

Basic to Advanced Java programs

1. Reverse a String

```
public class ReverseString {  
    public static void main(String[] args) {  
        String str = "Automation";  
        StringBuilder reversed = new StringBuilder(str).reverse();  
        System.out.println(reversed);  
    }  
}
```

2. Check for Palindrome

```
public class Palindrome {  
    public static void main(String[] args) {  
        String str = "madam";  
        String reversed = new StringBuilder(str).reverse().toString();  
        System.out.println(str.equals(reversed));  
    }  
}
```

3. Fibonacci Series

```
public class Fibonacci {  
    public static void main(String[] args) {  
        int n = 10, num1 = 0, num2 = 1;  
        System.out.print("Fibonacci Series: " + num1 + ", " + num2);  
        for (int i = 2; i < n; i++) {  
            int num3 = num1 + num2;  
            System.out.print(", " + num3);  
            num1 = num2; num2 = num3;  
        }  
    }  
}
```

4. Factorial of a Number

```
public class Factorial {  
    public static void main(String[] args) {  
        int num = 5, factorial = 1;  
        for (int i = 1; i <= num; i++) {  
            factorial *= i;  
        }  
        System.out.println(factorial);  
    }  
}
```

5. Prime Number Check

```
public class PrimeCheck {  
    public static void main(String[] args) {  
        int num = 11;  
        boolean isPrime = true;  
        for (int i = 2; i <= Math.sqrt(num); i++) {  
            if (num % i == 0) {  
                isPrime = false;  
                break;  
            }  
        }  
        System.out.println(isPrime);  
    }  
}
```

6. Count Vowels and Consonants

```
public class VowelConsonantCount {  
    public static void main(String[] args) {  
        String str = "Automation";
```

```

int vowels = 0, consonants = 0;
for (char c : str.toCharArray()) {
    if ("aeiouAEIOU".indexOf(c) != -1) {
        vowels++;
    } else if (Character.isLetter(c)) {
        consonants++;
    }
}

System.out.println("Vowels: " + vowels + ", Consonants: " + consonants);
}
}

```

7. Sort an Array

```

import java.util.Arrays;

public class SortArray {

    public static void main(String[] args) {

        int[] arr = {5, 2, 8, 1, 3};

        Arrays.sort(arr);

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

    }
}

```

8. Merge Two Arrays

```

import java.util.Arrays; public class
MergeArrays {
    public static void main(String[] args) {

        int[] arr1 = {1, 3, 5}; int[] arr2 = {2, 4, 6}; int[] merged = new
int[arr1.length + arr2.length]; System.arraycopy(arr1, 0,
merged, 0, arr1.length); System.arraycopy(arr2, 0, merged,
arr1.length, arr2.length);

```

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

9. Find the Largest Element in an Array

```
public class LargestInArray {  
    public static void main(String[] args) {  
        int[] arr = {1, 3, 5, 7, 9};  
        int largest = arr[0]; for  
        (int num : arr) {  
            if (num > largest) {  
                largest = num;  
            }  
        }  
        System.out.println(largest);  
    }  
}
```

10. Remove Duplicates from an Array

```
import java.util.HashSet;  
  
public class RemoveDuplicates {  
    public static void main(String[] args) {  
        int[] arr = {1, 2, 2, 3, 4, 4};  
        HashSet<Integer> set = new HashSet<>();  
        for (int num : arr) {  
            set.add(num);  
        }  
        System.out.println(set);  
    }  
}
```

11. Check if a Number is Armstrong

```
public class ArmstrongNumber {  
    public static void main(String[] args) {  
        int num = 153, sum = 0, temp = num;  
        while (temp != 0) {  
            int digit = temp % 10;  
            sum += Math.pow(digit, 3);  
            temp /= 10;  
        }  
        System.out.println(num == sum);  
    }  
}
```

12. Reverse a Number

```
public class ReverseNumber {  
    public static void main(String[] args) {  
        int num = 12345, reversed = 0;  
        while (num != 0) {  
            reversed = reversed * 10 + num % 10;  
            num /= 10;  
        }  
        System.out.println(reversed);  
    }  
}
```

13. Calculate GCD of Two Numbers

```
public class GCD {  
    public static void main(String[] args) {  
        int a = 60, b = 48;  
        while (b != 0) {  
            int temp = b;
```

```

        b = a % b;
        a = temp;
    }
    System.out.println(a);
}
}

```

14. Check for Anagram

```

import java.util.Arrays;

public class AnagramCheck {

    public static void main(String[] args) {
        String str1 = "listen", str2 = "silent";
        char[] arr1 = str1.toCharArray();
        char[] arr2 = str2.toCharArray();
        Arrays.sort(arr1);
        Arrays.sort(arr2);
        System.out.println(Arrays.equals(arr1, arr2));
    }
}

```

15. Count the Number of Digits in a Number

```

public class CountDigits {

    public static void main(String[] args) {
        int num = 12345;
        int count = String.valueOf(num).length();
        System.out.println(count);
    }
}

```

16. Print the Prime Numbers in a Range

```

public class PrimeInRange {

    public static void main(String[] args) {

```



```

int start = 10, end = 50;
for (int num = start; num <= end; num++) {
    boolean isPrime = true;
    for (int i = 2; i <= Math.sqrt(num); i++) {
        if (num % i == 0) {
            isPrime = false;
            break;
        }
    }
    if (isPrime && num > 1) {
        System.out.print(num + " ");
    }
}
}
}

```

17. Find the Second Largest Element in an Array

```

public class SecondLargest {
    public static void main(String[] args) {
        int[] arr = {12, 35, 1, 10, 34, 1};
        int first = Integer.MIN_VALUE, second = Integer.MIN_VALUE;
        for (int num : arr) {
            if (num > first) {
                second = first;
                first = num;
            } else if (num > second && num != first) {
                second = num;
            }
        }
        System.out.println(second);
    }
}

```

```
}
```

18. Swap Two Numbers

```
public class SwapNumbers {
```

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

```
        int a = 5, b = 10; a = a + b; b = a - b; a = a
```

```
        - b; System.out.println("a: " + a + ", b: " +
```

```
        b);
```

```
    }
```

```
}
```

19. Print the Pascal's Triangle

```
public class PascalsTriangle {
```

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

```
        int rows = 5;
```

```
        for (int i = 0; i < rows; i++) {
```

```
            int num = 1;
```

```
            System.out.format("%" + (rows - i) * 2 + "s", "");
```

```
            for (int j = 0; j <= i; j++) {
```

```
                System.out.format("%4d", num);
```

```
                num = num * (i - j) / (j + 1);
```

```
            }
```

```
            System.out.println();
```

```
        }
```

```
    }
```

```
}
```

20. Find the Missing Number in an Array

```
public class MissingNumber {
```

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



```

int[] arr = {1, 2, 4, 5, 6};

int n = arr.length + 1;

int total = n * (n + 1) / 2;

for (int num : arr) {
    total -= num;
}

System.out.println(total);
}
}

```

21. Convert Decimal to Binary

```

public class DecimalToBinary {
    public static void main(String[] args) {

        int num = 10;

        String binary = Integer.toBinaryString(num);

        System.out.println(binary);
    }
}

```

22. Check for Perfect Number

```

public class PerfectNumber {

    public static void main(String[] args) {

        int num = 28, sum = 0;

        for (int i = 1; i <= num / 2; i++) {

            if (num % i == 0) {

                sum += i;

            }

        }

        System.out.println(num == sum);

    }

}

```