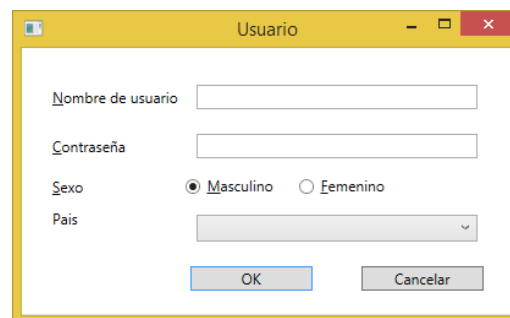


EJERCICIOS 2

Ejercicios avanzados

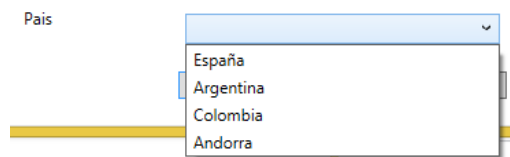
NOTA: Intenta reproducir las capturas de pantalla de forma lo más fidedigna posible.

1. Crea un dialogo como el siguiente (llámalo **Ejercicio1Dialogo**). El diálogo se encargará de encapsular datos relativos a un usuario en un objeto subyacente de clase **Usuario** (que tendrá los campos **Nombre**, **Password**, **Sexo** y **Pais**, todos ellos de tipo **String**):

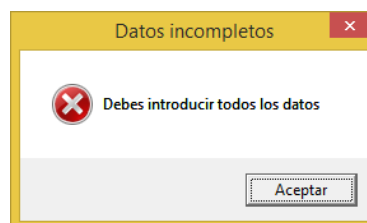


Establece mnemónicos en las etiquetas, y haz que el botón por defecto sea el de **OK**.

Los elementos del ComboBox *Pais* se definirán en XAML:

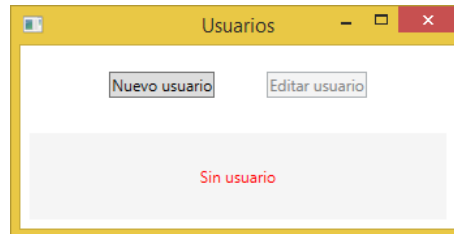


Al pulsar **OK** se comprobará si todos los campos están marcados. En caso contrario, se mostrará un mensaje indicativo como el siguiente:

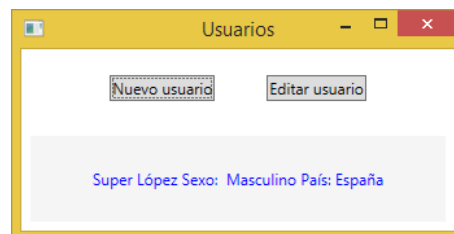


Al pulsar el botón **OK**, los datos de la interfaz se guardarán “manualmente” en el objeto de tipo **Usuario**. Es decir, los datos que contienen los controles tendrán que asignarse a los campos del objeto **Usuario**. Ten en cuenta que tendrás que hacer una conversión entre los datos contenidos en el **ComboBox Pais** (de tipo **ComboBoxItem**) con una cadena.

2. Realiza una ventana llamada **Ejercicio2** que te permita probar el diálogo anterior. La pantalla estará formada por dos botones en la parte superior y una etiqueta en la parte inferior y tendrá el siguiente aspecto. Como vemos, el botón de la derecha está deshabilitado inicialmente:

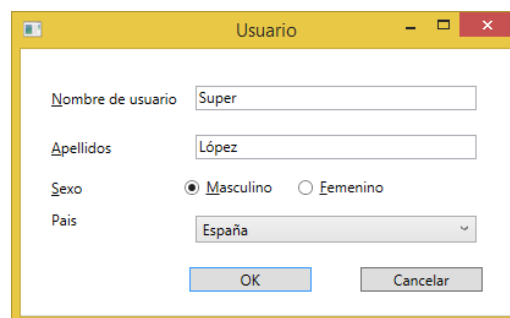


Al pulsar en *Nuevo Usuario* se mostrará el diálogo del ejercicio 1 que nos permitirá introducir los datos de un usuario. Si éstos se introducen correctamente, los datos se mostrarán en la etiqueta inferior (que además cambia de color):



Para mostrar los datos del usuario en la etiqueta, tendremos que redefinir el método **ToString()** en la clase **Usuario**.

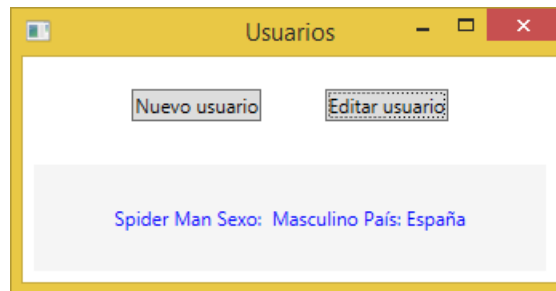
Si pulsamos en el botón *Editar usuario* (que se habilita cuando hemos creado un nuevo usuario) se abrirá la ventana del diálogo mostrando los datos del usuario que introducimos anteriormente:



Este comportamiento se logra guardando una instancia del usuario en la ventana Ejercicio2. Al pulsar el botón de *Editar* se instanciará el diálogo pasándole dicha instancia (luego tendremos que modificar el diálogo añadiendo un nuevo constructor que recibe un parámetro). En el constructor se rellenarán los datos usando el parámetro que hemos recibido.

