	RAMA:	Informática	CICLO:	DAM		
	MÓDULO	Programación				CURSO: 1º
	PROTOCOLO:	Boletín de ejercicios	AVAL:		DATA:	
	AUTOR	Francisco Bellas Aláez (Curro)				

## Boletín 9 – GUI 2ª Parte

**Nota:** Cuando no se indique lo contrario se usará un **layout null**.

### Ejercicio básicos:

Para realizar durante los videos o cuando se indique en clase. No es necesaria validación, pero se aconseja su realización para entender bien lo explicado en los videos.

V1. Realiza un programa en Swing con un Botón que ponga: Púlsame. Cada vez que el usuario accede con el ratón a dicho botón este cambia de posición de forma aleatoria en un rango de 600 en X y 400 en Y. Hazlo con una clase interna.

V2. Modifica el ejercicio anterior para realizarlo con un adaptador.

V3. Amplía el ejercicio anterior de forma que al pulsar la tecla A también salte a una posición aleatoria.

V4. Haz que al pulsar el icono de cierre (X) pregunte si desea cerrar la aplicación o no.

V5. Realiza un formulario en el que aparezcan 100 JLabel numeradas de 1 a 100 colocando 20 etiquetas por fila. Mediante un mouse adapter haz que el texto de las etiquetas cambie a rojo al pasar el ratón sobre ellas y se restaure a negro cuando se salga.

### Ejercicios para validación:

1. En un formulario con el título "Control de Ratón" coloca dos botones y una label Con el texto "Teclas". Actuará de la siguiente forma:

- Cada vez que el ratón esté dentro del formulario se debe indicar la coordenada añadida al título (P. ej. "Control de ratón – (X:100, Y:200)". También lo realizará aunque esté encima de los botones. El (0,0) será la esquina superior izquierda del contentpane (No del jframe)
- Cuando el puntero esté fuera del formulario se reestablecerá el título sin coordenadas.
- Los botones emularán los botones del ratón (izquierdo y derecho) de forma que cambian de color si el botón correspondiente está pulsado y se reestablece al soltarlo. Es suficiente con que se vea el efecto al pulsar sobre el formulario.

COLEXIO <b>VIVAS</b> S.L.	RAMA:	Informática	CICLO:	DAM		
	MÓDULO	Programación				CURSO: 1º
	PROTOCOLO:	Boletín de ejercicios	AVAL:		DATA:	
	AUTOR	Francisco Bellas Aláez (Curro)				

- Al pulsar una tecla se mostrará en la *label* tanto la letra Unicode pulsada como su código de teclado.
- Si se pulsa la combinación de teclas CTRL+C se abrirá un formulario secundario modal de configuración donde se puede elegir el título estándar del formulario principal (con un textfield) y el color al que cambian los botones (en un combobox con al menos 4 opciones). Puede ser FlowLayout.
- Al cerrar el formulario principal se preguntará al usuario si está seguro. En caso negativo se cancela el cierre de la aplicación.

## 2. Realizar un formulario que simule el teclado de un teléfono móvil clásico.

- Debe tener al menos 12 botones con los números del 1 al 9 en las tres primeras filas y en la última la almohadilla (#), el 0 y el asterisco (\*). Se inicializarán los 12 mediante un bucle (no habrá 12 variables JButton).
- Cuando el ratón pase por encima de los botones cambiará de color y si se pulsan quedarán permanentemente con un tercer color.
- Además se dispondrá de un jtextfield donde se mostrarán los números pulsados. Será no editable.
- Si se pulsa una tecla que corresponda a un número también funcionará.
- Finalmente se creará otro botón más de reset (este no es necesario que se haga en el bucle) mediante el cual se borra el textfield y se restaura el color de todos los botones.
- Crea una barra de menú con tres entradas. El primera denominada Archivo, habrá las siguientes opciones:
  - Grabar número: se permitirá añadir el número actual a un archivo.
  - Leer números: Muestra en un JOptionPane los números guardados en el archivo.

En la segunda denominada Móvil tendrá como ítems Reset, un separador y Salir. La tercera será Otros con un único ítem Acerca de... que sacará un JOptionPane con información sobre la aplicación y el autor. Debe responder a teclas ALT.

