.引用构造方法

格式：类名::new

范例：Student::new

• 4.使用类名引用成员方法

格式：类名::成员方法

范例：String::substring

• 特殊规则

- 1.需要有函数式接口
- 2.被引用的方法必须已经存在
- 3.被引用方法的形参要和抽象方法的第二个形参到最后一个形参保持一致，返回值需要保持一致。
- 4.被引用方法的功能需要满足当前的要求。

• 抽象方法形参的详解：

- **第一个参数：**表示被引用方法的调用者，决定了可以引用哪些类中的方法（**不是所有类都可以引用**）。在Stream流当中，第一个参数一般都表示流里面的每一个数据。假设流里面的数据是字符串，那么使用这种方式进行方法引用，只能引用string这个类中的方法
- **第二个参数到最后一个参数：**跟被引用方法的形参保持一致，如果没有第二个参数，说明被引用的方法需要是**无参**的成员方法。
- **局限性：**不能引用所有类中的成员方法。是和抽象方法的第一个参数有关，这个参数是什么类型的，那么就只能引用这个类中的方法。
- **用处：**把流里面的数据变成其他类的对象

• 5.引用数组的构造方法

格式：数据类型[]::new

范例：int[]::new

- 细节：数组的类型要和流中数据的类型保持一致。
- 用处：把流里面的数据放到数组里面

• 总结：

1. 什么是方法引用?

把已经存在的方法拿过来用, 当做函数式接口中抽象方法的方法体

2. :: 是什么符号?

方法引用符

3. 方法引用时要注意什么?

- 需要有函数式接口
- 被引用方法必须已经存在
- 被引用方法的形参和返回值需要跟抽象方法保持一致
- 被引用方法的功能要满足当前的需求

1. 引用静态方法

类名::静态方法

2. 引用成员方法

对象::成员方法

this::成员方法

super::成员方法

3. 引用构造方法

类名::new

4. 使用类名引用成员方法

类名::成员方法

不能引用所有类中的成员方法
如果抽象方法的第一个参数是A类型的
只能引用A类中的方法

5. 引用数组的构造方法

数据类型[]::new

● 技巧 (怎么用方法引用) :

- 1. 现在有没有一个方法符合我当前的需求
- 2. 如果有这样的方法, 这个方法是否满足引用的规则
- 3. 是静态 /成员(类名::方法名/ 对象名::方法名) /构造