

Forum:[Hack Forums](#)

Thread: [ImageToCode \(and CodeToImage\): Embed images in class files!](#)

Meerkat.? - June 11, 2012, 12:21 pm

Hello everyone. I've always wanted to embed small images in my themes (icons, logo's, whatever) but I hate hardcoding resource names in my classes.

Of course you can give the user some .png files and let them add them to the resources, but I wanted something better :)

This class can convert images into code (suitable for Visual Studio, I'll explain that later) and convert them back into images.

This way you can embed little icons in your class file!

Visual studio has a maximum line length of 65530 (or something like that). To prevent errors from happening I've added an option to output a VS-compatible string.

To use this, just add the class and use this code:

Convert an image into code:

```
Code Dim ITC As New ImageToCodeClass
```

```
Dim Code as String = ITC.ImageToCode(YourImage)
```

Convert the code into an image:

```
Code Dim ITC As New ImageToCodeClass
```

```
PictureBox1.Image = ITC.CodeToImage("<YOUR IMAGE CODE HERE>")
```

Download:

Spoiler ([Click to View](#))Code Public Class ImageToCodeClass

```
Public Function ImageToCode(ByVal Img As Bitmap) As String
```

```
Return
```

```
Convert.ToBase64String(DirectCast(System.ComponentModel.TypeDescriptor.GetConverter(Img).
```

```
ConvertTo(Img, GetType(Byte())), Byte()))
```

```
End Function
```

```
Public Function CodeToImage(ByVal Code As String) As Image
```

```
Return Image.FromStream(New System.IO.MemoryStream(Convert.FromBase64String(Code)))
```

```
End Function
```

```
End Class
```

Note: This isn't supposed to be top-notch coding. It's made to work, not made to be 100% perfect and professional. I made it for personal purposes and just decided to share this with you guys.

EDIT: Completely new class, super small yet much better. Better performance, smaller string,...

Thanks for the suggestions!

aTastyUnicorn - June 11, 2012, 12:31 pm

saw this on youtube, really nice work

keep up mate

DotNetNut - June 11, 2012, 12:56 pm

Where did you find out about these image codes? Never herd of them before and still not really sure why this is better?

I mean nice app and video but how? , why? lol :P

Deceptive - June 11, 2012, 12:59 pm

You're just getting each pixel, and logging their RGB right? :3

DotNetNut - June 11, 2012, 1:02 pm

(Jun 11, 2012 12:59 PM)Deceptive Wrote: [\[CITE\]](#) You're just getting each pixel, and logging their RGB right? :3

Yea thats what I got from the pastebin link, But why? whats the benefit?

I mean why not into a byte array or something?

Deceptive - June 11, 2012, 1:12 pm

(Jun 11, 2012 01:02 PM)DotNetNut Wrote: [\[CITE\]](#) Yea thats what I got from the pastebin link, But why? whats the benefit?

I mean why not into a byte array or something?

Evidently so he doesn't have to store his images in resources.

DotNetNut - June 11, 2012, 1:18 pm

(Jun 11, 2012 01:12 PM)Deceptive Wrote: [\[CITE\]](#) Evidently so he doesn't have to store his images in resources.

Oh.. ok lol, I thought it would of been easier to convert from Image > Bytearray > string?

I take it doing it that way, as i suggested above would make it a massive sting.. And also i wonder why he dose not want to store in resources? I mean is that not what their for?

I guess i just want to know if this method is actually better or ? lol.

confused... :P

Meerkat.? - June 11, 2012, 1:20 pm

(Jun 11, 2012 01:12 PM)Deceptive Wrote: [\[CITE\]](#) Evidently so he doesn't have to store his images in resources.

Exactly. I want to include small images into themes and still be able to just post a pastebin link.

There's probably lots of better ways to do this, and my method with splitting things probably isn't the most efficient, but it works :D

DotNetNut - June 11, 2012, 1:21 pm

(Jun 11, 2012 01:20 PM)Mavamaarten~ Wrote: [\[CITE\]](#) Exactly. I want to include small images into themes and still be able to just post a pastebin link.

