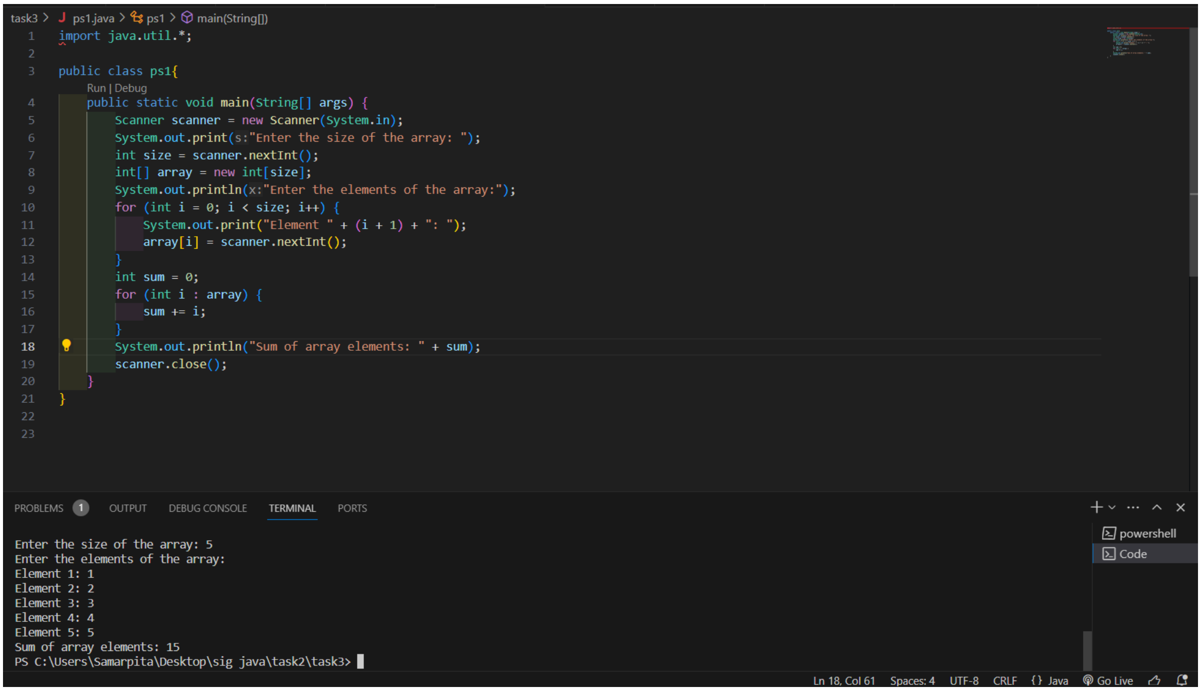
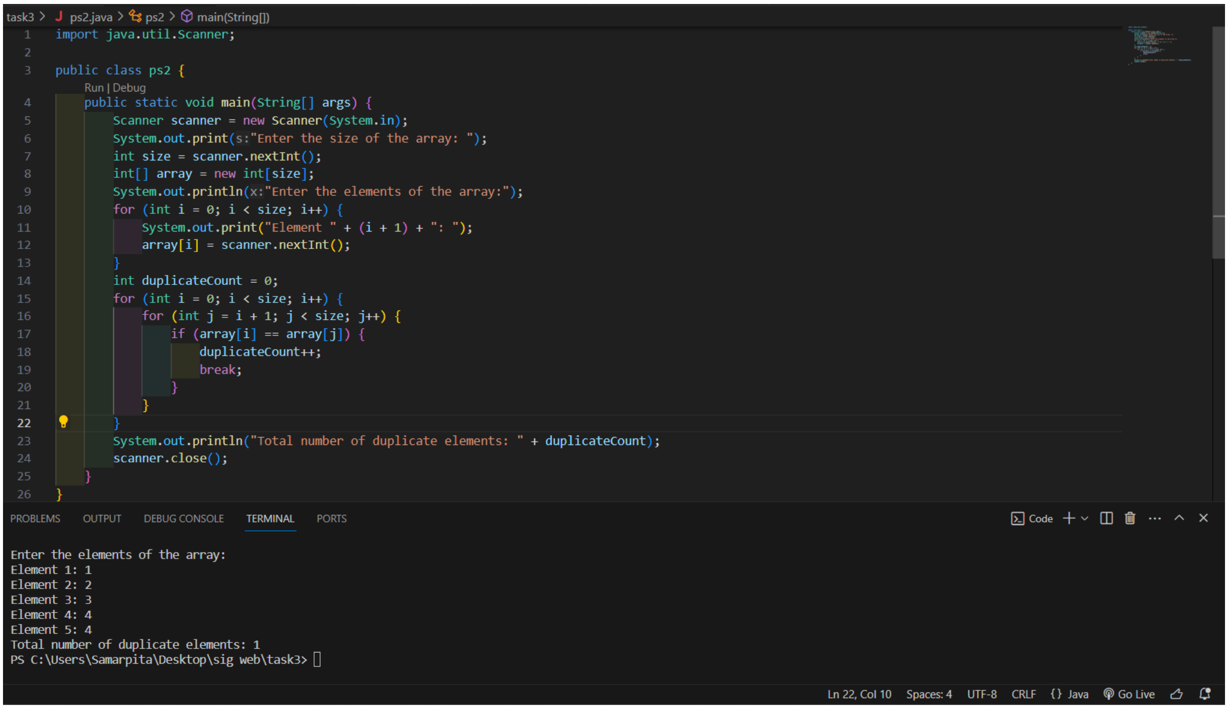
1. WAP to find the sum of all elements of the array.



```java
import java.util.*;

public class ps1{
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print(s:"Enter the size of the array: ");
        int size = scanner.nextInt();
        int[] array = new int[size];
        System.out.println(x:"Enter the elements of the array:");
        for (int i = 0; i < size; i++) {
            System.out.print("Element " + (i + 1) + ": ");
            array[i] = scanner.nextInt();
        }
        int sum = 0;
        for (int i : array) {
            sum += i;
        }
        System.out.println("Sum of array elements: " + sum);
        scanner.close();
    }
}
```

```
Enter the size of the array: 5
Enter the elements of the array:
Element 1: 1
Element 2: 2
Element 3: 3
Element 4: 4
Element 5: 5
Sum of array elements: 15
PS C:\Users\Samarpita\Desktop\sig java\task2\task3>
```

