

Comparator + ??? ? ??? ??

1. ?? ??

```
Arrays.sort(arr, (a, b) -> {  
    return a - b;  
});
```

- Arrays.sort(arr, Comparator) 使用 Comparator 接口实现排序。
- Comparator 接口实现 Integer 类型的排序，int[] → Integer[] 转换。

1.1 Comparator 比较规则

返回值	含义	排序结果
< 0	a < b	a 排在 b 前面
0	a == b	a 和 b 顺序不变
> 0	a > b	b 排在 a 前面

2. 自定义 Comparator

2.1 升序排序 (a - b)

```
Arrays.sort(arr, (a, b) -> a - b);
```

2.2 降序排序 (b - a)

```
Arrays.sort(arr, (a, b) -> b - a);
```

2.3 ??? ?? ????? ??

```
Arrays.sort(arr, (a, b) -> Math.abs(a) - Math.abs(b));
```

2.4 ??? ?? + ?? ??? ? ? ??

```
Arrays.sort(arr, (a, b) -> {  
    int diff = Math.abs(a) - Math.abs(b);  
    if (diff == 0) return b - a; // ??? ??? ? ?  
    return diff;  
});
```

3. ??

??	??
?? ???	(a, b) -> a - b
?? ???	(a, b) -> b - a
?? ?	(a, b) -> Math.abs(a) - Math.abs(b)
?? + ? ? ? ? ?	(a, b) -> { ... if ??? }

4. List ??

4.1 ?? ????? ??

```
List<Integer> list = Arrays.asList(3, 1, 5, 2);  
Collections.sort(list); // ?? ???
```

4.2 ????? ?? (??? ??)

```
list.sort((a, b) -> b - a); // ?? Collections.sort(list, (a, b) -> b - a);
```

4.3 ??? ?? ??

```
list.sort((a, b) -> Math.abs(a) - Math.abs(b));
```

5. ?? ??

5.1 ?? 1: Comparable ????? (?? ?? ?? ??)

?? ???

```
class Person implements Comparable<Person> {
    String name;
    int age;

    public Person(String name, int age) {
        this.name = name;
        this.age = age;
    }

    @Override
    public int compareTo(Person other) {
        return this.age - other.age; // ?? ???? ??
    }
}
```

?? ??

```
List<Person> list = new ArrayList<>();
list.add(new Person("Alice", 25));
list.add(new Person("Bob", 20));
Collections.sort(list); // compareTo() ???? ??
```

5.2 ?? 2: Comparator ????? (?? ?? ??)

?? 1: ?? ?? ?? (????)

```
list.sort(Comparator.comparing(p -> p.name));
```

?? 2: ?? ????? ??

```
list.sort((p1, p2) -> p2.age - p1.age);
```

6. ?? ??

??	????	?? ??	?? ?? ??
Comparable	compareTo()	??? ?? ??	?? (?? ??)
Comparator	compare()	???? ??	??? (??? ?? ??)

- Comparable: "?? ?? ??? ?? ?? !" (??? ??? ??)
- Comparator: "?? ??? ??? ??? !" (??? ?? ???)

Revision #2
Created 12 June 2025 14:33:31 by Dain
Updated 12 June 2025 14:37:30 by Dain