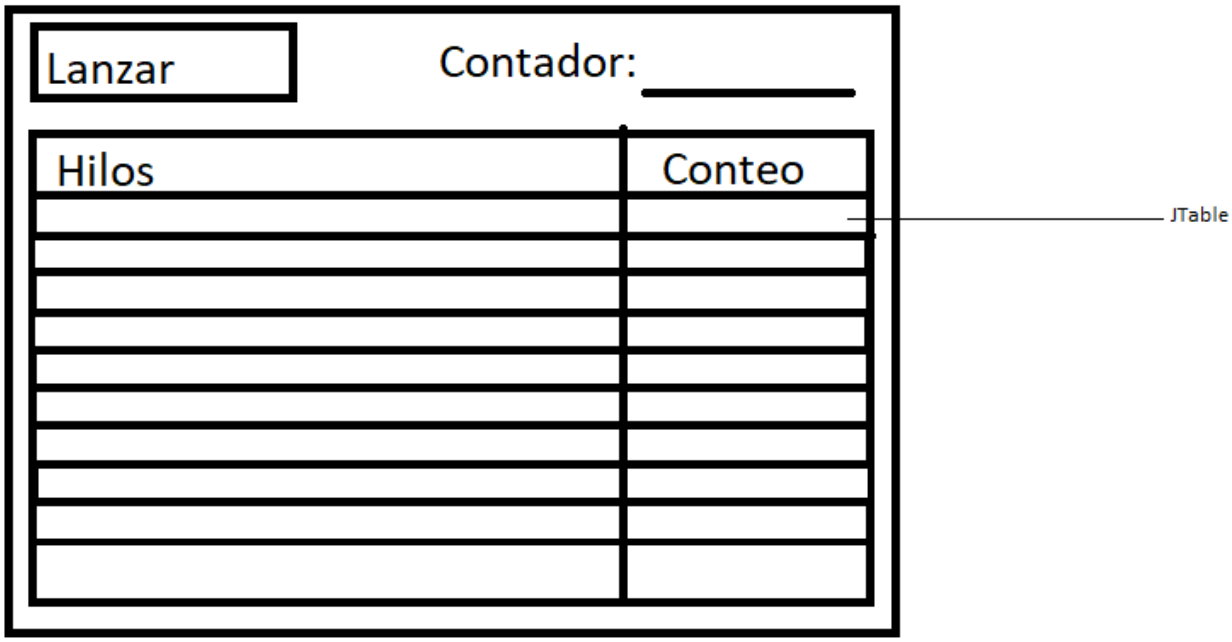
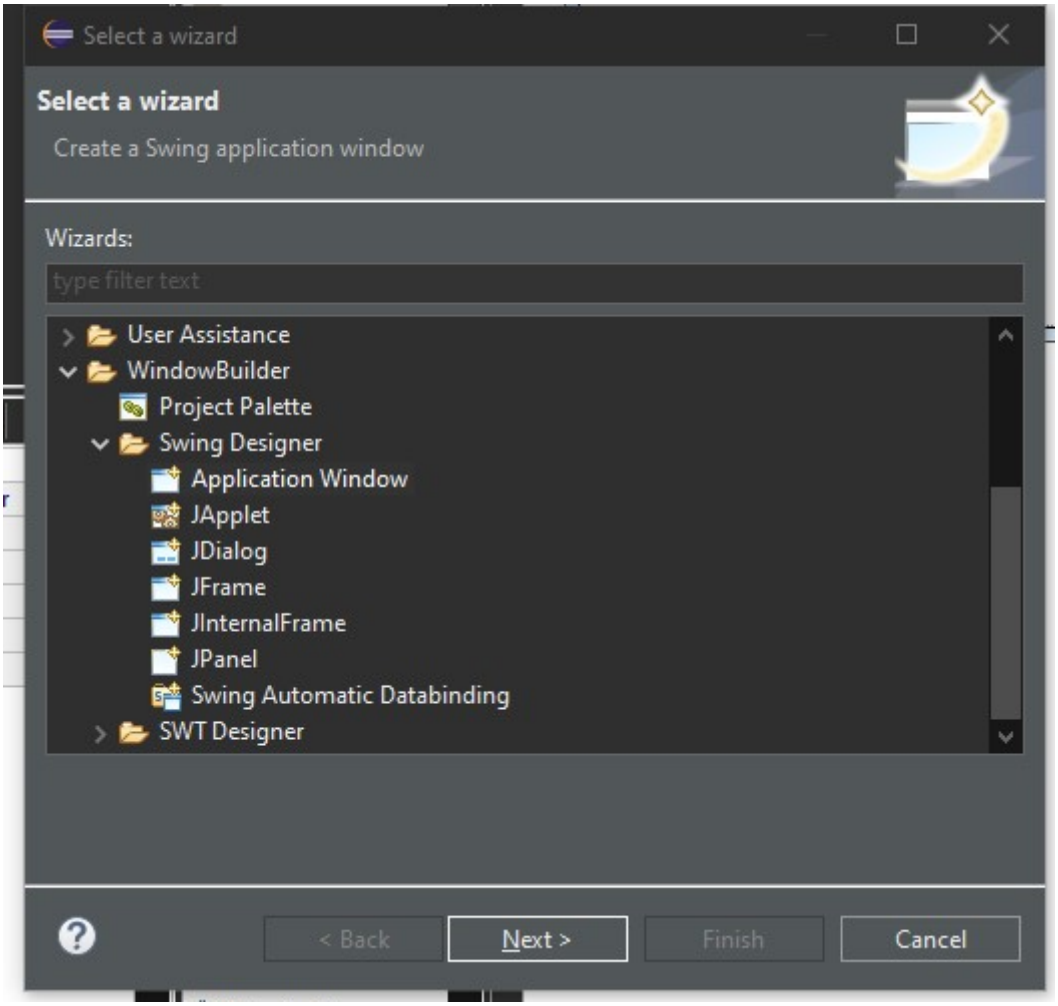
JTable

