



Universidad Nacional del Litoral
Facultad de Ingeniería y Ciencias Hídricas
Departamento de Informática

Tecnicatura en Informática Aplicada al Diseño Multimedia y Sitios Web

Elementos de Programación

Unidad 3: Interfaz Gráfica

Docente: Ing. Risso, Oscar Luis
Año: 2024

Índice

| | |
|---|----------|
| 1. Preparacion de un Proyecto | 3 |
| 1.1. Interfaz de GUI Builder | 5 |
| 1.2. Paleta | 6 |
| 1.3. Propiedades de elementos | 8 |
| 1.4. Nociones básicas de diseño | 8 |
| 1.5. Ejemplo.S2.3.1 | 11 |

Unidad 3: Interfaz Gráfica

Segunda Sesión de Estudios

En este Tema

Esta unidad está pensada para ser utilizada como guía de referencia a la hora de crear interfaces de usuario en JAVA utilizando la herramienta “GUI Builder” que viene incluida en NetBeans. Los métodos aquí descritos no son ni mucho menos exhaustivos, pues lo que se pretende es dar a conocer unas pautas básicas que sirvan como base de la creación de interfaces simples, dejando al interesado la tarea de investigar en mayor profundidad a la hora de utilizar técnicas más avanzadas..

A través de los ejercicios de práctica iremos integrando muchos de los conceptos estudiados antes para resolver situaciones más complejas.

Objetivos

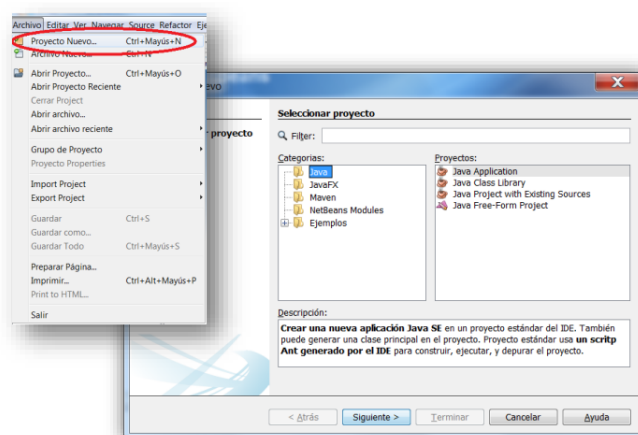
Al finalizar este tema Ud. debe lograr:

- Conocer los principales elementos de NetBeans y la filosofía de trabajo de esta herramienta de programación.
- Crear aplicaciones o proyectos JAVA sencillos, diseñar interfaces gráficas usando componentes visuales y programar eventos para estos componentes.
- Comprender la organización de un proyecto Java y los archivos empleados.

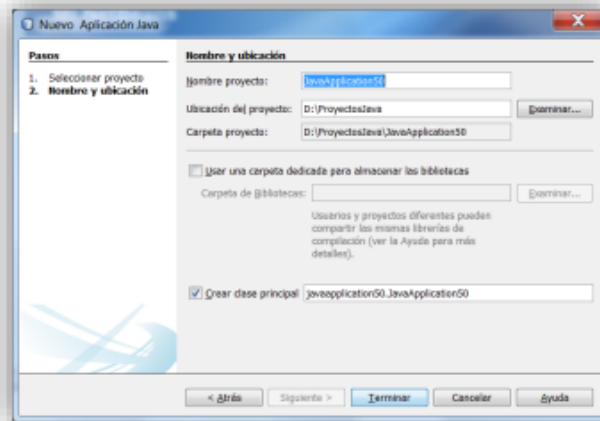
1. Preparacion de un Proyecto

Para poder utilizar GUI Builder en un proyecto, éste se debe crear de una manera específica, siguiendo los pasos que se detallan a continuación:

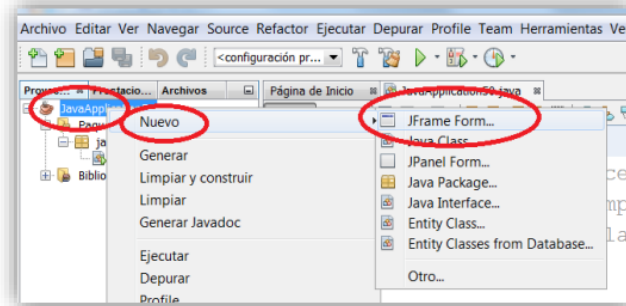
1. Elegimos “Archivo > Nuevo Proyecto” (File > New Project, en la versión en Inglés) o pulsamos directamente en el botón de “New Project” (Nuevo Proyecto) en la barra de herramientas.
2. Seleccionamos la carpeta que pone “Java”, y de ahí la opción que dice “Java Application”. Pulsamos “Next”. (Siguiente)



