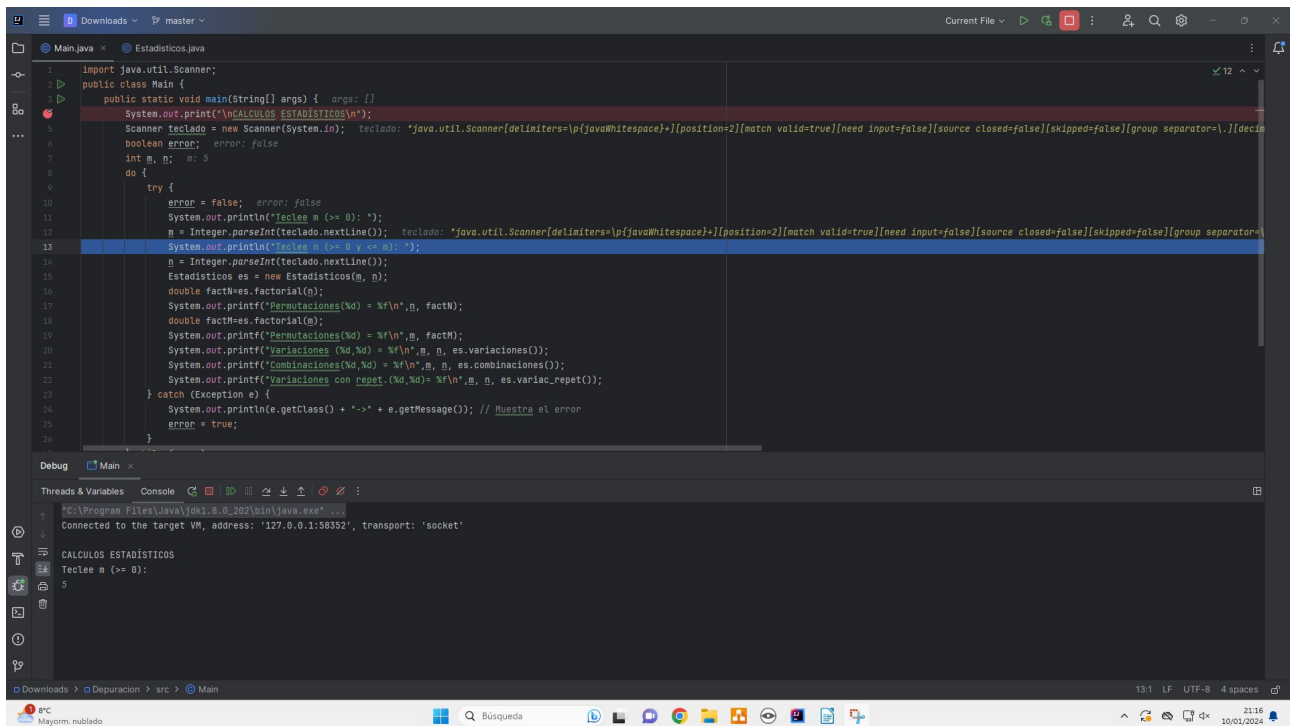


1)



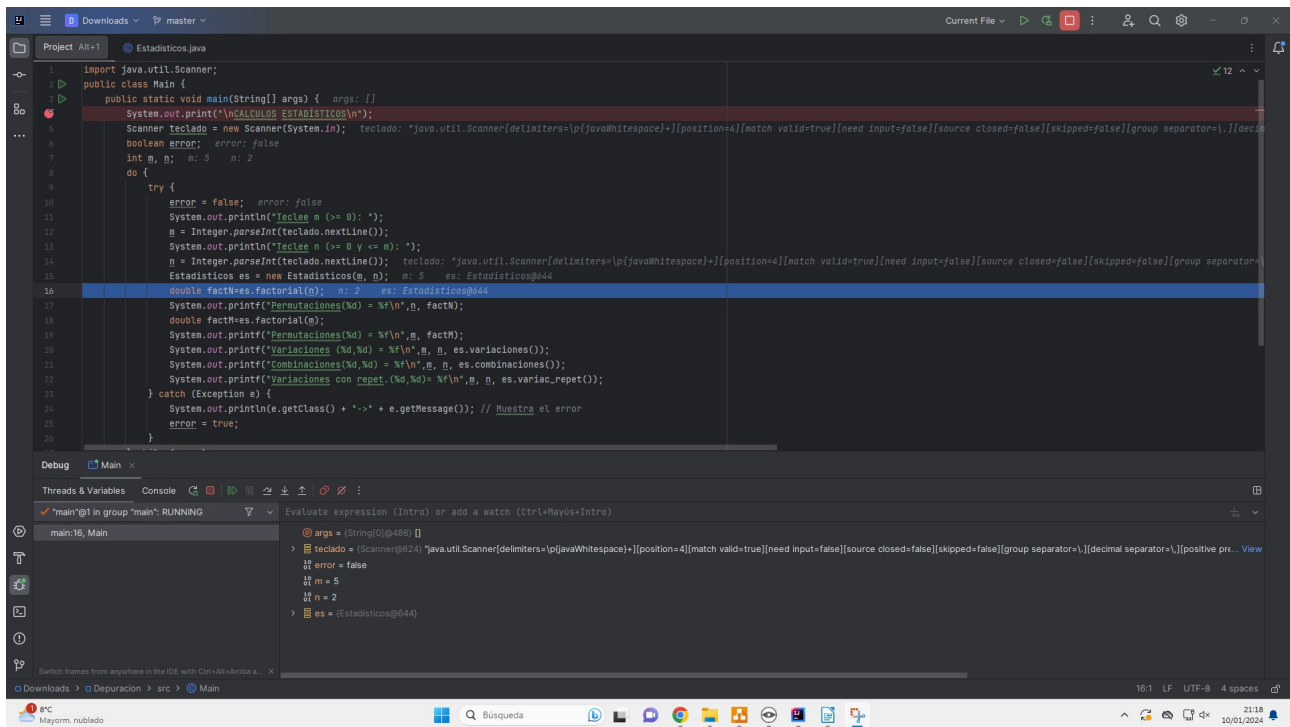
```
import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        System.out.println("CALCULOS ESTADÍSTICOS");
        Scanner teclado = new Scanner(System.in);
        boolean error = false;
        int m, n;
        do {
            try {
                error = false;
                System.out.println("Teclee m (>= 0): ");
                m = Integer.parseInt(teclado.nextLine());
                System.out.println("Teclee n (>= 0 y <= m): ");
                n = Integer.parseInt(teclado.nextLine());
                Estadisticos es = new Estadisticos(m, n);
                double factMes.factorial(n);
                System.out.printf("Permutaciones(%d) = %f\n", n, factMes);
                double factMes.factorial(m);
                System.out.printf("Permutaciones(%d) = %f\n", m, factMes);
                System.out.printf("Variaciones (%d,%d) = %f\n", m, n, es.variaciones());
                System.out.printf("Combinaciones(%d,%d) = %f\n", m, n, es.combinaciones());
                System.out.printf("Variaciones con repet. (%d,%d)= %f\n", m, n, es.variac_repet());
            } catch (Exception e) {
                System.out.println(e.getClass() + " -> " + e.getMessage()); // Muestra el error
                error = true;
            }
        } while (error);
    }
}
```

