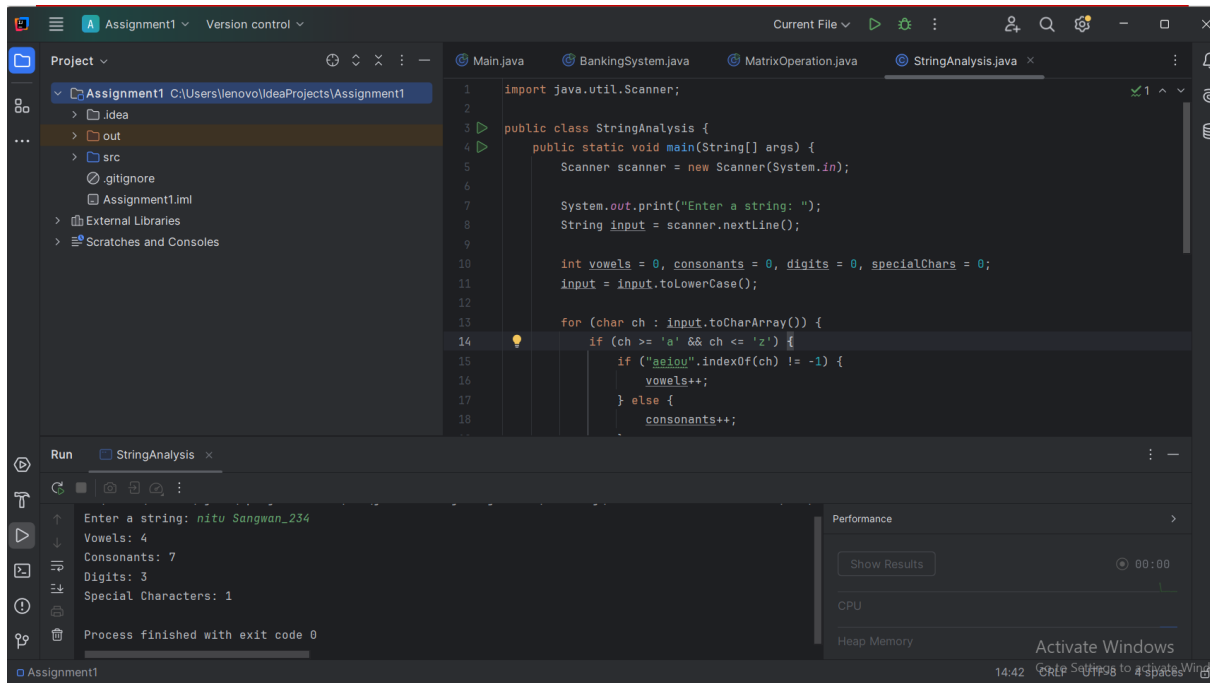


Input_output_examples:

StringAnalysis.java



The screenshot shows the IntelliJ IDEA IDE with the `StringAnalysis.java` file open. The code defines a `StringAnalysis` class with a `main` method that takes command-line arguments, reads a string from the user, and counts the number of vowels, consonants, digits, and special characters. The output window shows the results for the input string "nitu Sangwan_234".

