Python Imaging Library

The Python Imaging Library adds image processing capabilities to your Python interpreter.

This library provides extensive file format support, an efficient internal representation, and fairly powerful image processing capabilities.

The core image library is designed for fast access to data stored in a few basic pixel formats. It should provide a solid foundation for a general image processing tool.

Note: This documentation is based on the Python Imaging Library Handbook which you can also read online for more detailed information about concepts and supported file formats.

The version of PIL that is bundled with Pythonista includes some modifications to make it work well on iOS:

- You can use Pythonista's built-in images with the **`Image.open()`** function without needing to know the full file path. Example: `img = Image.open('Test_Lenna')`.
- The **`Image.Image.show()`** method can be used to view images directly in the console output area of Pythonista. This uses the **`console.show_image_file()`** function internally. On other platforms, this method usually shows an image in an external application.
- The **`ImageFont.truetype()`** function can also be used to load built-in system fonts in Pythonista. This also works on Windows, but not on most other platforms. When you're loading a built-in font, don't add a file extension, just use the font name, e.g. `font = ImageFont.truetype('Helvetica', 20)`.

Module Reference