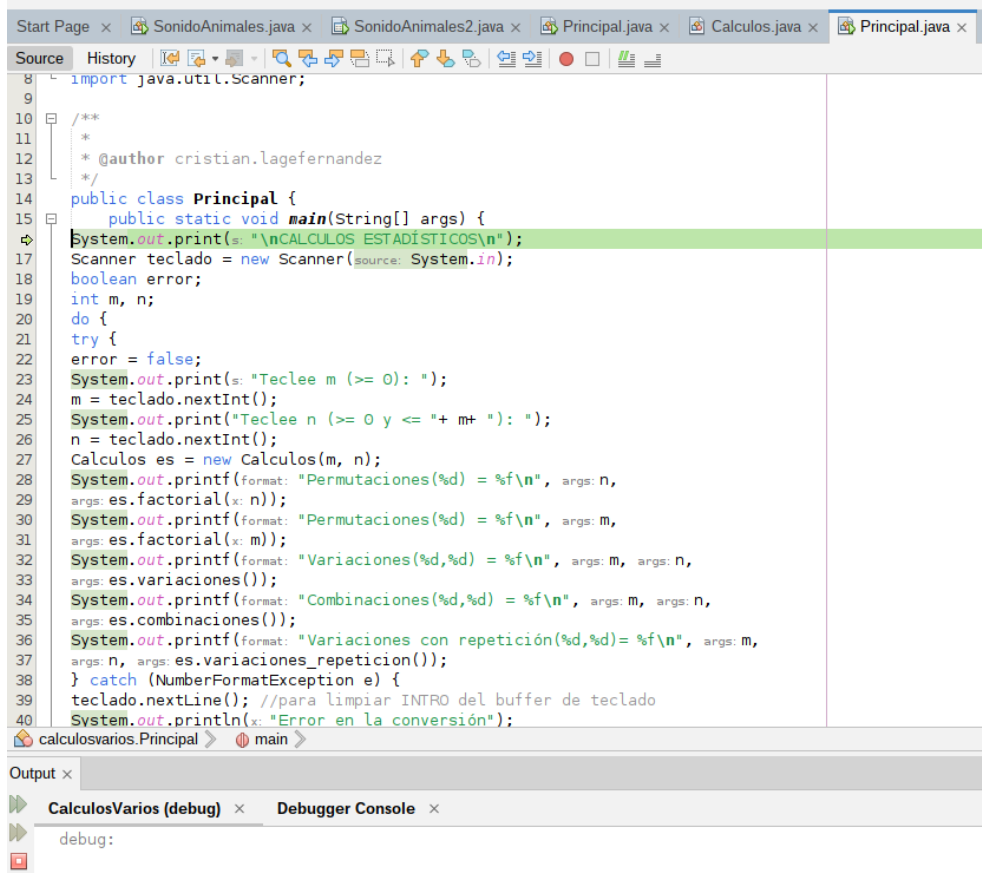


Cristian Lage Fernández



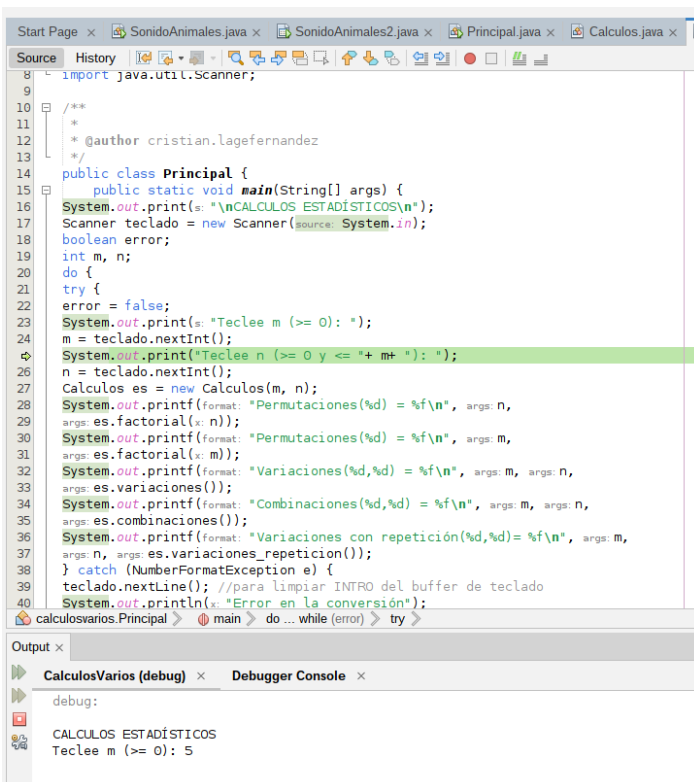
```
8  import java.util.Scanner;
9
10 /**
11  *
12  * @author cristian.lagefernandez
13  */
14 public class Principal {
15     public static void main(String[] args) {
16         System.out.print(s: "\nCALCULOS ESTADÍSTICOS\n");
17         Scanner teclado = new Scanner(source: System.in);
18         boolean error;
19         int m, n;
20         do {
21             try {
22                 error = false;
23                 System.out.print(s: "Teclee m (>= 0): ");
24                 m = teclado.nextInt();
25                 System.out.print("Teclee n (>= 0 y <= "+ m+ "): ");
26                 n = teclado.nextInt();
27                 Calculos es = new Calculos(m, n);
28                 System.out.printf(format: "Permutaciones(%d) = %f\n", args: n,
29                 args: es.factorial(x: n));
30                 System.out.printf(format: "Permutaciones(%d) = %f\n", args: m,
31                 args: es.factorial(x: m));
32                 System.out.printf(format: "Variaciones(%d,%d) = %f\n", args: m, args: n,
33                 args: es.variaciones());
34                 System.out.printf(format: "Combinaciones(%d,%d) = %f\n", args: m, args: n,
35                 args: es.combinaciones());
36                 System.out.printf(format: "Variaciones con repetición(%d,%d)= %f\n", args: m,
37                 args: n, args: es.variaciones_repeticion());
38             } catch (NumberFormatException e) {
39                 teclado.nextLine(); //para limpiar INTRO del buffer de teclado
40                 System.out.println(x: "Error en la conversión");
41             }
42         } while (error);
43     }
44 }
```

