



链滴

java 中 list 的 sort() 功能如何使用?

作者: [ruige](#)

原文链接: <https://ld246.com/article/1604644791136>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)



java中list的sort()功能如何使用?

排序时正序/倒序处理起来可能会混淆，可以用更简单的方法。

可以使用java.util自带的比较器来做

`Comparator.comparingInt(Integer::intValue).reversed()` // 倒序

`Comparator.comparingInt(Integer::intValue)` // 正序

示例

```
public class Main {
    public static void main(String[] args) {
        List<Integer> list = Lists.newArrayList(1, 2, 3);
        list.sort(Comparator.comparingInt(Integer::intValue).reversed()); // 倒序
        print(list);
        list.sort(Comparator.comparingInt(Integer::intValue)); // 默认正序
        print(list);
        List<Integer> list2 = Lists.newArrayList(1, 2, 3);
        list2 = list2.stream().sorted(Comparator.comparingInt(Integer::intValue).reversed())
            .collect(Collectors.toList()); // 倒序
        print(list2);
        list2 = list2.stream().sorted(Comparator.comparingInt(Integer::intValue))
            .collect(Collectors.toList()); // 默认正序
        print(list2);
    }
    /**
     * 输出
     */
    private static <T> void print(List<T> list) {
        list.forEach(e -> System.out.print(e + " "));
        System.out.println();
    }
}
```

```
}  
}
```

执行输出为:

```
3,2,1,  
1,2,3,  
3,2,1,  
1,2,3,
```

对于复杂对象, 比如User,想通过年龄来比较

```
public class User {  
    public String name; // 姓名  
    public int age; // 年龄  
    public int getAge() {  
        return age;  
    }  
}
```

```
Comparator.comparingInt(User::getAge).reversed() // 倒序  
Comparator.comparingInt(User::getAge) // 正序
```