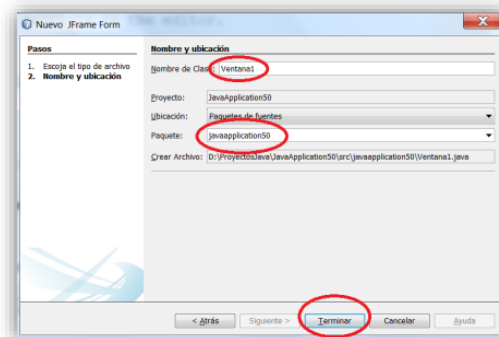
3. Rellenamos con los datos deseados el formulario, pero dejando marcada la casilla “Create Main Class”, (Crear Clase Principal). Pulsamos “Finish”, (Terminar).



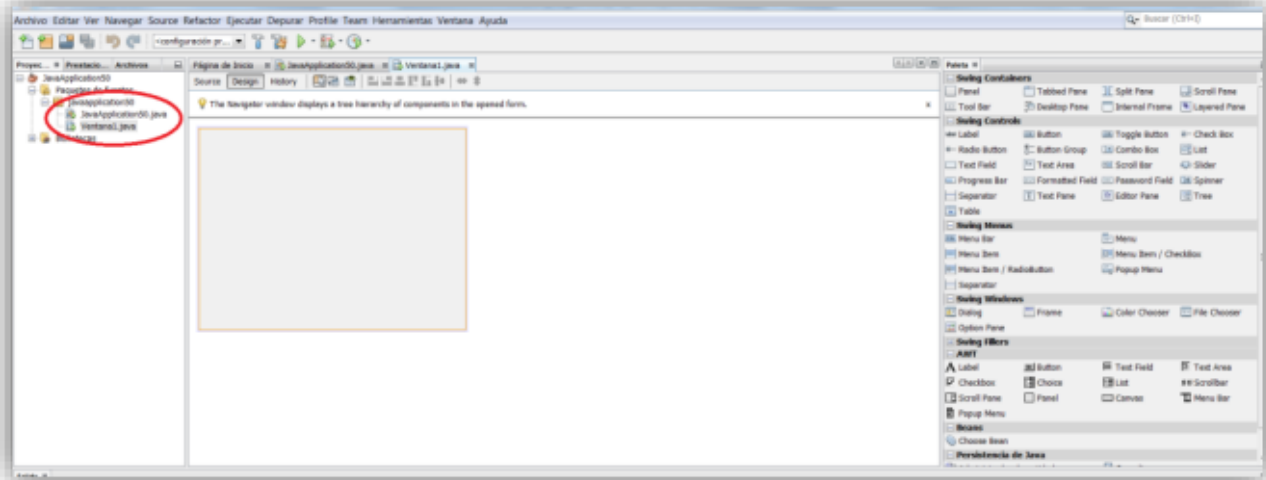
4. Ahora tenemos que crear el archivo que contendrá la GUI. Para ello pulsamos con el botón derecho sobre el icono del proyecto que hemos creado y elegimos “New > JFrame Form”. (Nuevo > JFrame Form)



5. Rellenamos el formulario poniéndole el nombre que deseemos y el paquete que contendrá la interfaz. Pulsamos “Finish”. (Terminar)

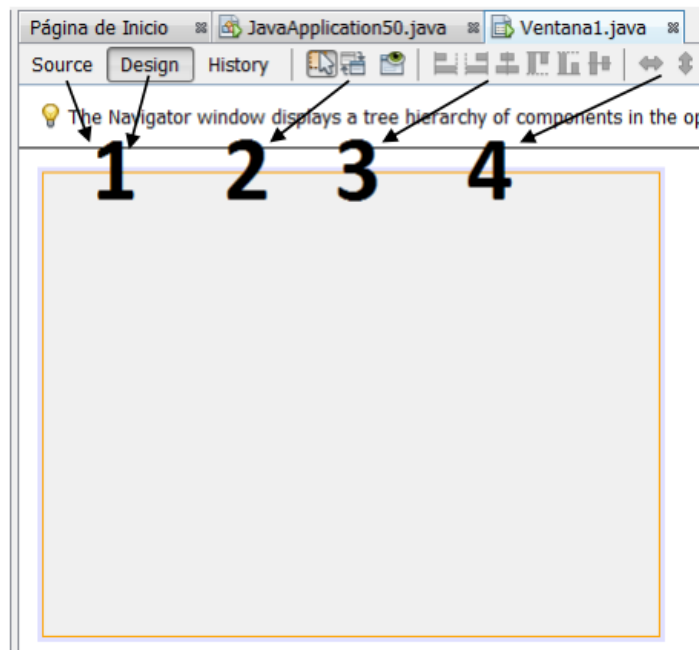


6. Se nos debería abrir el archivo que hemos creado, mostrando la interfaz del GUI Builder.



1.1. Interfaz de GUI Builder

Esta zona en un comienzo tan sólo contendrá un rectángulo vacío. Sirve para colocar los distintos elementos de la interfaz que estemos creando, cambiando sus parámetros (color, tamaño, etc) y, en definitiva, amoldándolos a los que queramos conseguir. En la parte superior tiene una barra de herramientas con diferentes elementos:



1. Source – Design: estos dos botones nos permiten cambiar entre modo de vista de código y gráfico. En el Gráfico podremos añadir los distintos elementos de la interfaz, mientras que en el de vista de código les daremos funcionalidad. Nótese que en la parte de código aparecerán zonas sombreadas de azul, lo que significará que ese trozo de código sólo puede ser editado mediante el GUI Builder y no de forma

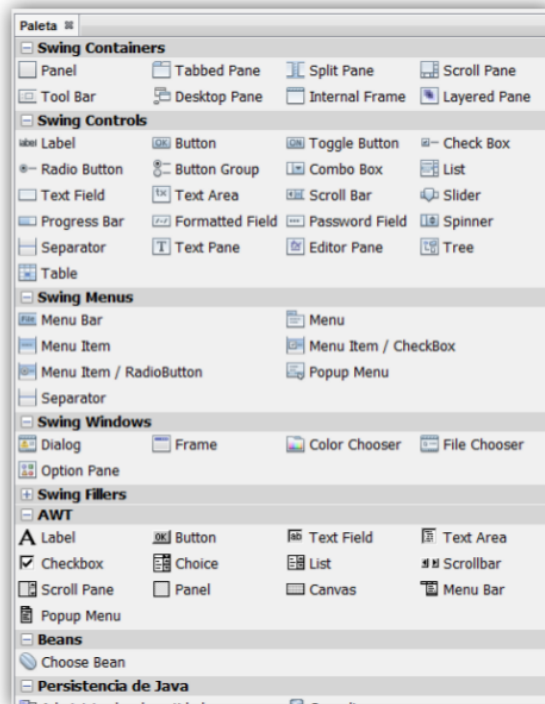
manual.

2. En esta parte tenemos 3 botones muy útiles.
 - a) Modo Selección: nos permite seleccionar los elementos de la interfaz y moverlos o cambiar su tamaño.
 - b) Modo Conexión: nos deja definir la relación entre dos elementos de la interfaz, sin tener que entrar en la vista de código.
 - c) Vista previa: aparece una interfaz preliminar para que podamos evaluar su funcionalidad antes de compilar.
3. Botones de autoajuste: con ellos podemos alinear los elementos de la interfaz de forma automática.
4. Pautas de autoredimensionamiento: indican si al ampliar la ventana principal de la interfaz los elementos que contenga se redimensionan con ella o no.

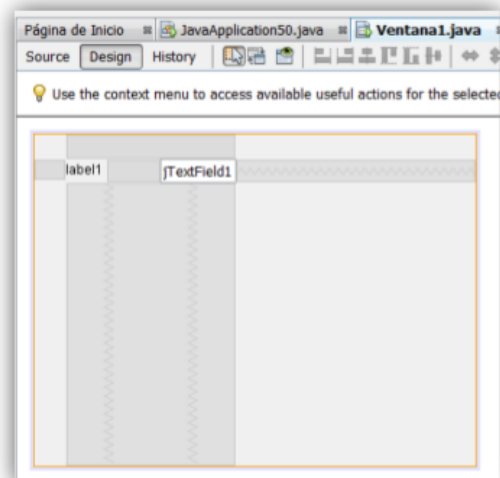
1.2. Paleta

Es la parte que contenida en el rectángulo de la derecha de la pantalla. En ella podremos elegir qué nuevo elemento queremos añadir al diseño de la interfaz y colocarlo según nuestras preferencias. Contiene varios apartados diferenciados:

- Swing Containers: en esta parte se encuentran los elementos que sólo sirven para contener a otros, pero que por sí mismos no hacen nada. Ejemplos de ello son las barras de herramientas o los menús tabulares. Si queremos que contengan algo, debemos añadirlo en su interior.
- Swing Controls: aquí se almacenan los elementos mediante los que se crea o almacena información y órdenes, y que pueden estar contenidos en los descritos anteriormente o no. Como ejemplos, tenemos botones, etiquetas, barras deslizantes o tablas.
- Swing Menus: aquí hay distintos elementos que ayudan a la creación de barras de menús, añadiendo los menús propiamente dichos y sus elementos.
- Swing Windows: en esta sección tenemos una serie de ventanas que aparecen como diálogos, y que son útiles a la hora de crear eventos de advertencia y similares.
- AWP: estos elementos son similares a los visto con anterioridad en Swing Containers y Swing Controls, pero con un diseño anticuado y simple, pues AWP es una biblioteca de clases para el desarrollo de interfaces anterior a Swing, la cual no es ni más ni menos que una extensión de la Segunda.
- Java Persistence: en este apartado se hallan los elementos correspondientes a consultas sobre bases de datos.

