**Challenge 12**

[www.pythonchallenge.com/pc/return/evil.html](http://www.pythonchallenge.com/pc/return/evil.html)



This picture is a fuzzy picture of somebody dealing out cards into (I think) 5 hands. The title is ‘dealing evil’. Clicking on it didn’t do anything. The clue in the page source is the name of the image ‘evil1.jpg’. So are there evil2, 3, . . . images? Yes, there are.

. . . evil2.jpg is a picture that says ‘not jpg \_.gfx’

When I typed in .gfx instead in the browser, it just stays on the .jpg page. Evil3 says ‘no more evils. . .’ There is evil4 – a black screen with little white square in the middle.

This one completely flummoxed me. I googled .gfx, which there was a python thing for but I couldn’t make much of it. Had to look at just about every solution site out there. After lots of research and playing, finally boiled it all down like this:

1. Took requests.get(‘. . .evil2.gfx’).content, got bytes, e.g. \xff, . . .
2. This data is actually 5 different image files in different formats interleaved. In order, there’s a jpg, a png, a gif, a png, and a jpg.
3. So the first file is gfx[0::5], second is gfx[1::5], etc.

I thought all this was really interesting, since I’ve never gotten into the different image formats before. The is a website for a recovery project that I thought was interesting. It has a user guide with the formats for a bunch of file types: [*https://www.file-recovery.com/xxx-signature-format.htm*](https://www.file-recovery.com/xxx-signature-format.htm)

1. I’m sure there are other places, too, where this stuff can be found.

The ‘headers’ I printed out looked like this:

Header 0: b'\xff\xd8\xff\xe0\x00\x10JFIF\x00\x01\x01'

Header 1: b'\x89PNG\r\n\x1a\n\x00\x00\x00\rI'

Header 2: b'GIF87a@\x01\xf0\x00\xe7\x00\x00'

Header 3: b'\x89PNG\r\n\x1a\n\x00\x00\x00\rI'

Header 4: b'\xff\xd8\xff\xe0\x00\x10JFIF\x00\x01'

*# jpg header: FF D8 -- FF trailer is FF D9  
# png \211 P N G \r \n \032 \n  
# GIF8 -- next two are version -- like 9a*

1. After I created the five image files, I used **cv2** to try to display them, but bombed after a couple. In pycharm editor, I could see all but one. PCPaint showed all of them, except on the second png file, it looked like the bottom half of the lettering was cut off. So, I think there must be some weird character in there, but I just don’t feel like tracking that down. There’s enough to read an answer that takes me to the next challenge.
2. Putting the messages from the pictures together: dis-pro-port-ional-~~ity~~