## 3. Formulario con 2 cajas de texto una etiqueta entre ellas con un símbolo de operación (por defecto +) y una etiqueta a la derecha de ambas con el símbolo =. Debajo de las etiquetas 4 radiobutton con las opciones suma (marcada por defecto la primera vez), resta, multiplicación y división. Finalmente un botón de Operación.

Cuando se seleccione una operación cambiará la etiqueta con el símbolo correspondiente. Y si hubiera algún resultado se borra y se deja el =.

COLEXIO <b>VIVAS</b> S.L.	RAMA:	Informática	CICLO:	DAM		
	MÓDULO	Programación				CURSO: 1º
	PROTOCOLO:	Boletín de ejercicios	AVAL:		DATA:	
	AUTOR	Francisco Bellas Aláez (Curro)				

Al pulsar Operación en la etiqueta de resultado aparecerá el igual (=) y el resultado de la operación. Si la operación no es posible, en una etiqueta que se encuentre al lado del botón Operación en color rojo se informará del problema.

Mediante un combobox se podrá además seleccionar la cantidad de decimales a mostrar entre 0 y 5.

Cuando el programa finalice, debe guardar los datos numéricos que haya en las cajas de texto y la operación seleccionada, en un archivo de configuración que estará en el directorio de usuario pero oculto, de forma que si se vuelve a arrancar el programa lee ese archivo y aparece tal cual estaba en la anterior ejecución.

**Recomendación:** Echa un vistazo al apéndice de gestión de los radiobutton para manejar la selección.

4. Coloca dos combobox (denominadas cbA y cbB), 4 botones, 2 textfield, 3 label,

- Cada botón es una acción (Añadir, quitar, traspasar (2 botones))
- Añadir: Añade el texto del primer textfield (si hay) en cbA. En el textfield se pueden meter varias entradas separadas por punto y coma (;). Se deben obviar entradas vacías (por ejemplo varios ; seguidos o solo con espacios). También añade si se pulsa la tecla Enter estando en dicho textfield.
- Quitar: Elimina el elemento seleccionado en cbA.

El texto del botón se pondrá rojo cuando el ratón pase por encima y se restaurará al salir.

Si en el segundo textfield hay algo, borrará todas las entradas de cbA que empiecen por el string que hay en dicho textfield.

- Botones de traspasar: Traspasa el elemento seleccionado de cbA a cbB y viceversa.
- En una etiqueta indica la cantidad de elementos que hay en cbA y en la otra el índice del elemento seleccionado.
- Mediante un timer se debe comprobar que si la aplicación lleva más de un minuto sin usarse (es decir, no se efectuó ninguno de los eventos programados) debe hacer un reset borrando todos los elementos de los combo y los textfield.
- Meter ToolTip para los distintos componentes. En el caso de cbB, el tooltip indicará el índice del elemento seleccionado (será por tanto variable).
- Al cerrar el formulario se preguntará al usuario si está seguro. En caso negativo se cancela el cierre de la aplicación.

COLEXIO <b>VIVAS</b> S.L.	RAMA:	Informática	CICLO:	DAM		
	MÓDULO	Programación				CURSO: 1º
	PROTOCOLO:	Boletín de ejercicios	AVAL:		DATA:	
	AUTOR	Francisco Bellas Aláez (Curro)				

**Opcional:** Sustituye el primer combobox por un JList multiselección. Los botones quitar y traspasar de este al combo actuarán sobre todos los elementos seleccionados.

5. Realizar un juego de lotería primitiva. En el formulario principal aparecen 49 números (hazlo con checkbox y bucle) que el usuario puede marcar o desmarcar hasta 6, ni más ni menos. Habrá un botón Jugar que se habilita cuando el usuario escoge los 6 números. Una vez pulsado el ordenador sacará 6 números aleatorios y marcará en rojo los que salieron y no se acertaron y en verde los que se acertaron. Haz un menú con las opciones:

Guardar última partida: Sacará un formulario secundario que pide el nombre y se añade a un archivo de texto dicho nombre y los números que acertó. Todo en una línea. También podrá cancelar. El archivo estará en la carpeta de usuario y se denominará .records.txt .

Ver records: Sacará un Formulario secundario con un textarea y mostrará la lista de records que haya en el archivo.