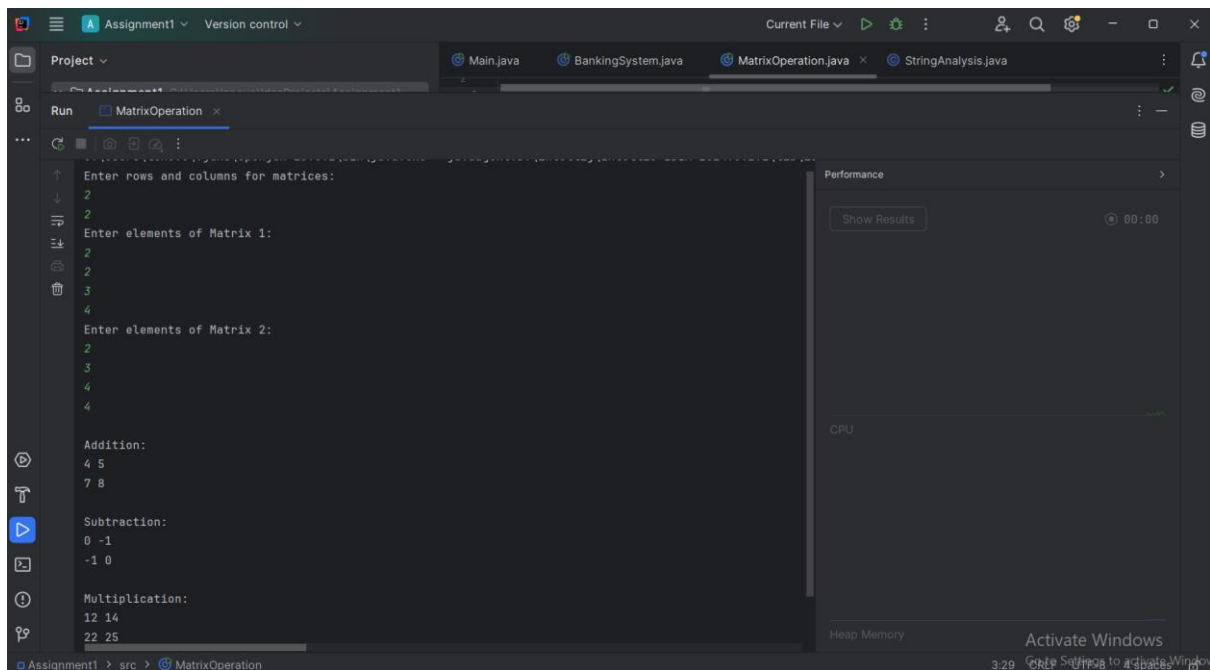
```
1 import java.util.Scanner;
2
3 public class StringAnalysis {
4     public static void main(String[] args) {
5         Scanner scanner = new Scanner(System.in);
6
7         System.out.print("Enter a string: ");
8         String input = scanner.nextLine();
9
10        int vowels = 0, consonants = 0, digits = 0, specialChars = 0;
11        input = input.toLowerCase();
12
13        for (char ch : input.toCharArray()) {
14            if (ch >= 'a' && ch <= 'z') {
15                if ("aeiou".indexOf(ch) != -1) {
16                    vowels++;
17                } else {
18                    consonants++;
19                }
20            }
21        }
22
23        System.out.println("Vowels: " + vowels);
24        System.out.println("Consonants: " + consonants);
25        System.out.println("Digits: " + digits);
26        System.out.println("Special Characters: " + specialChars);
27    }
28 }
```

Run StringAnalysis x

Enter a string: nitu Sangwan_234
Vowels: 4
Consonants: 7
Digits: 3
Special Characters: 1
Process finished with exit code 0

Performance
Show Results 00:00
CPU
Heap Memory
Activate Windows

MatrixOperations.java



The screenshot shows the IntelliJ IDEA IDE with the `MatrixOperations.java` file open. The code defines a `MatrixOperations` class with methods for adding, subtracting, and multiplying matrices. The output window shows the results for two 2x2 matrices.