Debug Console:

```
Connected to the target VM, address: '127.0.0.1:58352', transport: 'socket'

CALCULOS ESTADÍSTICOS
Teclee m (>= 0):
5
```

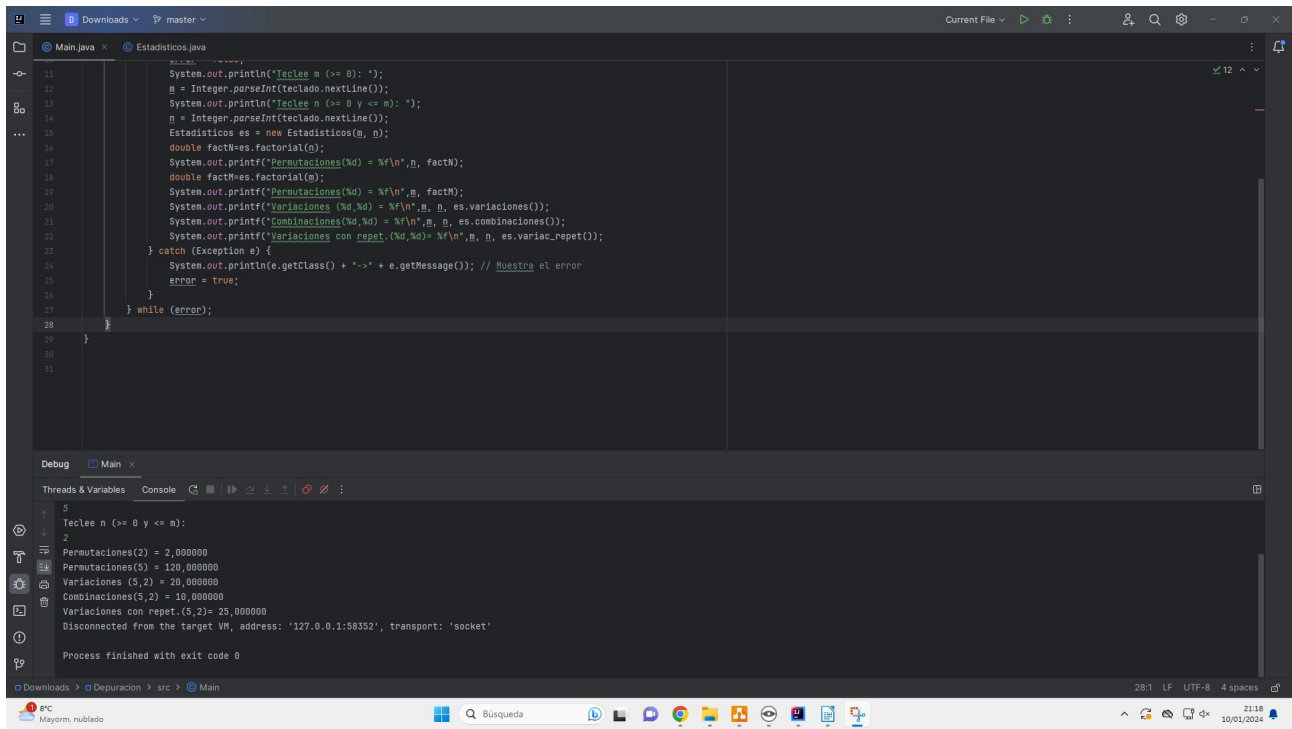


```
import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        System.out.println("CALCULOS ESTADÍSTICOS");
        Scanner teclado = new Scanner(System.in);
        boolean error = false;
        int m, n;
        do {
            try {
                error = false;
                System.out.println("Teclee m (>= 0): ");
                m = Integer.parseInt(teclado.nextLine());
                System.out.println("Teclee n (>= 0 y <= m): ");
                n = Integer.parseInt(teclado.nextLine());
                Estadisticos es = new Estadisticos(m, n);
                double factMes.factorial(n);
                System.out.printf("Permutaciones(%d) = %f\n", n, factMes);
                double factMes.factorial(m);
                System.out.printf("Permutaciones(%d) = %f\n", m, factMes);
                System.out.printf("Variaciones (%d,%d) = %f\n", m, n, es.variaciones());
                System.out.printf("Combinaciones(%d,%d) = %f\n", m, n, es.combinaciones());
                System.out.printf("Variaciones con repet. (%d,%d)= %f\n", m, n, es.variac_repet());
            } catch (Exception e) {
                System.out.println(e.getClass() + " -> " + e.getMessage()); // Muestra el error
                error = true;
            }
        } while (error);
    }
}
```

