


INSTRUCCIONES:

- 1) Crea una carpeta **Apellido1_Apellido2_Nombre_DI02_ExamenFebrero**.
- 2) Dentro de esta carpeta deberás crear una nueva, que la nombrarás como se te especificará en el enunciado del ejercicio y que contendrá el desarrollo del mismo.
- 3) Debes asegurarte que se **guarda el proyecto completo** y/o los ficheros, tal y como lo has hecho con las tareas, para la correcta corrección.
- 4) **No olvides generar el proyecto en NetBeans**. La imposibilidad de poder abrir el proyecto no pudiendo acceder al código fuente para su revisión y calificación **supondrá la no puntuación del ejercicio**.
- 5) Una vez finalizado el examen **deberás comprimir la carpeta** (Creada en el punto 1) nombrando el fichero comprimido con el mismo nombre que has creado la carpeta y luego subirlo plataforma. En caso que no fuera posible la conexión a la plataforma, una vez comunicado al profesorado, se almacenará tal y como se indica en el medio que le suministre el profesorado.

Recursos para el examen presencial

 [Tarea para subir el Examen práctico Febrero](#)

- Es necesario obtener al menos un **4 sobre 10** en este examen para que se realice la media con el examen teórico.
- Este examen supone el **70% de la nota** final del examen.
- Se permite el uso de material de consulta impreso y/o en soporte magnético y el uso de internet.
- Recuerda, si necesitas reiniciar tu equipo almacena tus ficheros en un almacenamiento externo. Puede ser que el equipo esté congelado y perderías tus ficheros.

PUNTUACIÓN

Se puntuarán los ejercicios de acuerdo a la siguiente tabla.

TABLA DE PUNTUACIONES

	EJ1	EJ2	EJ3	EJ4
PUNTUACIÓN MÁXIMA	3	2	4	1
PUNTUACIÓN OBTENIDA				

NOTA: _____/10

Consideraciones previas

- Nombre del proyecto: ExamenFebrero
- Nombre de Packages dentro del proyecto:
 - a. MisControles
 - b. Prueba

EJERCICIO

Dentro del package MisControles

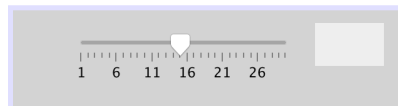
1. Crea un componente de texto con una única propiedad con el nombre **longitud**
Los valores por defecto que debe tener el componente al momento de su creación son los siguientes:
 - a. Longitud de valor 7
 - b. Color de fondo RGB(240, 180, 180)

El componente debe poder gestionar internamente los eventos necesarios para que al ingresar texto se cumplan las siguientes especificaciones:

- c. Si el componente contiene texto cambia su color de fondo a blanco.
- d. Si el componente no contiene texto el color de fondo será el mismo utilizado al momento de su creación.
- e. La cantidad de caracteres del componente está limitada por su longitud.

Nota: Es posible que tengas que utilizar más de un evento de teclado.

2. Crea un editor personalizado con un JSlider y un JLabel como se muestra en la imagen



- a. Limita el Slider entre 1 y 30
 - b. Agrega en el mismo JPanel el código necesario para que se modifique el valor del JLabel según el valor establecido en el JSlider tanto al abrir el JPanel o al modificar el valor de JSlider.
3. Crea el editor de propiedades personalizado con el nombre **LongPropertyEditor** que permita establecer la longitud del componente:
 - a. desde la hoja de propiedades con una lista desplegable con los valores 1, 5, 10, 15, 20, 25, 30
 - b. desde el editor gráfico con los valores que permita el JSlider.

Dentro del package Prueba

4. Crea el siguiente diálogo con el nombre DatosPersonales utilizando el componente creado.

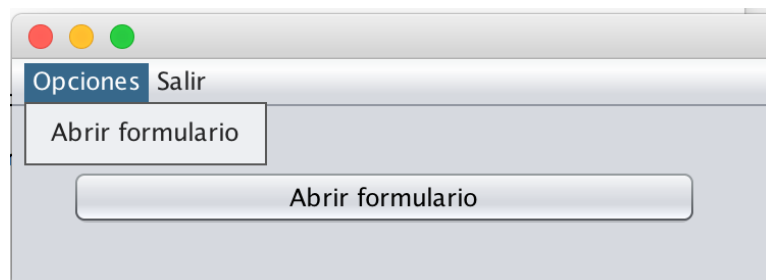
- El botón **Aceptar** debe verificar que se hayan completado todos los datos
 - a. Si falta algún dato por completar debe informarlo con un mensaje de diálogo.



- b. Si se han completado todos los datos debe informarlo con otro mensaje de diálogo y volver a la pantalla principal.



- c. Si se pulsa el botón cancela debe volver a la pantalla principal.
- Crea un formulario principal con un menú de opciones y un botón como se muestra en la imagen siguiente:



- a. Al seleccionar **Salir** debe cerrarse la aplicación
 - b. Al pulsar el **botón Abrir formulario** o la **opción de menú Abrir formulario** debe abrir el diálogo de **Datos Personales**.