

Forum:[Hack Forums](#)

Thread: [ProgressBar control with text / percentage!](#)

Delirium™ - October 27, 2014, 9:29 pm

Hello.

I was in need of a progress bar with the ability to display the percentage on it and using a label onto a default progress bar sounded like a dumb method therefore, I wrote my own custom progress bar control with the ability to display text or percentage built-in.

Here's the source:

```
Code using System;
using System.ComponentModel;
using System.Drawing;
using System.Runtime.InteropServices;
using System.Windows.Forms;

public enum ProgressBarDisplayStyle
{
    Percentage,
    Text
}

class TextProgressBar : ProgressBar
{
    [DllImport("uxtheme.dll")]
    private static extern int SetWindowTheme(IntPtr hWnd, string appname, string idlist);

    private ProgressBarDisplayStyle _DisplayStyle = ProgressBarDisplayStyle.Text;

    [Browsable(true)]
    [Category("ProgressBar Style")]
    public ProgressBarDisplayStyle DisplayStyle
    {
        get { return _DisplayStyle; }
        set
        {
            _DisplayStyle = value;
            Invalidate();
        }
    }

    private string _Text = null;

    [Browsable(true)]
    [Category("ProgressBar Style")]
    public string Text
```

```

{
    get { return _Text; }
    set { _Text = value; Invalidate(); }
}

protected override void OnHandleCreated(EventArgs e)
{
    SetWindowTheme(Handle, "", "");
    base.OnHandleCreated(e);
}

public TextProgressBar()
{
    SetStyle(ControlStyles.UserPaint | ControlStyles.AllPaintingInWmPaint, true);
    Text = Name;
}

protected override void OnPaint(PaintEventArgs e)
{
    Rectangle R = ClientRectangle;
    Graphics G = e.Graphics;

    ProgressBarRenderer.DrawHorizontalBar(G, R);

    if (Value > 0)
    {
        Rectangle R2 = new Rectangle(R.X, R.Y, (int)Math.Round(((float)Value / Maximum) *
R.Width), R.Height);
        ProgressBarRenderer.DrawHorizontalChunks(G, R2);
    }

    string ProgressText = (DisplayStyle == ProgressBarDisplayStyle.Text) ? Text : Value + "%";

    Font F = new Font("Segoe UI", 10);
    SizeF TextSize = G.MeasureString(ProgressText, F);
    Point Center = new Point(Convert.ToInt32((Width / 2) - TextSize.Width / 2),
Convert.ToInt32((Height / 2) - TextSize.Height / 2));

    G.DrawString(ProgressText, F, Brushes.Black, Center);
    F.Dispose();
}
}

```

Nothing significant at all but could be useful to a couple of people. Enjoy.

Hiiro - October 28, 2014, 5:59 pm

What is uxtheme.dll?

Should I already have this library on my computer?

EDIT: Found it in System32, Thanks :P

Delirium™ - October 28, 2014, 7:39 pm

(Oct 28, 2014 05:59 PM)Austrix Wrote: [\[CITE\]](#) What is uxtheme.dll?

Should I already have this library on my computer?

EDIT: Found it in System32, Thanks :P

It's part of the Windows OS which handles painting and stuff.

The code which utilizes the Uxtheme.dll prevents the progress bar from flickering on the designer and the form.

WZDY - October 29, 2014, 12:45 pm

A couple of days ago I made a progressbar with % and it was like a few lines of code.

How come your be so much? Anyways, thanks for the share but you could've made this alot easier.

Hiiro - October 29, 2014, 1:02 pm

(Oct 29, 2014 12:45 PM)The Flash ? Wrote: [\[CITE\]](#) A couple of days ago I made a progressbar with % and it was like a few lines of code.

How come your be so much? Anyways, thanks for the share but you could've made this alot easier. It just looks like a lot because he's calling OnHandleCreated and centering his text, etc.

WZDY - October 29, 2014, 1:12 pm

He could just do it in the Properties of the textbox?

Only3 - October 29, 2014, 1:23 pm

(Oct 29, 2014 01:12 PM)The Flash ? Wrote: [\[CITE\]](#) He could just do it in the Properties of the textbox?

? what textbox

want short code

```
Code class TextProgressBar : ProgressBar
{
```

```
    public TextProgressBar()
    {
        SetStyle(ControlStyles.UserPaint | ControlStyles.AllPaintingInWmPaint |
ControlStyles.OptimizedDoubleBuffer, true);
    }

    protected override void OnPaint(PaintEventArgs e)
    {
        Graphics G = e.Graphics;
        ProgressBarRenderer.DrawHorizontalBar(G, ClientRectangle);
        if (Value > 0) {
            ProgressBarRenderer.DrawHorizontalChunks(G, new Rectangle(ClientRectangle.X,
ClientRectangle.Y, Convert.ToInt32(Math.Round((Convert.ToSingle(Value) / Maximum) *
ClientRectangle.Width)), ClientRectangle.Height));
        }
        SizeF TextSize = G.MeasureString(Value, new Font("Segoe UI", 10));
        G.DrawString(Value, new Font("Segoe UI", 10), Brushes.Black, new Point((Width / 2) -
(TextSize.Width / 2), (Height / 2) - (TextSize.Height / 2)));
    }
}
```

Delirium™ - November 2, 2014, 2:57 am

(Oct 29, 2014 01:12 PM)The Flash ? Wrote: [\[CITE\]](#) He could just do it in the Properties of the

textbox?

I am not using any textbox. I am drawing and centering the text manually. The code looks long probably because of the newlines I'm using to make the code cleaner and more beautiful. Or maybe because I have also implemented the ability to add custom text instead of percentage which takes up a few more lines.