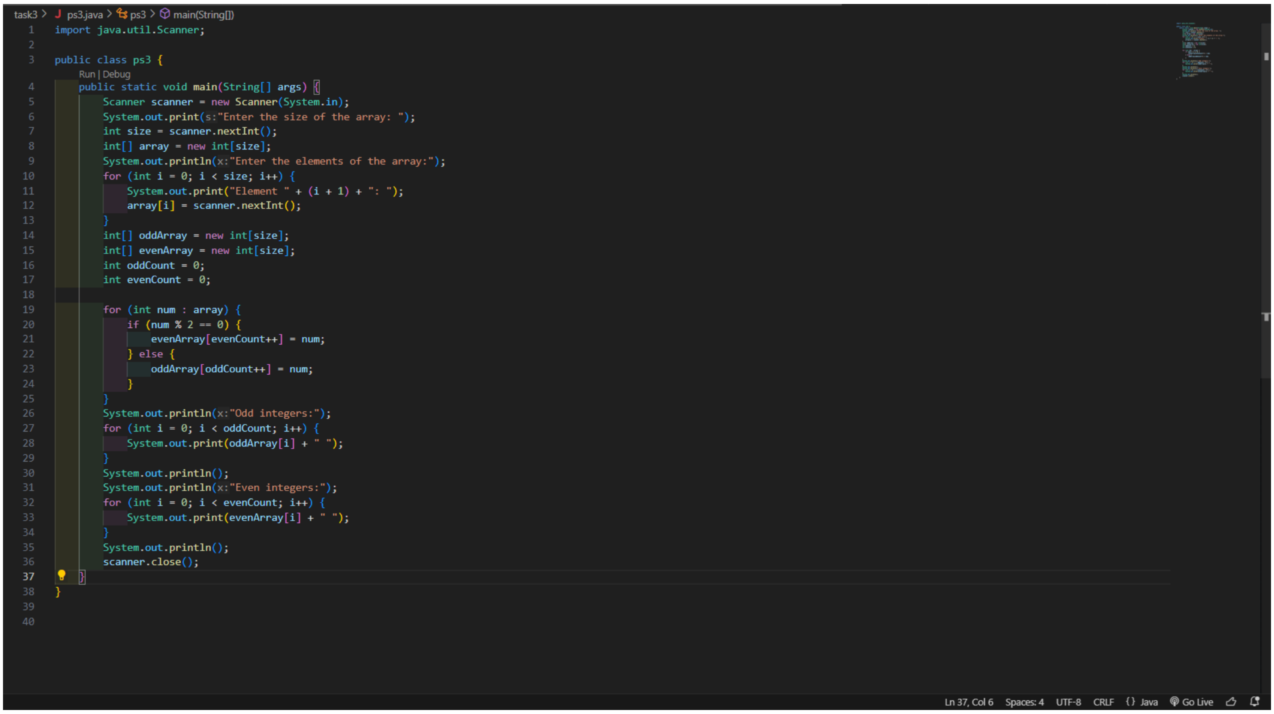
2. WAP to count the total number of duplicate elements in an array.



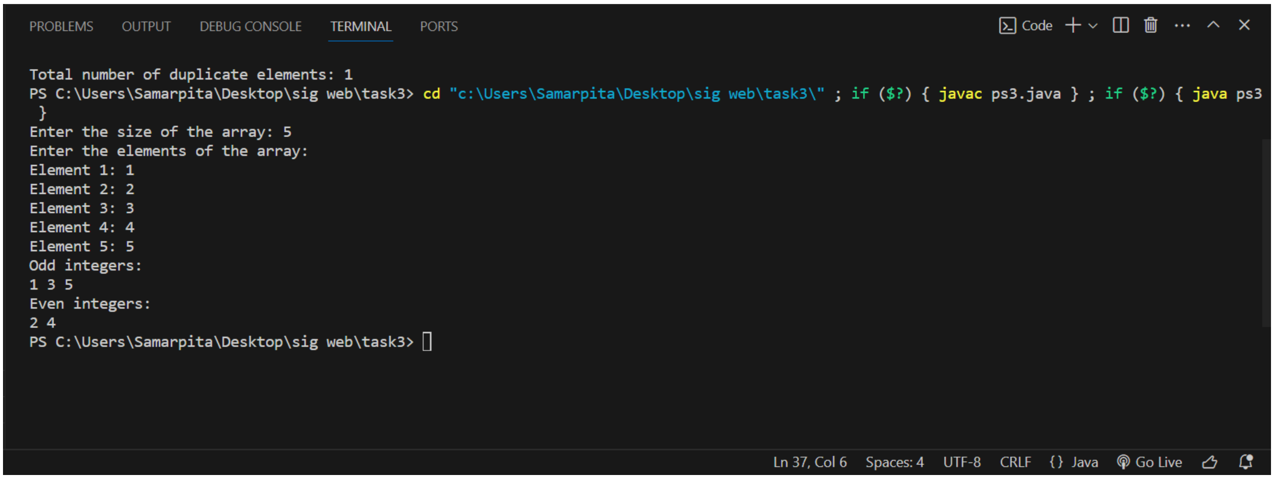
```java
import java.util.Scanner;

public class ps2 {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print(s:"Enter the size of the array: ");
        int size = scanner.nextInt();
        int[] array = new int[size];
        System.out.println(x:"Enter the elements of the array:");
        for (int i = 0; i < size; i++) {
            System.out.print("Element " + (i + 1) + ": ");
            array[i] = scanner.nextInt();
        }
        int duplicateCount = 0;
        for (int i = 0; i < size; i++) {
            for (int j = i + 1; j < size; j++) {
                if (array[i] == array[j]) {
                    duplicateCount++;
                    break;
                }
            }
        }
        System.out.println("Total number of duplicate elements: " + duplicateCount);
        scanner.close();
    }
}
```

```
Enter the elements of the array:
Element 1: 1
Element 2: 2
Element 3: 3
Element 4: 4
Element 5: 4
Total number of duplicate elements: 1
PS C:\Users\Samarpita\Desktop\sig web\task3>
```

