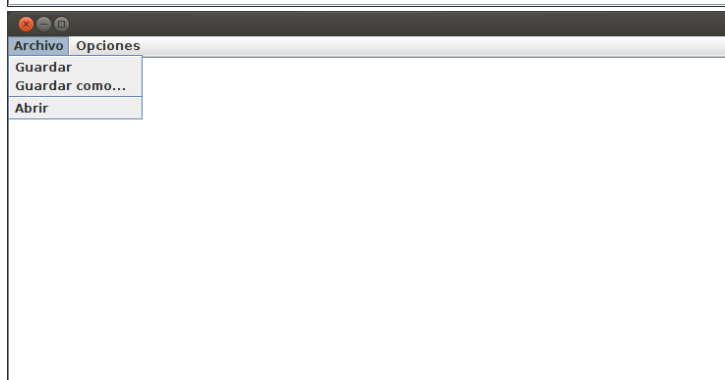
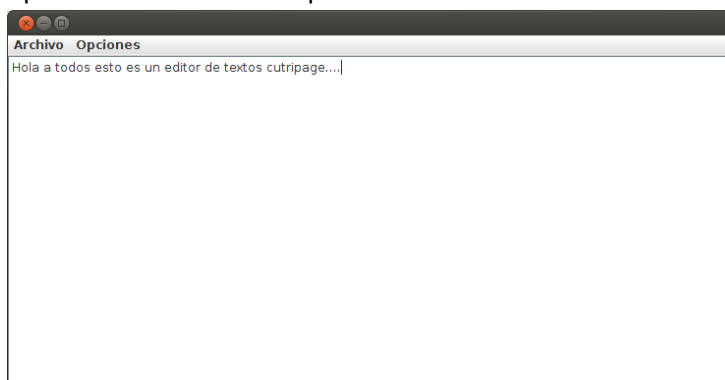


Todos estos ejercicios serán realizados:

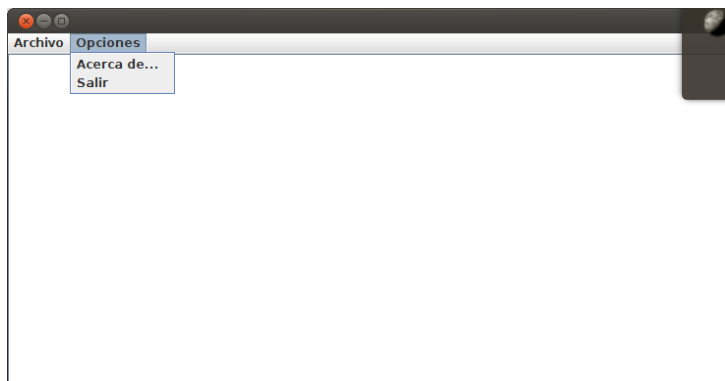
- 1º) Almacenando los datos en ficheros de texto.
- 2º) Gestionando la información en ficheros serializados (menos el 1).
- 3º) Usando bases de datos (menos el 1).

## 1.- Editor de textos.

Realiza un editor de textos sencillos. Deberás utilizar un `textArea` para el texto, con las opciones de menú correspondientes.



Y

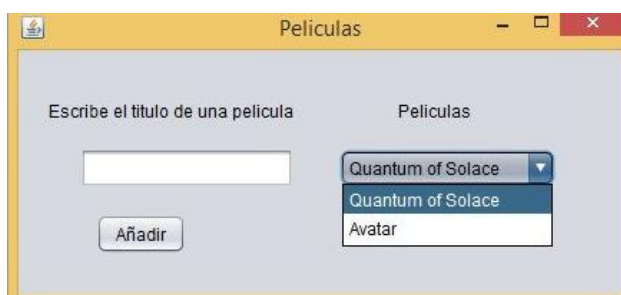


Investiga los tipos de contenedores de texto: **`textArea`**, **`editorPane`** y **`textPane`**. Comprueba cual es el que mejor te vendría y añade algún menú contextual que permita cambiar a negrita

o cursiva. Las opciones guardar y abrir lanzarán las **ventanas estándar** de swing. La opción **Acerca de...** una ventana diseñada por ti con la versión del programa y el creador.