Threads & Variables:

```
main:16, Main
  @ args = (String[]@486) []
  @ teclado = (Scanner@624) "java.util.Scanner[delimiters=[p[\\javaWhitespace]+][position=4][match valid=true][need input=false][source closed=false][skipped=false][group separator=\\.][decim...
```



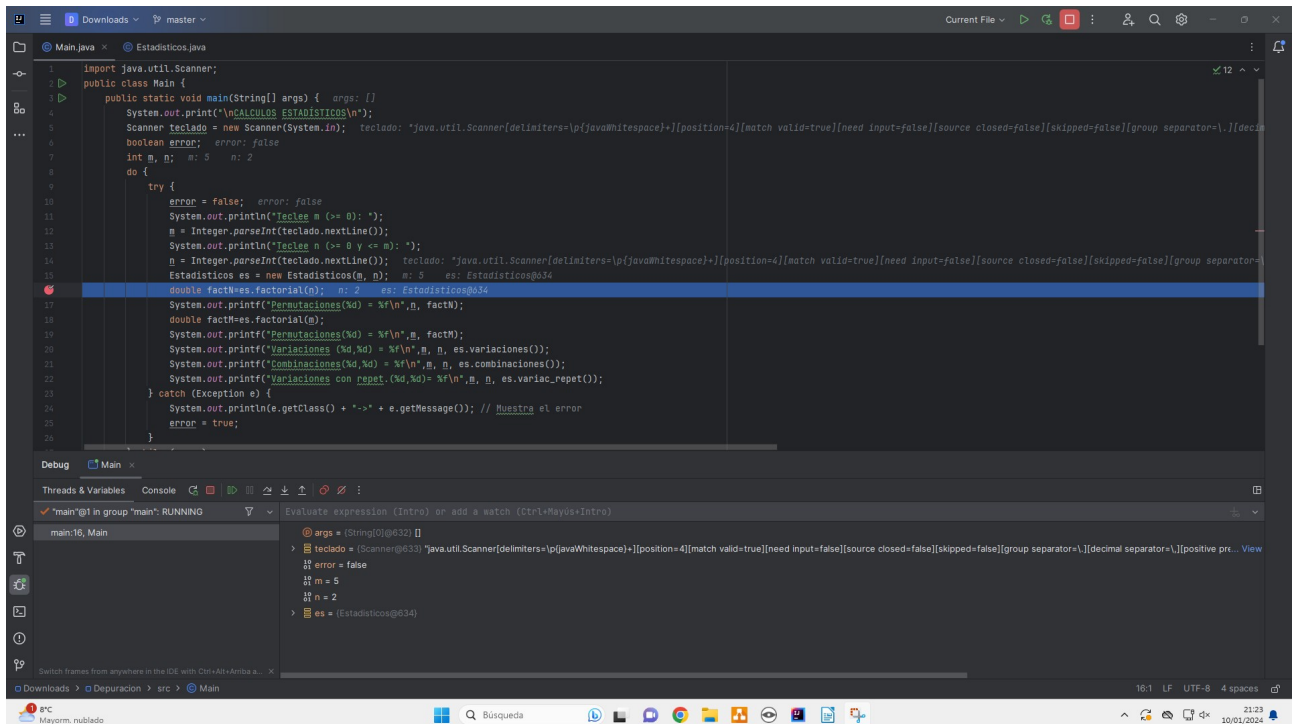
```
11 System.out.println("Teclee m (>= 0): ");
12 m = Integer.parseInt(teclado.nextLine());
13 System.out.println("Teclee n (>= 0 y <= m): ");
14 n = Integer.parseInt(teclado.nextLine());
15 Estadisticos es = new Estadisticos(m, n);
16 double factN=es.factorial(n);
17 System.out.printf("Permutaciones(%d) = %f\n",n, factN);
18 double factM=es.factorial(m);
19 System.out.printf("Permutaciones(%d) = %f\n",m, factM);
20 System.out.printf("Variaciones (%d,%d) = %f\n",m, n, es.variaciones());
21 System.out.printf("Combinaciones(%d,%d) = %f\n",m, n, es.combinaciones());
22 System.out.printf("Variaciones con repet. (%d,%d)= %f\n",m, n, es.variac_repet());
23 } catch (Exception e) {
24 System.out.println(e.getClass() + "->" + e.getMessage()); // Muestra el error
25 error = true;
26 }
27 } while (error);
28 }
29 }
30 }
31 }
```