calculosvarios.Principal > main >

Output x

CalculosVarios (debug) x Debugger Console x

debug:



```
8  import java.util.Scanner;
9
10 /**
11  *
12  * @author cristian.lagefernandez
13  */
14 public class Principal {
15     public static void main(String[] args) {
16         System.out.print(s: "\nCALCULOS ESTADÍSTICOS\n");
17         Scanner teclado = new Scanner(source: System.in);
18         boolean error;
19         int m, n;
20         do {
21             try {
22                 error = false;
23                 System.out.print(s: "Teclee m (>= 0): ");
24                 m = teclado.nextInt();
25                 System.out.print("Teclee n (>= 0 y <= "+ m+ "): ");
26                 n = teclado.nextInt();
27                 Calculos es = new Calculos(m, n);
28                 System.out.printf(format: "Permutaciones(%d) = %f\n", args: n,
29                 args: es.factorial(x: n));
30                 System.out.printf(format: "Permutaciones(%d) = %f\n", args: m,
31                 args: es.factorial(x: m));
32                 System.out.printf(format: "Variaciones(%d,%d) = %f\n", args: m, args: n,
33                 args: es.variaciones());
34                 System.out.printf(format: "Combinaciones(%d,%d) = %f\n", args: m, args: n,
35                 args: es.combinaciones());
36                 System.out.printf(format: "Variaciones con repetición(%d,%d)= %f\n", args: m,
37                 args: n, args: es.variaciones_repeticion());
38             } catch (NumberFormatException e) {
39                 teclado.nextLine(); //para limpiar INTRO del buffer de teclado
40                 System.out.println(x: "Error en la conversión");
41             }
42         } while (error);
43     }
44 }
```

calculosvarios.Principal > main > do ... while (error) > try >

Output x

CalculosVarios (debug) x Debugger Console x

debug:

CALCULOS ESTADÍSTICOS
Teclee m (>= 0): 5

Start Page x SonidoAnimales.java x SonidoAnimales2.java x Principal.java x Calculos.java x Principal.java x

Source History

```
8 import java.util.Scanner;
9
10 /**
11  *
12  * @author cristian.lagefernandez
13  */
14 public class Principal {
15     public static void main(String[] args) {
16         System.out.print(s: "\nCALCULOS ESTADÍSTICOS\n");
17         Scanner teclado = new Scanner(source: System.in);
18         boolean error;
19         int m, n;
20         do {
21             try {
22                 error = false;
23                 System.out.print(s: "Teclee m (>= 0): ");
24                 m = teclado.nextInt();
25                 System.out.print("Teclee n (>= 0 y <= "+ m+ "): ");
26                 n = teclado.nextInt();
27                 Calculos es = new Calculos(m, n);
28                 System.out.printf(format: "Permutaciones(%d) = %f\n", args: n,
29                 args: es.factorial(x: n));
30                 System.out.printf(format: "Permutaciones(%d) = %f\n", args: m,
31                 args: es.factorial(x: m));
32                 System.out.printf(format: "Variaciones(%d,%d) = %f\n", args: m, args: n,
33                 args: es.variaciones());
34                 System.out.printf(format: "Combinaciones(%d,%d) = %f\n", args: m, args: n,
35                 args: es.combinaciones());
36                 System.out.printf(format: "Variaciones con repetición(%d,%d)= %f\n", args: m,
37                 args: n, args: es.variaciones_repeticion());
38             } catch (NumberFormatException e) {
39                 teclado.nextLine(); //para limpiar INTRO del buffer de teclado
40                 System.out.println(x: "Error en la conversión");
41             }
42         } while (error);
43     }
44 }
```