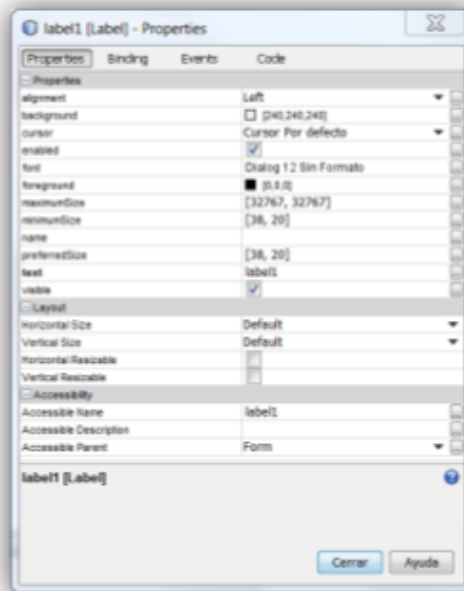


Agregamos algunos componentes al JFrame



1.3. Propiedades de elementos

Aquí se mostrarán las propiedades del elemento que tengamos seleccionado en ese momento, permitiendo a su vez su alteración. Hacemos click derecho sobre el elemento a inspeccionar, por ejemplo en label1 y seleccionamos propiedades:



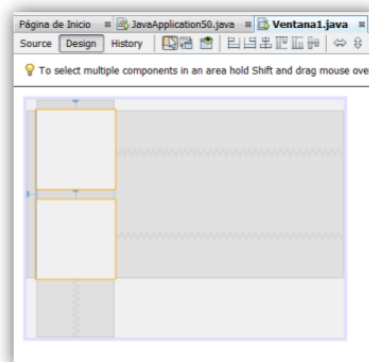
Propiedades: Es aquí donde podemos cambiar sus propiedades pinchando en opción adecuada al hacer clic con el botón izquierdo o derecho del ratón, sobre las propiedades.

1.4. Nociones básicas de diseño

A la hora de diseñar en GUI Builder, hay 3 pasos que son claves para dar forma a la interfaz que se desea crear: añadir, modificar propiedades y definir el anclaje.

Añadir un elemento

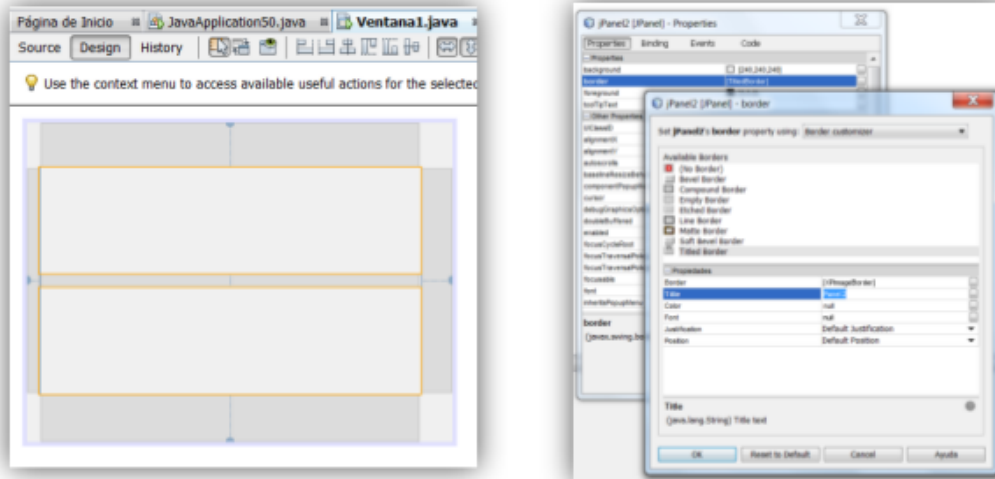
Basta con seleccionar el que queramos y arrastrarlo hasta la zona de diseño. En un principio suele ser mejor insertar primero los elementos contenedores, tales como paneles o menús, para luego ir añadiendo los demás. En el ejemplo hemos añadido 2 paneles:



Nota: En la imagen se ven ambos paneles porque están seleccionados, pero de no ser así no se vería nada, pues los paneles por defecto no traen borde.

