



How to convert webpage into PDF by using Python



I was finding solution to print webpage into local file PDF, using Python. one of the good solution is to use Qt, found here, <https://bharatikunal.wordpress.com/2010/01/>.

It didn't work at the beginning as I had problem with the installation of PyQt4 because it gave error messages such as 'ImportError: No module named PyQt4.QtCore', and 'ImportError: No module named PyQt4.QtCore'.

It was because PyQt4's not installed properly. I used to have the libraries located at C:\Python27\Lib however it's not for PyQt4.

In fact, it simply needs to download from <http://www.riverbankcomputing.com/software/pyqt/download> (mind the correct Python version you are using), and install it to C:\Python27 (my case). That's it.

Now the scripts runs fine so I want to share it. for more options in using Qprinter, please refer to <http://qt-project.org/doc/qt-4.8/qprinter.html#Orientation-enum>.

python pdf webpage qprinter

asked Apr 29 '14 at 8:10



Mark K

1,208 1 8 22

5 Answers

You also can use [pdftkit](#):

```
import pdftkit
pdftkit.from_url('http://google.com', 'out.pdf')
```

answered May 20 '14 at 13:24



NorthCat

3,799 13 20 32

thanks, NorthCat :) – [Mark K](#) May 22 '14 at 4:52

This is awesome, way easier than messing around with reportlab or using a print drive to convert. Thanks so much. – [Dowlers](#) May 27 '15 at 18:56

@NorthCat can u give another example about converting html tables with pdftkit ? – [Egzona](#) Aug 12 '15 at 14:44

The render is very good and can convert form very well. – [Laurent LAPORTE](#) Nov 9 '15 at 9:56



thanks to below posts, and I am able to add on the webpage link address to be printed and present time on the PDF generated, no matter how many pages it has.

[Add text to Existing PDF using Python](#)

https://github.com/disflux/django-mtr/blob/master/pdfgen/doc_overlay.py

To share the script as below:

```
import time
from pyPdf import PdfFileWriter, PdfFileReader
import StringIO
from reportlab.pdfgen import canvas
from reportlab.lib.pagesizes import letter
from xhtml2pdf import pisa
import sys
from PyQt4.QtCore import *
from PyQt4.QtGui import *
from PyQt4.QtWebKit import *

url = 'http://www.yahoo.com'
tem_pdf = "c:\\tem_pdf.pdf"
final_file = "c:\\younameit.pdf"

app = QApplication(sys.argv)
web = QWebView()
#Read the URL given
web.load(QUrl(url))
printer = QPrinter()
#setting format
printer.setPageSize(QPrinter.A4)
printer.setOrientation(QPrinter.Landscape)
printer.setOutputFormat(QPrinter.PdfFormat)
#export file as c:\\tem_pdf.pdf
printer.setOutputFileName(tem_pdf)

def convertIt():
    web.print_(printer)
    QApplication.exit()

QObject.connect(web, SIGNAL("loadFinished(bool)"), convertIt)

app.exec_()
sys.exit

# Below is to add on the weblink as text and present date&time on PDF generated

outputPDF = PdfFileWriter()
packet = StringIO.StringIO()
# create a new PDF with Reportlab
can = canvas.Canvas(packet, pagesize=letter)
can.setFont("Helvetica", 9)
# Writing the new line
oknow = time.strftime("%a, %d %b %Y %H:%M")
can.drawString(5, 2, url)
can.drawString(605, 2, oknow)
can.save()

#move to the beginning of the StringIO buffer
packet.seek(0)
new_pdf = PdfFileReader(packet)
# read your existing PDF
existing_pdf = PdfFileReader(file(tem_pdf, "rb"))
pages = existing_pdf.getNumPages()
output = PdfFileWriter()
# add the "watermark" (which is the new pdf) on the existing page
for x in range(0,pages):
    page = existing_pdf.getPage(x)
    page.mergePage(new_pdf.getPage(0))
    output.addPage(page)
# finally, write "output" to a real file
outputStream = file(final_file, "wb")
output.write(outputStream)
outputStream.close()

print final_file, 'is ready.'
```

answered Apr 30 '14 at 7:31



Mark K
1,208 ● 1 ● 8 ● 22

Thanks for sharing your code! Any advice for making this work for local pdf files? Or is it as easy as prepending "file:/" to the url? I'm not very familiar with these libraries... thanks – [user2426679](#) Oct 31 '14 at 18:02

@user2426679, you mean convert online PDF into local PDF files? – [Mark K](#) Nov 25 '14 at 1:48

thanks for your reply... sorry for my tardiness. I ended up using wkhtmltopdf since it was able to handle what I was throwing at it. But I was asking how to load a pdf that was local to my hdd. Cheers – [user2426679](#) Dec 28 '14 at 23:15

@user2426679 sorry I still don't get you. maybe because I am a newbie to Python too. You meant read local PDF files in Python? – [Mark K](#) Jan 22 '15 at 8:05

here is the one working fine:

```
import sys
from PyQt4.QtCore import *
from PyQt4.QtGui import *
from PyQt4.QtWebKit import *

app = QApplication(sys.argv)
web = QWebView()
web.load(QUrl("http://www.yahoo.com"))
```

```

printer = QPrinter()
printer.setPageSize(QPrinter.A4)
printer.setOutputFormat(QPrinter.PdfFormat)
printer.setOutputFileName("fileOK.pdf")

def convertIt():
    web.print_(printer)
    print "Pdf generated"
    QApplication.exit()

QObject.connect(web, SIGNAL("loadFinished(bool)"), convertIt)
sys.exit(app.exec_())

```

answered Apr 29 '14 at 8:11



Mark K

1,208 ● 1 ● 8 ● 22

Interestingly, the web page links are generated as text rather than links in the generated PDF. – [amergin](#)
Nov 24 '14 at 17:16

WeasyPrint

```

pip install weasyprint

python
>>> pdf = weasyprint.HTML('http://www.google.com').write_pdf()
>>> len(pdf)
92059
>>> file('google.pdf', 'w').write(pdf)

```

answered Dec 23 '15 at 15:04



JohnMudd

5,782 ● 2 ● 11 ● 14

Here is a simple solution using QT. I found this as part of an answer to a different question on StackOverflow. I tested it on Windows.

```

from PyQt4.QtGui import QTextDocument, QPrinter, QApplication

import sys
app = QApplication(sys.argv)

doc = QTextDocument()
location = "c://apypthon//Jim//html//notes.html"
html = open(location).read()
doc.setHtml(html)

printer = QPrinter()
printer.setOutputFileName("foo.pdf")
printer.setOutputFormat(QPrinter.PdfFormat)
printer.setPageSize(QPrinter.A4);
printer.setPageMargins (15,15,15,15,QPrinter.Millimeter);

doc.print_(printer)
print "done!"

```

edited Mar 12 '15 at 13:31

answered Jan 20 '15 at 20:38



Jim Paul

61 ● 3