

Primera Clase

```
package ejercicios;

import java.util.Random;
import java.util.Scanner;

public class sumaFilasTotal {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.println("Ingrese el número de filas");
        int f = sc.nextInt();
        System.out.println("Ingrese el número de columnas");
        int c = sc.nextInt();

        int [][] matriz = llenarMatriz(f,c);
        int [] sumas = sumarFilasTotal(matriz);
        int total = sumarTotal(sumas);

        System.out.println("Matriz          >> Sumas por filas");
        for (int i = 0; i < f; i++) {
            for (int j = 0; j < c; j++) {
                System.out.print(matriz[i][j] + " ");
            }
            System.out.println("          >> " + sumas[i]);
        }

        /*
        System.out.println("Sumas por Filas");
        for (int xd: sumas) {
            System.out.println(">> " + xd);
        }
        */

        System.out.println("Suma total:" + total);
    }

    public static int[] [] llenarMatriz(int f, int c) {
        Random r = new Random();
        int [][] matriz = new int[f][c];
        for (int i = 0; i < f; i++) {
            for (int j = 0; j < c; j++) {
                matriz[i][j] = r.nextInt(9) + 1;
            }
        }

        return matriz;
    }

    public static int[] sumarFilasTotal(int[][] matriz) {
        int[] sumasFilas = new int[matriz.length];

        for (int i = 0; i < matriz.length; i++) {
            for (int j = 0; j < matriz[0].length; j++) {
                sumasFilas[i] += matriz[i][j];
            }
        }

        return sumasFilas;
    }

    public static int sumarTotal(int[] vector) {
        int total = 0;
    }
}
```

```

        for (int i = 0; i < vector.length; i++) {
            total += vector[i];
        }
        return total;
    }
}

```

Segunda Clase

```

package ejercicios;

import java.util.Scanner;

public class sumaFilasTotalHilos2 implements Runnable {
    private int[] row;
    private static int[] sumaTotal;
    private int index = 0;

    public sumaFilasTotalHilos2(int[] row, int index) {
        this.row = row;
        this.index = index;
    }

    public int[] getRow() {
        return row;
    }

    public static int[] getSumaTotal() {
        return sumaTotal;
    }

    @Override
    public void run() {
        int sum = 0;
        for (int num : row) {
            sum += num;
        }

        /* ESTO ES PARA SINCRONIZAR EL ACCESO AL VECTOR Y EVITAR
        INCONSISTENCIA
        synchronized (sumaFilasTotalHilos2.class) {
            sumaTotal[index] = sum;
        }
        */
        sumaTotal[index] = sum;
        System.out.printf("%s trabajando. Total: %d\n",
        Thread.currentThread().getName(), sum);
    }

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.println("Ingrese el número de filas");
        int f = sc.nextInt();
        System.out.println("Ingrese el número de columnas");
        int c = sc.nextInt();

        int[][] matriz = sumaFilasTotal.llenarMatriz(f, c);

        sumaTotal = new int[f]; // para no estar inicializando el suma total
        por cada hilo
    }
}