## 2.- Añadir pelis.

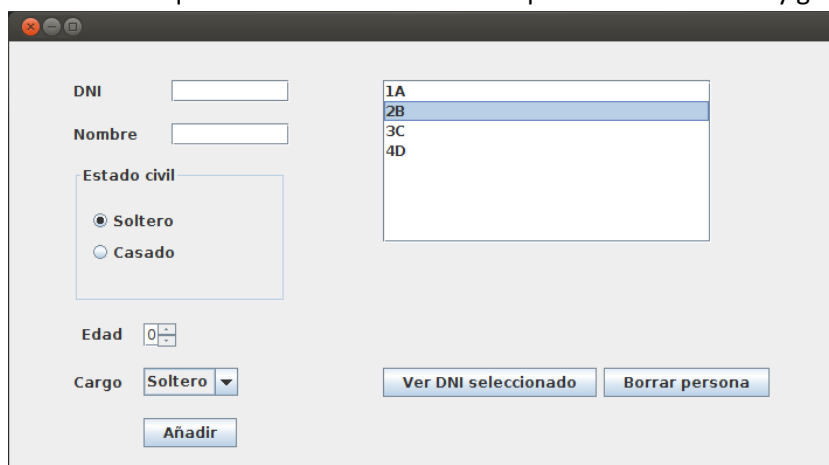
Crea una simple lista de películas. Tendremos un *JComboBox*, donde almacenaremos las películas, que vayamos almacenando en un campo de texto. Al pulsar el botón Añadir la película que hayamos metido, se introducirá en el *JComboBox*. Este ejercicio lo realizamos en el tema anterior. Añade un botón más que sea listar que en otra ventana muestre el listado de las películas introducidas.



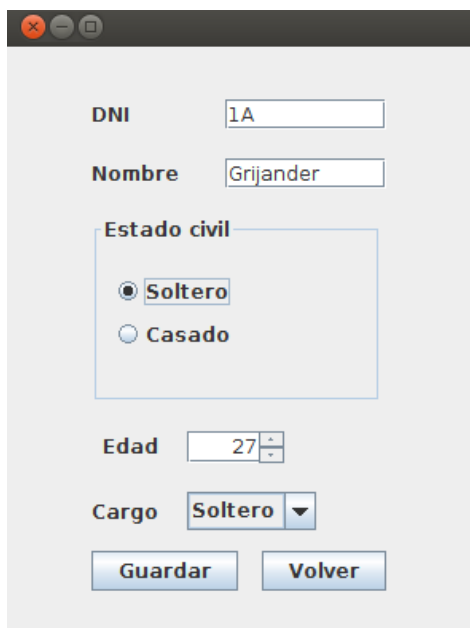
Como opción añade algún sistema que permita borrar las películas que queramos de forma permanente.

## 3.- Registro de usuarios.

Realiza una ventana en la que tendremos un formulario para añadir usuarios y gestionarlos.



Puedes usar alguna de las colecciones que conoces para almacenar la información. Cada vez que le demos a *Añadir* se añadirá a dicha colección. En los botones *Ver* y *Borrar* se maneja la lista. Con *Ver* se desplegará otra ventana en la que se mostrará la información de esa persona:



The screenshot shows a web form with the following fields and controls:

- DNI**: Text input field containing "1A".
- Nombre**: Text input field containing "Grijander".
- Estado civil**: A container with two radio buttons:
  - ☒ **Soltero**
  - ☐ **Casado**
- Edad**: Spin box containing the number "27".
- Cargo**: Dropdown menu with "Soltero" selected.
- Buttons**: Two buttons at the bottom, "Guardar" (Save) and "Volver" (Back).

Si pulsamos Guardar los cambios que hagamos serán actualizados en la lista de la ventana principal. Al pulsar Volver volveremos a la ventana principal.

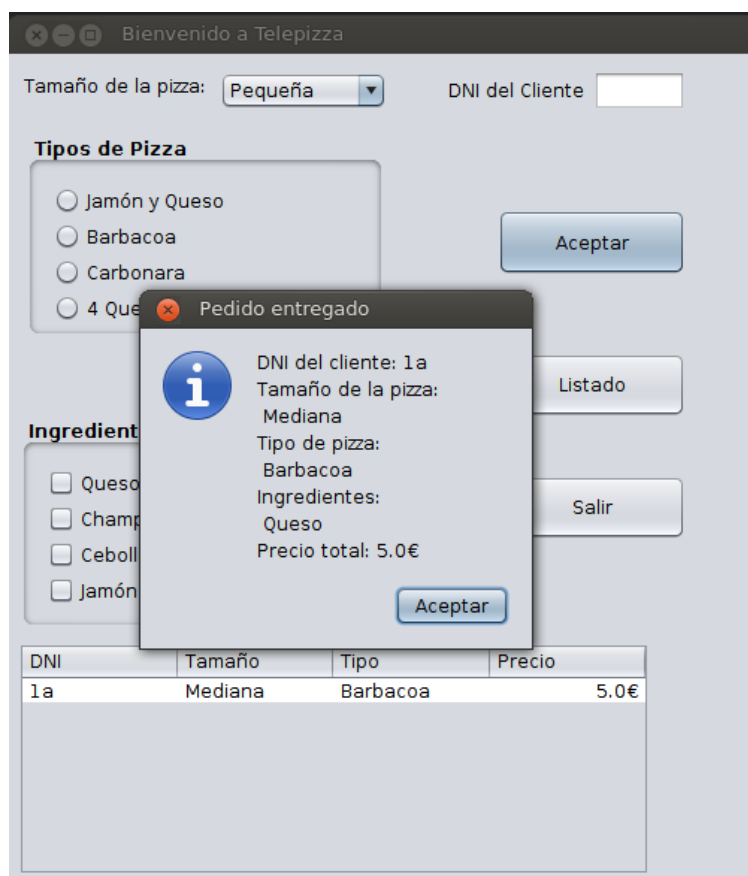
**Ampliación de lo realizado anteriormente:** Este sistema debe incorporar un sistema de almacenamiento externo para que no se pierdan datos. Se debe añadir también algún modo de borrado y edición.