Debug Main

Threads & Variables Console

5
Teclee m (>= 0 y <= m):
2
Permutaciones(2) = 2,000000
Permutaciones(5) = 120,000000
Variaciones (5,2) = 20,000000
Combinaciones(5,2) = 10,000000
Variaciones con repet.(5,2)= 25,000000
Disconnected from the target VM, address: '127.0.0.1:58352', transport: 'socket'
Process finished with exit code 0

2)



```
1 import java.util.Scanner;
2 public class Main {
3     public static void main(String[] args) { args= {}
4         System.out.print("\nCALCULOS ESTADISTICOS\n");
5         Scanner teclado = new Scanner(System.in); teclado: "java.util.Scanner[delimiters=lp{javaWhitespace}+][position=4][match valid=true][need input=false][source closed=false][skipped=false][group separator=\\.][decim
6         boolean error; error: false
7         int m, n; m: 5 n: 2
8         do {
9             try {
10                 error = false; error: false
11                 System.out.println("Teclee m (>= 0): ");
12                 m = Integer.parseInt(teclado.nextLine());
13                 System.out.println("Teclee n (>= 0 y <= m): ");
14                 n = Integer.parseInt(teclado.nextLine()); teclado: "java.util.Scanner[delimiters=lp{javaWhitespace}+][position=4][match valid=true][need input=false][source closed=false][skipped=false][group separator=\\.][decim
15                 Estadisticos es = new Estadisticos(m, n); m: 5 es: Estadisticos@634
16                 double factN=es.factorial(n); m: 2 es: Estadisticos@634
17                 System.out.printf("Permutaciones(%d) = %f\n",n, factN);
18                 double factM=es.factorial(m);
19                 System.out.printf("Permutaciones(%d) = %f\n",m, factM);
20                 System.out.printf("Variaciones (%d,%d) = %f\n",m, n, es.variaciones());
21                 System.out.printf("Combinaciones(%d,%d) = %f\n",m, n, es.combinaciones());
22                 System.out.printf("Variaciones con repet. (%d,%d)= %f\n",m, n, es.variac_repet());
23             } catch (Exception e) {
24                 System.out.println(e.getClass() + "->" + e.getMessage()); // Muestra el error
25                 error = true;
26             }
27         } while (error);
28     }
29 }
30 }
31 }
```