3. WAP to separate odd and even integers into separate arrays.

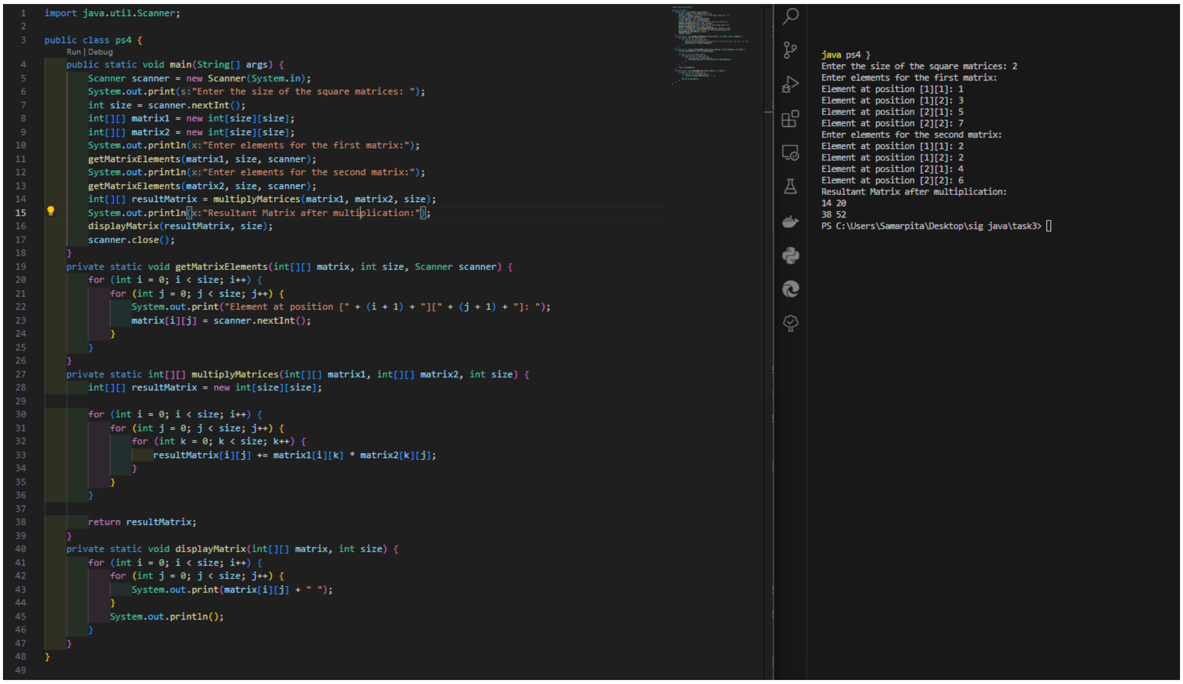


```java
import java.util.Scanner;

public class ps3 {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print(s:"Enter the size of the array: ");
        int size = scanner.nextInt();
        int[] array = new int[size];
        System.out.println(x:"Enter the elements of the array:");
        for (int i = 0; i < size; i++) {
            System.out.print("Element " + (i + 1) + ": ");
            array[i] = scanner.nextInt();
        }
        int[] oddArray = new int[size];
        int[] evenArray = new int[size];
        int oddCount = 0;
        int evenCount = 0;

        for (int num : array) {
            if (num % 2 == 0) {
                evenArray[evenCount++] = num;
            } else {
                oddArray[oddCount++] = num;
            }
        }
        System.out.println(x:"Odd integers:");
        for (int i = 0; i < oddCount; i++) {
            System.out.print(oddArray[i] + " ");
        }
        System.out.println();
        System.out.println(x:"Even integers:");
        for (int i = 0; i < evenCount; i++) {
            System.out.print(evenArray[i] + " ");
        }
        System.out.println();
        scanner.close();
    }
}
```



```
Total number of duplicate elements: 1
PS C:\Users\Samarpita\Desktop\sig web\task3> cd "c:\Users\Samarpita\Desktop\sig web\task3\" ; if ($?) { javac ps3.java } ; if ($?) { java ps3
 }
Enter the size of the array: 5
Enter the elements of the array:
Element 1: 1
Element 2: 2
Element 3: 3
Element 4: 4
Element 5: 5
Odd integers:
1 3 5
Even integers:
2 4
PS C:\Users\Samarpita\Desktop\sig web\task3> 
```

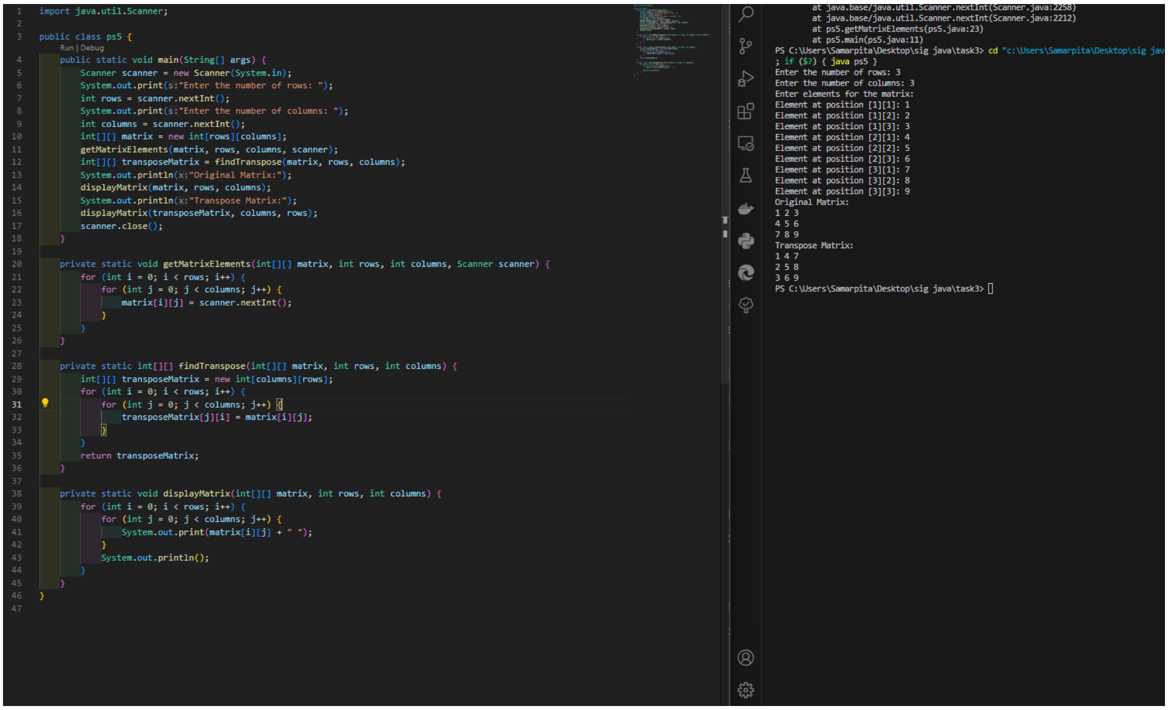
4. WAP for the multiplication of two square matrices.



```java
import java.util.Scanner;

public class ps4 {
    Run | Debug
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print(s:"Enter the size of the square matrices: ");
        int size = scanner.nextInt();
        int[][] matrix1 = new int[size][size];