```

```

    // simplemente se hace que este sumaTotal sea static y se lo
    // inicializa con el tamaño de filas
    // que también será la cantidad de hilos generados
    Thread[] threads = new Thread[f];
    for (int i = 0; i < f; i++) {
        threads[i] = new Thread(new sumaFilasTotalHilos2(matriz[i], i));
    // mando el índice para evitar
    // crear un contador static e ir sumándolo
        threads[i].start();
    }

    for (int i = 0; i < f; i++) {
        try {
            threads[i].join();
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
    }

    int sT = 0;
    for (int xd : getSumaTotal()) {
        sT += xd;
    }

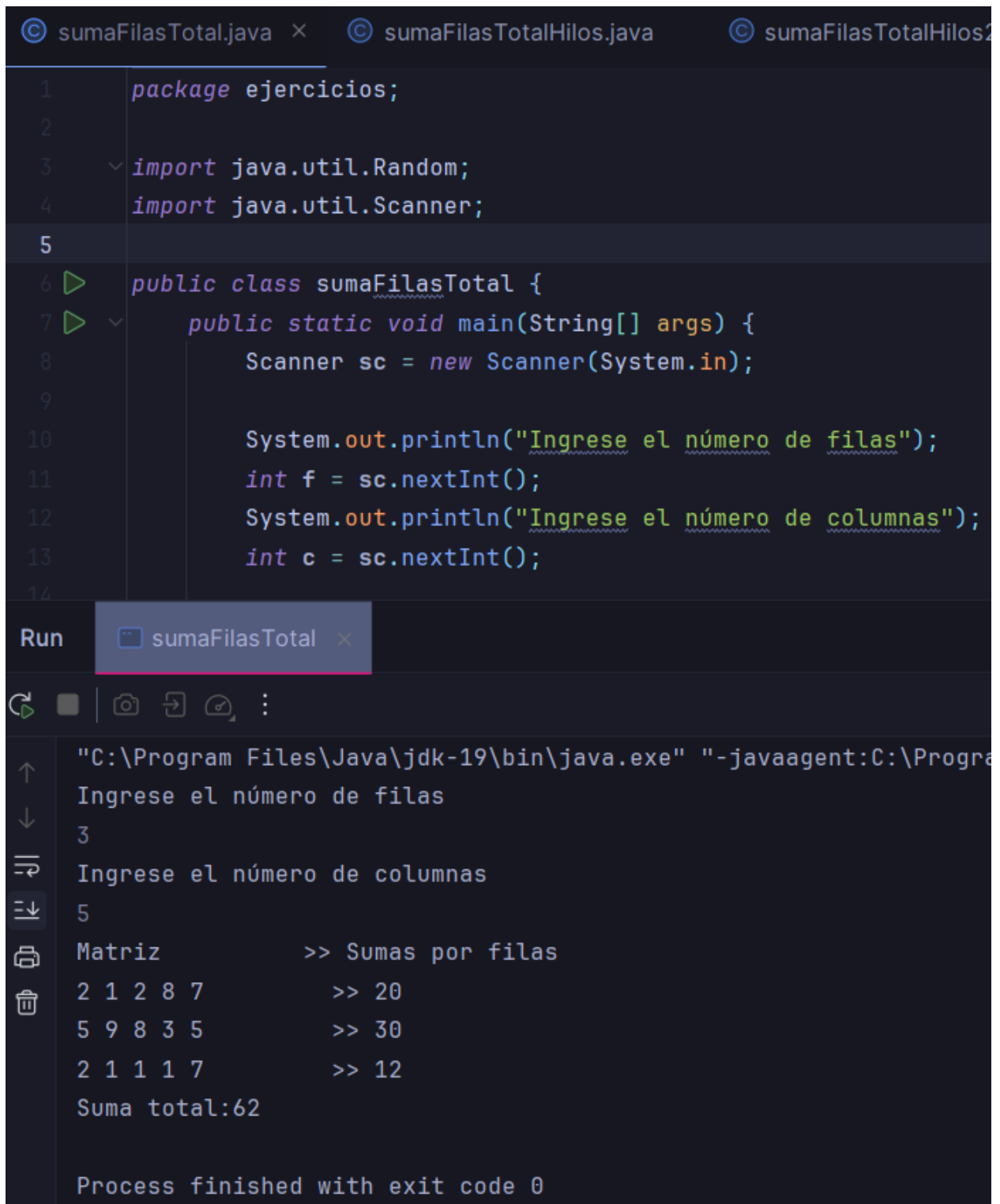
    System.out.println("Suma total: " + sT);
    presentarMatriz(matriz);
}

public static void presentarMatriz(int[][] matriz) {
    System.out.println(">>> PRESENTAR MATRIZ <<<");
    for (int i = 0; i < matriz.length; i++) {
        for (int j = 0; j < matriz[0].length; j++) {
            System.out.print(matriz[i][j] + " ");
        }
        System.out.println();
    }
}
}

```

Capturas:

Primera clase



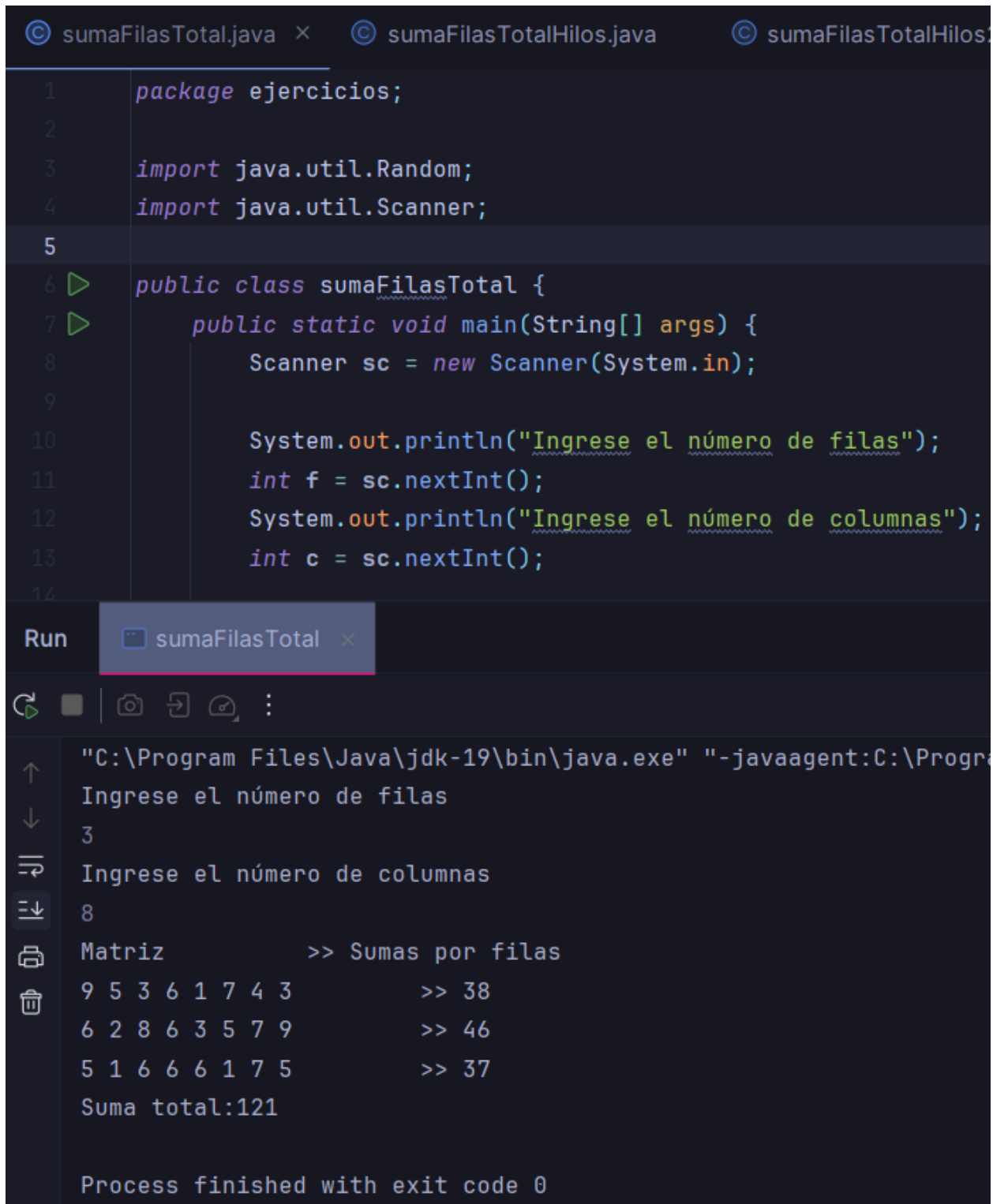
The image shows a screenshot of an IDE with three tabs: `sumaFilasTotal.java`, `sumaFilasTotalHilos.java`, and `sumaFilasTotalHilos2`. The active tab is `sumaFilasTotal.java`, which contains the following Java code:

```
1 package ejercicios;
2
3 import java.util.Random;
4 import java.util.Scanner;
5
6 public class sumaFilasTotal {
7     public static void main(String[] args) {
8         Scanner sc = new Scanner(System.in);
9
10        System.out.println("Ingrese el número de filas");
11        int f = sc.nextInt();
12        System.out.println("Ingrese el número de columnas");
13        int c = sc.nextInt();
14    }
```