Vamos a implementar una JTable que muestre los hilos lanzados con un contador de iteraciones por Hilo y otro contador de iteraciones totales:

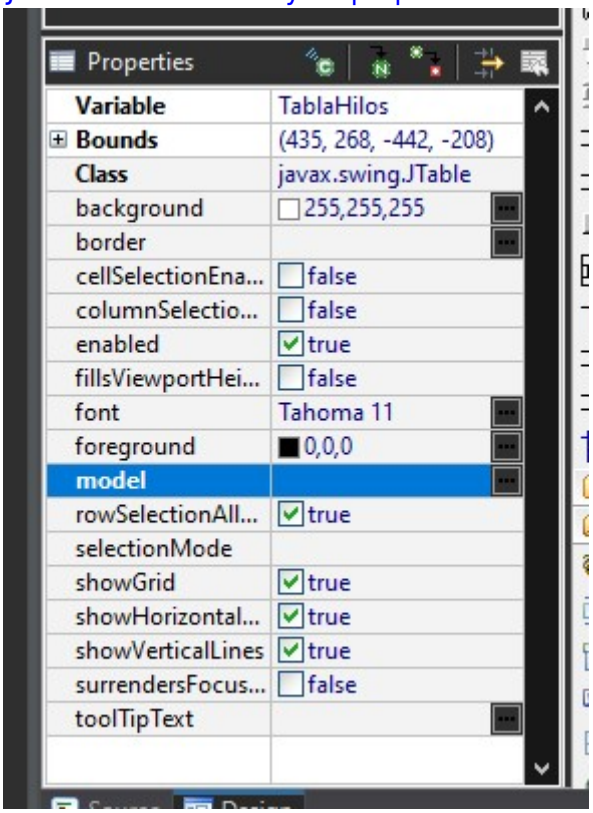


Implementación de la GUI

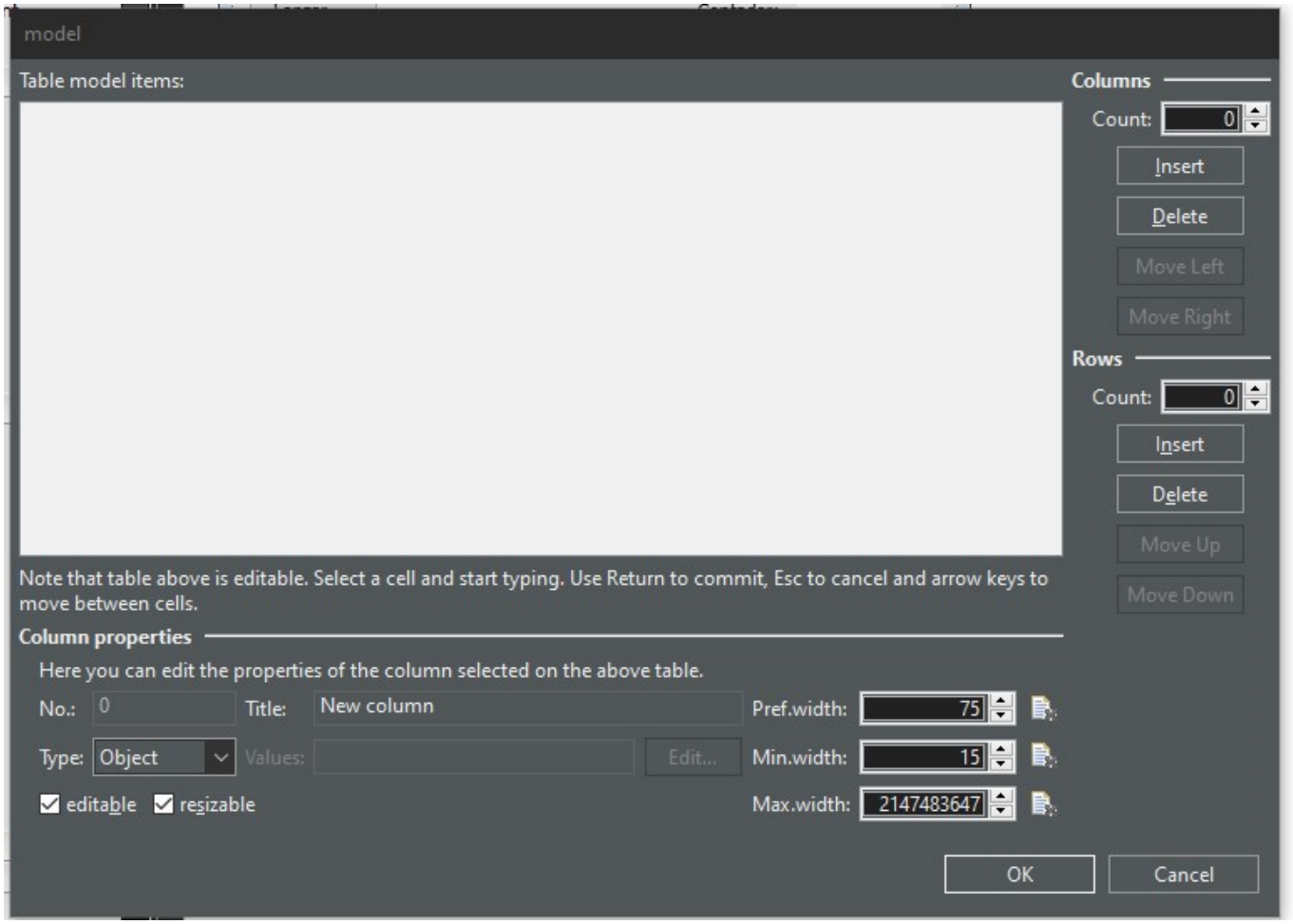
Primero creamos una nueva ventana desde el Window Builder:



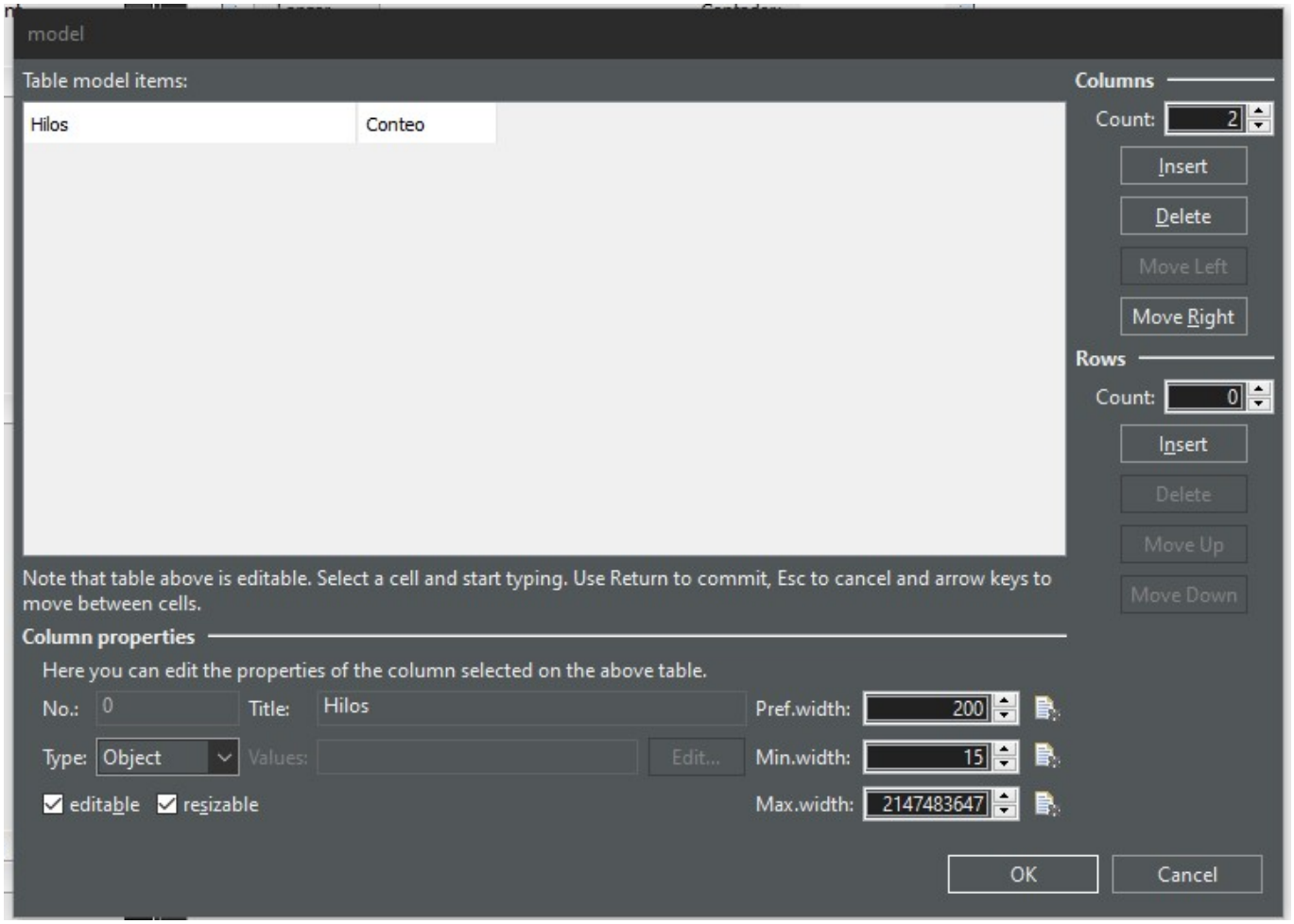
A continuación insertamos la JTable en la ventana y en propiedades de esta vamos a model:



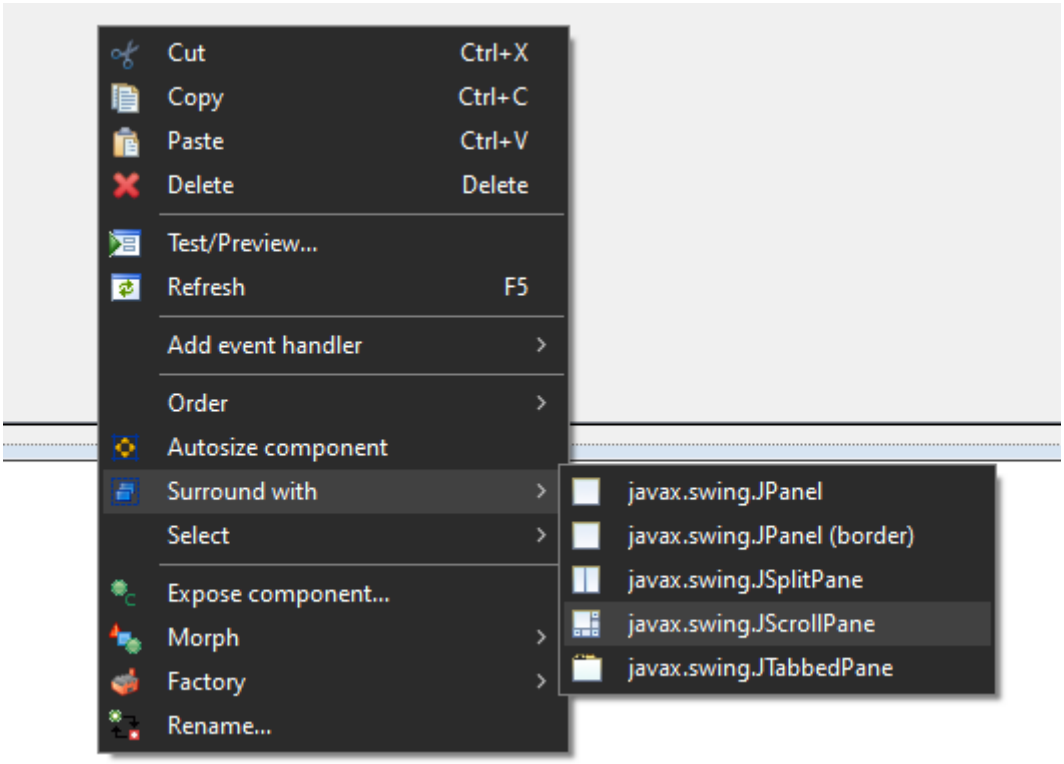
Cuando le demos a model nos saldrá la siguiente ventana con el editor de la tabla:



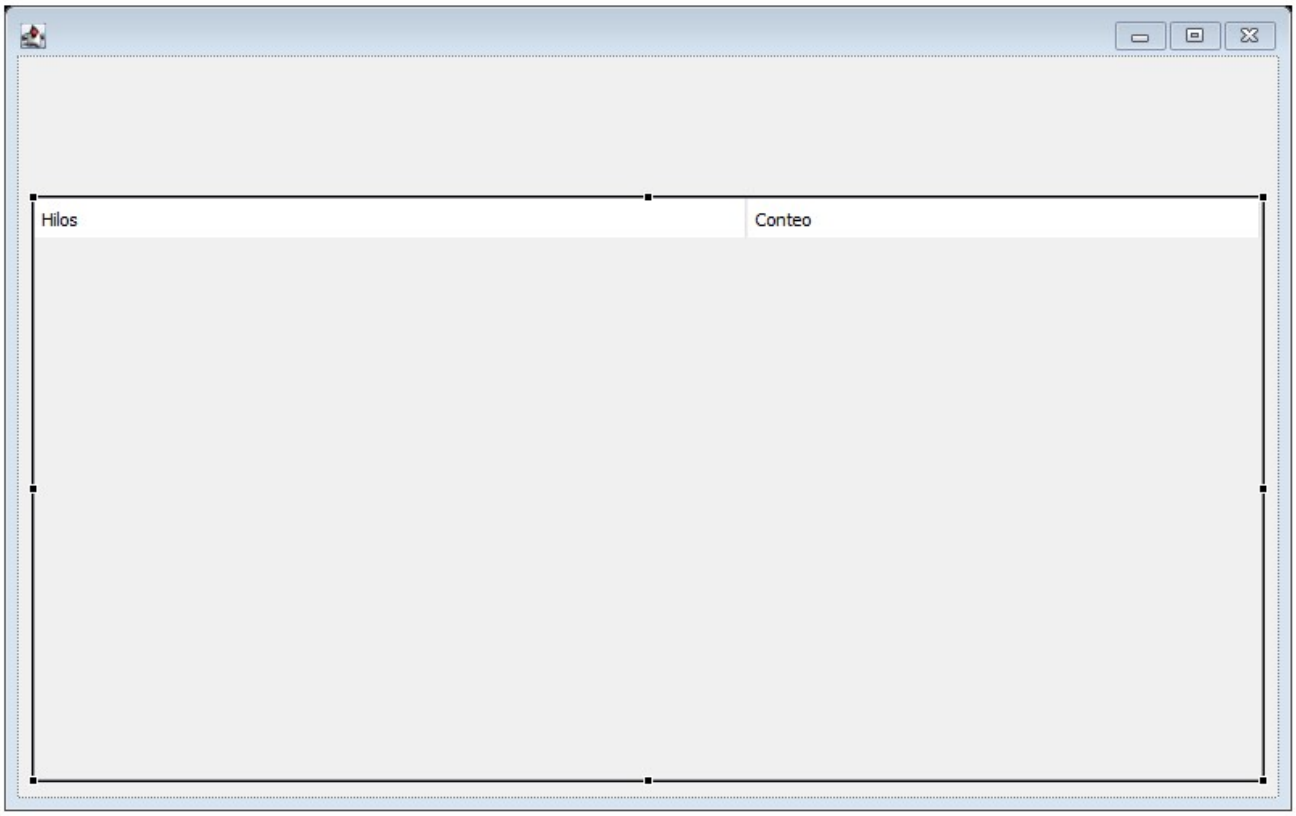
En la derecha le daremos a insertar una vez, colocaremos abajo el nombre de la columna, en este caso conteo, le daremos a insertar de nuevo y pondremos como nombre Hilos. Podemos ajustar el tamaño de las columnas abajo, en este caso la columna Hilos será de tamaño 200:



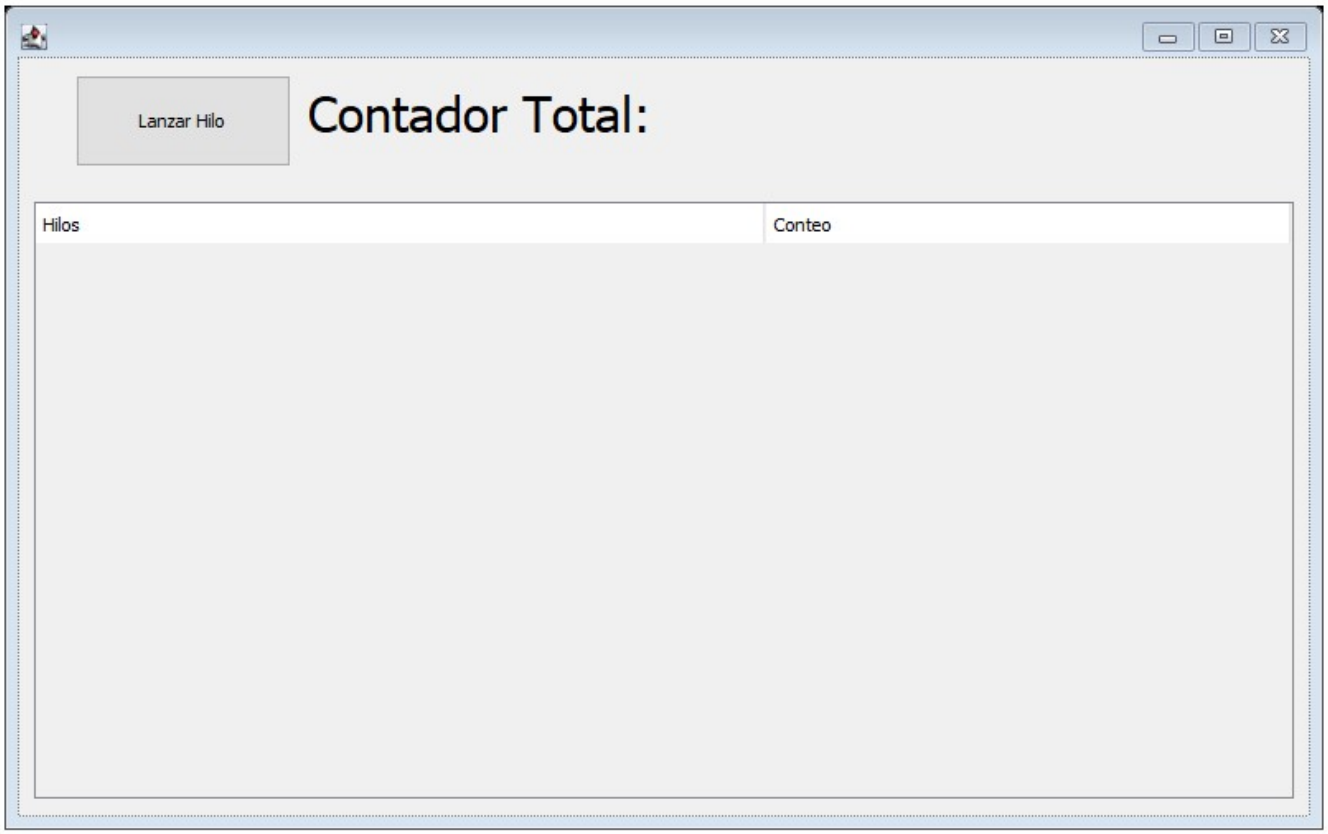
Tras darle a aceptar, no veremos nada en donde hemos insertado la tabla, ahora le daremos click derecho al recuadro donde debería estar la tabla e iremos a Surround With y seleccionaremos JScrollPane:



