

wx.iconfrombitmap(wx.bitmap(path)) python で使用する画像を埋め込む方法

質問 7年前、6か月前 7年前、5か月前に変更されました 2k回 閲覧



3



[私はここ](#)から入手し、必要に応じて変更した Python アプリを持っていますが、アイコンに使用する画像をスクリプト内に配置したいと考えています。パフォーマンスの問題が発生する可能性があることは承知していますが、現時点では問題ではありません。必要なのは、すべてのファイルを1つにまとめておくことだけです。そうすれば、友人やコンピューター間で簡単に変更できます。そこで私の質問は、ファイル自体の代わりにファイルへのパスを取得するこのコード行で、埋め込まれたアイコン画像(png ファイル、任意のタイプ)を使用するにはどうすればよいかということです。

```
def set_icon(self, path):  
    icon = wx.IconFromBitmap(wx.Bitmap(path))  
    self.SetIcon(icon, TRAY_TOOLTIP)
```

img2py、base64 を試しましたが、何も機能しませんでした。そのため、必要なのはファイルではなくパスである可能性があると考えました。パスではなく埋め込みファイルで動作するようにスクリプト (または wx ライブラリ) を変更/修正するにはどうすればよいですか?

パイソン

wxpython

埋め込み

共有 この質問を改善する

フォローする

2017年5月23日 11:53に編集



コミュニティ

ボット

1 • 1

2017年2月25日 13:31に 質問



セルソ・カルデラ

31 • 4

1 回答

並び替え: 最高得点 (デフォルト)



4



私が通常これを実現する方法は、base64とzlibを組み合わせることです

埋め込み画像データを生成するには

```
from zlib import compress  
from base64 import b64encode  
with open("image.png", "rb") as fileobj:  
    data = b64encode(compress(fileobj.read()))
```

三重引用符を使用して「data」をスクリプトにコピーし、必要に応じて改行を挿入します。

```
data = """Tm90IGFjdHVhbCBwaWN0dXJlIGRhdGE="""
```

画像データを再作成する必要がある場合は、エンコードと圧縮を逆にするだけです

```
from base64 import b64decode
from zlib import decompress
image_data = decompress(b64decode(data))
```

ビットマップオブジェクトを再作成する方法は、wxpythonのバージョンによって異なります。フェニックスビルドを使用すると、ストリームからwx.Imageを作成し、次にイメージからwx.Bitmapを作成できます。

```
from io import BytesIO
stream = BytesIO(bytearray(image_data)) # just bytes() for py3
image = wx.Image(stream, wx.BITMAP_TYPE_ANY) # wx.ImageFromStream for legacy wx
bitmap = wx.Bitmap(image) # wx.BitmapFromImage for legacy wx
```

私の使用状況では、複数の埋め込みファイルを使用してもパフォーマンスへの影響は感じられませんでした。

共有 この回答を改善する

2017年3月12日 17:21に編集

2017年2月25日 16:30に 回答

フォローする



ユーザー2682863

3,187 ● 1 ● 30 ● 39

-
- 2 wxPythonには、画像ファイルをオブジェクトとしてPythonモジュールにエンコードするのに役立つツールがいくつか含まれています。このオブジェクトには、それを、など `PyEmbeddedImage` に変換するメソッドがあります。 `wx.Image wx.Bitmap python -m wx.tools.img2py --help`
- [ロビンダン](#) 2017年2月25日 21:04
-