Hay que tener en cuenta que para establecer el valor del **ComboBox** tendremos que recorrer todos los valores del mismo hasta encontrar aquel cuyo valor coincide con el país del usuario que le pasamos. Una vez encontrado, podemos seleccionar el elemento del **ComboBox** en función del índice.

Lógicamente, si cambiamos algún valor en el diálogo anterior, los datos se reflejarán en la pantalla principal:



3. Modifica el diálogo realizado en el ejercicio 1 para que los elementos del **ComboBox** sean de tipo **string** y se establezcan dinámicamente desde el *Code-Behind*. Llama al diálogo **Ejercicio3Dialogo**.

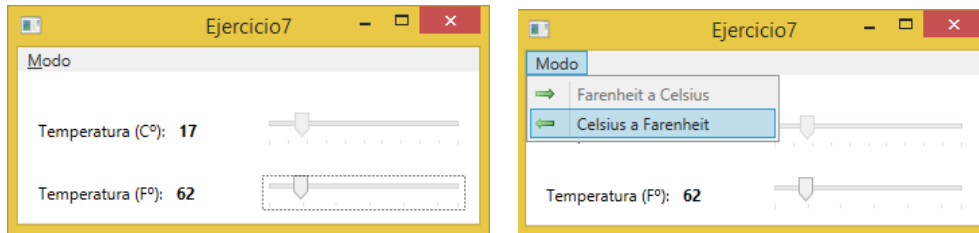
Nota: Al implementar esta versión del diálogo comprobarás que puedes asignar directamente la propiedad **País** al **ComboBox** usando el atributo **SelectedItem** del mismo, pues ambos son de tipo cadena y se pueden asignar directamente.

4. Modifica el diálogo realizado en el ejercicio anterior para que los controles de la interfaz de usuario se vinculen al modelo subyacente (el objeto **Usuario**) mediante un *binding*. Llama al diálogo **Ejercicio4Dialogo**.

A la hora de implementar el *Binding* tendrás que tener en cuenta que hay que convertir los valores del campo Sexo (que puede contener las cadenas *masculino* o *femenino*) a boolean que manejarán los radio button. Además, en cada radio button el valor se interpreta de manera distinta (no es lo mismo un true en el botón *Masculino* que en *Femenino*).

Para ello tendrás que implementar dos convertidores que llamarás **ConvertidorMasculino** y **ConvertidorFemenino** y que te permitirán realizar dicha traducción.

5. Implementa una ventana a la que llamarás **Ejercicio5** que implementará un convertidor de temperaturas como el realizado en la *Unidad Didáctica 1*:



La aplicación está formada por un menú, dos etiquetas y dos barras de desplazamiento:

- Las barras de desplazamiento muestran una temperatura expresada tanto en grados Celsius (barra superior) como en Farenheit (barra inferior). Los rangos de los sliders son 0-100 y 32-212 respectivamente. Se trata de los valores de congelación y ebullición del agua en las respectivas escalas.
- Cada etiqueta mostrará el valor que contiene su barra de desplazamiento correspondiente. Para ello se usará un *Binding* que asocie la etiqueta con la barra.
- Por defecto, la barra superior está activada y la inferior desactivada.
- Al mover la barra superior se actualizará el valor de la barra inferior. Dicha asociación también se realizará mediante un *Binding* entre ambas barras. Ten en cuenta que tendrás que implementar un convertidor de valores (*IValueConverter*) para pasar de un valor a otro, teniendo en cuenta las fórmulas para la conversión de Celsius a Farenheit y viceversa:

Fahrenheit a Celsius:

$$C = (F - 32) \cdot \frac{5}{9}$$

Celsius a Fahrenheit:

$$F = C \cdot \frac{9}{5} + 32$$

NOTA: Las fracciones decimales se implementarán como los números 0.55 y 1.8 respectivamente.

- El menú de la parte superior permite escoger el modo en que vamos a convertir la temperatura:
 - Si se escoge la opción “Fahrenheit a Celsius” se deshabilitará el slider correspondiente a Celsius y se habilita el de Farenheit. También se habilitará la opción del menú “Celsius a Fahrenheit” y se deshabilitará “Fahrenheit a Celsius”).
 - Si se escoge la opción “Celsius a Fahrenheit” se realizarán las operaciones opuestas.

¿Cuál es tu opinión respecto a Swing tras haber realizado el mismo ejercicio usando ambas tecnologías?