

Bloque 4: Entorno gráfico

1. ¿Qué es un Component?
2. ¿Qué es un Container?
3. ¿Qué relación existe entre ellos?
4. ¿Qué es un Listener?
5. Explica qué estamos haciendo con el siguiente código:

```
Panel1.setLayout(new GridLayout(3, 5))
```
6. Haz una aplicación que tenga 2 botones. Estos no harán nada por el momento. Nota: Usa JButton y JFrame.
7. Modifica el ejercicio anterior, y añade una caja de texto. Cuando pulsemos el botón1, debe aparecer el texto "hola!" y cuando pulsemos el 2, "adios!".
8. Vamos a hacer una pequeña calculadora, para ello crea un programa que tenga 4 botones: (+ - * /) y 3 cajas de texto. Las 2 primeras contendrán los numeros iniciales, y la 3ª el resultado. Para colocar los elementos en pantalla, usa GridLayout.
9. Añade un botón para cerrar a la aplicación anterior. Nota: usa System.exit(0).
10. Añade 2 JCheckBox al ejercicio anterior.
 - Cuando el 1º esté activo, el tipo de letra de los JTextField será rojo, cuando esté inactivo negro.
 - Cuando el 2º esté activo, todos los otros componentes (menos el checkbox actual) se bloquearán.
11. Añade un JList al ejercicio anterior, en él guardaremos cada uno de los resultados que obtengamos de la calculadora. Por ejemplo, cuando pulses en "+" el resultado se guardará en la lista. Si luego pulsas en "-", se guardará a continuación en la misma lista y así sucesivamente. El formato en que se guardará será el siguiente:
$$3 \times 5 = 15$$
$$6 - 9 = -3$$

...

12. Añade un JComboBox al ejercicio anterior, con las siguientes opciones:

- escoge una opción
- borrar textFields
- borrar lista

Cuando escojas la opción 2, los textFields deben quedar en blanco. Cuando escojas la 3, debe vaciarse la lista. La 1 no hace nada.

13. Añade una nueva opción al JComboBox anterior, que se llamará “mostrar lista como texto”. Cuando la utilices, se copiará todo el contenido de la lista a un nuevo campo de texto que crearemos en la parte de abajo de la aplicación.

Como el texto puede ser largo, en lugar de usar JTextField, usaremos JTextArea, y cada línea debe corresponder con una entrada de la lista.

14. Para hacer más entendible nuestra aplicación, añade 4 labels con una descripción al lado (o arriba/abajo) de los textFields y del textArea.

15. Modifica (una vez más) el ejercicio anterior, y mueve todo el contenido que hemos creado dentro de un JPanel. Una vez lo tengas, haz que en JFrame use BorderLayout, y añade en el centro nuestro panel con el programa, arriba un label con el nombre de nuestra aplicación (calculadora 1.0) y abajo la info de copyright.

16. Añade la opción de cambiar el color de fondo. Para ello, añade un botón nuevo a la aplicación, y en el listener del mismo usa la clase JColorChooser.

17. Vamos a mejorar el aspecto de nuestro programa, en lugar de tener un boton “salir” vamos a crear un menú con esa opción. Así que elimina el boton indicado, y crea un menú que tendrá la siguiente estructura:

archivo ->exportar como TXT
->salir

Cuando pulsemos en salir deberá salir de la aplicación.

18. Pon el código necesario en la opcion exportar para que se guarde en un fichero.txt toda la información del JTextArea. Deberemos pedir al usuario la ruta en la que guardar el fichero (usa JFileChooser).

19. Añade la opción de cambiar el tipo de letra, entre 3 disponibles. Para eso, añade 3 JRadioButtons a la aplicación con esas descripciones, y al pulsar en cada uno debe cambiar el tipo.

20. Añade un JSpinner que nos permita modificar el tamaño del texto.
21. Nuestro programa ha crecido mucho, y ahora hay demasiadas opciones en pantalla. Vamos a mover las opciones de configuración (color de fondo, radioButtons y spinner) dentro de otro panel, que mostraremos/ocultaremos según el estado de un checkbox que añadiremos, que cuando esté activo mostrará el panel.
22. (opcional) Modifica el ejercicio de la calculadora, para que funcione como applet. Elimina del applet la exportación en TXT.