

Universidad Abierta Para Adultos



Formación General

CARRERA

Ingeniería en Software

ASIGNATURA

Programación II

FACILITADOR

Diógenes Amaury Martínez Silverio

PARTICIPANTE

Vladimir Núñez Calderón

MATRICULA

100057484

TEMA

Semana IV

FECHA

13 de febrero del 2023

Santiago de los Caballeros, República Dominicana

INTRODUCCION

El componente ComboBox es un elemento común en la interfaz de usuario de muchas aplicaciones de software. Es una combinación de una lista desplegable y un cuadro de texto, lo que permite al usuario seleccionar una opción de una lista o escribir una entrada personalizada. Este componente se utiliza en muchas situaciones diferentes, como la selección de una fecha, una categoría, un estado, una ubicación, entre otros.

INVESTIGAR EN LA WEB ACERCA DEL COMPONENTE COMBOBOX, VENTAJAS, DESVENTAJAS, FORMA DE USO

El componente ComboBox es un elemento común en la interfaz de usuario de muchas aplicaciones de software. Es una combinación de una lista desplegable y un cuadro de texto, lo que significa que el usuario puede seleccionar una opción de una lista o escribir una entrada personalizada.

Ventajas:

- Permiten una selección fácil y rápida de opciones, lo que puede mejorar la eficiencia del usuario.
- Pueden ahorrar espacio en la pantalla al compararse con una lista larga de opciones.
- Permiten la entrada de una opción personalizada si la opción deseada no está disponible en la lista.
- Proporcionan una forma intuitiva de seleccionar una opción, lo que puede mejorar la accesibilidad para usuarios con discapacidades.

Desventajas:

- Pueden ser confusos para algunos usuarios si no se utilizan de manera clara y coherente en la interfaz de usuario.
- Pueden requerir más tiempo para ser completados si la lista de opciones es larga o si se requiere la entrada de una opción personalizada.
- Pueden ser menos accesibles para usuarios con discapacidades visuales si la lista de opciones no está bien implementada.

En cuanto a la forma de uso, el ComboBox se puede utilizar en muchas situaciones diferentes, como la selección de una fecha, una categoría, un estado, una ubicación, entre otros. Se puede implementar en muchas plataformas y lenguajes de programación diferentes, y su uso y apariencia puede variar dependiendo de la plataforma y el lenguaje. Sin embargo, en general, un ComboBox se puede utilizar de la siguiente manera:

1. Hacer clic en el ComboBox para abrir la lista de opciones.

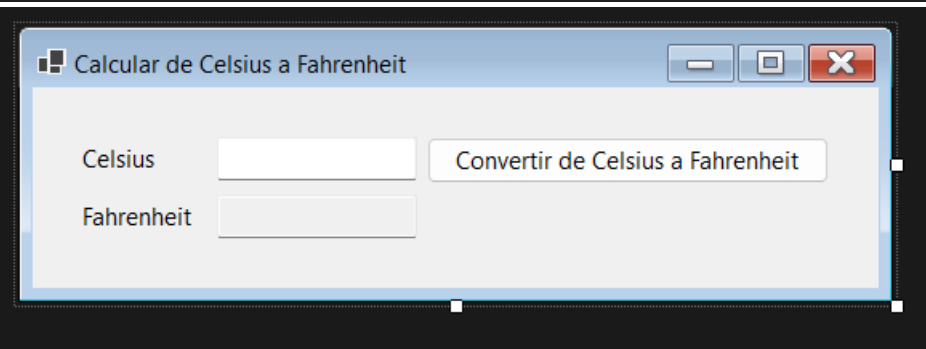
2. Hacer clic en una opción de la lista o escribir una entrada personalizada en el cuadro de texto.
3. Hacer clic en un botón o pulsar la tecla Enter para confirmar la selección.

Realiza un programa que permita convertir de Celsius a Fahrenheit. Para lograr esto, tenemos que multiplicar primero Celsius por $9/5$ y luego sumarle 32. Lo único que el usuario debe hacer para utilizar este programa es digitar la cantidad de Celsius y al pulsar clic en el botón Convertir a Fahrenheit, se mostrará el resultado.

```
1 referencia
private void btnConvertir_Click(object sender, EventArgs e)
{
    float celcius, fahrenheit;
    celcius = Convert.ToInt32(txtCelcius.Text);

    fahrenheit = (celcius * 9 / 5) + 32;

    txtFahrenheit.Text = fahrenheit.ToString();
}
```

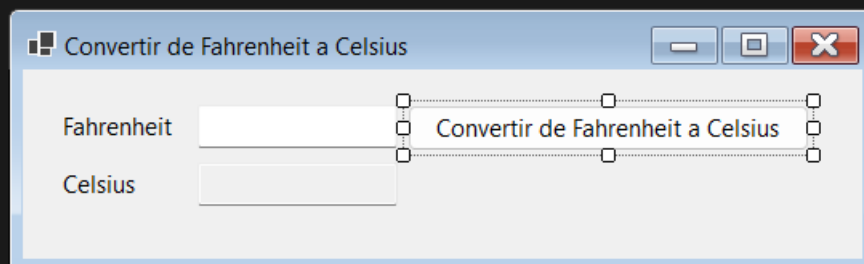


The screenshot shows a Windows application window titled "Calcular de Celsius a Fahrenheit". The window has a standard Windows title bar with minimize, maximize, and close buttons. Inside the window, there are two text input fields. The first field is labeled "Celsius" and the second field is labeled "Fahrenheit". To the right of the "Celsius" field is a button labeled "Convertir de Celsius a Fahrenheit".

