Desarrollo de Aplicaciones Web/Desarrollo de Aplicaciones Multiplataforma

Serialización y creación de interfaces de usuario

Programación I

Actividad

Serialización y creación de interfaces de usuario

**Objetivos**

* Crear programas que accedan a la información de ficheros.
* Desarrollar interfaces gráficas utilizando las librerías adecuadas.
* Programar controladores de eventos.

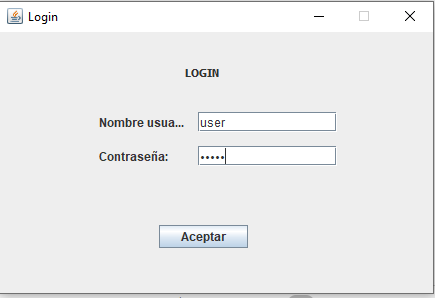
***Se debe entregar un zip que contenga todos los archivos. java que has creado. Para poder aprobar un ejercicio, éste debe poder ejecutarse sin errores. Crea los archivos .java dentro de una carpeta de nombre actividad08***

Crea un programa con una interfaz gráfica que nos permita almacenar en un archivo de texto un listado de trabajadores, añadir nuevos trabajadores y visualizar los trabajadores creados. Para poder gestionar los trabajadores primero tendremos que hacer login con el usuario (“user”) y password(“12345”) almacenados en un archivo de texto.

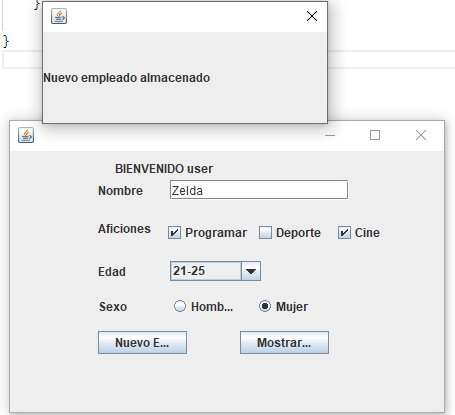
1. Crea una ventana que con dos campos de texto: uno para introducir el nombre de usuario, otro para la contraseña (tipo JPasswordField). También habrá un botón que comprobará si los datos introducidos corresponden con un usuario. Además, contendrá el texto “CREDENCIALES INCORRECTAS” en color rojo. Éste estará oculto, y sólo se mostrará si el usuario le da al botón y los datos introducidos no coinciden con los de ningún usuario. También habrá un mensaje oculto “DEBES INTRODUCIR LOS DATOS” en color rojo, que aparecerá si el usuario le da al botón sin haber escrito el nombre o la contraseña.
   1. Para comprobar que los datos introducidos son correctos, se utilizará un fichero de nombre “**datos\_login.txt**” en el que se almacenarán el nombre de usuario (“user”) y password (“12345”). Se deja a elección del alumno el formato del fichero de datos.
2. Si los datos son correctos (concuerdan con los almacenados en el archivo de texto) se accederá a una segunda ventana en donde se mostrará:
   1. El nombre de la persona que se ha logueado.
   2. Una caja de texto para introducir el nombre del nuevotrabajador.
   3. Un grupo de casillas (CheckBox) que permita seleccionar aficiones.
   4. Una lista desplegable (ComboBox) que permita seleccionar una franja de edad: 10-15, 15-20, 20-25…
   5. Dos botones de selección (RadioButton) para seleccionar el sexo.
   6. Un botón **Nuevo** **Empleado** y otro botón **Mostrar Empleados**
3. Crea una nueva clase Empleado con la siguiente información: un nombre, una edad, un sexo y un listado de aficiones. Crea una clase con los atributos necesarios para almacenar los datos de cada empleado y los métodos constructores, getters y setters teniendo en cuenta que la **clase Empleado debe poder serializarse**.
4. Al clicar en el botón **Nuevo** **Empleado** :
   1. Comprueba si existen empleados previamente serializados en un archivo. Si es así crea un ArrayList con ésos empleados.
   2. Crea un nuevo empleado a partir de los datos introducidos por el usuario.
   3. Añade al ArrayList de empleados el nuevo empleado.
   4. Serializa la lista de empleados y guárdalos en un archivo de nombre “**datos\_empleados**”.
   5. Si todo es correcto, se abrirá un diálogo (JDialog) con el texto “Gracias por usar la aplicación”. Si el usuario no rellena algún campo, se abrirá un diálogo mostrando un texto que describa el error producido.
5. Al clicar en el botón **Muestra Empleado:**
   1. Obtiene el listado de empleados serializados en el archivo “**datos\_empleados”**.
   2. Si todo es correcto, se abrirá un diálogo (JDialog) con la información de todos los empleados.

A continuación se muestran unos ejemplos de funcionamiento:

Pantalla de login



Creación de un nuevo empleado:



Mostrando todos los empleados serializados y almacenados