        int[][] matrix2 = new int[size][size];
        System.out.println(x:"Enter elements for the first matrix:");
        getMatrixElements(matrix1, size, scanner);
        System.out.println(x:"Enter elements for the second matrix:");
        getMatrixElements(matrix2, size, scanner);
        int[][] resultMatrix = multiplyMatrices(matrix1, matrix2, size);
        System.out.println(x:"Resultant Matrix after multiplication:");
        displayMatrix(resultMatrix, size);
        scanner.close();
    }
    private static void getMatrixElements(int[][] matrix, int size, Scanner scanner) {
        for (int i = 0; i < size; i++) {
            for (int j = 0; j < size; j++) {
                System.out.print("Element at position [" + (i + 1) + "][" + (j + 1) + "]: ");
                matrix[i][j] = scanner.nextInt();
            }
        }
    }
    private static int[][] multiplyMatrices(int[][] matrix1, int[][] matrix2, int size) {
        int[][] resultMatrix = new int[size][size];

        for (int i = 0; i < size; i++) {
            for (int j = 0; j < size; j++) {
                for (int k = 0; k < size; k++) {
                    resultMatrix[i][j] += matrix1[i][k] * matrix2[k][j];
                }
            }
        }

        return resultMatrix;
    }
    private static void displayMatrix(int[][] matrix, int size) {
        for (int i = 0; i < size; i++) {
            for (int j = 0; j < size; j++) {
                System.out.print(matrix[i][j] + " ");
            }
            System.out.println();
        }
    }
}
```

```
java ps4 }
Enter the size of the square matrices: 2
Enter elements for the first matrix:
Element at position [1][1]: 1
Element at position [1][2]: 3
Element at position [2][1]: 5
Element at position [2][2]: 7
Enter elements for the second matrix:
Element at position [1][1]: 2
Element at position [1][2]: 2
Element at position [2][1]: 4
Element at position [2][2]: 6
Resultant Matrix after multiplication:
14 20
38 52
PS C:\Users\Samarpita\Desktop\sig java\task3> 
```

