

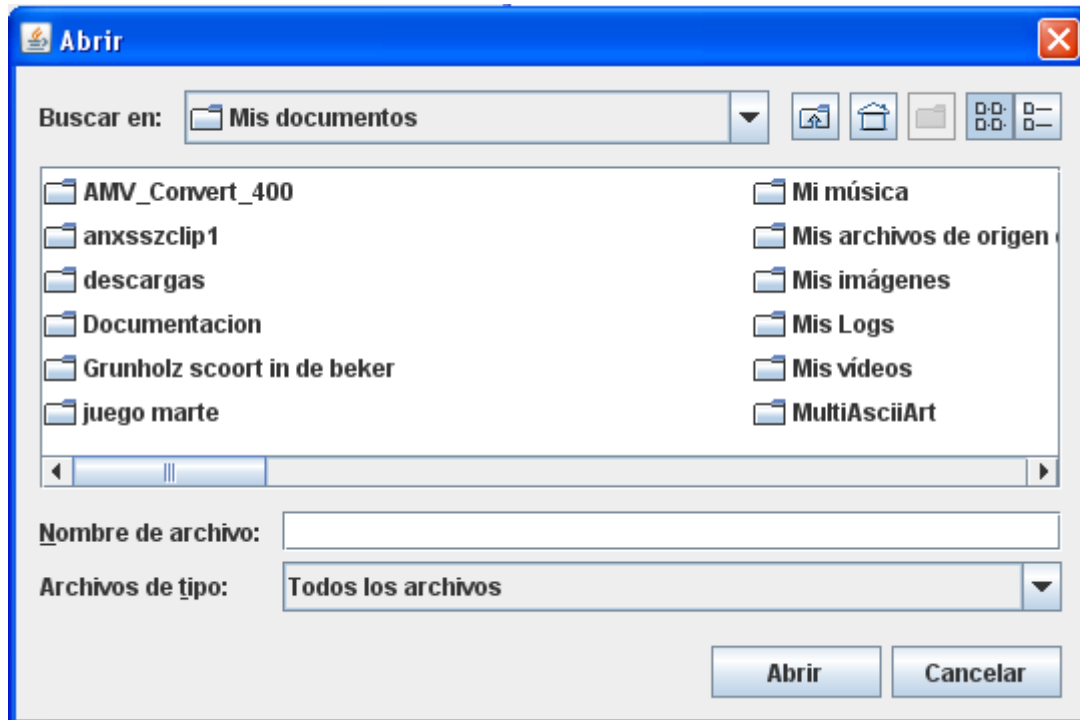
## EJERCICIO GUIADO. JAVA: FILECHOOSER

### Cuadros de diálogo Abrir y Guardar

Las opciones Abrir y Guardar son opciones muy comunes en las aplicaciones. Estas opciones permiten buscar en el árbol de carpetas del sistema un fichero en concreto y abrirlo, o bien guardar una información dentro de un fichero en alguna carpeta.

Java proporciona una clase llamada JFileChooser (*elegir fichero*) que permite mostrar la ventana típica de Abrir o Guardar:

Ventana Abrir fichero:

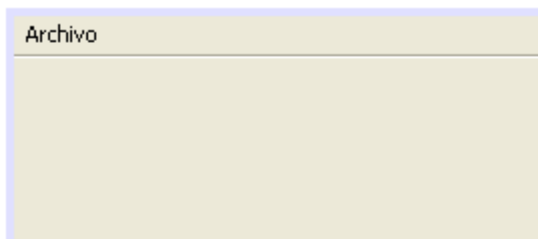


(La ventana de guardar es la misma, solo que muestra en su barra de título la palabra *Guardar*)

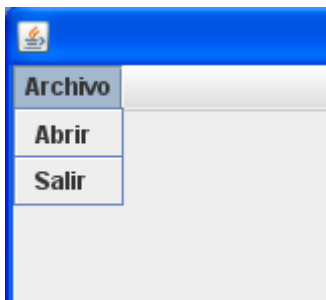
El objeto JFileChooser nos facilita la labor de elegir el fichero, pero no realiza la apertura o la acción de guardar la información en él. Esto tendrá que ser programado.

