

```

1 #!/usr/bin/env python
2 # -*- coding: latin-1 -*-
3 # *****
4 # * Software: FPDF for python *
5 # * Version: 1.7.1 *
6 # * Date: 2010-09-10 *
7 # * Last update: 2012-08-16 *
8 # * License: LGPL v3.0 *
9 # *
10 # * Original Author (PHP): Olivier PLATHEY 2004-12-31 *
11 # * Ported to Python 2.4 by Max (maxpat78@yahoo.it) on 2006-05 *
12 # * Maintainer: Mariano Reingart (reingart@gmail.com) et al since 2008 est. *
13 # * NOTE: 'I' and 'D' destinations are disabled, and simply print to STDOUT *
14 # *****
15
16 from __future__ import division, with_statement
17
18 from datetime import datetime
19 from functools import wraps
20 import math
21 import errno
22 import os, sys, zlib, struct, re, tempfile, struct
23
24 from .ttfonts import TTFontFile
25 from .fonts import fpdf_charwidths
26 from .php import substr, sprintf, print_r, UTF8ToUTF16BE, UTF8StringToArray
27 from .py3k import PY3K, pickle, urlopen, BytesIO, Image, basestring, unicode, exception, b
28
29 # Global variables
30 FPDF_VERSION = '1.7.2'
31 FPDF_FONT_DIR = os.path.join(os.path.dirname(__file__), 'font')
32 FPDF_CACHE_MODE = 0 # 0 - in same folder, 1 - none, 2 - hash
33 FPDF_CACHE_DIR = None
34 SYSTEM_TTFONTS = None
35
36 PAGE_FORMATS = {
37     "a3": (841.89, 1190.55),
38     "a4": (595.28, 841.89),
39     "a5": (420.94, 595.28),
40     "letter": (612, 792),
41     "legal": (612, 1008),
42 }
43
44 class FPDF(object):

```

```
47 def small_1(self):
48     print('this function is small')
49
50 def big_no_call(self):
51     print('This function is big')
52     print('This function is big')
53     print('This function is big')
54     print('This function is big')
55     print('This function is big')
56     print('This function is big')
57     print('This function is big')
58
59 def small_2(self):
60     print('this function is also small')
61     print('another line')
62
63 def big_with_one_call(self):
64     print('This function is big with one call')
65     print('This function is big with one call')
66     print('This function is big with one call')
67     print('This function is big with one call')
68     self.small_2()
69     print('This function is big with one call')
70     print('This function is big with one call')
71     self.big_no_call()
72
73 def small_3(self):
74     print('this function is also small')
75
76 def big_with_three_call(self):
77     print('This function is big with three call')
78     print('This function is big with three call')
79     print('This function is big with three call')
80     print('This function is big with three call')
81     self.small_1()
82     print('This function is big with three call')
83     print('This function is big with three call')
84     print('This function is big with three call')
85     print('This function is big with three call')
86     self.small_2()
87     self.small_1()
88     self.small_3()
```