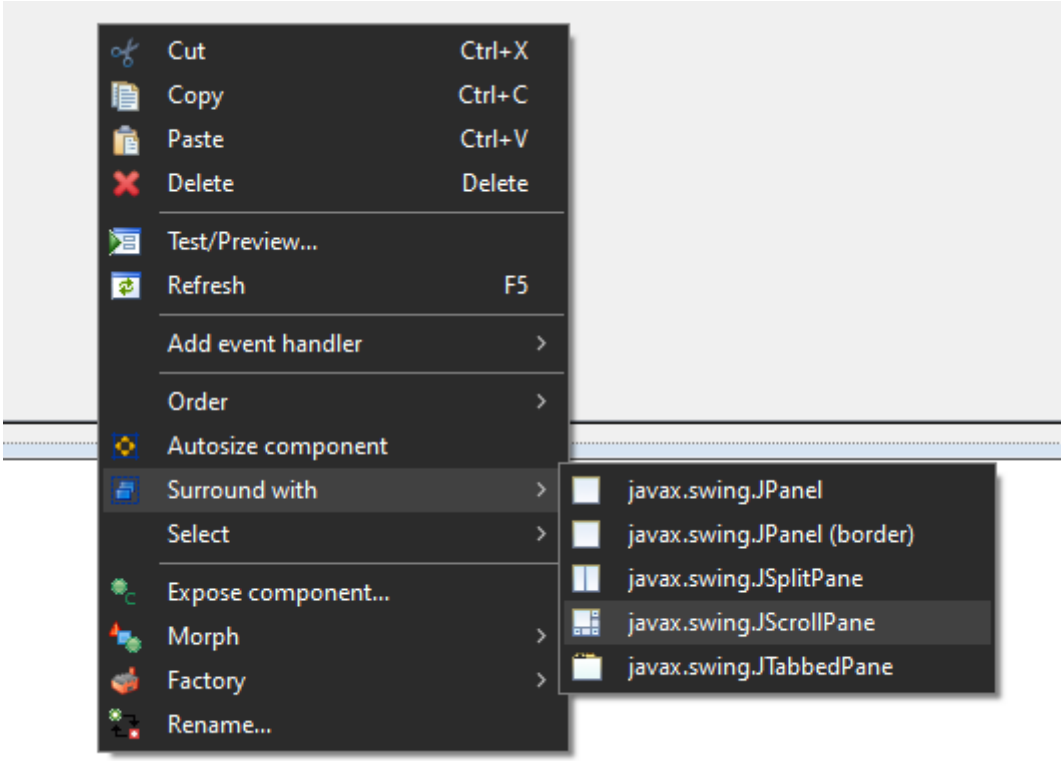
Tras eso tendremos algo como esto:



Ahora que hemos terminado con la inserción de la JTable insertaremos el Botón Lanzar y la Jlabel que irá al lado del contador de iteraciones totales:



Finalmente insertaremos un JTextField no editable para que muestre los conteos totales:



Implementación Funcional

Variables

Hilos

Para implementar los hilos que lanzaremos crearemos una nueva clase a la que llamaremos SimpleThread:

```
1  package ss;
2
3  public class SimpleThread extends Thread{//Extendemos la clase Thread
4      public static final int FOR_EVER = -1;//Constante para ejecución sin fin
5      protected long delay;//variable para el retraso de ejecución
6      protected int times;//Variable para el número de repeticiones
7      //Constructor del hilo, indicamos el retraso que trendrá en re-ejecutarse
8      //y cuantas veces se volerá a ejecutar
9      public SimpleThread (long delay, int times) {
10         System.out.println("Constructor SimpleThread");//Texto a mostrar en consola cuando se construye un hilo
11         this.delay = delay;
12         this.times = times;
13     }
14     public void run() {
15         try {
16             for(int aux = times; (times>=0)||(aux == FOR_EVER); times--) {
17                 System.out.println("Mi Delay es:"+delay);//Mostramos en consola el delay
18                 sleep(delay);//Para la ejecución por el tiempo de retardo especificado
19             }
20         }catch(Exception e) {
21             System.out.println("Error.");//En caso de que algo falle se mostrará este error
22         }
23     }
24
25
26 }
```

From:
https://knoppia.net/ - **Knoppia**

Permanent link:
https://knoppia.net/doku.php?id=dad:jtable&rev=1695855210

Last update: **2023/09/27 22:53**