Modificar atributos

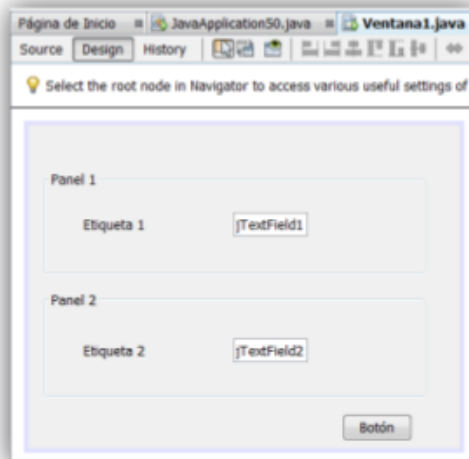
Lo primero que debemos hacer es redimensionarlos hasta estar conformes con el resultado. Después, seleccionamos el que queramos y en la zona de propiedades los modificamos a nuestro gusto. En este caso, hemos añadido un borde con título a las paletas, y de paso unos cuantos elementos más, como botones o etiquetas:



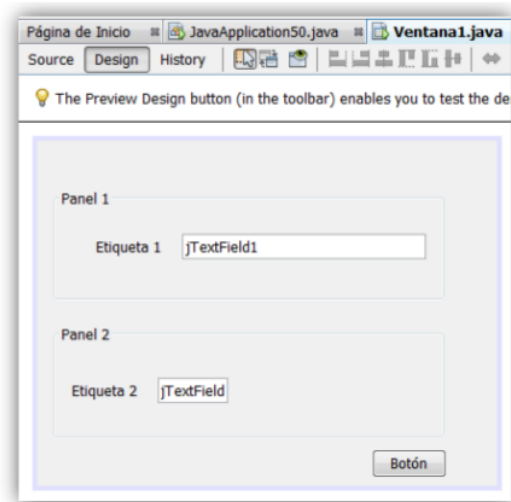
Click derecho en el borde del panel y luego propiedades

Anclaje

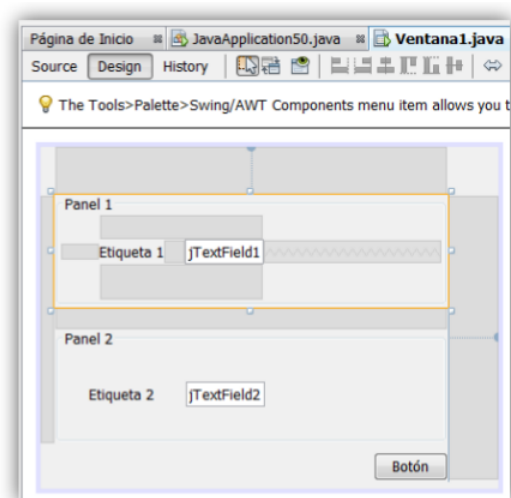
EL anclaje se refiere a la capacidad de un elemento de adaptarse al tamaño de la ventana o del objeto que la contiene (algo así como una interfaz dinámica en cuanto a tamaño). Se puede realizar de forma automática si acercamos el elemento hasta el borde de la tabla de diseño o del contenedor, cuando aparece una línea punteada. Para hacerlo de forma manual simplemente basta con seleccionar o deseleccionar las opciones que vimos en la zona de diseño número 4, y que relacionan contenedor con contenido en ancho, alto, o ambos a la vez.



Se puede comprobar cómo el Panel 1 tiene definida la propiedad de autorredimensionarse con el ancho de la ventana, mientras que el Panel 2 queda intacto sea cual sea.



Del mismo modo, jTextField1 está relacionado con Panel 1, puesto que si no lo estuviera ocurriría lo que se ve en la imagen.



Otro tipo de anclaje es el que se da entre un elemento y otro al mismo nivel (ninguno contiene al otro), que hace que se conserve la posición relativa entre ambos (como pasa con el Botón del ejemplo, que conserva la posición con respecto al borde). Esto se consigue usando las guías que aparecen mientras se arrastra un objeto: si lo ponemos a la distancia de una de las líneas que aparecen en otro elemento, esa distancia no se alterará nunca:

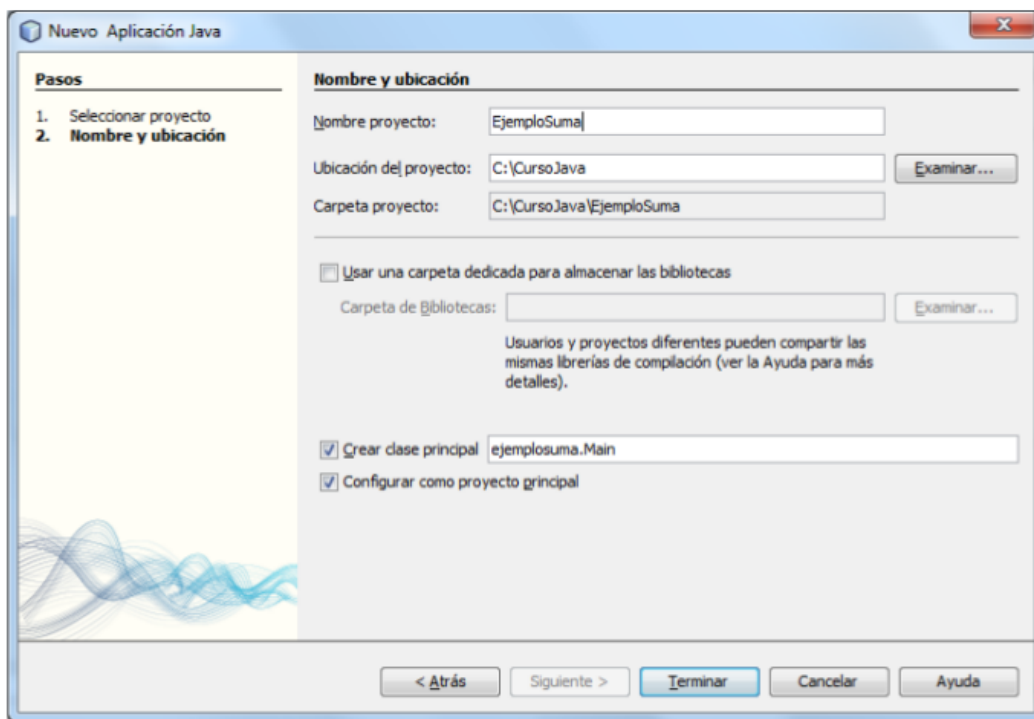


1.5. Ejemplo.S2.3.1

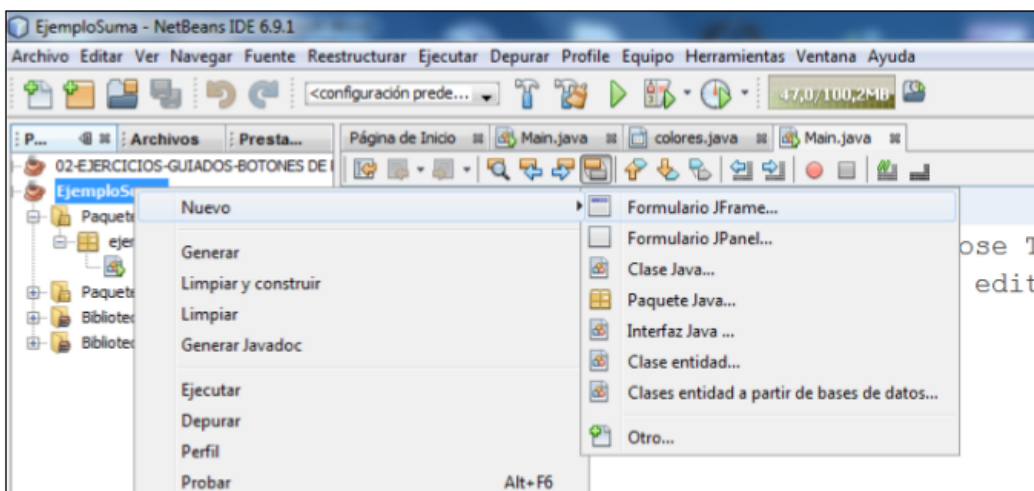
Diseñe un formulario donde se ingresen 2 números enteros y por medio de un botón realice la suma de esos números y muestre el resultado en el formulario.

Resolución paso a paso

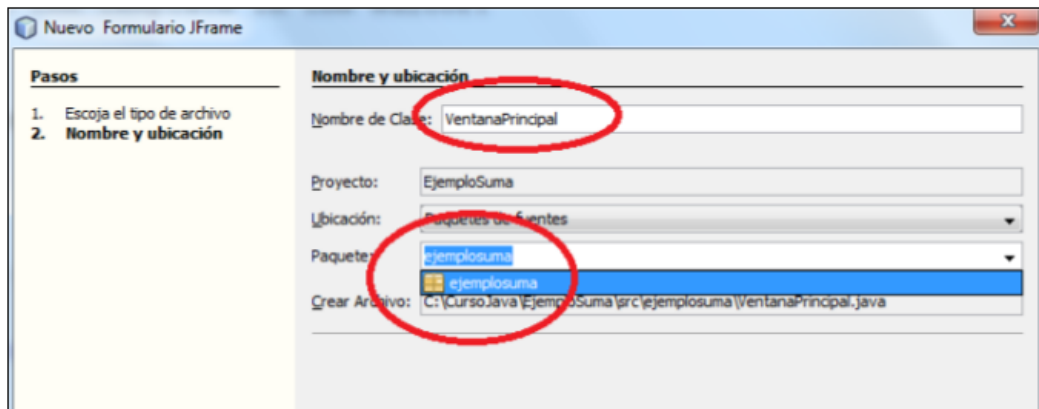
1. Primero debemos crear nuestro nuevo proyecto desde el menú Archivo / Proyecto Nuevo (Ctrl+Mayúsculas+N).
Nombre: “EjemploSuma”



