Below are **all 45 Java practical questions** written in a **very simple way**, exactly in this format:

---

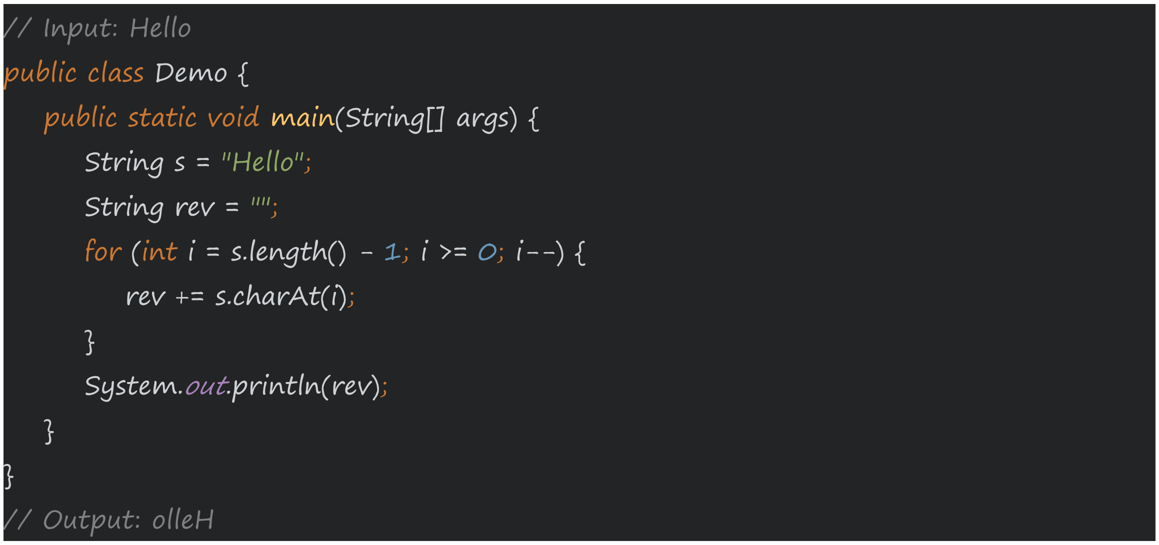
1. Reverse a String

Steps

Take a string

Loop from last index to first

Append characters to a new string



```
// Input: Hello
public class Demo {
    public static void main(String[] args) {
        String s = "Hello";
        String rev = "";
        for (int i = s.length() - 1; i >= 0; i--) {
            rev += s.charAt(i);
        }
        System.out.println(rev);
    }
}
// Output: olleH
```

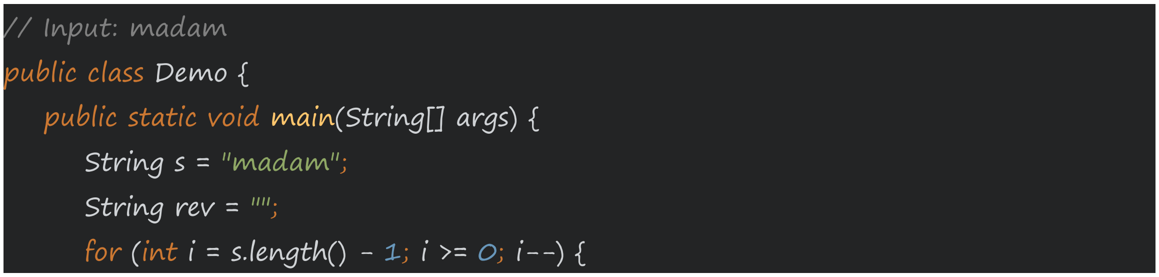
---

2. Check for Palindrome

Steps:

Reverse the string

Compare original and reversed string



```
// Input: madam
public class Demo {
    public static void main(String[] args) {
        String s = "madam";
        String rev = "";
        for (int i = s.length() - 1; i >= 0; i--) {
            rev += s.charAt(i);
        }
        System.out.println(s.equals(rev));
    }
}
// Output: true
```

Output

```
Palindrome
```

---

3. Fibonacci Series

In short:

Add previous two numbers to get next number.

Steps:

Initialize first two numbers

Loop and calculate next number

```
// Input: 5
public class Demo {
    public static void main(String[] args) {
        int a = 0, b = 1;
        for (int i = 1; i <= 5; i++) {
            System.out.print(a + " ");
            int c = a + b;
            a = b;
            b = c;
        }
    }
}
// Output: 0 1 1 2 3
```

---

4. Factorial Number

In short:

Multiply numbers from 1 to N.

```
// Input: 5
public class Demo {
    public static void main(String[] args) {
        int n = 5, fact = 1;
        for (int i = 1; i <= n; i++) {
            fact *= i;
        }
        System.out.println(fact);
    }
}
// Output: 120
```

5. Count Vowels and Consonants

In short:

Check each character.

```
// Input: hello
public class Demo {
    public static void main(String[] args) {
        String s = "hello";
        int v = 0, c = 0;
        for (char ch : s.toCharArray()) {
            if ("aeiou".indexOf(ch) != -1) v++;
            else c++;
        }
        System.out.println("Vowels=" + v + " Consonants=" + c);
    }
}
// Output: Vowels=2 Consonants=3
```