Below the code editor, there is a "Run" button and a console window. The console output shows the execution of the program:

```
"C:\Program Files\Java\jdk-19\bin\java.exe" "-javaagent:C:\Progra
Ingrese el número de filas
3
Ingrese el número de columnas
5
Matriz          >> Sumas por filas
2 1 2 8 7       >> 20
5 9 8 3 5       >> 30
2 1 1 1 7       >> 12
Suma total:62

Process finished with exit code 0
```



```
sumaFilasTotal.java x sumaFilasTotalHilos.java sumaFilasTotalHilos.java
1 package ejercicios;
2
3 import java.util.Random;
4 import java.util.Scanner;
5
6 public class sumaFilasTotal {
7     public static void main(String[] args) {
8         Scanner sc = new Scanner(System.in);
9
10        System.out.println("Ingrese el número de filas");
11        int f = sc.nextInt();
12        System.out.println("Ingrese el número de columnas");
13        int c = sc.nextInt();
14    }
}

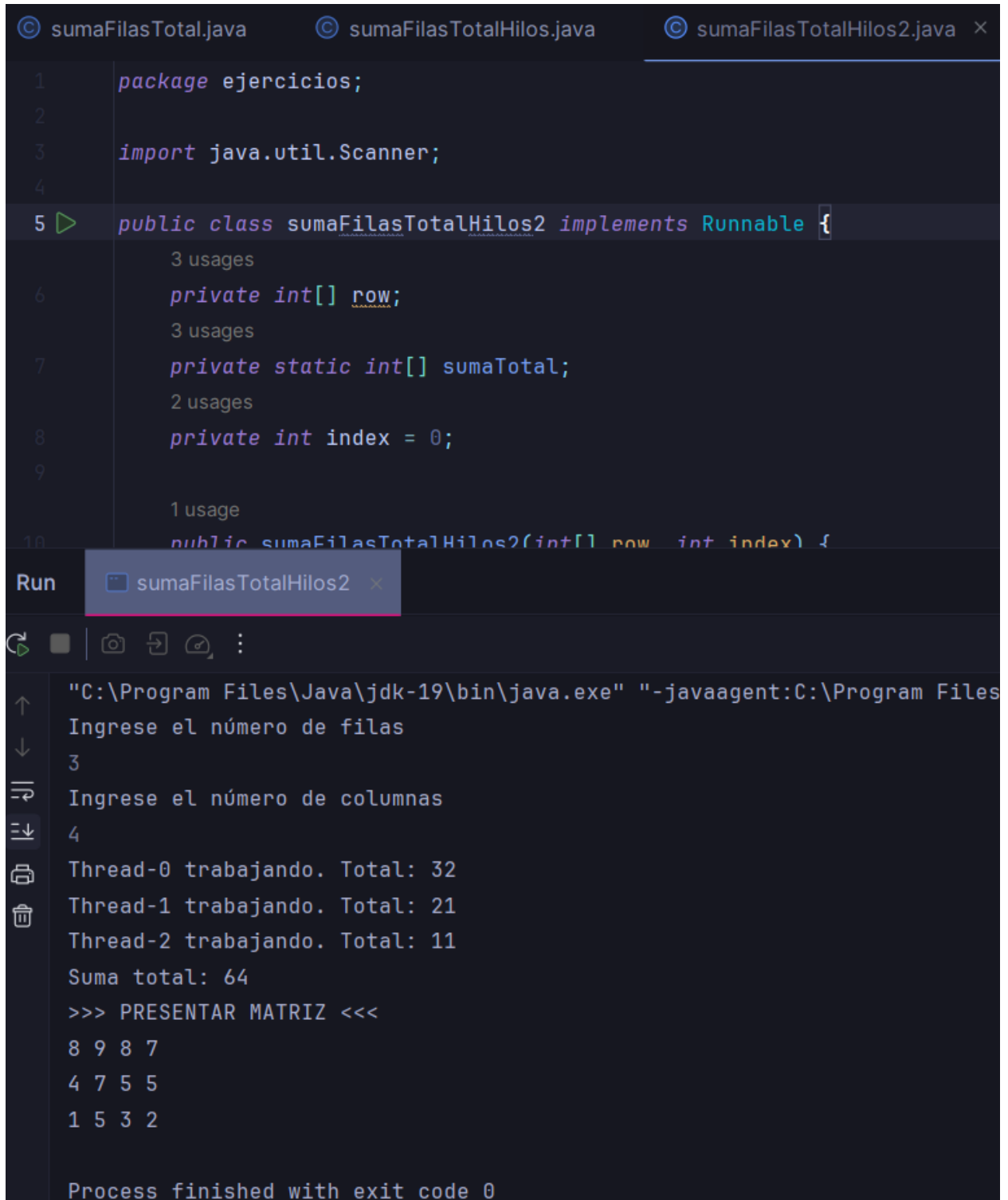
Run sumaFilasTotal x

"C:\Program Files\Java\jdk-19\bin\java.exe" "-javaagent:C:\Progra
Ingrese el número de filas
3
Ingrese el número de columnas
8
Matriz          >> Sumas por filas
9 5 3 6 1 7 4 3      >> 38
6 2 8 6 3 5 7 9      >> 46
5 1 6 6 6 1 7 5      >> 37
Suma total:121

Process finished with exit code 0
```