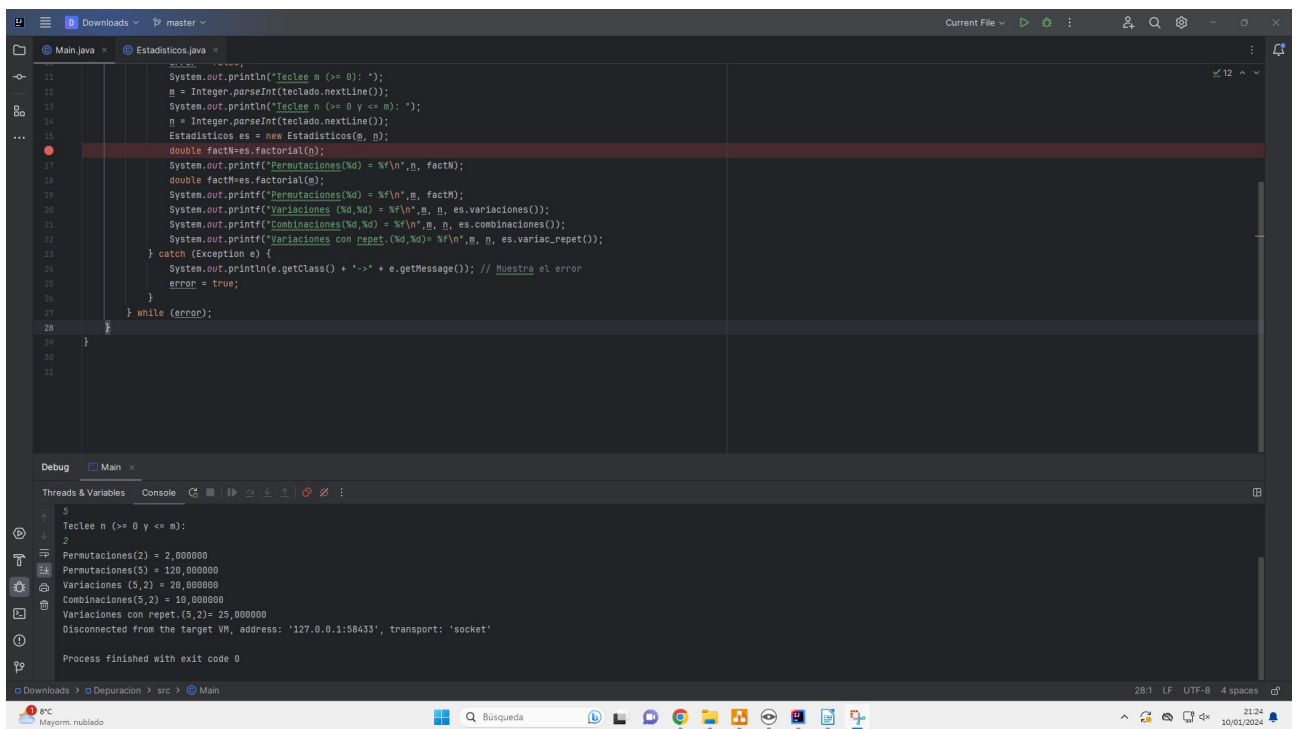
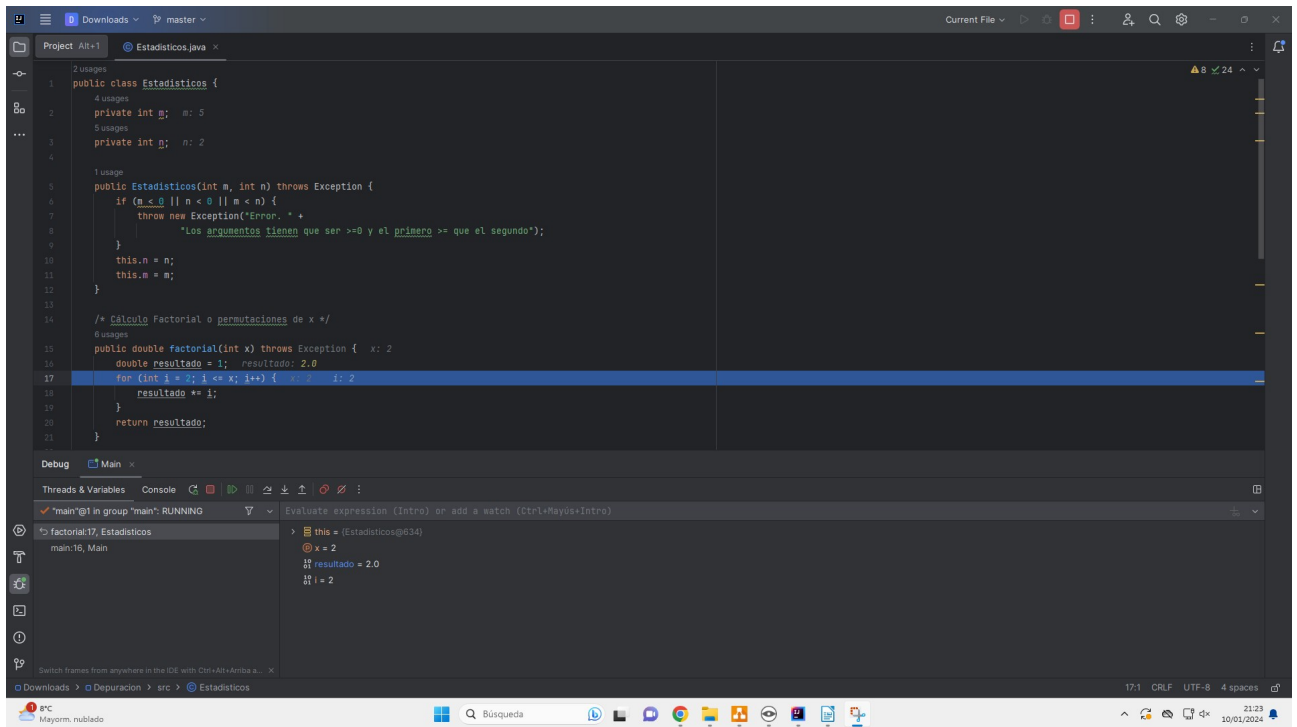
Debug Main