There's probably lots of better ways to do this, and my method with splitting things probably isn't the most efficient, but it works :D

Right i got you know :D lol, I thought this was just a way for you to store them when making new apps lulz, But i can see the benefit of it now :P

DotNetNut - June 11, 2012, 1:38 pm

I just tried a few of these and omg the string are fucking huge!

I tried to post one string as an example but it was apparently 16300 chars over the limit on a 60kb image lul..

EDIT : And i see you have gone to the dark side.. Enjoy your uber :D

Support. - June 12, 2012, 3:36 am

Ahhh, another good release, will definitely have to have a play with this!

marvinn - June 12, 2012, 3:40 am

Why not just using a binarySerializer and serialize it to a byte array and convert that to a base64 string.. I think it could take like 5 lines of code..

Anyways, gj

?ndromeda - June 12, 2012, 3:43 am

I can't call this Stenography and still. This is amazing. Thank you.

lolcakes131 - June 12, 2012, 3:52 am

(Jun 12, 2012 03:40 AM)marvinn Wrote: [\[CITE\]](#) Why not just using a binarySerializer and serialize it to a byte array and convert that to a base64 string.. I think it could take like 5 lines of code..

Anyways, gj

This is what I'd suggest too (perhaps not serializing, but just converting the image to it's byte representation and then to a Base64 string), it's how resources in Visual Studio are stored in RESX files. Iterating through each pixel isn't very efficient and uses a lot of resources, not to mention the string this produces can be ridiculously large, but then again, you did explicitly mention "this isn't supposed to be top-notch coding".

Washi - June 12, 2012, 5:06 am

I like the idea, but the strings become loooonng..

Maybe you can take a look at compressing the strings with a GZip or Deflate Stream.

Meerkat.? - June 12, 2012, 7:11 am

(Jun 12, 2012 05:06 AM)TheUnknownProgrammer Wrote: [\[CITE\]](#) I like the idea, but the strings become loooonng..

Maybe you can take a look at compressing the strings with a GZip or Deflate Stream.

Hmm, good idea. As I said before, I didn't want the best possible solution but you're certainly right. BTW, I just rewrote the whole class with a Base64 string. It works faster and most of all, the string is MUCH shorter :)

(40k chars for a 32x32 image, it used to be 200k chars)

hoocrax - June 12, 2012, 7:48 am

This is just simply jaw dropping ! Good Job ! I will definitely try to use it some time !

Machine gun kelly - June 12, 2012, 8:11 am

really nice work... keep it up man!

marvinn - June 12, 2012, 8:51 am

(Jun 12, 2012 07:11 AM)Mavamaarten~ Wrote: [\[CITE\]](#) Hmm, good idea. As I said before, I didn't want the best possible solution but you're certainly right.

BTW, I just rewrote the whole class with a Base64 string. It works faster and most of all, the string is MUCH shorter :)

(40k chars for a 32x32 image, it used to be 200k chars)

So you did what I said? Lol :)

```
Code public string img2str(Image img)
{
    System.IO.MemoryStream m = new System.IO.MemoryStream();
    new BinaryFormatter().Serialize(m, img);
    return Convert.ToBase64String(m.ToArray());
}
```

Clean simple & easy :) voila.

Meerkat.? - June 12, 2012, 9:13 am

(Jun 12, 2012 08:51 AM)marvinn Wrote: [\[CITE\]](#) So you did what I said? Lol :)

Clean simple & easy :)

I did, a little bit different and even shorter haha

```
Code Public Class ImageToCodeClass
```

```
    Public Function ImageToCode(ByVal Img As Bitmap) As String
```

```
    Return
```

```
Convert.ToBase64String(DirectCast(System.ComponentModel.TypeDescriptor.GetConverter(Img).
```

```
ConvertTo(Img, GetType(Byte())), Byte()))
```

```
    End Function
```

```
    Public Function CodeToImage(ByVal Code As String) As Image
```

```
    Return Image.FromStream(New System.IO.MemoryStream(Convert.FromBase64String(Code)))
```

```
    End Function
```

```
End Class
```

ubanooba - June 12, 2012, 10:23 am

Why convert to a Base64 string? Greatly increases the required memory when performing the conversion. Better to use something like the following:

```
Code Public NotInheritable Class InlineImaging
```

```
    Private Shared _converter As New ImageConverter()
```

```
    Public Shared Function ImageToCode(ByVal image As Image) As Byte()
```

```
    Return _converter.ConvertTo(image, GetType(Byte()))
```

```
    End Function
```

```
    Public Shared Function CodeToImage(ByVal code As Byte()) As Image
```

```
    Return New Bitmap(New IO.MemoryStream(code))
```

```
    End Function
```

```
End Class
```

