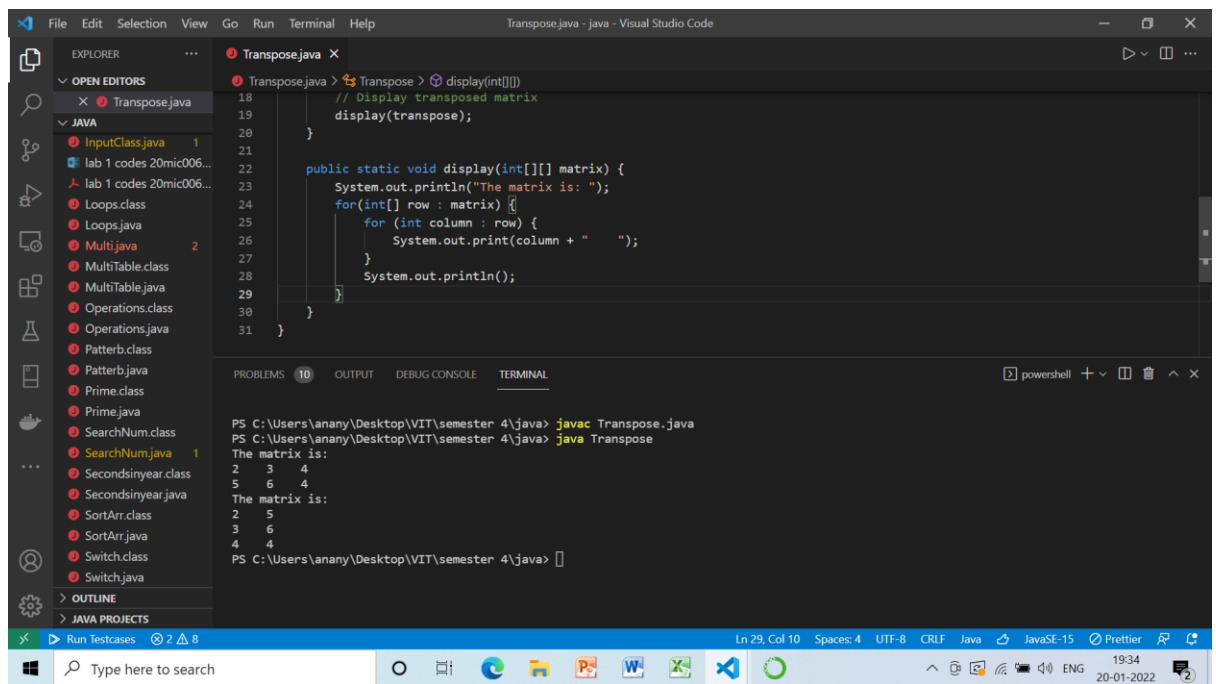


1. Find the transpose of a matrix.

```
public class Transpose {  
  
    public static void main(String[] args) {  
        int row = 2, column = 3;  
        int[][] matrix = { {2, 3, 4}, {5, 6, 4} };  
  
        // Display current matrix  
        display(matrix);  
  
        // Transpose the matrix  
        int[][] transpose = new int[column][row];  
        for(int i = 0; i < row; i++) {  
            for (int j = 0; j < column; j++) {  
                transpose[j][i] = matrix[i][j];  
            }  
        }  
  
        // Display transposed matrix  
        display(transpose);  
    }  
  
    public static void display(int[][] matrix) {  
        System.out.println("The matrix is: ");  
        for(int[] row : matrix) {  
            for (int column : row) {  
                System.out.print(column + " ");  
            }  
            System.out.println();  
        }  
    }  
}
```



2. Given a 2D array, re-arrange the rows in the ascending order of the row sum. Row sum is the sum of all the elements in the row.

import java.util.Arrays;

public class Rearrange {
//Java to find positions of zeroes flipping which
// produces maximum number of consecutive 1's

static int arr[] = new int[]{50, 40, 70, 60, 90};
static int index[] = new int[]{3, 0, 4, 1, 2};

// Method to reorder elements of arr[] according
// to index[]

static void reorder()

{

int temp[] = new int[arr.length];

// arr[i] should be present at index[i] index

for (int i=0; i<arr.length; i++)

temp[index[i]] = arr[i];

// Copy temp[] to arr[]

for (int i=0; i<arr.length; i++)

{

arr[i] = temp[i];

index[i] = i;

}

```

    }

    // Driver method to test the above function
    public static void main(String[] args)
    {

        reorder();

        System.out.println("Reordered array is: ");
        System.out.println(Arrays.toString(arr));
        System.out.println("Modified Index array is:");
        System.out.println(Arrays.toString(index));

    }
}

```

The screenshot shows the Visual Studio Code editor with the file `Rearrange.java` open. The code defines a `Rearrange` class with a `reorder` method. The `main` method calls `reorder` and prints the results. The terminal output shows the execution of the program, displaying the reordered array and the modified index array.

```

Rearrange.java
1  import java.util.Arrays;
2
3  public class Rearrange {
4      //Java to find positions of zeroes flipping which
5      // produces maximum number of consecutive 1's
6
7
8      static int arr[] = new int[]{50, 40, 70, 60, 90};
9      static int index[] = new int[]{3, 0, 4, 1, 2};
10
11     // Method to reorder elements of arr[] according
12     // to index[]
13     static void reorder()
14     {
15         int temp[] = new int[arr.length];

```

```

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\anany\Desktop\VIT\semester 4\java> cd "C:\Users\anany\Desktop\VIT\semester 4\java"; if ($?) { javac
Rearrange.java } ; if ($?) { java Rearrange }
Reordered array is:
[40, 60, 90, 50, 70]
Modified Index array is:
[0, 1, 2, 3, 4]
PS C:\Users\anany\Desktop\VIT\semester 4\java>

