Largest Element in an Array :

→ arr = 10, 50, 70, 90, 30

→ Output = 90

① Brute force approach :

(i) Sort the Array

(ii) And return the last element.

```
int[] arr = {10,50,70,90,30}
Arrays.sort(arr);
int largest = arr[arr.length-1];
```

T.C ↓ O(n log n)

Dry Run :→

arr = 10, 50, 70, 90, 30

Arrays.sort(arr) ⇒ 10, 30, 50, 70, 90

largest = arr[5-1];

= arr[4]

↓

90

② Optimized Approach:

↳ (i) Assume the first element of array as largest.

```
int[] arr = {10,50,70,90,30}
int largest = arr[0];
```

(ii) Traverse through all the elements of array using for loop and check if we get element which is greater than first.

(iii) If we get the element which is greater than first, we will return that as largest otherwise, we will return first element as largest.

```
for (int i=0; i<arr.length; i++)
{
  if (arr[i] > largest)
  {
    largest = arr[i];
  }
}
Sop("Largest element: " + largest)
```

Dry Run:

(i) 10, 50, 70, 90, 30

largest = 10

Compare → arr[i] > largest

(i) 10 > 10 → no change, largest = 10

(ii) 50 > 10 → updated, largest = 50

(iii) 70 > 50 → updated, largest = 70

(iv) 90 > 70 → updated, largest = 90

(v) 30 > 90 → X no change

Largest = 90