calculosvarios.Principal > main > do ... while (error) > try > es >

Output x

CalculosVarios (debug) x Debugger Console x

debug:

```
CALCULOS ESTADÍSTICOS
Teclee m (>= 0): 5
Teclee n (>= 0 y <= 5): 2
```

Start Page x SonidoAnimales.java x SonidoAnimales2.java x Principal.java x Calculos.java x

Source History

```
1 /**
2  * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to
3  * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this
4  */
5 package calculosvarios;
6
7 /**
8  *
9  * @author cristian.lagefernandez
10  */
11 public class Calculos {
12     private int m;
13     private int n;
14     public Calculos(int m, int n) throws Exception {
15         if (m < 0 || n < 0 || m < n) {
16             throw new Exception("Error. " +
17             "Los argumentos tienen que ser >=0 y el primero >= que el segundo");
18         }
19         this.n = n;
20         this.m = m;
21     }
22     /* Cálculo Factorial o permutaciones de x */
23     public double factorial(int x) throws Exception {
24         double resultado = 1;
25         for (int i = 2; i <= x; i++) {
26             resultado *= i;
27         }
28         return resultado;
29     }
30     /* Cálculo Combinaciones de m elementos tomados de n en n*/
31     public double combinaciones() throws Exception {
32         double combi = factorial(x: m)/(factorial(x: n)*factorial(m-n));
33         return combi;
34     }
35 }
```

calculosvarios.Calculos > factorial > resultado >

Output x

CalculosVarios (debug) x Debugger Console x

debug:

```
CALCULOS ESTADÍSTICOS
Teclee m (>= 0): 5
Teclee n (>= 0 y <= 5): 2
Permutaciones(2) = 2,000000
```

```
...ge SonidoAnimales.java x SonidoAnimales2.java x Principal.java x Calculos.java x Principal.java x
Source History
6
7 import java.util.InputMismatchException;
8 import java.util.Scanner;
9
10 /**
11  *
12  * @author cristian.lagefernandez
13  */
14 public class Principal {
15     public static void main(String[] args) {
16         System.out.print(s: "\nCALCULOS ESTADÍSTICOS\n");
17         Scanner teclado = new Scanner(System.in);
18         boolean error;
19         int m, n;
20         do {
21             try {
22                 error = false;
23                 System.out.print(s: "Teclee m (>= 0): ");
24                 m = teclado.nextInt();
25                 System.out.print("Teclee n (>= 0 y <= m+ 1): ");
26                 n = teclado.nextInt();
27                 Calculos es = new Calculos(m, n);
28                 System.out.printf(format: "Permutaciones(%d) = %f\n", args: n,
29                     args: es.factorial(x: n));
30                 System.out.printf(format: "Permutaciones(%d) = %f\n", args: m,
31                     args: es.factorial(x: m));
32                 System.out.printf(format: "Variaciones(%d,%d) = %f\n", args: m, args: n,
33                     args: es.variaciones());
34                 System.out.printf(format: "Combinaciones(%d,%d) = %f\n", args: m, args: n,
35                     args: es.combinaciones());
36                 System.out.printf(format: "Variaciones con repetición(%d,%d) = %f\n", args: m,
37                     args: n, args: es.variaciones_repeticion());
38             } catch (NumberFormatException e) {
39                 calculosvarios.Principal > main >
40             }
41         } while (error);
42     }
43 }
44
45 Output x
46 CalculosVarios (debug) x Debugger Console x
47 debug:
48
49 CALCULOS ESTADÍSTICOS
50 Teclee m (>= 0): 5
51 Teclee n (>= 0 y <= 5): 2
52 Permutaciones(2) = 2,000000
53 Permutaciones(5) = 120,000000
54 Variaciones(5,2) = 20,000000
55 Combinaciones(5,2) = 10,000000
56 Variaciones con repetición(5,2)= 25,000000
57 BUILD SUCCESSFUL (total time: 6 minutes 7 seconds)
```