6. Sort an Array

In short:

Compare and swap elements.

```
// Input: [5,3,1,4]
public class Demo {
    public static void main(String[] args) {
        int[] arr = {5,3,1,4};
        for(int i=0;i<arr.length;i++){
            for(int j=i+1;j<arr.length;j++){
                if(arr[i]>arr[j]){
                    int t=arr[i]; arr[i]=arr[j]; arr[j]=t;
                }
            }
        }
        for(int n:arr) System.out.print(n+" ");
    }
}
// Output: 1 3 4 5
```

7. Merge Two Arrays

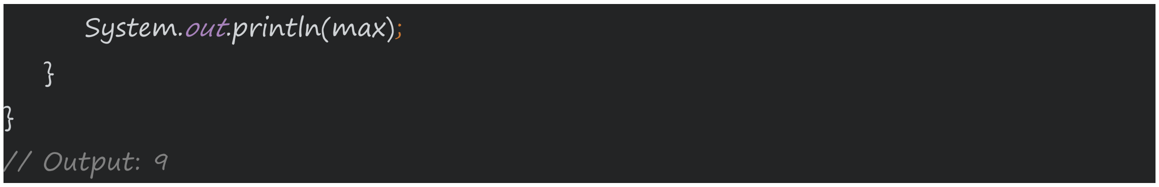
In short:

Copy elements of both arrays into a new array.

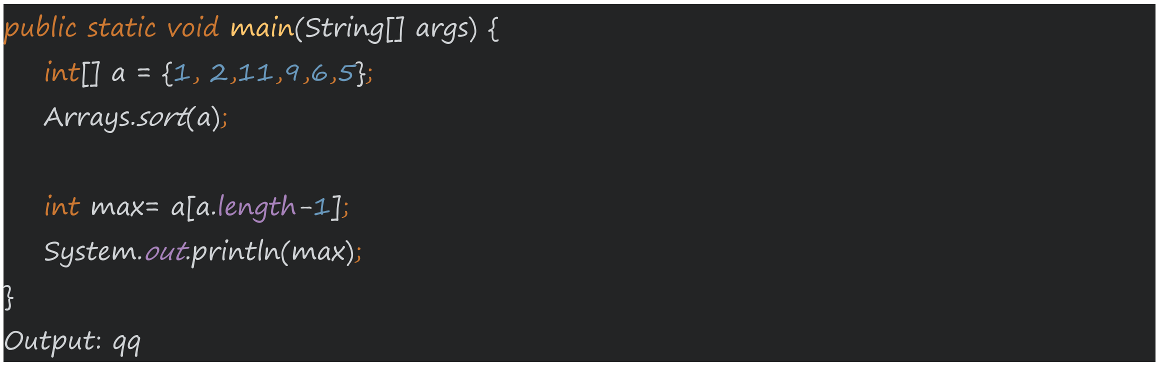
```
// Input: [1,2] [3,4]
public class Demo {
    public static void main(String[] args) {
        int[] a={1,2}, b={3,4};
        int[] c=new int[a.length+b.length];
        int k=0;
        for(int i:a) c[k++]=i;
        for(int i:b) c[k++]=i;
        for(int i:c) System.out.print(i+" ");
    }
}
// Output: 1 2 3 4
```

8. Find Largest Element

```
// Input: [2,9,4]
public class Demo {
    public static void main(String[] args) {
        int[] arr={2,9,4};
        int max=arr[0];
        for(int i:arr) if(i>max) max=i;
```



```
        System.out.println(max);
    }
}
// Output: 9
```



```
public static void main(String[] args) {
    int[] a = {1, 2,11,9,6,5};
    Arrays.sort(a);

    int max= a[a.length-1];
    System.out.println(max);
}
Output: qq
```

---

9. Remove Duplicates from Array

Remove duplicate elements from a sorted array

In short:

We move all non-equal (unique) elements to the beginning of the same array.

Steps:

1. Use an index variable to track position of unique elements

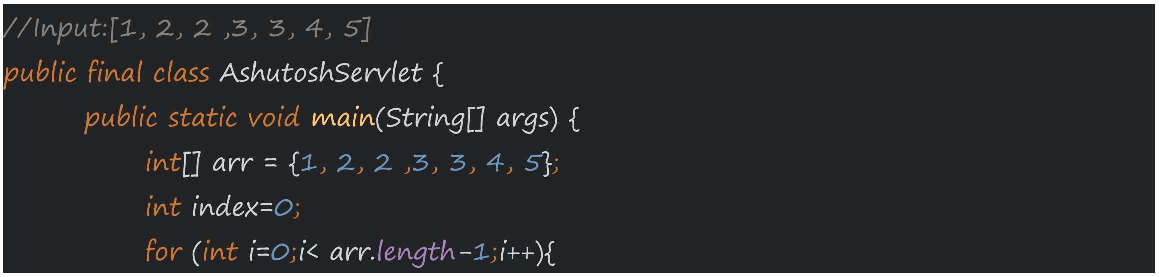
2. Loop through array till length - 1

3. Compare current element with next element

4. If both are different, store current element at index position

5. Increase index

6. After loop, add the last element (it is always unique)



```
//Input:[1, 2, 2 ,3, 3, 4, 5]
public final class AshutoshServlet {
        public static void main(String[] args) {
                int[] arr = {1, 2, 2 ,3, 3, 4, 5};
                int index=0;
                for (int i=0;i< arr.length-1;i++){
```

```
                if(arr[i]!=arr[i+1]){
                    arr[index++] = arr[i];
                 }
              }
              arr[index++] = arr[arr.length-1];
                for(int j=0;j<index;j++){
                  System.out.println(arr[j]);
                }
        }
}
// output: 1 2 3 4 5
```