Then, to put it into your project do something like:

```
Code Dim blcon As Byte() = { 1, 1, 1, 1 } 'output from ImageToCode
```

To easily work out the Byte array just do something like:

```
Code Clipboard.SetText(String.Format("{ {0} }", String.Join(", ",
```

```
Array.ConvertAll(InlineImaging.ImageToCode(Image.FromFile("B:\example.png")), Function(b As Byte) b.ToString()))
```

Meerkat.? - June 12, 2012, 10:34 am

(Jun 12, 2012 10:23 AM)ubanooba Wrote: [\[CITE\]](#) Why convert to a Base64 string? Greatly increases the required memory when performing the conversion. Better to use something like the following:

Because it's fucking meant to embed images to themes. Theme files (class files) are text only, so you simply cannot store a byte array in there -."

ubanooba - June 12, 2012, 11:06 am

(Jun 12, 2012 10:34 AM)Mavamaarten~ Wrote: [\[CITE\]](#) Because it's fucking meant to embed images to themes. Theme files (class files) are text only, so you simply cannot store a byte array in there -." Sigh, please open your eyes. You write your classes in text, yes, but that doesn't mean you have to define everything as a string. If you actually read my post, you convert the image to bytes, and then you can turn that byte array into an initializer list for a byte array. More memory efficient both ways.

PI3 - June 22, 2012, 12:54 pm

What the fuck is this? What a useless tool. And in vb.net. Feel ashamed kid.

fillobotto - June 22, 2012, 12:58 pm

(Jun 22, 2012 12:54 PM)PI3 Wrote: [\[CITE\]](#) What the fuck is this? What a useless tool. And in vb.net. Feel ashamed kid.

Deceptive:

Evidently so he doesn't have to store his images in resources.

simonbenyo - June 22, 2012, 1:04 pm

Thanks Br0 its so cool ,, Keep up the good work ?

Meerkat.? - June 23, 2012, 12:49 am

(Jun 22, 2012 12:54 PM)PI3 Wrote: [\[CITE\]](#) What the fuck is this? What a useless tool. And in vb.net. Feel ashamed kid.

You probably don't even know what a byte is yet you're talking shit on my thread, lol.

gungunfebrianza - November 8, 2012, 2:47 am

whooooaaa i love this mavamarteen ?
thanks for sharing :)

432 - November 8, 2012, 7:14 pm

Can i get the source please ?

gungunfebrianza - November 8, 2012, 7:18 pm

(Nov 8, 2012 08:14 PM)432 Wrote: [\[CITE\]](#) Can i get the source please ?
hei you can see the source of class in the first page sir ?
?

Secret Tiddmore - November 8, 2012, 7:52 pm

This snippet code helped me make the images to my theme faster instead of drawing lines and so forth. Good job.

Thank yer,
S. Tiddmore

432 - November 9, 2012, 7:34 am

(Nov 8, 2012 08:18 PM)gungunfebrianza Wrote: [\[CITE\]](#) hei you can see the source of class in the first page sir ?
?
it's doesn't work

longtrenton1242 - November 9, 2012, 8:01 am

Really cool, But not what i expected when you said imageTOtext.

Meerkat.? - November 9, 2012, 8:05 am

(Nov 9, 2012 08:34 AM)432 Wrote: [\[CITE\]](#) it's doesn't work

If that's true, you have no idea what you're doing.

Shaxine - November 16, 2012, 6:03 pm

Thanks Mavamaarten~

I have already made a program similar to yours to help me in my future projects, and I will give you credit when I use your Technique! ;)

eSp0rt - November 17, 2012, 2:51 am

Good Release dude ! I think using this in my future program.

li0nsar3c00l - February 13, 2013, 10:08 am

because im using it now, i want to say thanks one more time to your awesome work :D