Min range → Integer.MIN_VALUE;

max range → Integer.MAX_VALUE;

Second largest

Largest → MIN-VALUE   arr[i] > largest   largest = 9

Second largest → MIN-VALUE   largest = arr[i]

3.
0 1 2 3

9 6 9 7

arr[i] > SL && arr[i] != largest

arr[0] > MIN && arr[0] != 9

9 > min && 9 != 9
T   F
     F

6 > min && 6 != 9
   T   T

7 > 6 && 7 != 9
   T
     T

SL = 9 (F)

HW_Second Largest in array 2

$n = 4$

	0	1	2	3
	9	6	9	7

⑥ 6 6

$4 \leq 1 F$

Largest = min

$SL = min$

$i = 0 < 4$

arr[0] ; $9 > min$

$SL = min$

$L = 9$

$i = 1 < 4$ T

arr[1]

$6 > 9 F$

```java
public static int findsecondLargest(int arr[]){
    int n = arr.length;

    if(n <= 1){
        return Integer.MIN_VALUE;
    }

    int largest = Integer.MIN_VALUE;
    int secondLargest = Integer.MIN_VALUE;

    for(int i =0; i <n;i++){
        if(arr[i] > largest){
            secondLargest = largest;
            largest = arr[i];
        }
        else if(arr[i] > secondLargest && arr[i] != largest){
            secondLargest = arr[i];
        }
    }
    // if all the values are same ie. no SL found
    if(secondLargest == Integer.MIN_VALUE){
        return Integer.MIN_VALUE;
    }
    return secondLargest;
}
```

elsif

arr[i] > min && arr[i] != 9

$6 > min$ && $6 != 9$

T   T

SL = 6 (F) — off

$i = 2 < 4$ T

$9 > 9 F$

$9 > 6$ && $9 != 9$

T   F Ⓔ

$i = 3 < 4$ T

$7 > 9 F$

$7 > 6$ && $7 != 9$

T   T