## Ejercicio guiado

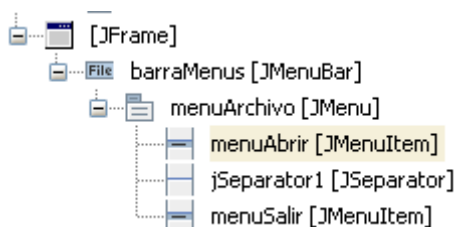
1. Vamos a practicar con el JFileChooser. Para ello, crea un nuevo proyecto.
2. Añade en el proyecto los siguientes elementos:
  - a. Una barra de menús. Llámala *barraMenus*.
  - b. Dentro de ella una opción “Archivo” llamada *menuArchivo*.
  - c. Dentro de la opción “Archivo”, introduce los siguientes elementos:
    - Una opción “Abrir”, llamada *menuAbrir*.
    - Un separador (llámalo como quieras)
    - Una opción “Salir”, llamada *menuSalir*.
3. Una vez hecho esto tu formulario tendrá la siguiente forma:



4. Si ejecutas el programa el menú se verá así:



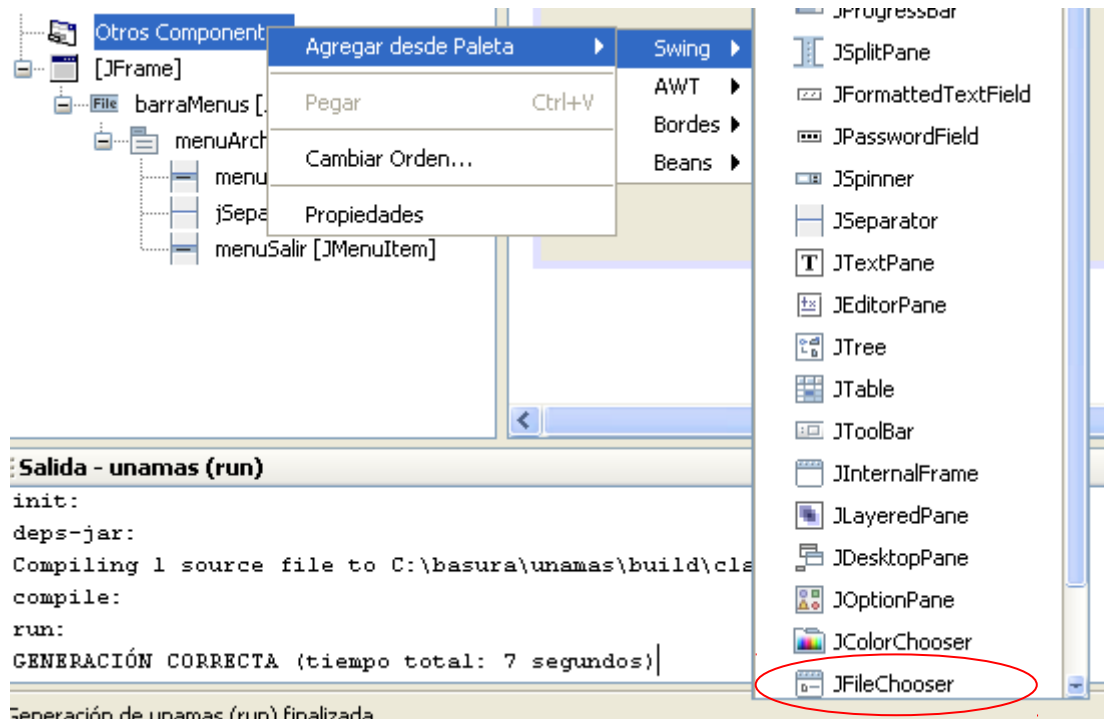
5. Si observas el *Inspector*, tendrá un aspecto parecido al siguiente:



6. Para que al pulsar la opción “Abrir” de nuestro programa aparezca el diálogo de apertura de ficheros, es necesario añadir a nuestro programa un objeto del tipo JFileChooser.