Threads & Variables Console

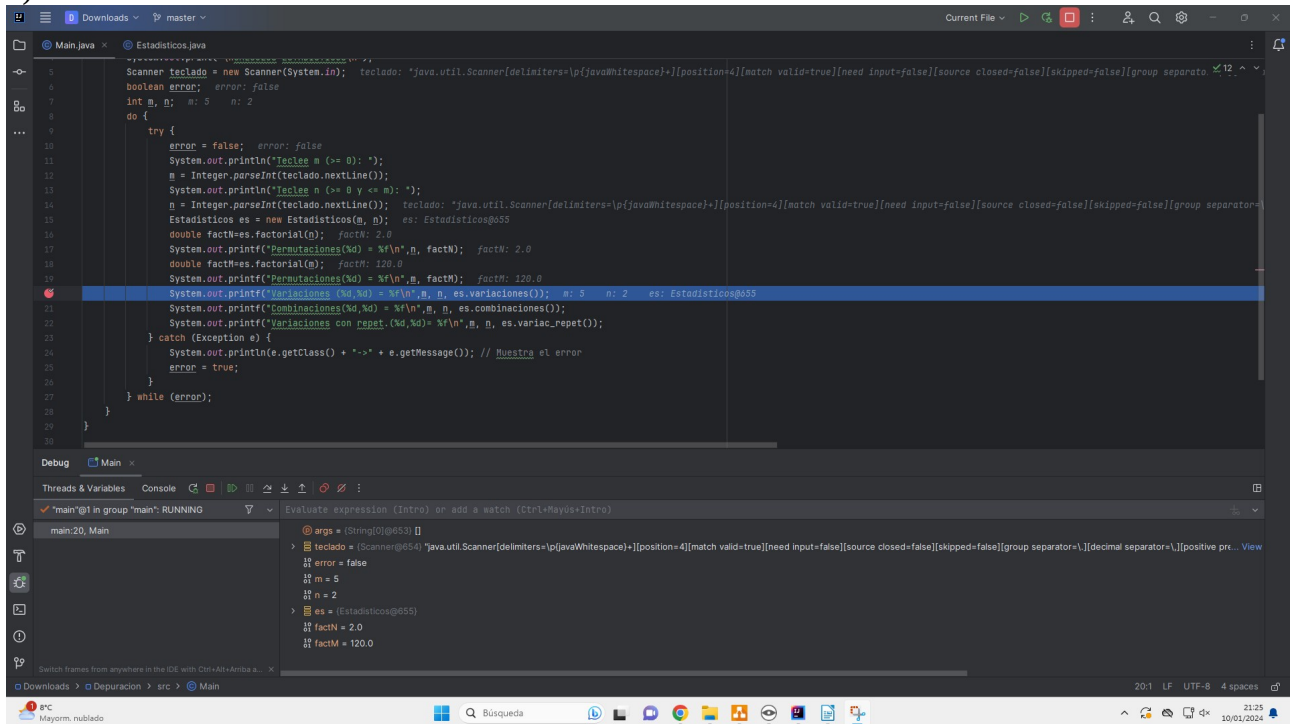
main:16, Main

main:16, Main

args = (String[])@632 []
teclado = (Scanner@633) "java.util.Scanner[delimiters=lp{javaWhitespace}+][position=4][match valid=true][need input=false][source closed=false][skipped=false][group separator=\\.][decimal separator=\\.][positive pr... View
error = false
m = 5
n = 2
es = (Estadisticos@634)

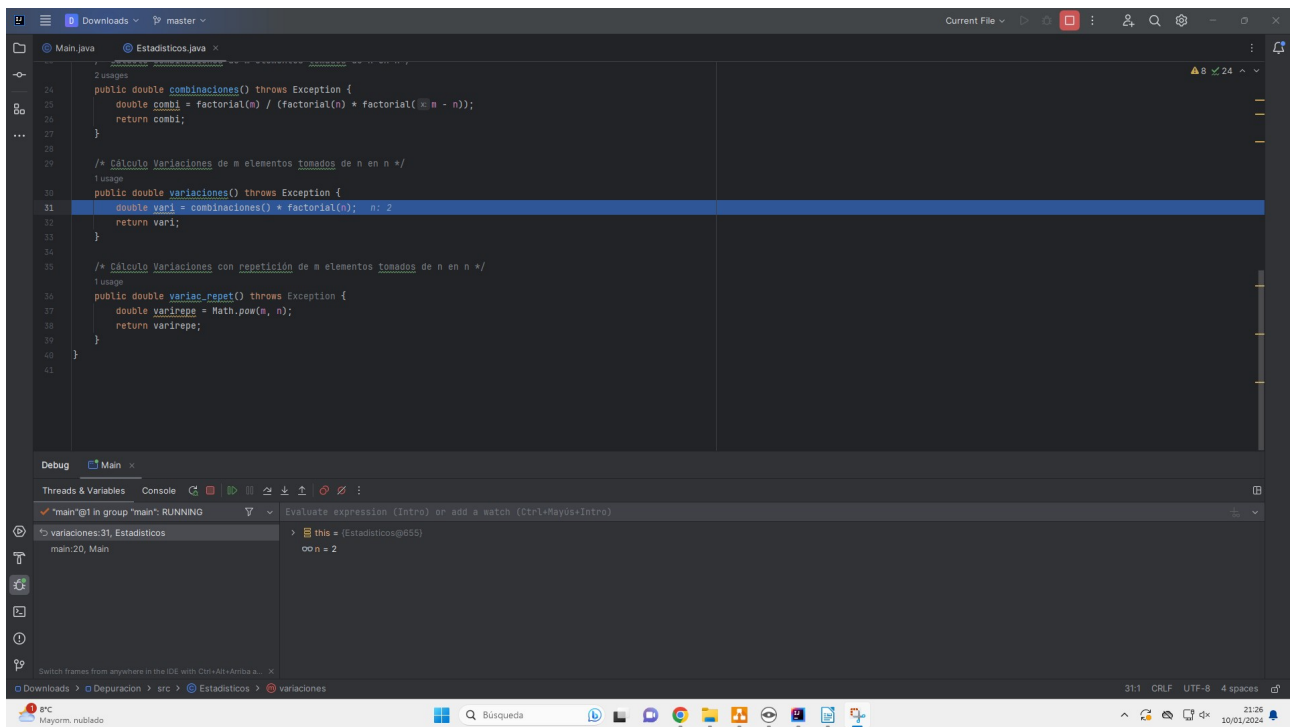


3)