Además el título del formulario será animado mediante un Timer de forma que cada 300ms va apareciendo parte del título como si fuera un scroll (Por ejemplo si el título es Lotería aparecería a, ía, ería, tería,...). Cuando se completa espera un poco y vuelve a empezar.

***A partir de este ejercicio si prefieres puedes plantearte coger un proyecto del apartado Ideas para proyectos en lugar de seguir con estos ejercicios. Informa de ello al profesor .***

6 . Coge el juego del tema anterior de la bomba y añádele:


- Funcionalidad con teclas de cursor para mover.
- Un registro de récords con puntuación, nombre y fecha en archivos.

Por supuesto añade elementos que ayuden a realizar lo anterior como formularios secundarios de datos, récords, etc..).

7. Editor de texto: Diseña un editor de textos similar al notepad. El objetivo es dejar el formulario principal sólo con el textarea y un menú de opciones (Mira el Apéndice de menús). De hecho no cambies el layout, deja el que viene por defecto para que el textarea con el scroll siempre ocupe todo el formulario. El resto del diseño lo harás tú usando formularios secundarios en caso de necesidad.

Las características serán las siguientes:


- Debes crear un menú con las siguientes opciones:

	RAMA:	Informática	CICLO:	DAM		
	MÓDULO	Programación				CURSO: 1º
	PROTOCOLO:	Boletín de ejercicios	AVAL:		DATA:	
	AUTOR	Francisco Bellas Aláez (Curro)				

- Archivo: Nuevo, Guardar, Abrir, Separador y Salir.
  - Edición: Seleccionar todo. Mostrar selección (en un JOptionPane).
  - Otros: Configuración, Idiomas y Cuadro de Acerca de... (en un JOptionPane).
- La opción de Configuración saca un formulario secundario **modal** donde aparecerán las siguientes posibilidades:
    - Radiobutton para seleccionar al menos 3 colores distintos de texto y 3 de fondo.
    - Combobox para seleccionar algún tipo de letra de los instalados en el ordenador (mira en el primer tema de Swing como se obtienen).
    - Textfield para indicar tamaño de tabulador.
    - Checkbox para indicar si se activa el ajuste de línea y para especificar si el TextArea será editable o solo lectura.
  - Si se hace doble clic sobre la textarea selecciona todo. También funcionará al pulsar la combinación CTRL+A.
  - Con internacionalización en dos idiomas. Los textos que aparecen en los componentes estarán almacenados en distintos archivos de texto según el idioma pero en el mismo orden. Se escogerá mediante sendas opciones del submenú Idiomas el idioma deseado.
  - Grabar en un archivo temporal el contenido del textarea cada minuto (en un subdirectorío de la carpeta de usuario).
  - Pedirá confirmación antes de salir del programa.
  - Todas las opciones de menú deben tener acceso mediante tecla ALT.

8. Realizar un prototipo funcional en Java de un programa de cambio de divisas. En un mismo formulario se debe poder elegir entre:

- Al menos 3 tipos de monedas (€, \$, Libra) , excluyentes entre sí.
- Tres formas de pago (Transferencia, cheque o metálico) (excluyentes)
- Posibilidad de Imprimir un justificante, Ver estados de cuentas (este solo se permite si el anterior está marcado), Incluir copia a central. Estos tres no son excluyentes.
- Por último, el cambio se puede seleccionar de 4 cajeros distintos y excluyentes.

	RAMA:	Informática	CICLO:	DAM		
	MÓDULO	Programación				CURSO: 1º
	PROTOCOLO:	Boletín de ejercicios	AVAL:		DATA:	
	AUTOR	Francisco Bellas Aláez (Curro)				

Tendrá una etiqueta donde se muestra un resumen de lo seleccionado (debes actualizarla cada vez que cambia un radiobutton)).

Un botón salir cerrará la aplicación previa petición de confirmación. Si se pulsa a la vez que se aprieta la tecla CTRL saldrá sin preguntar.

9. Toma la clase Aula del boletín de Arrays y realiza un interface de usuario en modo gráfico mostrando una tabla de Jtextfield donde estén las notas de los alumnos y luego mete botones u opciones de menú y otros componentes que decidas para poder mostrar distintos resultados y realizar distintas modificaciones.