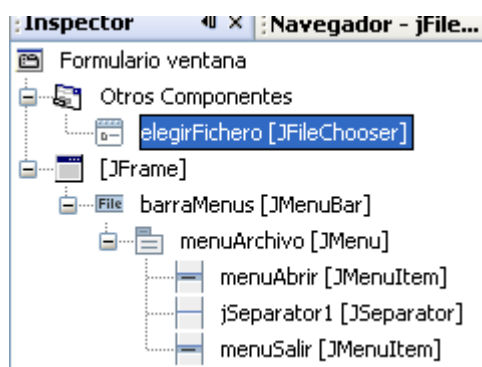
Los objetos JFileChooser se añadirán el la zona de “Otros Componentes” del inspector.

7. Haz clic con el derecho sobre la zona de “otros componentes” y activa la opción *Agregar desde Paleta – Swing – JFileChooser*:



8. Aparecerá entonces un objeto JFileChooser dentro de *Otros Componentes*. Aprovecha para cambiarle el nombre a este objeto. Su nombre será *elegirFichero*.

El *inspector* quedará así:



9. Una vez hecho esto, ya podemos programar la opción Abrir del menú. Activa el evento *actionPerformed* de la opción “Abrir” y programa dentro de él lo siguiente:

```

int resp;
resp=elegirFichero.showOpenDialog(this);
if (resp==JFileChooser.APPROVE_OPTION) {
    JOptionPane.showMessageDialog(null,elegirFichero.getSelectedFile(
    ).toString());
} else if (resp==JFileChooser.CANCEL_OPTION) {
    JOptionPane.showMessageDialog(null,"Se pulsó la opción Cancelar");
}

```

10. Ejecuta el código y prueba la opción “Abrir” del menú. Prueba a elegir algún fichero y abrirlo. Prueba a cancelar la ventana de apertura. Etc

11. Analicemos el código anterior:

```

int resp;
resp=elegirFichero.showOpenDialog(this);

```

- Estas dos líneas crean una variable entera *resp* (respuesta) y a continuación hacen que se muestre la ventana “Abrir Fichero”. Observa que para conseguirlo hay que usar el método *showOpenDialog* del objeto *elegirFichero*. Este método lleva como parámetro la ventana actual (*this*)
- El método *showOpenDialog* no solo muestra la ventana “Abrir Fichero” sino que también devuelve un valor entero según el botón pulsado por el usuario en esta ventana. Esto es: botón “Abrir” o botón “Calcelar”.
- Se pueden usar dos *if* para controlar lo que sucede si el usuario pulsó el botón “Abrir” o el botón “Calcelar” de la ventana “Abrir Fichero”:

```

if (resp==JFileChooser.APPROVE_OPTION) {
    JOptionPane.showMessageDialog(null,elegirFichero.getSelectedFile(
    ).toString());
} else if (resp==JFileChooser.CANCEL_OPTION) {
    JOptionPane.showMessageDialog(null,"Se pulsó la opción Cancelar");
}

```

- En el primer *if* se compara la variable *resp* con la constante *JFileChooser.APPROVE\_OPTION*, para saber si el usuario pulsó “Abrir”.
- En el segundo *if* se compara la variable *resp* con la constante *JFileChooser.CANCEL\_OPTION*, para saber si el usuario pulsó “Calcelar”.
- En el caso de que el usuario pulsara “Abrir”, el programa usa el método *getSelectedFile* del objeto *elegirFichero* para recoger el camino del fichero elegido. Este camino debe ser convertido a cadena con el método *toString*.
- El programa aprovecha esto para mostrar dicho camino en pantalla gracias al típico *JOptionPane*.

- En el caso del que el usuario pulsara el botón “Cancelar” el programa muestra un mensaje indicándolo.
12. Hay que volver a dejar claro que el cuadro de diálogo “Abrir” realmente no abre ningún fichero, sino que devuelve el camino del fichero elegido usando el código:

```
elegirFichero.getSelectedFile().toString()
```

Luego queda en manos del programador el trabajar con el fichero correspondiente de la forma que desee.

## **CONCLUSIÓN**

**Los objetos JFileChooser permiten mostrar el cuadro de diálogo “Abrir Fichero” o “Guardar Fichero”.**

**Estos objetos no abren ni guardan ficheros, solo permiten al usuario elegir el fichero a abrir o guardar de forma sencilla.**

**El JFileChooser devuelve el camino del fichero elegido, y luego el programador trabajará con dicho fichero como mejor le interese.**