

Forum:[Hack Forums](#)

Thread: [How to change background colour across multiple forms?](#)

Thoma\$ - August 10, 2016, 4:12 am

So i'm trying to create a series of radio buttons on one form which will then change the background colour of all forms in the program to the corresponding selection. this.backcolor only applies the colour to the form it's located on so that's not what im looking for. Any help would be appreciated.

Zilog80 - August 10, 2016, 6:12 am

Can you provide some source code to check what are you doing?

Nikola ?esla - August 10, 2016, 6:46 am

Form1.BackColor , Form2.BackColor will work.

GuRuHaX - August 10, 2016, 9:43 am

The above would work and its an EZ solution however i think there is another way as well ,

As i did not tested following but it would seem logical and more elegant way of doing this , is to make a BaseForm , let all the other forms you want to change color of inherit from that BaseForm , when you change the color of the BaseForm => all derived forms will change as well.

You can also work with an event that is triggered on the change of the color , pass the color (code or whatever it is your using) as parameter and that should work as well.

More about inheritance :

[https://msdn.microsoft.com/en-us/library/bx1155fz\(v=vs.110\).aspx](https://msdn.microsoft.com/en-us/library/bx1155fz(v=vs.110).aspx)

?

Nick (Name) - August 10, 2016, 12:35 pm

I'm guessing your trying to change back color of all forms that are open and visible to the user, here's an easy foreach loop to change the background of all open forms...

PHP Code

```
<?php
foreach (Form f in Application.OpenForms)
{
    f.BackColor = System.Drawing.Color.Grey
    return;
}
```

Chappie - August 10, 2016, 7:27 pm

It's this stuff that makes me wonder why people still use WinForms. In WPF this could be easily accomplished without needing any code-behind directly manipulating the user controls.

0xDEAD10CC - August 10, 2016, 7:27 pm

If it's an option that you just set one time, then instead of setting them at some later point in runtime, why not just create your own container/form?.

(Aug 10, 2016 07:27 PM)Schlemiel Wrote: [\[CITE\]](#) It's this stuff that makes me wonder why people still use WinForms. In WPF this could be easily accomplished without needing any code-behind. Who cares? You still need XAML.

+Haze - August 10, 2016, 7:30 pm

(Aug 10, 2016 06:46 AM)Nikola ?esla Wrote: [\[CITE\]](#) Form1.BackColor , Form2.BackColor will work. No it won't actually, that only works in vb. That shouldn't even be a thing because it confuses developers who are used to OOP programming.

Chappie - August 10, 2016, 7:33 pm

(Aug 10, 2016 07:27 PM)0xDEAD10CC Wrote: [\[CITE\]](#) Who cares? You still need XAML. You don't need the unnecessary smell. You can have static resources for setting control properties. I just don't see the point in using something out-dated by 20 years when objectively better things are available. Especially if you are going to be manipulating the view like OP is.

0xDEAD10CC - August 10, 2016, 8:17 pm

(Aug 10, 2016 07:33 PM)Schlemiel Wrote: [\[CITE\]](#) You don't need the unnecessary smell. You can have static resources for setting control properties. I just don't see the point in using something out-dated by 20 years when objectively better things are available. Especially if you are going to be manipulating the view like OP is.

All he's doing is changing the color of the background lol... Honestly for something this minimal in terms of graphic manipulation, what benefit will WPF have over WinForms? There's absolutely none. It's not like there's massive amount of graphics processing to be done here. WPF does better with more intricate GUI's, but there's hardly any difference with something this insignificant.

Btw, WinForms is not outdated and probably never will be unless Microsoft makes considerable change to the core of their operating system because they would break a lot of backwards compatible features and existing ones for the desktop if they were to remove the components that WinForms rely on. We can all take this "out-dated" vocabulary as a matter of of your misinformed opinion in this case; you can't consider something out-dated just because it's been around for a long time and it still works, and especially when the internals of .NET 4.0 have written components for GDI showing that Microsoft is still in no way ready to give up support for it.

Chappie - August 10, 2016, 9:45 pm

(Aug 10, 2016 08:17 PM)0xDEAD10CC Wrote: [\[CITE\]](#) All he's doing is changing the color of the background lol... Honestly for something this minimal in terms of graphic manipulation, what benefit will WPF have over WinForms? There's absolutely none. It's not like there's massive amount of graphics processing to be done here. WPF does better with more intricate GUI's, but there's hardly any difference with something this insignificant.

He's changing the background color across multiple forms which leads me to believe he is also doing other things involving the view. It's not just the simple task, it's that based on even wanting to do this I am assuming much more is also being done or will be done. WinForms is good for fast development. WPF is good for a more interactive(For lack of a better word) interface. Based on the OP, I believe it is reasonable to assume he is most likely aiming for something that is, or is going to be, quiet involved with the UI - Which WPF is great for.

Side Note

Spoiler ([Click to View](#))

Look. I'm not trying to be one of those faggots who come in preaching about MVVM and try to force people to implement a bunch of ICommand's to make a simple button click fire off a function. All I am saying is even if you don't want to fully embrace the patterns WPF tries to encourage, you can still use it practically the same as WinForms but be using something with other features that can be

used if need be.

Thoma\$ - August 10, 2016, 11:37 pm

Thought i'd give you guys some context. I'm making a simple game for my school project and theres a customization menu which contains all the radio buttons which will change the form's colours. Gonna try some of the solutions mentioned now.

+Haze - August 11, 2016, 11:52 am

(Aug 10, 2016 12:35 PM)Nick (Name) Wrote: [\[CITE\]](#) I'm guessing your trying to change back color of all forms that are open and visible to the user, here's an easy foreach loop to change the background of all open forms...

PHP Code

```
<?php
foreach (Form f in Application.OpenForms)
{
    f.BackColor = System.Drawing.Color.Grey
    return;
}
```

Why make it return after the first iteration?