### Opcionales:

**10.** Programa *Datos Personales*: Crea una clase denominada Persona con los campos Nombre, apellido, DNI (con letra), sexo (integer o enumerado, al menos para hombre, mujer, otro) y edad (int). Sobreescribe la función toString de forma que devuelva un String con nombre y DNI (El nombre ocupa siempre 10 caracteres. Si fuera más largo se reduce a 7 y se añaden 3 puntos suspensivos ).

Realizar un programa en el que se tenga una Jlist con una lista de personas y los siguientes botones:

Botón añadir que lanza un formulario secundario modal que tendrá los componentes necesarios para pedir todos los datos de persona (Para el sexo usa radiobuttons y para edad un combobox con valores de 1 a 100). Tendrá un botón de Añadir y otro de Cerrar.

Si se pulsa Añadir en el secundario, añade los datos dentro de un objeto Persona con los campos antes citados y luego añade el objeto al JList. Al menos los campos Nombre y DNI deben contener algún valor. Y el DNI debe ser válido y no repetido.


Un botón Eliminar que elimina todos los elementos seleccionados previa confirmación con JOptionPane.

Si se hace click sobre un elemento de la lista (solo uno), en una label en el mismo formulario principal aparecerán todos los datos de la persona seleccionada. Usa html en la label para dar un formato agradable al usuario.

Cerrar cierra la aplicación.

Debe tener tooltips explicativos.

11. Realizar un programa que lo primero que vea el usuario sea una ventana dónde se le pida un usuario y una contraseña. Validará el usuario y la contraseña de un archivo en el que están guardados los usuarios y sus contraseñas (en cada línea un usuario y separado por : la password. Por ejemplo *admin:1234*). Si la contraseña es correcta pasa a la aplicación principal, si no, permite intentarlo de nuevo. Debe haber un botón cancelar que si se pulsa se saldrá del programa. Lógicamente en el campo

	RAMA:	Informática	CICLO:	DAM		
	MÓDULO	Programación				CURSO: 1º
	PROTOCOLO:	Boletín de ejercicios	AVAL:		DATA:	
	AUTOR	Francisco Bellas Aláez (Curro)				

contraseña aparecerán asteriscos a medida que se escribe (Revisa el Apéndice del *JPasswordChar*).

En el formulario principal es una aplicación simple que permita añadir nuevos pares usuario-contraseña al archivo. Lo único que se debe tener en cuenta es que el usuario es único. No tendrá labels, en el propio textfield indicará usuario y contraseña borrándose en el momento que se pincha con el ratón.

Este formulario principal tendrá además un menú con las mismas opciones que los botones y además la posibilidad de seleccionar un nuevo archivo de contraseñas.

*Nota: Para hacer la primera parte tienes, entre otras, las dos siguientes posibilidades:*

1. En el constructor de la aplicación principal lanzas el *JDialog* de login de forma modal. Esto hace que no salga la aplicación principal pues se detiene la ejecución dentro del constructor. Luego haces la comprobación y si es incorrecta vuelves a hacer visible el modal (un bucle) y si es correcta dejas que termine el constructor con lo que se dará paso a la principal.
2. En el main lanzas primero el *JDialog* de login de forma modal, luego haces la comprobación y si es correcta lanzas el *JFrame* de aplicación, si no mediante un bucle vuelves a hacer visible el formulario de login.

### ***Ideas para proyectos:***

- Realiza una calculadora del estilo de la que viene incluida con el sistema operativo.
- Arkanoid: Realizar un 'arkanoid simple' (<https://es.wikipedia.org/wiki/Arkanoid>). Los gráficos serán labels con una imagen en su interior (puedes usar GIMP para diseñar raquetas y bloques mediante el generador de botones).
- Añádele manejo con teclas (y ratón sobre el escenario de juego) al del campo de minas de los temas anteriores además de lista de records, niveles, bonus y otros elementos que se te ocurran.
- Jawbreak/CandyCrush: Realizar el juego Jawbreaker ou similar [https://en.wikipedia.org/wiki/Jawbreaker\\_\(Windows\\_Mobile\\_game\)](https://en.wikipedia.org/wiki/Jawbreaker_(Windows_Mobile_game))
- Damas/othello/Oca/parchis...
- Space Invaders o similar.
- Gestión de almacén: Stocks, Proveedores, clientes, entradas, salidas,...
- Worms/Angry Birds.
- Juego de plataformas sencillo.
- Otro juego o aplicación de tu interés.