```
1. public class SeparateOddEvenArrays {
    public static void main(String[] args) {
        int[] inputArray = {10, 3, 5, 12, 17, 22};

        // Create arrays for odd and even elements
        int[] evenArray = new int[inputArray.length];
        int[] oddArray = new int[inputArray.length];

        int evenCount = 0;
        int oddCount = 0;

        // Separate odd and even elements
        for (int num : inputArray) {
            if (num % 2 == 0) {
                evenArray[evenCount++] = num;
            } else {
                oddArray[oddCount++] = num;
            }
        }

        // Trim the arrays to remove unused space
        evenArray = trimArray(evenArray, evenCount);
        oddArray = trimArray(oddArray, oddCount);

        // Print the results
        System.out.println("Even Array:");
        printArray(evenArray);

        System.out.println("\nOdd Array:");
        printArray(oddArray);
    }
```

```
    // Helper method to trim the array to the actual size
    private static int[] trimArray(int[] array, int size) {
        int[] trimmedArray = new int[size];
        System.arraycopy(array, 0, trimmedArray, 0, size);
        return trimmedArray;
    }

    // Helper method to print an array
    private static void printArray(int[] array) {
        for (int num : array) {
            System.out.print(num + " ");
        }
        System.out.println();
    }
}
```

2.
```
public class StringCompression {
    public static void main(String[] args) {
        String input1 = "AAABBC";
        String input2 = "AAABBCCCDE";

        String compressed1 = compressString(input1);
        String compressed2 = compressString(input2);

        System.out.println("Input: " + input1);
        System.out.println("Output: " + compressed1);

        System.out.println("\nInput: " + input2);
        System.out.println("Output: " + compressed2);
    }
```

```
    private static String compressString(String input) {
        StringBuilder compressed = new StringBuilder();

        int count = 1;
        for (int i = 0; i < input.length(); i++) {
            if (i < input.length() - 1 && input.charAt(i) == input.charAt(i + 1)) {
                count++;
            } else {
                compressed.append(input.charAt(i));
                if (count > 1) {
                    compressed.append(count);
                }
                count = 1;
            }
        }

        return compressed.toString();
    }
}
```

3.
```
public class ZigZagPattern {
    public static void main(String[] args) {
        String input = "zohocorporationteam";
        printZigZagPattern(input);
    }

    private static void printZigZagPattern(String input) {
        int n = input.length();

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

```
            if (i == j) {
                System.out.print(input.charAt(j));
            } else if (i + j == n - 1) {
                System.out.print("  " + input.charAt(j));
            } else {
                System.out.print("   ");
            }
        }
        System.out.println();
    }
  }
}
```