Segunda clase

Matriz Secuencial

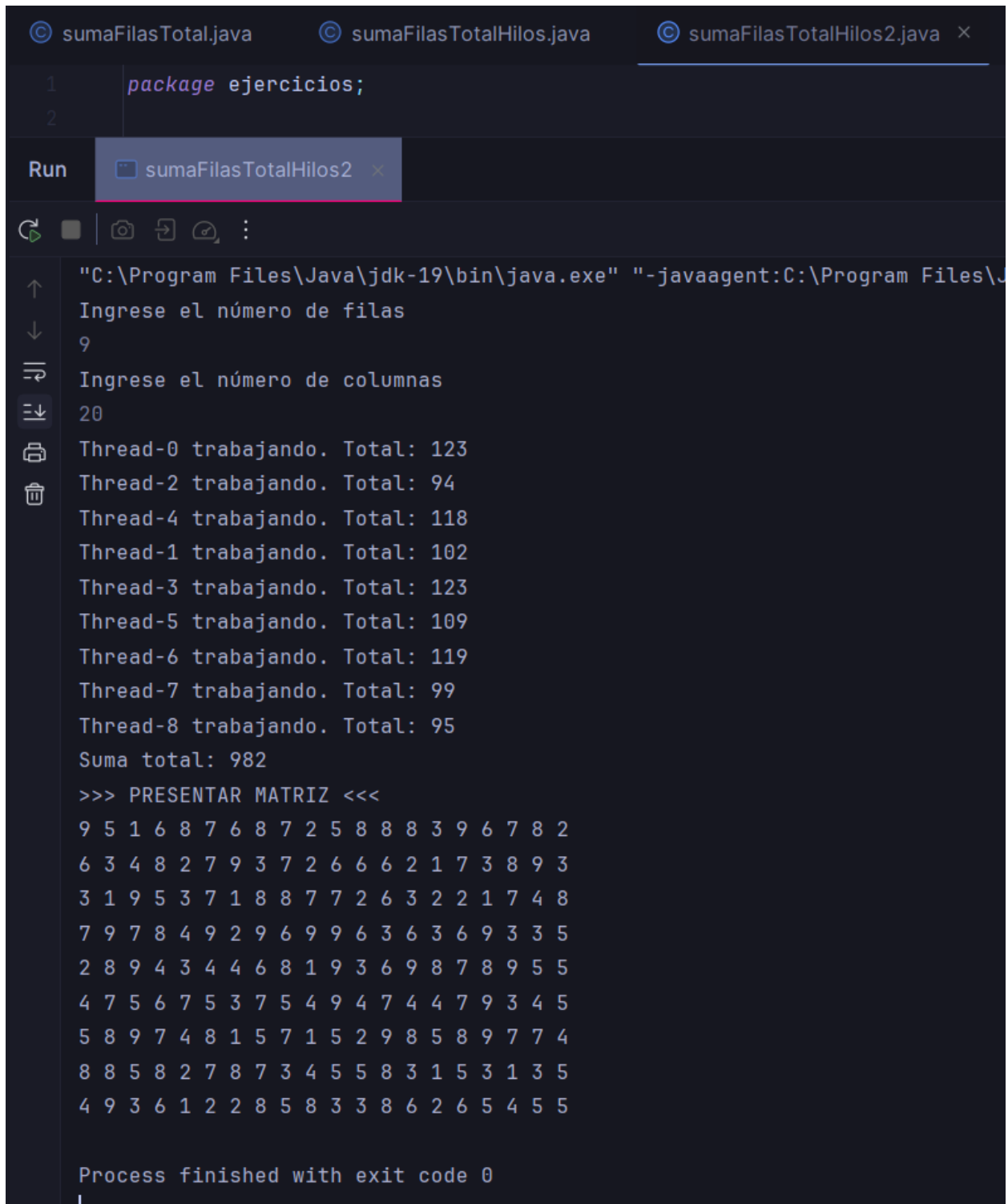


```
© sumaFilasTotal.java  © sumaFilasTotalHilos.java  © sumaFilasTotalHilos2.java x
1  package ejercicios;
2
3  import java.util.Scanner;
4
5  public class sumaFilasTotalHilos2 implements Runnable {
6      private int[] row;
7      private static int[] sumaTotal;
8      private int index = 0;
9
10     public sumaFilasTotalHilos2(int[] row, int index) {
11
12     }
13
14     public void run() {
15
16     }
17 }

Run sumaFilasTotalHilos2 x

"C:\Program Files\Java\jdk-19\bin\java.exe" "-javaagent:C:\Program Files
Ingrese el número de filas
3
Ingrese el número de columnas
4
Thread-0 trabajando. Total: 32
Thread-1 trabajando. Total: 21
Thread-2 trabajando. Total: 11
Suma total: 64
>>> PRESENTAR MATRIZ <<<
8 9 8 7
4 7 5 5
1 5 3 2

Process finished with exit code 0
```



