→ In-built functions

1) Arrays.sort(arr)

// sort in ↑ing order by default

2) Arrays.sort(arr, Collections.reverseOrder());

// sort in ↓ing order by default.

TC. → $O(n \log n)$ ← merge, quick

Primitive data types.

object classes

int → Integer

boolean → Boolean

float → Float

char → character

double → Double

→ Custom Sort (alters the properties of in-built fun)

	0	1	2	3	4
arr =	-5	3	-2	8	0
	(25)	(9)	(4)	(64)	(0)

0	-2	3	-5	8

it will not get affected.

Syntax

```
Arrays.sort(arr, new myComparator());
```

Implementation

```
public static class myComparator implements Comparator<Integer> {
    @override
    public int compare(Integer a, Integer b){
        return a-b;
    }
}
```

interface

return a - b; // arrange elements in ↑ing order
asc. order.

return b - a; // in ↓ing order. ▷ desc. order

a = myself

b = other

```java
        Scanner s = new Scanner(System.in);
        int n = s.nextInt();
        Integer arr[] = new Integer[n];
        for(int i = 0; i < n;i ++){
            arr[i] = s.nextInt();
        }

        Arrays.sort(arr, new myComparator());

        // print
        for(int i = 0; i <n; i++){
            System.out.print(arr[i] + " ");
        }
    }
    public static class myComparator implements Comparator<Integer>{
        @Override
        public int compare(Integer a, Integer b){
            return a*a - b*b;
        }
    }
```

5 2

a - b = 5 - 2 => 3

b - a = 2 - 5 => -3

→ Lambda function (Imp).

(alters the properties of in-built func.)

Arrays.sort(arr, (a,b) → {

return a-b;

});

```
        Scanner s = new Scanner(System.in);
        int n = s.nextInt();
        Integer arr[] = new Integer[n];
        for(int i = 0; i < n;i ++){
            arr[i] = s.nextInt();
        }

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

        // print
        for(int i = 0; i <n; i++){
            System.out.print(arr[i] + " ");
        }
    }
```

Imp.

→ return a - b : ascending order

→ return b - a : descending order

→ return +1 : a value will be place later

→ return -1 : a value will be place first

Ex.

arr =

0	1	2	3	4	5
5	2	1	-2	8	-7

(a, b)

a = myself

b = other

→ b - a = [5, -7]
  -7 5

→ a - b = [-7, 5]
  5 7

→ -1 = [5, -7]

→ +1 = [-7, 5]