Realiza un programa que permita convertir de Fahrenheit a Celsius. Para lograr esto, tenemos que restar 32 Fahrenheit, luego multiplicarlo por 5 y dividir el resultado entre 9 al. Lo único que el usuario debe hacer para utilizar este programa es digitar la cantidad de Fahrenheit y al pulsar clic en el botón Convertir a Celsius, se mostrará el resultado.

```
1 referencia
private void btnConvertir_Click(object sender, EventArgs e)
{
    float fahrenheit, celcius;
    fahrenheit = Convert.ToInt32(txtFahrenheit.Text);
    celcius = (fahrenheit - 32) * 5 / 9;

    txtCelsius.Text = celcius.ToString();
}
```



The screenshot shows a Windows application window with the title "Convertir de Fahrenheit a Celsius". The window has a standard Windows XP-style title bar with minimize, maximize, and close buttons. Inside the window, there are two text input fields. The first field is labeled "Fahrenheit" and is empty. The second field is labeled "Celsius" and is also empty. To the right of these fields is a button with the text "Convertir de Fahrenheit a Celsius". The button has a light blue background and a thin border. The window itself has a light blue border and a white background.

Realiza un programa que permita convertir dependiendo del tipo seleccionado. Lo único que el usuario debe hacer para utilizar este programa es Seleccionar el tipo de conversión, digitar la cantidad de grados y pulsar clic en el botón Convertir, se mostrará el resultado.

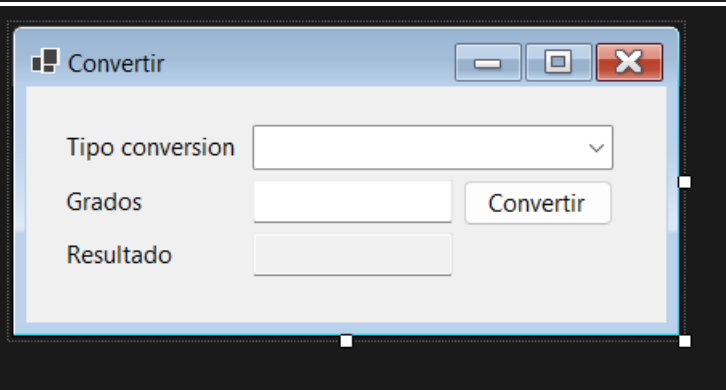
```
1 referencia
private void btnConvertir_Click(object sender, EventArgs e)
{
    string tipo;
    float grados, resultado;

    tipo = sltTipoConversion.Text;
    grados = Convert.ToInt32(txtGrados.Text);

    switch (tipo)
    {
        case "Celsius a Fahrenheit":
            resultado = (grados * 9 / 5) + 32;

            txtResultado.Text = resultado.ToString();
            break;
        case "Fahrenheit a Celsius":
            resultado = (grados - 32) * 5 / 9;

            txtResultado.Text = resultado.ToString();
            break;
    }
}
}
```



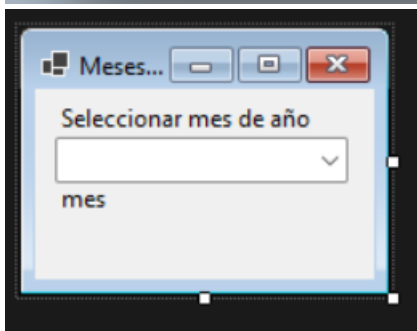
The screenshot shows a Windows application window titled "Convertir". Inside the window, there is a dropdown menu labeled "Tipo conversion" with a downward arrow. Below it is a text input field labeled "Grados". To the right of the "Grados" field is a button labeled "Convertir". Below the "Grados" field is another text input field labeled "Resultado". The window has standard Windows window controls (minimize, maximize, close) in the top right corner.

Utilizando un comboBox mostrar los numero de los meses del año 1 al 12, luego asignarle un evento al objeto, de manera que si por ejemplo seleccione el 1 en un label muestre Enero, si es 2 Febrero, si es 3 Marzo y así sucesivamente.

```
private void frmMeses_Load(object sender, EventArgs e)
{
    lblMes.Text = "";
    for(int i = 1; i <= 12; i++)
    {
        sltNumeroMes.Items.Add(i);
    }
}

private void sltNumeroMes_SelectedIndexChanged(object sender, EventArgs e)
{
    string mes;
    mes = sltNumeroMes.Text;

    switch (mes)
    {
        case "1":
            lblMes.Text = "Enero";
            break;
        case "2":
            lblMes.Text = "Febrero";
            break;
        case "3":
            lblMes.Text = "Marzo";
            break;
        case "4":
            lblMes.Text = "Abril";
            break;
        case "5":
            lblMes.Text = "Mayo";
            break;
        case "6":
            lblMes.Text = "Junio";
            break;
        case "7":
            lblMes.Text = "Julio";
            break;
        case "8":
            lblMes.Text = "Agosto";
            break;
        case "9":
            lblMes.Text = "Septiembre";
            break;
        case "10":
            lblMes.Text = "Octubre";
            break;
        case "11":
            lblMes.Text = "Noviembre";
            break;
        case "12":
            lblMes.Text = "Diciembre";
            break;
    }
}
```



CONCLUSIÓN

En general, el ComboBox es una herramienta útil y versátil en la interfaz de usuario que puede mejorar la eficiencia y accesibilidad para los usuarios. Sin embargo, también hay algunas desventajas, como la posibilidad de confusión para algunos usuarios y la necesidad de más tiempo para completar si la lista de opciones es larga. En última instancia, la implementación efectiva del ComboBox depende del contexto y las necesidades específicas de la aplicación y los usuarios.