#### 4.- TeleDominosPizza.

Vamos a diseñar una aplicación que gestione los pedidos del TeleDominosPizza. La ventana principal es como la siguiente. Cuando llega un cliente se anota su DNI y se elige el tamaño de la pizza del combo. Se marcará el tipo de pizza y los ingredientes añadidos (los precios quedan a libertad del programador). Al pulsar el botón **Aceptar** se añade a una cola de pedidos.

Si pulsamos **Listado** se mostrará el estado de la cola en la tabla inferior.

Cada x segundos se saca el primer pedido de la cola y se sirve; se mostrará un mensaje como el que se muestra en la figura a modo de ticket de entrega. El pedido, lógicamente, se sacará de la cola.



The screenshot shows a web application titled "Bienvenido a Telepizza". It has a form for placing an order with the following fields and controls:

- Tamaño de la pizza:** A dropdown menu currently set to "Pequeña".
- DNI del Cliente:** A text input field.
- Tipos de Pizza:** A group of radio buttons with options: "Jamón y Queso", "Barbacoa", "Carbonara", and "4 Quesos".
- Ingredientes:** A group of checkboxes with options: "Queso", "Champ", "Cebolla", and "Jamón".
- Buttons:** "Aceptar" (Accept), "Listado" (List), and "Salir" (Exit).

A modal dialog box titled "Pedido entregado" (Order delivered) is displayed in the center. It contains the following information:

- DNI del cliente:** 1a
- Tamaño de la pizza:** Mediana
- Tipo de pizza:** Barbacoa
- Ingredientes:** Queso
- Precio total:** 5.0€

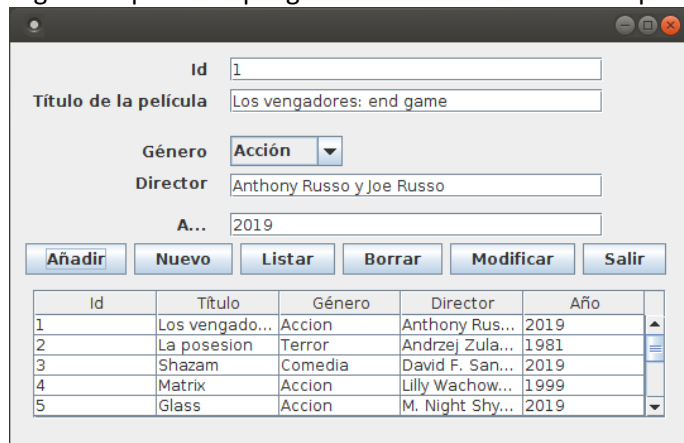
At the bottom of the application, there is a table showing the order list:

DNI	Tamaño	Tipo	Precio
1a	Mediana	Barbacoa	5.0€

Se debe guardar una copia periódicamente de lo que ocurre, de forma que si se sufriera un corte de electricidad el sistema siguiera por donde estuviera al reiniciarse. También se enviará a un fichero de bitácora todo aquello que va ocurriendo en el sistema: añadidos de pedidos y entrega de los mismos, con fecha y hora.

### 5.- Películas.

Vamos a realizar la siguiente pantalla que gestionará nuestra videoteca personal:

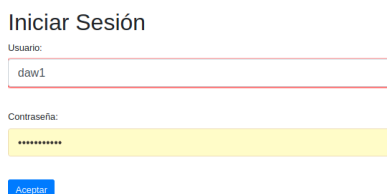


The screenshot shows a web application window for managing a personal video library. It includes input fields for 'Id' (1), 'Título de la película' (Los vengadores: end game), 'Género' (Acción), 'Director' (Anthony Russo y Joe Russo), and 'A...' (2019). Below these are buttons: 'Añadir', 'Nuevo', 'Listar', 'Borrar', 'Modificar', and 'Salir'. At the bottom is a table with columns: Id, Título, Género, Director, and Año.