10. Check Armstrong Number

```
// Input: 153
public class Demo {
    public static void main(String[] args) {
        int n=153,sum=0,temp=n;
        while(n>0){
            int d=n%10;
            sum+=d*d*d;
            n/=10;
        }
        System.out.println(sum==temp);
    }
}
// Output: true
```

11. Reverse a Number

```
// Input: 123
public class Demo {
    public static void main(String[] args) {
        int n=123, rev=0;
        while(n>0){
            rev=rev*10+n%10;
            n/=10;
        }
        System.out.println(rev);
    }
}
// Output: 321
```

12. GCD of Two Numbers

```
// Input: 12, 18
public class Demo {
    public static void main(String[] args) {
        int a=12,b=18;
        while(b!=0){
            int t=b;
            b=a%b;
            a=t;
        }
        System.out.println(a);
    }
}
// Output: 6
```

13. Prime Number Check

```
// Input: 7
public class Demo {
    public static void main(String[] args) {
        int n=7,count=0;
        for(int i=1;i<=n;i++){
            if(n%i==0) count++;
        }
        System.out.println(count==2);
    }
}
// Output: true
```

14. Check for Anagram

```
// Input: listen, silent
import java.util.Arrays;
public class Demo {
    public static void main(String[] args) {
        char[] a="listen".toCharArray();
        char[] b="silent".toCharArray();
        Arrays.sort(a);
        Arrays.sort(b);
        System.out.println(Arrays.equals(a,b));
    }
}
// Output: true
```

---

15. Count Digits

```
// Input: 12345
public class Demo {
    public static void main(String[] args) {
        int n=12345,count=0;
        while(n>0){ count++; n/=10; }
        System.out.println(count);
    }
}
// Output: 5
```

---

16. Prime Numbers in a Range

```
// Input: 1 to 10
public class Demo {
    public static void main(String[] args) {
        for(int i=2;i<=10;i++){
            int c=0;
            for(int j=1;j<=i;j++){
                if(i%j==0) c++;
            }
            if(c==2) System.out.print(i+" ");
        }
    }
}
// Output: 2 3 5 7
```

---

17. Second Largest Element

```
// Input: [3,5,1,4]
public class Demo {
    public static void main(String[] args) {
        int[] arr={3,5,1,4};
        int max=Integer.MIN_VALUE, sec=Integer.MIN_VALUE;
        for(int i:arr){
            if(i>max){ sec=max; max=i; }
            else if(i>sec && i!=max) sec=i;
        }
        System.out.println(sec);
    }
}
// Output: 4
```

---

18. Swap Two Numbers

```
// Input: a=3, b=5
public class Demo {
    public static void main(String[] args) {
        int a=3,b=5;
        a=a+b; b=a-b; a=a-b;
        System.out.println(a+" "+b);
    }
}
// Output: 5 3
```

---

19. Pascal’s Triangle

```
// Input: 4
public class Demo {
    public static void main(String[] args) {
        for(int i=0;i<4;i++){
            int num=1;
            for(int j=0;j<=i;j++){
                System.out.print(num+" ");
                num=num*(i-j)/(j+1);
            }
            System.out.println();
        }
    }
}
```

```
1
1 1
1 2 1
1 3 3 1
```

---

20. Missing Number in Array

```
// Input: [1,2,4,5]
public class Demo {
    public static void main(String[] args) {
        int[] arr={1,2,4,5};
        int n=5,sum=n*(n+1)/2;
        int s=0;
        for(int i:arr) s+=i;
        System.out.println(sum-s);
    }
}
// Output: 3
```

---

21. Convert Decimal to Binary

```
// Input: 5
public class Demo {
    public static void main(String[] args) {
        int n=5;
        System.out.println(Integer.toBinaryString(n));
    }
}
// Output: 101
```

---

22. Check Perfect Number

```
// Input: 6
public class Demo {
    public static void main(String[] args) {
        int n=6,sum=0;
        for(int i=1;i<n;i++){
            if(n%i==0) sum+=i;
        }
        System.out.println(sum==n);
    }
}
// Output: true
```

23. Sum of Digits

```
// Input: 123
public class Demo {
    public static void main(String[] args) {
        int n=123,sum=0;
        while(n>0){
            sum+=n%10;
            n/=10;
        }
        System.out.println(sum);
    }
}
// Output: 6
```

24. Length of a String

```
// Input: Hello
public class Demo {
    public static void main(String[] args) {
        String s="Hello";
        System.out.println(s.length());
    }
}
// Output: 5
```

25. Check if String is Empty

```
// Input: ""
public class Demo {
    public static void main(String[] args) {
        String s="";
        System.out.println(s.isEmpty());
    }
}
// Output: true
```