The screenshot shows an IDE window with the file `Estadisticos.java` open. The code defines a `Scanner` `teclado` and a `do-while` loop that calculates factorial, permutations, combinations, and variations. The debug console at the bottom shows the current state of the program, with the `main` method running. The console output includes the values of `m` (5), `n` (2), and the calculated values for `factN` (2.0) and `factM` (120.0).

```
1 Scanner teclado = new Scanner(System.in); teclado: "java.util.Scanner[delimiters=\p{javaWhitespace}+][position=4][match valid=true][need input=false][source closed=false][skipped=false][group separator=\\n][decimal separator=\\n][positive pre-lookahead=false][unescaped backslashes=false][automatic escaping=false][current line=1][current column=1]"
2 boolean error; error: false
3 int m, n; m: 5 n: 2
4 do {
5     try {
6         error = false; error: false
7         System.out.println("Teclée m (>= 0): ");
8         m = Integer.parseInt(teclado.nextLine());
9         System.out.println("Teclée n (>= 0 y <= m): ");
10        n = Integer.parseInt(teclado.nextLine()); teclado: "java.util.Scanner[delimiters=\p{javaWhitespace}+][position=4][match valid=true][need input=false][source closed=false][skipped=false][group separator=\\n][decimal separator=\\n][positive pre-lookahead=false][unescaped backslashes=false][automatic escaping=false][current line=1][current column=1]"
11        Estadisticos es = new Estadisticos(m, n); es: Estadisticos@655
12        double factNes.factorial(n); factN: 2.0
13        System.out.printf("Permutaciones(%d) = %f\n", n, factN); factN: 2.0
14        double factMes.factorial(m); factM: 120.0
15        System.out.printf("Permutaciones(%d) = %f\n", m, factM); factM: 120.0
16        System.out.printf("Variaciones (%d,%d) = %f\n", m, n, es.variaciones()); m: 5 n: 2 es: Estadisticos@655
17        System.out.printf("Combinaciones(%d,%d) = %f\n", m, n, es.combinaciones());
18        System.out.printf("Variaciones con repet. (%d,%d)= %f\n", m, n, es.variac_repet());
19    } catch (Exception e) {
20        System.out.println(e.getClass() + "->" + e.getMessage()); // Muestra el error
21        error = true;
22    } while (error);
23 }
```



The screenshot shows the same IDE window, but now the `combinaciones` method is highlighted. The debug console shows the current state of the program, with the `combinaciones` method being executed. The console output includes the values of `m` (5), `n` (2), and the calculated value for `combi` (10.0).

```
24 public double combinaciones() throws Exception {
25     double combi = factorial(m) / (factorial(n) * factorial(m - n));
26     return combi;
27 }
28
29 /* Cálculo Variaciones de m elementos tomados de n en n */
30 public double variaciones() throws Exception {
31     double vari = combinaciones() * factorial(n); n: 2
32     return vari;
33 }
34
35 /* Cálculo Variaciones con repetición de m elementos tomados de n en n */
36 public double variac_repet() throws Exception {
37     double variac_repe = Math.pow(n, n);
38     return variac_repe;
39 }
40
41 }
```

```
1 Scanner teclado = new Scanner(System.in);
2 boolean error;
3 int m, n;
4 do {
5     try {
6         error = false;
7         System.out.println("Teclee m (>= 0): ");
8         m = Integer.parseInt(teclado.nextLine());
9         System.out.println("Teclee n (>= 0 y <= m): ");
10        n = Integer.parseInt(teclado.nextLine());
11        Estadisticos es = new Estadisticos(m, n);
12        double factMes.factorial(n);
13        System.out.printf("Permutaciones(%d) = %f\n", n, factM);
14        double factMes.factorial(m);
15        System.out.printf("Permutaciones(%d) = %f\n", m, factM);
16        System.out.printf("Variaciones (%d,%d) = %f\n", m, n, es.variaciones());
17        System.out.printf("Combinaciones(%d,%d) = %f\n", m, n, es.combinaciones());
18        System.out.printf("Variaciones con repet. (%d,%d)= %f\n", m, n, es.variac_repet());
19    } catch (Exception e) {
20        System.out.println(e.getClass() + "->" + e.getMessage()); // Muestra el error
21        error = true;
22    } while (error);
23 }
24 }
```

Debug Main

Threads & Variables Console

5
Teclee m (>= 0 y <= m):
2
Permutaciones(2) = 2,000000
Permutaciones(5) = 120,000000
Variaciones (5,2) = 20,000000
Combinaciones(5,2) = 10,000000
Variaciones con repet. (5,2)= 25,000000
Disconnected from the target VM, address: '127.0.0.1:58439', transport: 'socket'
Process finished with exit code 0

4)