The screenshot shows an IDE with three tabs: `sumaFilasTotal.java`, `sumaFilasTotalHilos.java`, and `sumaFilasTotalHilos2.java`. The `sumaFilasTotalHilos2.java` tab is active, showing the following code:

```
1 package ejercicios;
2
```

Below the code editor, the **Run** button is highlighted, and a dropdown menu shows `sumaFilasTotalHilos2` with a close button. The output console displays the execution results:

```
"C:\Program Files\Java\jdk-19\bin\java.exe" "-javaagent:C:\Program Files\J
Ingrese el número de filas
9
Ingrese el número de columnas
20
Thread-0 trabajando. Total: 123
Thread-2 trabajando. Total: 94
Thread-4 trabajando. Total: 118
Thread-1 trabajando. Total: 102
Thread-3 trabajando. Total: 123
Thread-5 trabajando. Total: 109
Thread-6 trabajando. Total: 119
Thread-7 trabajando. Total: 99
Thread-8 trabajando. Total: 95
Suma total: 982
>>> PRESENTAR MATRIZ <<<
9 5 1 6 8 7 6 8 7 2 5 8 8 8 3 9 6 7 8 2
6 3 4 8 2 7 9 3 7 2 6 6 6 2 1 7 3 8 9 3
3 1 9 5 3 7 1 8 8 7 7 2 6 3 2 2 1 7 4 8
7 9 7 8 4 9 2 9 6 9 9 6 3 6 3 6 9 3 3 5
2 8 9 4 3 4 4 6 8 1 9 3 6 9 8 7 8 9 5 5
4 7 5 6 7 5 3 7 5 4 9 4 7 4 4 7 9 3 4 5
5 8 9 7 4 8 1 5 7 1 5 2 9 8 5 8 9 7 7 4
8 8 5 8 2 7 8 7 3 4 5 5 8 3 1 5 3 1 3 5
4 9 3 6 1 2 2 8 5 8 3 3 8 6 2 6 5 4 5 5

Process finished with exit code 0
```