26. Count Occurrences of a Character

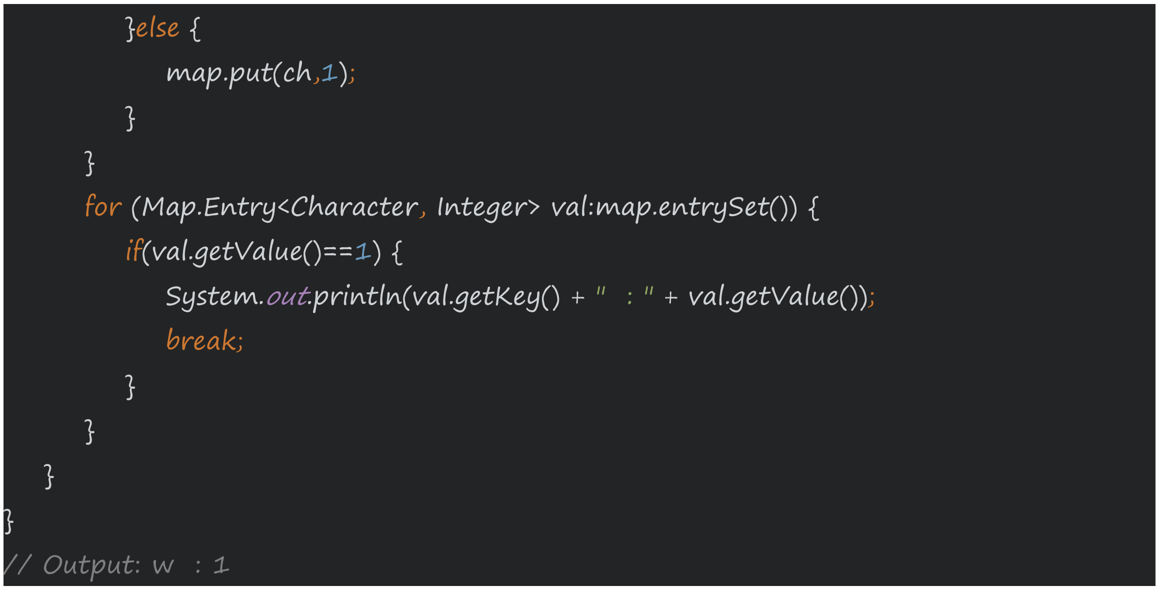
```
// Input: hello
public class Demo {
    public static void main(String[] args) {
        String s="hello";
        Map<Character,Integer> map = new HashMap<>();
        for(char ch:s.toCharArray()){
            if(map.containsKey(ch)){
                map.put(ch,map.get(ch)+1);
            }else {
                map.put(ch,1);
            }
        }

        for (Map.Entry<Character, Integer> val:map.entrySet()) {
                System.out.println(val.getKey() + "  : " + val.getValue());
        }

    }
}
 /*Output: e  : 1
         h  : 1
         l  : 2
         o  : 1*/
```

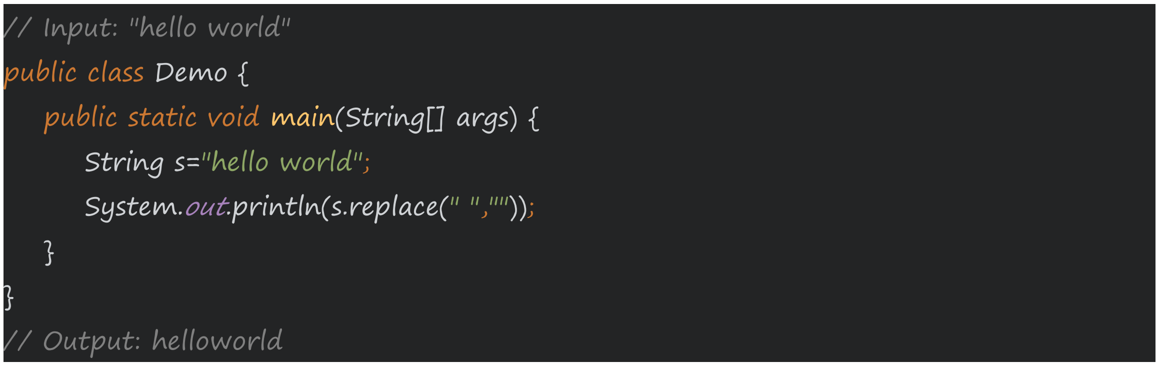
27. First Non-Repeated Character

```
// Input: swiss
public class Demo {
    public static void main(String[] args) {
        String s="swiss";
        Map<Character,Integer> map = new HashMap<>();
        for(char ch:s.toCharArray()){
            if(map.containsKey(ch)){
                map.put(ch,map.get(ch)+1);
```