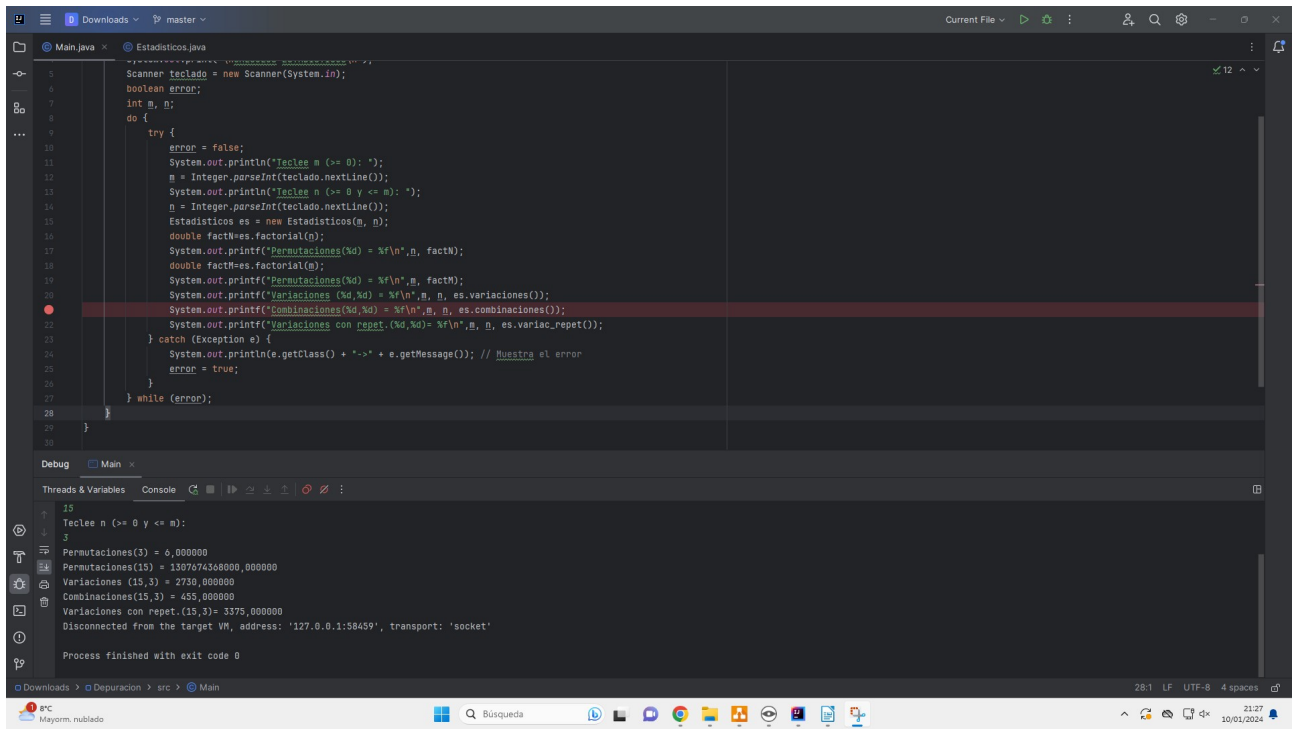
```
1 Scanner teclado = new Scanner(System.in); teclado = "java.util.Scanner[delimiters=[p(javaWhitespace)+][position=5][match valid=true][need input=false][source closed=false][skipped=false][group separator=
2 boolean error; error: false
3 int m, n; m: 15 n: 3
4 do {
5     try {
6         error = false; error: false
7         System.out.println("Teclee m (>= 0): ");
8         m = Integer.parseInt(teclado.nextLine());
9         System.out.println("Teclee n (>= 0 y <= m): ");
10        n = Integer.parseInt(teclado.nextLine()); teclado = "java.util.Scanner[delimiters=[p(javaWhitespace)+][position=5][match valid=true][need input=false][source closed=false][skipped=false][group separator=
11        Estadisticos es = new Estadisticos(m, n); es: Estadisticos@655
12        double factMes.factorial(n); factM: 0.0
13        System.out.printf("Permutaciones(%d) = %f\n", n, factM); factM: 0.0
14        double factMes.factorial(m); factM: 1.307674368E12
15        System.out.printf("Permutaciones(%d) = %f\n", m, factM); factM: 1.307674368E12
16        System.out.printf("Variaciones (%d,%d) = %f\n", m, n, es.variaciones());
17        System.out.printf("Combinaciones(%d,%d) = %f\n", m, n, es.combinaciones()); m: 15 n: 3 es: Estadisticos@655
18        System.out.printf("Variaciones con repet. (%d,%d)= %f\n", m, n, es.variac_repet());
19    } catch (Exception e) {
20        System.out.println(e.getClass() + "->" + e.getMessage()); // Muestra el error
21        error = true;
22    } while (error);
23 }
24 }
```

Debug Main

Threads & Variables Console

main[21]: Main

args = (String[]@653) []
teclado = (Scanner@654) "java.util.Scanner[delimiters=[p(javaWhitespace)+][position=5][match valid=true][need input=false][source closed=false][skipped=false][group separator=\\][decimal separator=\\][positive pr... View
error = false
m = 15
n = 3
es = (Estadisticos@655)
factM = 0.0
factM = 1.307674368E12



```
1 Scanner teclado = new Scanner(System.in);
2 boolean error;
3 int m, n;
4 do {
5     try {
6         error = false;
7         System.out.println("Telee m (>= 0): ");
8         m = Integer.parseInt(teclado.nextLine());
9         System.out.println("Telee n (>= 0 y <= m): ");
10        n = Integer.parseInt(teclado.nextLine());
11        Estadisticos es = new Estadisticos(m, n);
12        double factMes.factorial(n);
13        System.out.printf("Permutaciones(%d) = %f\n", n, factM);
14        double factMes.factorial(m);
15        System.out.printf("Permutaciones(%d) = %f\n", m, factM);
16        System.out.printf("Variaciones (%d,%d) = %f\n", m, n, es.variaciones());
17        System.out.printf("Combinaciones (%d,%d) = %f\n", m, n, es.combinaciones());
18        System.out.printf("Variaciones con repet. (%d,%d)= %f\n", m, n, es.variac_repet());
19    } catch (Exception e) {
20        System.out.println(e.getClass() + "->" + e.getMessage()); // Muestra el error
21        error = true;
22    } while (error);
23 }
24 }
```

Debug Main

Threads & Variables Console

```
25 Telee n (>= 0 y <= m):
3
Permutaciones(3) = 6.000000
Permutaciones(15) = 1307674368000.000000
Variaciones (15,3) = 2730.000000
Combinaciones(15,3) = 455.000000
Variaciones con repet. (15,3)= 3375.000000
Disconnected from the target VM, address: '127.0.0.1:58459', transport: 'socket'
Process finished with exit code 0
```

Andrés Amado Cibreiro 1ºDAM