Id	Título	Género	Director	Año
1	Los vengado...	Accion	Anthony Rus...	2019
2	La posesion	Terror	Andrzej Zula...	1981
3	Shazam	Comedia	David F. San...	2019
4	Matrix	Accion	Lilly Wachow...	1999
5	Glass	Accion	M. Night Shy...	2019

### 6.- Juego de multiplicar y CRUD.

Vamos a realizar una aplicación que funcione de forma análoga a una aplicación WEB. La **pantalla principal** mostrará un **login** (pedirá un usuario y contraseña para ingresar a la aplicación).



The screenshot shows a login form titled 'Iniciar Sesión'. It has two input fields: 'Usuario:' with the value 'daw1' and 'Contraseña:' with masked characters. Below the fields is a blue 'Aceptar' button.

Tendremos dos tipos de usuarios: estándar y administradores. Los usuarios estándar accederán a una ventana en la que podrán jugar al juego de multiplicar que hicimos en el tema anterior.

Los usuarios administradores, al entrar, tendrán que elegir como quieren acceder al sistema: como usuarios estándar (irán al juego) o como usuarios administrador. Si acceden como administradores aparecerá una pantalla que, en diseño web, se llama CRUD (Create Read Update Delete) que permitirá ver a todos los usuarios que tienen acceso al sistema y su tipo de usuario. A través de esta ventana también podremos eliminar, editar y crear usuarios.

Gestión de usuarios

DNI	Correo	Telefono	Nombre	Rol	Edad			
2B	l@l.l	2	Laura	17	Administrador	X	Editar	
3C	n@n.n	3	Nathan	27	Usuario	X	Editar	
4D	m@m.m	4	Marina	9	Administrador	X	Editar	
5E	i@i.i	5	Isabel	25	Usuario	X	Editar	
6F	r@r.r	6	Rafa	12	Usuario	X	Editar	



# PROGRAMACIÓN 1º CFGS DAW Almacenamiento externo – Ficheros y BBDD



## 7.- Alquiler de coches.

Realizaremos una aplicación que permita gestionar **alquiler de coches**. Tendrán tres partes diferenciadas: **gestión de usuarios**, **gestión de coches** y **alquileres**.

Tendremos dos tipos de usuarios con acceso al sistema: **gestores** y **administradores**. Los administradores se encargarán de la gestión de usuarios y coches (altas, bajas, modificaciones). Las gestiones se realizarán con CRUDS.

El CRUD de usuarios será similar a:

DNI	Correo	Nombre	Telefono	Edad		
2B	b	Laura	2	17	Modificar	Eliminar
3C	c	Nathan	3	27	Modificar	Eliminar
4D	d	Marina	4	9	Modificar	Eliminar
5E	e	Isabel	5	25	Modificar	Eliminar

[Añadir](#)

[Volver Inicio](#)

El de coches:

100A	Citroen	C3	Eliminar	Modificar
200B	Citroen	C5	Eliminar	Modificar
300C	Peugeot	205	Eliminar	Modificar
400D	Peugeot	405	Eliminar	Modificar
500E	Renault	Megane	Eliminar	Modificar
600F	Renault	Laguna	Eliminar	Modificar

[Añadir](#)

[Volver Inicio](#)

Los gestores se encargarán de los alquileres con ventanas maestro/esclavo.

DNI:  Nombre:

100A	Citroen	C3	<input type="button" value="Devolver"/>
200B	Citroen	C5	<input type="button" value="Devolver"/>

[Inicio](#) [Volver](#)

En estos formularios según lo que esté relleno para el usuario, así se desplegará abajo, se mostrarán los coches alquilados por esa persona. Al pulsar '*Devolver*' es coche pasará a estar libre.

El botón '*Ver*' es optativa, sería para ver a esa persona en detalle.

Al pulsar '*Add*' (añadir alquiler) se mostrarán los coches no alquilados disponibles:

500E	Renault	Megane	<input type="button" value="Alquilar"/>
600F	Renault	Laguna	<input type="button" value="Alquilar"/>

[Volver](#)

Con la opción de alquilar se alquilarán esos coches por ese usuario.

**Importante:** dentro de mi sistema de almacenamiento tendré que distinguir entre gestores, administradores y clientes. Los clientes no tienen acceso al sistema pero tendré que guardar información de ellos y de qué coches tienen.