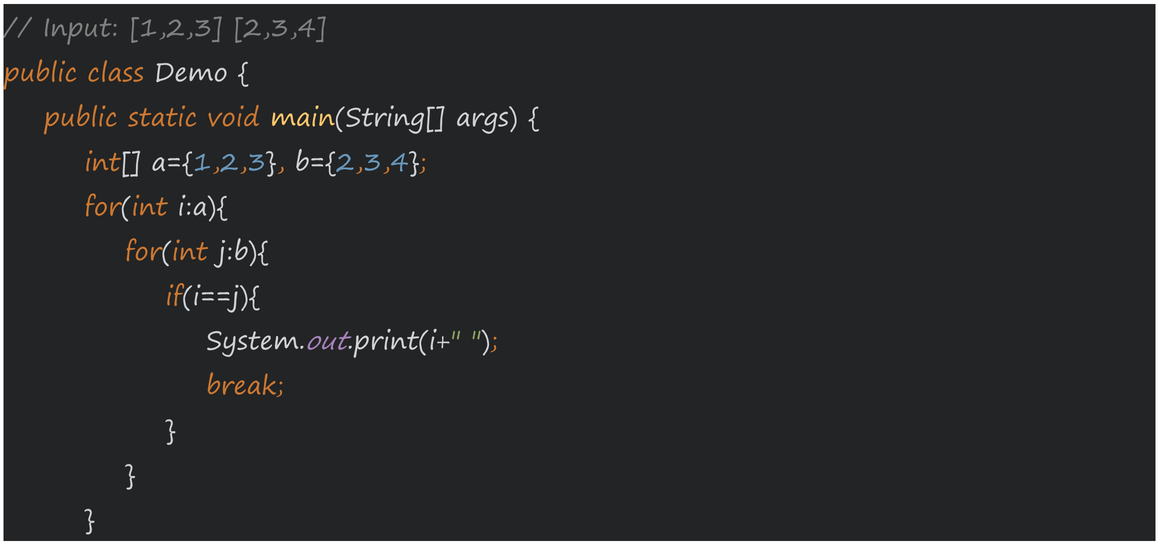
```java
            }else {
                map.put(ch,1);
            }
        }
        for (Map.Entry<Character, Integer> val:map.entrySet()) {
            if(val.getValue()==1) {
                System.out.println(val.getKey() + "  : " + val.getValue());
                break;
            }
        }
    }
}
// Output: w  : 1
```

28. Remove White Spaces from String



```java
// Input: "hello world"
public class Demo {
    public static void main(String[] args) {
        String s="hello world";
        System.out.println(s.replace(" ",""));
    }
}
// Output: helloworld
```

29. Common Elements in Two Arrays



```java
// Input: [1,2,3] [2,3,4]
public class Demo {
    public static void main(String[] args) {
        int[] a={1,2,3}, b={2,3,4};
        for(int i:a){
            for(int j:b){
                if(i==j){
                    System.out.print(i+" ");
                    break;
                }
            }
        }
    }
}
// Output: 2 3
```

30. Factorial Using Recursion

```
// Input: 5
public class Demo {
    public static void main(String[] args) {
        System.out.println(fact(5));
    }
    static int fact(int num){
        if(num==1) return 1;
        return num*fact(num-1);
    }
}
// Output: 120
```

31. Generate Random Number

```
public class Demo {
    public static void main(String[] args) {
        Random r = new Random();
        System.out.println(r.nextInt(100));
    }
}
```

Output

```
(any random number)
```

32. Leap Year Check

```
// Input: 2024
public class Demo {
    public static void main(String[] args) {
        int y=2024;
        System.out.println(y%4==0 && (y%100!=0 || y%400==0));
    }
}
// Output: true
```

---

33. Sum of First N Natural Numbers

```
// Input: 5
public class Demo {
    public static void main(String[] args) {
        int n=5;
        System.out.println(n*(n+1)/2);
    }
}
// Output: 15
```

---

34. Check if String Contains Another String

```
// Input: hello world, world
public class Demo {
    public static void main(String[] args) {
        System.out.println("hello world".contains("world"));
    }
}
// Output: true
```

---

35. Maximum Occurring Character

Solution

```
public class MaxChar {
    public static void main(String[] args) {
        String str = "java";
        int[] count = new int[256];
        char maxChar = str.charAt(0);

        for (char c : str.toCharArray())
            count[c]++;

        for (char c : str.toCharArray()) {
            if (count[c] > count[maxChar])
                maxChar = c;
        }
        System.out.println(maxChar);
    }
}
```

Output

```
a
```

---

36. Bubble Sort

```
// Input: [5,1,4]
public class Demo {
    public static void main(String[] args) {
        int[] arr={5,1,4};
        for(int i=0;i<arr.length;i++){
            for(int j=0;j<arr.length-i-1;j++){
                if(arr[j]>arr[j+1]){
                    int t=arr[j]; arr[j]=arr[j+1]; arr[j+1]=t;
                }
            }
        }
        for(int i:arr) System.out.print(i+" ");
    }
}
// Output: 1 4 5
```

---

37. Selection Sort

```
// Input: [3,1,2]
public class Demo {
    public static void main(String[] args) {
        int[] arr={3,1,2};
        for(int i=0;i<arr.length;i++){
            int min=i;
            for(int j=i+1;j<arr.length;j++){
                if(arr[j]<arr[min]) min=j;
            }
            int t=arr[min]; arr[min]=arr[i]; arr[i]=t;
        }
        for(int i:arr) System.out.print(i+" ");
    }
}
// Output: 1 2 3
```