5. WAP to find the transpose of a given matrix.



```java
import java.util.Scanner;

public class ps5 {
    Run | Debug
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print(s:"Enter the number of rows: ");
        int rows = scanner.nextInt();
        System.out.print(s:"Enter the number of columns: ");
        int columns = scanner.nextInt();
        int[][] matrix = new int[rows][columns];
        getMatrixElements(matrix, rows, columns, scanner);
        int[][] transposeMatrix = findTranspose(matrix, rows, columns);
        System.out.println(x:"Original Matrix:");
        displayMatrix(matrix, rows, columns);
        System.out.println(x:"Transpose Matrix:");
        displayMatrix(transposeMatrix, columns, rows);
        scanner.close();
    }

    private static void getMatrixElements(int[][] matrix, int rows, int columns, Scanner scanner) {
        for (int i = 0; i < rows; i++) {
            for (int j = 0; j < columns; j++) {
                matrix[i][j] = scanner.nextInt();
            }
        }
    }

    private static int[][] findTranspose(int[][] matrix, int rows, int columns) {
        int[][] transposeMatrix = new int[columns][rows];
        for (int i = 0; i < rows; i++) {
            for (int j = 0; j < columns; j++) {
                transposeMatrix[j][i] = matrix[i][j];
            }
        }
        return transposeMatrix;
    }

    private static void displayMatrix(int[][] matrix, int rows, int columns) {
        for (int i = 0; i < rows; i++) {
            for (int j = 0; j < columns; j++) {
                System.out.print(matrix[i][j] + " ");
            }
            System.out.println();
        }
    }
}
```

```
        at java.base/java.util.Scanner.nextInt(Scanner.java:2258)
        at java.base/java.util.Scanner.nextInt(Scanner.java:2212)
        at ps5.getMatrixElements(ps5.java:23)
        at ps5.main(ps5.java:11)
PS C:\Users\Samarpita\Desktop\sig java\task3> cd "c:\Users\Samarpita\Desktop\sig jav
; if ($?) { java ps5 }
Enter the number of rows: 3
Enter the number of columns: 3
Enter elements for the matrix:
Element at position [1][1]: 1
Element at position [1][2]: 2
Element at position [1][3]: 3
Element at position [2][1]: 4
Element at position [2][2]: 5
Element at position [2][3]: 6
Element at position [3][1]: 7
Element at position [3][2]: 8
Element at position [3][3]: 9
Original Matrix:
1 2 3
4 5 6
7 8 9
Transpose Matrix:
1 4 7
2 5 8
3 6 9
PS C:\Users\Samarpita\Desktop\sig java\task3> 
```

6. Define and compare single-dimensional and multi-dimensional arrays, providing examples for each.

comparison	**Single-Dimensional Array:**	**Multi-Dimensional Array:**
Definition:	Contains elements in a linear, one-dimensional sequence.	Contains elements organized in multiple dimensions, often represented as rows and columns.
Declaration:	Declared using one pair of square brackets [].	Declared using multiple pairs of square brackets, indicating the number of dimensions.
Initialization:	Elements are initialized in a single line.	Initialization involves nested arrays, with each inner array representing a row.
Accessing Elements:	Accessed using a single index.	Accessed using multiple indices, with one index for each dimension.
Example:	int[] scores = {85, 90, 75, 92, 88};	int[][] matrix = { {1, 2, 3}, {4, 5, 6}, {7, 8, 9} };