JAVA HANDS ON PRACTICE

6) Second Largest element In An Array.

I/P
Array

O/P
Int second largest element

```
 5  9  6  3  1  2
```
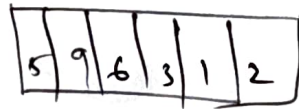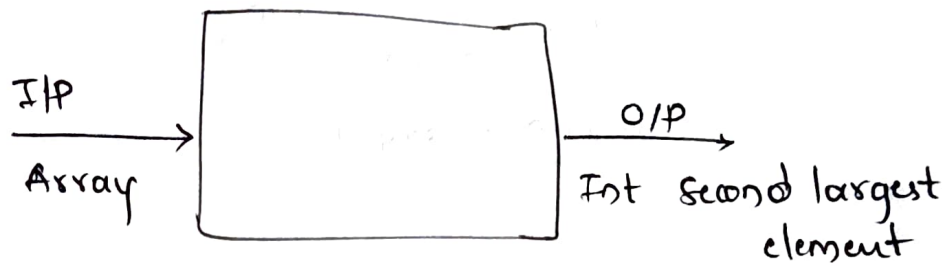
1) Method 1 :- Sort the Array and return second largest element

2) Method 2 :- Max Store values

Second Max according

return secondMax

```java
public class SecondLargest {
    public static void main(String[] args) {
        int[] arr = {12, 35, 1, 10, 34, 1};

        if (arr.length < 2) {
            System.out.println(x:"Array must have at least two elements.");
            return;
        }

        int first = Integer.MIN_VALUE;
        int second = Integer.MIN_VALUE;

        for (int num : arr) {
            if (num > first) {
                second = first;    // update second
                first = num;       // update first
            } else if (num > second && num != first) {
                second = num;      // update second only
            }
        }

        if (second == Integer.MIN_VALUE) {
            System.out.println(x:"No second largest element (all elements are same).");
        } else {
            System.out.println("The second largest element is: " + second);
        }
    }
}
```
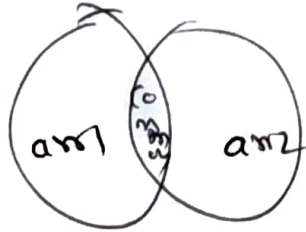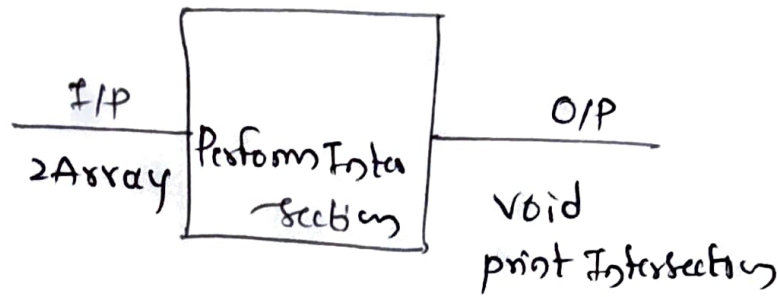
6) functions to print unique element In An Integer Array.
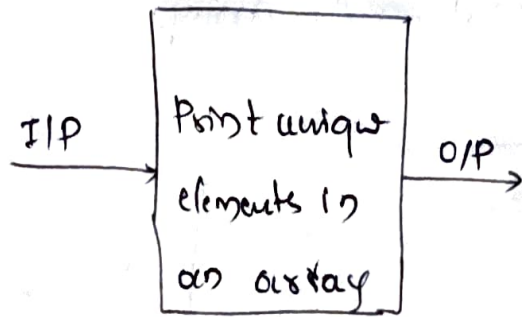
C: > java programs > **J** UniqueElements.java > ...

```java
public class UniqueElements {
    Run main | Debug main | Run | Debug
    public static void main(String[] args) {
        int[] arr = {1, 2, 2, 3, 4, 4, 5};

        System.out.print(s:"Unique elements: ");
        for (int i = 0; i < arr.length; i++) {
            boolean isUnique = true;

            for (int j = 0; j < arr.length; j++) {
                if (i != j && arr[i] == arr[j]) {
                    isUnique = false;
                    break;
                }
            }

            if (isUnique) {
                System.out.print(arr[i] + " ");
            }
        }
    }
}
```

Q> Print Common elements In the two Array Intersection

```
        ┌─────────────┐
  I/P   │ Perform Inta│   O/P
────────│   -section  │────────
 2Array │             │   void
        └─────────────┘   print Intersection
```


Taking Intersection Helps to find value

C: > java programs > J ArrayIntersection.java > ...

```java
public class ArrayIntersection {
    Run main | Debug main | Run | Debug
    public static void main(String[] args) {
        int[] arr1 = {1, 2, 3, 4, 5};
        int[] arr2 = {3, 4, 5, 6, 7};

        System.out.print(s:"Common elements: ");
        for (int i = 0; i < arr1.length; i++) {
            for (int j = 0; j < arr2.length; j++) {
                if (arr1[i] == arr2[j]) {
                    System.out.print(arr1[i] + " ");
                    break; // prevent duplicate printing
                }
            }
        }
    }
}
```