---

38. Check List Contains Only Odd Numbers

```
// Input: [1,3,5]
public class Demo {
    public static void main(String[] args) {
        int[] arr={1,3,5};
        boolean flag=true;
        for(int i:arr){
            if(i%2==0) flag=false;
        }
        System.out.println(flag);
    }
}
// Output: true
```

---

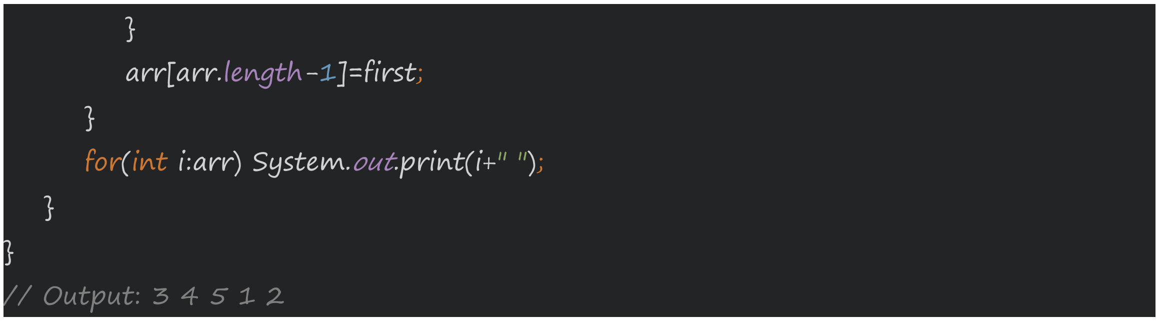
39. Two Arrays Have Same Elements

```
// Input: [1,2,3] [3,2,1]
import java.util.Arrays;
public class Demo {
    public static void main(String[] args) {
        int[] a={1,2,3}, b={3,2,1};
        Arrays.sort(a);
        Arrays.sort(b);
        System.out.println(Arrays.equals(a,b));
    }
}
// Output: true
```

---

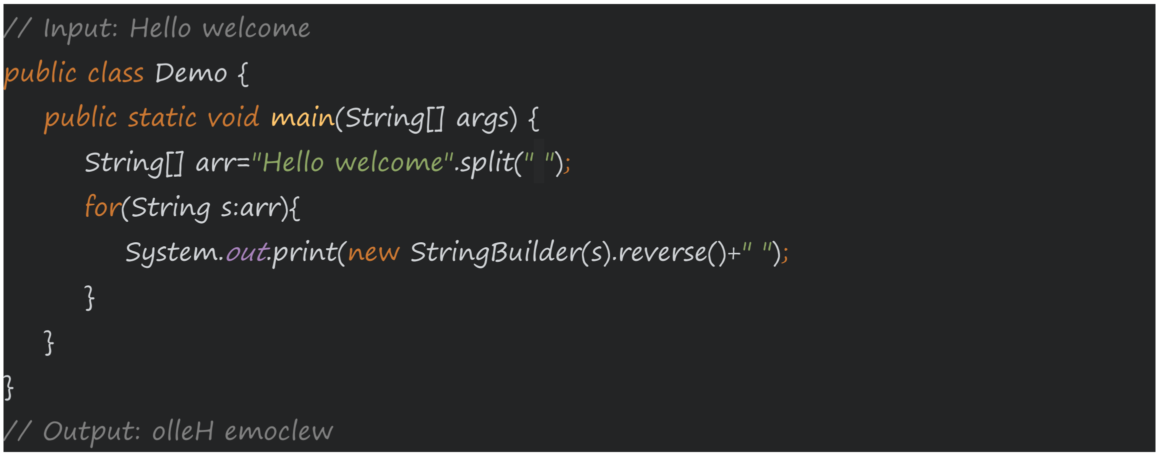
40. Rotate Array Left by N

```
// Input: [1,2,3,4,5], n=2
public class Demo {
    public static void main(String[] args) {
        int[] arr={1,2,3,4,5};
        int n=2;
        for(int i=0;i<n;i++){
            int first=arr[0];
            for(int j=0;j<arr.length-1;j++){
                arr[j]=arr[j+1];
```



```
            }
            arr[arr.length-1]=first;
        }
        for(int i:arr) System.out.print(i+" ");
    }
}
// Output: 3 4 5 1 2
```

41. Reverse Words in String



```
// Input: Hello welcome
public class Demo {
    public static void main(String[] args) {
        String[] arr="Hello welcome".split(" ");
        for(String s:arr){
            System.out.print(new StringBuilder(s).reverse()+" ");
        }
    }
}
// Output: olleH emoclew
```

42. Move All Zeros to End

/*

Move all zeros to the end of the array

In short:

We move all non-zero elements to the beginning of the same array

and push all zeros to the end.