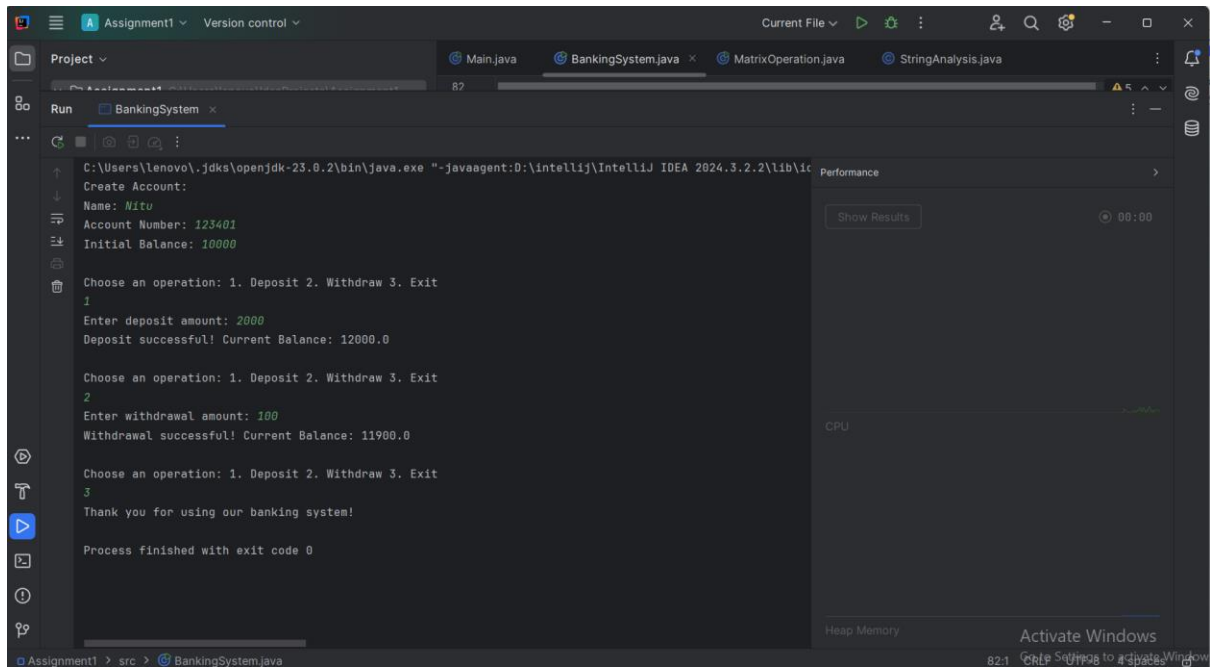
```
1 import java.util.Scanner;
2
3 public class MatrixOperations {
4     public static void main(String[] args) {
5         Scanner scanner = new Scanner(System.in);
6
7         System.out.print("Enter rows and columns for matrices: ");
8         int rows = scanner.nextInt();
9         int cols = scanner.nextInt();
10
11        System.out.print("Enter elements of Matrix 1: ");
12        int[][] matrix1 = new int[rows][cols];
13        for (int i = 0; i < rows; i++) {
14            for (int j = 0; j < cols; j++) {
15                matrix1[i][j] = scanner.nextInt();
16            }
17        }
18
19        System.out.print("Enter elements of Matrix 2: ");
20        int[][] matrix2 = new int[rows][cols];
21        for (int i = 0; i < rows; i++) {
22            for (int j = 0; j < cols; j++) {
23                matrix2[i][j] = scanner.nextInt();
24            }
25        }
26
27        System.out.println("Addition:");
28        int[][] result = add(matrix1, matrix2);
29        for (int i = 0; i < rows; i++) {
30            for (int j = 0; j < cols; j++) {
31                System.out.print(result[i][j] + " ");
32            }
33            System.out.println();
34        }
35
36        System.out.println("Subtraction:");
37        result = subtract(matrix1, matrix2);
38        for (int i = 0; i < rows; i++) {
39            for (int j = 0; j < cols; j++) {
40                System.out.print(result[i][j] + " ");
41            }
42            System.out.println();
43        }
44
45        System.out.println("Multiplication:");
46        result = multiply(matrix1, matrix2);
47        for (int i = 0; i < rows; i++) {
48            for (int j = 0; j < cols; j++) {
49                System.out.print(result[i][j] + " ");
50            }
51            System.out.println();
52        }
53    }
54
55    private static int[][] add(int[][] matrix1, int[][] matrix2) {
56        int rows = matrix1.length;
57        int cols = matrix1[0].length;
58        int[][] result = new int[rows][cols];
59        for (int i = 0; i < rows; i++) {
60            for (int j = 0; j < cols; j++) {
61                result[i][j] = matrix1[i][j] + matrix2[i][j];
62            }
63        }
64        return result;
65    }
66
67    private static int[][] subtract(int[][] matrix1, int[][] matrix2) {
68        int rows = matrix1.length;
69        int cols = matrix1[0].length;
70        int[][] result = new int[rows][cols];
71        for (int i = 0; i < rows; i++) {
72            for (int j = 0; j < cols; j++) {
73                result[i][j] = matrix1[i][j] - matrix2[i][j];
74            }
75        }
76        return result;
77    }
78
79    private static int[][] multiply(int[][] matrix1, int[][] matrix2) {
80        int rows1 = matrix1.length;
81        int cols1 = matrix1[0].length;
82        int rows2 = matrix2.length;
83        int cols2 = matrix2[0].length;
84        if (cols1 != rows2) {
85            System.out.println("Matrix multiplication is not possible.");
86            return null;
87        }
88        int rows = rows1;
89        int cols = cols2;
90        int[][] result = new int[rows][cols];
91        for (int i = 0; i < rows; i++) {
92            for (int j = 0; j < cols; j++) {
93                for (int k = 0; k < rows2; k++) {
94                    result[i][j] += matrix1[i][k] * matrix2[k][j];
95                }
96            }
97        }
98        return result;
99    }
100 }
```

Run MatrixOperation x

Enter rows and columns for matrices:
2
2
Enter elements of Matrix 1:
2
2
3
4
Enter elements of Matrix 2:
2
3
4
4
Addition:
4 5
7 8
Subtraction:
0 -1
-1 0
Multiplication:
12 14
22 25
Process finished with exit code 0

Performance
Show Results 00:00
CPU
Heap Memory
Activate Windows

BankSystem.java



```
C:\Users\lenovo\jdk-23.0.2\bin\java.exe "-javaagent:D:\intellij\IntelliJ IDEA 2024.3.2.2\lib\ic
Create Account:
Name: Nitu
Account Number: 123401
Initial Balance: 10000

Choose an operation: 1. Deposit 2. Withdraw 3. Exit
1
Enter deposit amount: 2000
Deposit successful! Current Balance: 12000.0

Choose an operation: 1. Deposit 2. Withdraw 3. Exit
2
Enter withdrawal amount: 100
Withdrawal successful! Current Balance: 11900.0

Choose an operation: 1. Deposit 2. Withdraw 3. Exit
3
Thank you for using our banking system!

Process finished with exit code 0
```

Performance

Show Results 00:00

CPU

Heap Memory

Activate Windows

82.1 CPU usage to page Window