Java Array Interview Questions (Most Asked)

1. ## Combine Two Array
   - *Input:* {1, 2, 3}, {4, 5, 6}
   - *Output:* [1, 2, 3, 4, 5, 6]

```java
package Array;
import java.util.Arrays;
public class CombinedOfTwoArrays {
    Run | Debug | Run main | Debug main
    public static void main(String[] args) {
        int[] arr1= {1, 2, 3};
        int[] arr2= {4, 5, 6};
        combinedArrays(arr1, arr2);
    }
    public static void combinedArrays(int[] arr1, int[] arr2) {
        int length =arr1.length+arr2.length;
        int[] newArr= new int[length];
        for(int i=0; i<arr1.length; i++) {
            newArr[i]=arr1[i];
        }
        for(int i=0; i<arr1.length; i++) {
            newArr[arr1.length+i]=arr2[i];
        }
        System.out.println("New Combined array is: "+Arrays.toString(newArr));
    }

}
```

2. ## Find Min and Max in Array
   - *Input:* {3, 6, 9, 5, 20, 43}
   - *Output:* Min: 3, Max: 43

```java
package Array;
public class FindMinAndMaxInGivenArray {
    Run | Debug | Run main | Debug main
    public static void main(String[] args) {
        int arr[]= {3, 6, 9, 5, 20, 43};
        minAndMax(arr);
    }
    public static void minAndMax(int[] arr) {
        int min=Integer.MAX_VALUE;
        int max=Integer.MIN_VALUE;
        for(int i=0; i<arr.length; i++) {
            if(arr[i]<min) {
                min=arr[i];
            }
            if(arr[i]>max) {
                max=arr[i];
            }
        }
        System.out.println("Minimum number in the given array is: "+min);
        System.out.println("Maximum number in the given array is: "+max);
    }

}
```

3. ## Find Pair with Target Sum

- *Input:* {1, 2, 3, 4, 5}, Target = 5
- *Output:* 3+2, 4+1

```java
package Array;
import java.util.ArrayList;
import java.util.List;
public class FindTargetSum {
    Run | Debug | Run main | Debug main
    public static void main(String[] args) {
        int[] input= {1, 2, 3, 4, 5};
        int target=5;
        findTarget(input, target);
    }
    public static void findTarget(int[] input, int target) {
        List<Integer> seen=new ArrayList<>();
        for(int i=0; i<input.length; i++) {
            int value=target-input[i];
            if(seen.contains(value)) {
                System.out.println("Given Target: "+ input[i] + " "+ value);
            }
            seen.add(input[i]);
        }
    }
}
```

4. ## Frequency of Each Element

- *Input:* {1, 2, 2, 3, 1, 4, 5, 1}
- *Output:* 1 → 3 times, etc

```java
package Array;
public class FrequencyOfElement {
    Run | Debug | Run main | Debug main
    public static void main(String[] args) {
        int arr[]=  {1, 2, 2, 3, 1, 4, 5, 1};
        countOfEachElement(arr);
    }
    public static void countOfEachElement(int[] arr) {
        for(int i=0; i<arr.length; i++) {
            boolean alreadyPresent=false;
            for(int j=0; j<i; j++) {
                if(arr[i]==arr[j]) {
                    alreadyPresent=true; //useful to skip same element in next iteration
                    break;
                }
            }
            if(alreadyPresent) {
                continue; //move to next element
            }
            int count=1;
            for(int j=i+1; j<arr.length; j++) {
                if(arr[i]==arr[j]) {
                    count++;
                }
            }
            System.out.println("Occurence of "+arr[i]+ " is: "+ count);
        }
    }

}
```

5. Kth Largest Element

- *Input: {2, 5, 7, 8, 9, 6}, K = 3*
- *Output: 7*

```java
package Array;
public class KthLargestDigit {
    Run | Debug | Run main | Debug main
    public static void main(String[] args) {
        int arr[]= {2, 5, 7, 8, 9, 6};
        int k=3;
        LargestDigit(arr, k);
    }
    public static void LargestDigit(int[] arr, int k) {
        int result=0;
        for(int i=0; i<arr.length; i++) {
            for(int j=i+1; j<arr.length; j++) {
                if(arr[i]>arr[j]) {
                    int temp=arr[j];
                    arr[j]=arr[i];
                    arr[i]=temp;
                }
            }
        }
        result=arr[arr.length-k];
        System.out.println("Kth Largest digit is: "+result);
    }

}
```

6. Remove Duplicates

- *Input: {4, 2, 3, 2, 4, 1, 5, 3, 5}*
- *Output: 1 2 3 4 5*

```java
package String;
import java.util.HashSet;
import java.util.Set;
public class RemoveDuplicates {
    Run | Debug | Run main | Debug main
    public static void main(String[] args) {
        removeDuplicateChar(input:"My Name is Vikash");
    }
    public static void removeDuplicateChar(String input) {
        StringBuilder result =new StringBuilder();
        Set<Character> seen= new HashSet<>();
        for(int i=0; i<input.length(); i++) {
            char currentChar=input.charAt(i);
            if(!seen.contains(currentChar)) {
                seen.add(currentChar);
                result.append(currentChar);
            }
        }
        System.out.println("After removing duplicates: "+result.toString());
    }
}
```

7. Find Second Largest Element

- *Input: {4, 19, 3, 4, 16}*
- *Output: 16*

```java
package Array;
public class secondLargestElement {
    Run | Debug | Run main | Debug main
    public static void main(String[] args) {
        int input[]= {4, 19, 3, 4, 16};
        System.out.println("Second largest digit is: "+secondLargestDigit(input));
    }
    public static int secondLargestDigit(int[] input) {
        int largest=0;
        int secondLargest=0;
        for(int i=0; i<input.length; i++) {
            if(input[i]>largest) {
                secondLargest=largest;
                largest=input[i];
            }
            else if(input[i]>secondLargest && input[i]<largest) {
                secondLargest=input[i];
            }
        }
        return secondLargest;
    }
}
```

8. Shift a Value to End

- *Input: {1, 4, 5, 4, 3, 0, 3, 2, 0, 1}, Value = 1*
- *Output: [4, 5, 4, 3, 0, 3, 2, 0, 1, 1]*

```java
package Array;
import java.util.Arrays;

public class shiftValueToLast {

    Run | Debug | Run main | Debug main
    public static void main(String[] args) {
        int[] input = { 1, 4, 5, 4, 3, 0, 3, 2, 0, 1 };
        System.out.println("After shifting values: "+shiftToLast(input, shiftValue:1));

    }
    public static String shiftToLast(int[] input, int shiftValue) {
        for (int i = 0; i < input.length; i++) {
            for (int j = i + 1; j < input.length; j++) {
                if (input[i] == shiftValue) {
                    int temp = input[i];
                    input[i]=input[j];
                    input[j]=temp;
                }
            }
        }
        return Arrays.toString(input);
    }

}
```