Steps:

1. Take an index variable to store position of non-zero elements

2. Loop through the entire array

3. If element is not zero, place it at index position

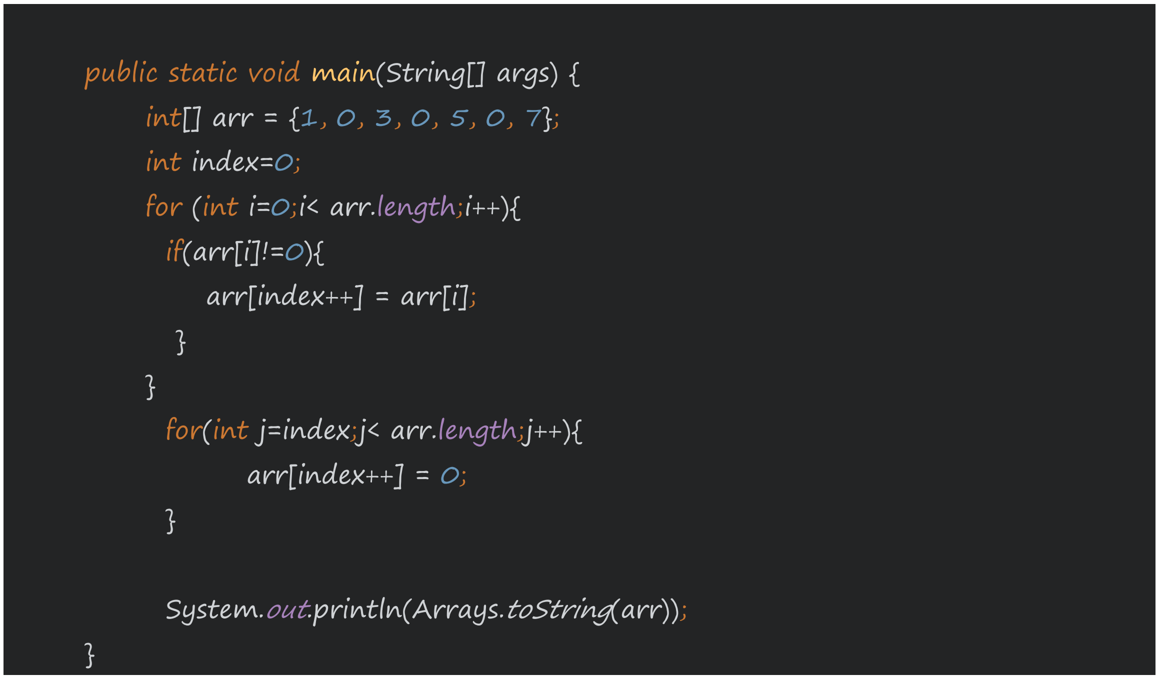
4. Increment index

5. After loop, fill remaining positions with zero

*/

//input:{1, 0, 3, 0, 5, 0, 7};

//output:[1, 3, 5, 7, 0, 0, 0]



```
public static void main(String[] args) {
    int[] arr = {1, 0, 3, 0, 5, 0, 7};
    int index=0;
    for (int i=0;i< arr.length;i++){
     if(arr[i]!=0){
        arr[index++] = arr[i];
      }
    }
     for(int j=index;j< arr.length;j++){
          arr[index++] = 0;
     }

     System.out.println(Arrays.toString(arr));
}
```

Output

1 3 5 7 0 0 0

43. Two Sum Problem

Two Sum using Two Pointer Approach(Array must be sorted)

In short:

We use two pointers from start and end of the array

and move them based on the sum.

Steps:

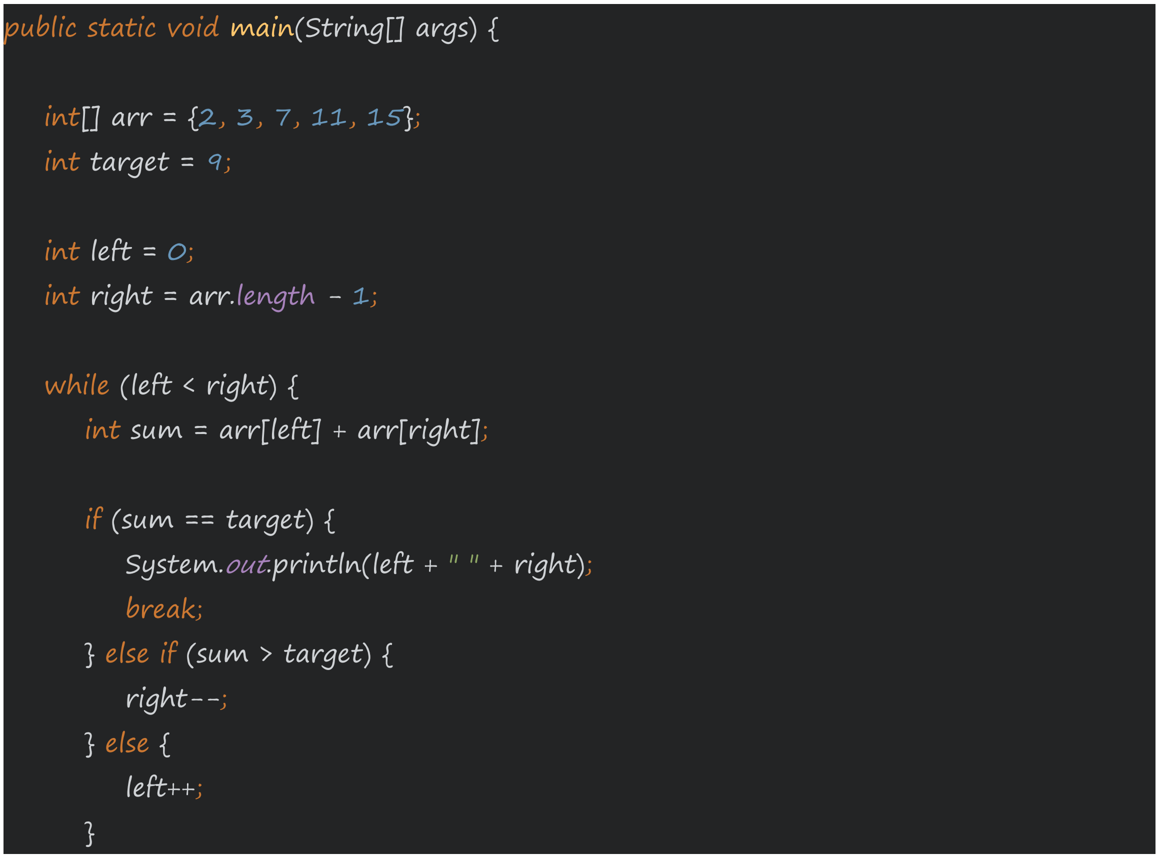
1. Take two pointers: left = 0, right = length - 1