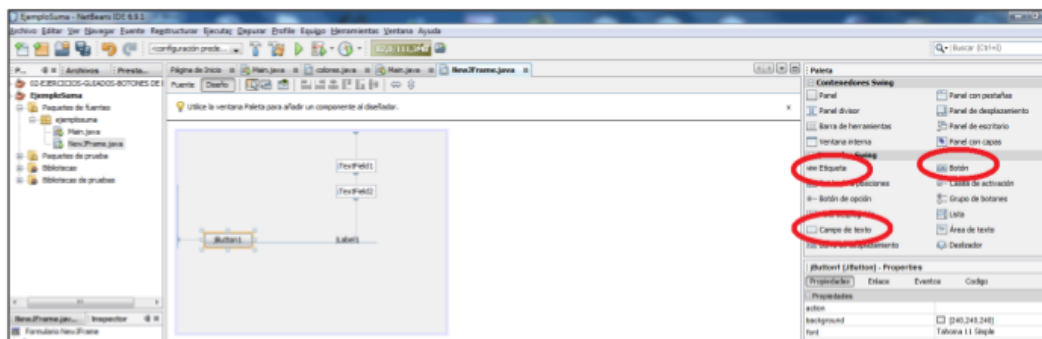
2. Posicionamos el puntero del Mouse sobre el nombre del proyecto, pulsamos el botón derecho, elegimos Nuevo y Formulario JFrame... .



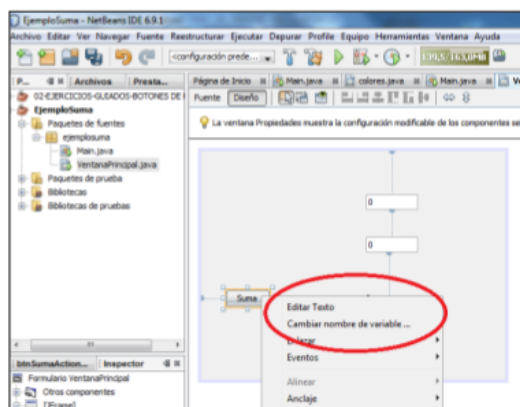
3. Cambiamos el Nombre de Clase por VentanaPrincipal y seleccionamos el paquete que corresponde al proyecto en curso.



4. A continuación agregue 2 campos de texto, un botón y una etiqueta.



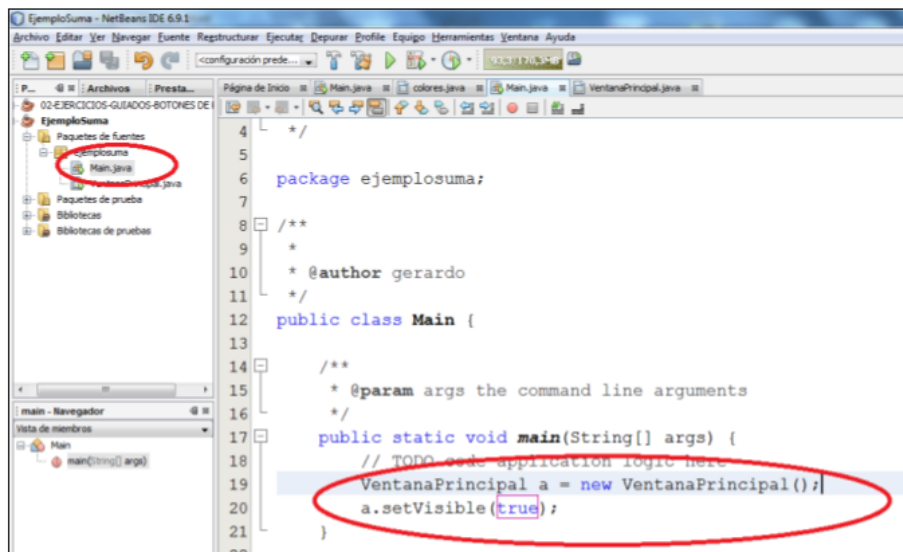
5. Me posiciono sobre el botón, pulso click derecho y cambio el Texto por “Suma” y el nombre del botón por btnSuma. Para el campo de texto cambio el nombre por txtA y el otro campo de texto lo llamo txtB, a la etiqueta la llamo etiResultado.