```

3. If a 2D array has n rows and each k row has k columns then it is called a lower-triangular matrix.

Example

```

1
2 3
4 5 6
7 8 9 10

```

The program should return a symmetric matrix as output.

output

```
1 2 4 7
2 3 5 8
4 5 6 9
7 8 9 10
```

```
// Java program to print Lower
// triangular and Upper triangular
// matrix of an array
class LowerTriangular
{
    // method to form lower
    // triangular matrix
    static void lower(int matrix[][],
                      int row, int col)
    {
        int i, j;
        for (i = 0; i < row; i++)
        {
            for (j = 0; j < col; j++)
            {
                if (i < j)
                {
                    System.out.print("0" + " ");
                }
                else
                    System.out.print(matrix[i][j] + " ");
            }
            System.out.println();
        }
    }

    // Method to form upper
    // triangular matrix
    static void upper(int matrix[][],
                      int row, int col)
    {
        int i, j;
        for (i = 0; i < row; i++)
        {
            for (j = 0; j < col; j++)
            {
                if (i > j)
                {
                    System.out.print("0" + " ");
                }
                else
                    System.out.print(matrix[i][j] + " ");
            }
            System.out.println();
        }
    }
}
```

```

    }
}

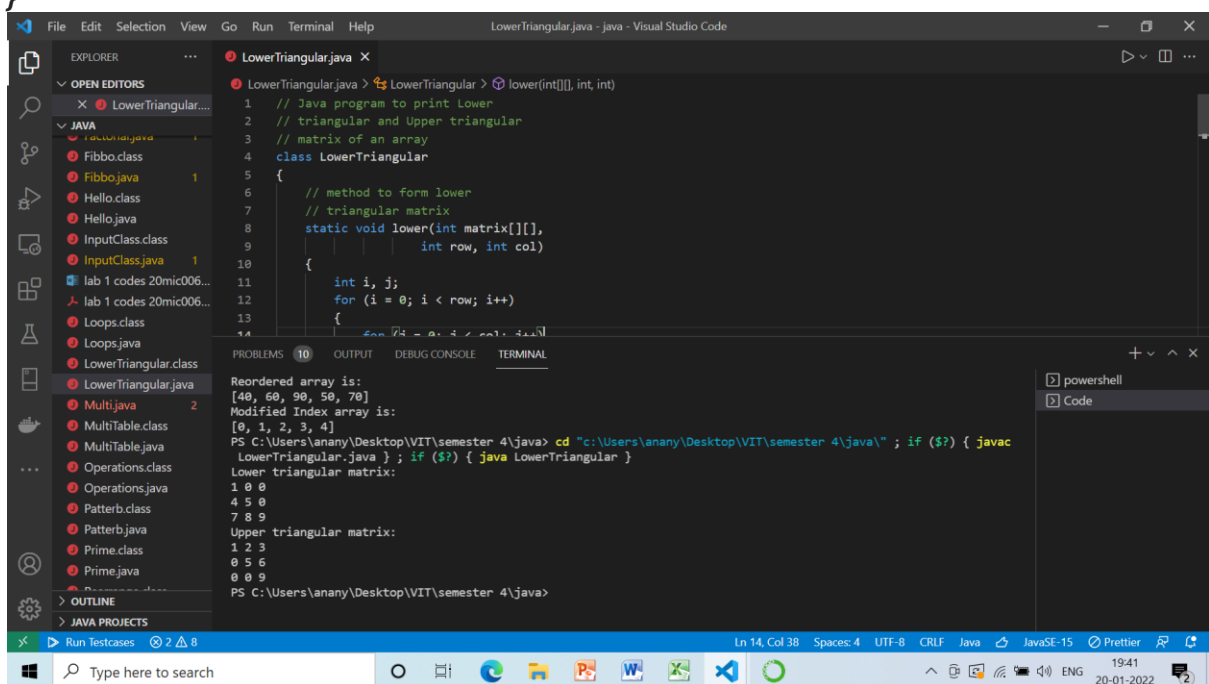
// Driver Code
public static void main(String args[])
{
    int matrix[][] = {{1, 2, 3},
                      {4, 5, 6},
                      {7, 8, 9}};

    int row = 3, col = 3;

    System.out.println("Lower triangular matrix: ");
    lower(matrix, row, col);

    System.out.println("Upper triangular matrix: ");
    upper(matrix, row, col);
}
}

```



```

LowerTriangular.java
1 // Java program to print Lower
2 // triangular and Upper triangular
3 // matrix of an array
4 class LowerTriangular
5 {
6     // method to form lower
7     // triangular matrix
8     static void lower(int matrix[][],
9                       int row, int col)
10    {
11        int i, j;
12        for (i = 0; i < row; i++)
13        {
14            for (j = 0; j < col; j++)
15            {
16                // Print the element at row i and column j
17                System.out.print(matrix[i][j] + " ");
18                // Print a space after each element
19                System.out.print(" ");
20            }
21            // Print a new line after each row
22            System.out.println();
23        }
24    }
25 }

```

Reordered array is:
[40, 60, 90, 50, 70]
Modified Index array is:
[0, 1, 2, 3, 4]
PS C:\Users\anany\Desktop\VIT\semester 4\java> cd "c:\Users\anany\Desktop\VIT\semester 4\java\" ; if (\$?) { javac LowerTriangular.java } ; if (\$?) { java LowerTriangular }
Lower triangular matrix:
1 0 0
4 5 0
7 8 9
Upper triangular matrix:
1 2 3
0 5 6
0 0 0
PS C:\Users\anany\Desktop\VIT\semester 4\java>