2. Add elements at both pointers

3. If sum equals target → print indices

4. If sum is greater → move right pointer left

5. If sum is smaller → move left pointer right



```
public static void main(String[] args) {

    int[] arr = {2, 3, 7, 11, 15};
    int target = 9;

    int left = 0;
    int right = arr.length - 1;

    while (left < right) {
        int sum = arr[left] + arr[right];

        if (sum == target) {
            System.out.println(left + " " + right);
            break;
        } else if (sum > target) {
            right--;
        } else {
            left++;
        }
    }
}
```

Output

```
0 2
```

---

44. Find Common Elements in Array

Find common elements in two arrays

In short:

We compare elements of both arrays and print elements that appear in both.

Steps:

1. Loop through first array

2. Loop through second array

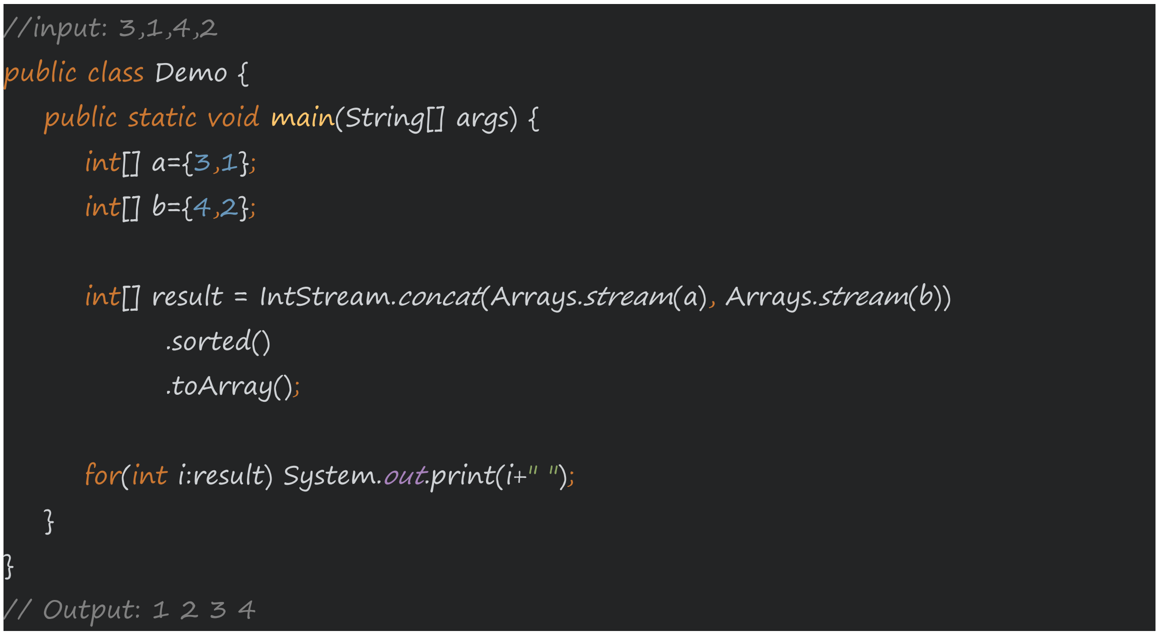
3. If elements are equal, print the element

4. Break inner loop to avoid duplicate printing

```
public static void main(String[] args) {
    int[] arr1 = {1, 2, 3, 4, 5};
    int[] arr2 = {3, 4, 5, 6, 7};
    System.out.print("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;
            }
        }
    }
}
```

OutPut: Common elements: 3 4 5

45. Merge Two Arrays & Sort (Streams)



```
//input: 3,1,4,2
public class Demo {
    public static void main(String[] args) {
        int[] a={3,1};
        int[] b={4,2};

        int[] result = IntStream.concat(Arrays.stream(a), Arrays.stream(b))
                .sorted()
                .toArray();

        for(int i:result) System.out.print(i+" ");
    }
}
// Output: 1 2 3 4
```