6. Hacemos doble click en el botón Suma para modificar “ActionPerformed” y copiamos el siguiente código.

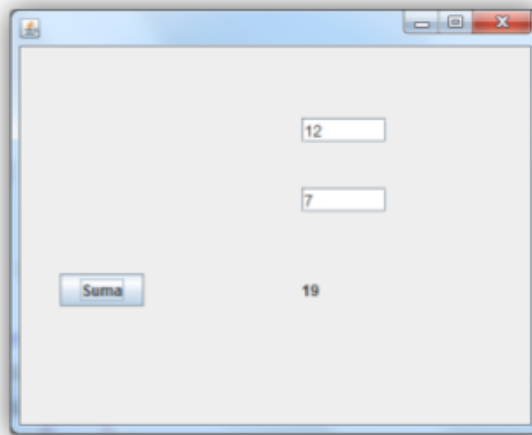
```
1 private void btnSumaActionPerformed(java.awt.event.ActionEvent evt) {
2     // TODO add your handling code here:
3     int a,b,c;
4     a=Integer.parseInt(txtA.getText()); //Convierte a entero el contenido de la
      caja
5     b=Integer.parseInt(txtB.getText()); //de texto.
6     c= a + b;
7     String salida="";
8     salida += c;
9     etiResultado.setText(salida); //Convierte salida en texto en la etiqueta.
10 }
```

7. Ahora modificamos el main del Proyecto, como muestra la figura:

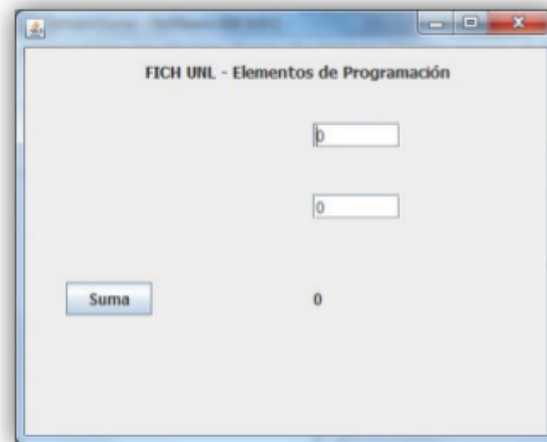


```
1 public class Main {
2     /**
3     * @param args the command line arguments
4     */
5     public static void main(String[] args) {
6         // TODO code application logic here
7         VentanaPrincipal a = new VentanaPrincipal();
8         a.setVisible(true);
9     }
}
```

8. Compilamos con F9 y ejecutamos con F6.



9. Podemos Mejorar un poco la interface colocando etiquetas para mostrar distintos textos, por ejemplo, una etiqueta (label) que diga: **FICH UNL Elementos de Programación**

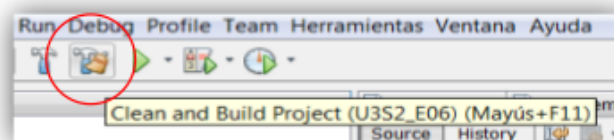


10. Distribución e Instalación de programas en Java

Una vez que se ha finalizado el programa, debemos prepararlo para la instalación en el ordenador cliente.

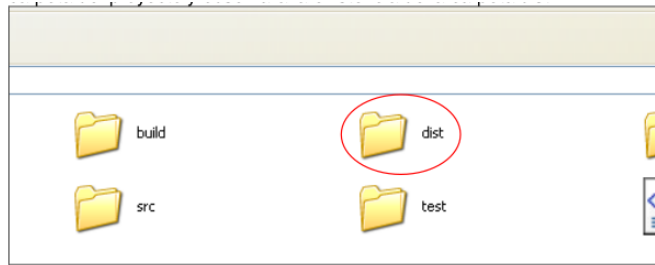
A continuación se indican las cosas a tener en cuenta a la hora de realizar una distribución de nuestro proyecto.

- a) En primer lugar es necesario pulsar el botón Limpiar y construir proyecto principal. Este botón compila el programa y crea una carpeta llamada dist dentro de la carpeta del proyecto, la cual será la carpeta “distribuable”.



(Si hubiera un error de compilación el proceso se detendría y aparecería un mensaje de error)

- b) Si realiza esta operación con alguno de sus proyectos, puede acudir a la carpeta del proyecto y observará la existencia de la carpeta dist:



- c) Entre en esta carpeta y observe su contenido:



Observara la existencia de un fichero con extensión `.jar`. Este fichero se puede decir que es el “ejecutable” del programa.

El fichero `.jar` contiene todas las clases que ha programado en su proyecto.

Debe recordar que los programas java pueden ejecutarse en cualquier entorno, por ello, los ejecutables no tienen una extensión `.exe`. Para que un ejecutable `.jar` pueda ser ejecutado en un determinado sistema, debe estar instalado en él, la máquina virtual de java.