

Tvorba pdf dokumentů - Python



příklady použití

- Reporty
- Faktury
- Technické zprávy
- Emaily
- Generování velmi podobných dokumentů stále dokola



Jak vytvořit pdf

1. Vytvoření pdf pomocí knihovny **reportlab**
2. Vlastní šablony (**HTML, LaTeX**) knihovna **Jinja2** a konverze na pdf
3. Vlastní šablony **google docs**, API
4. Šablony vytvořené v **online službě** (třetí strana) a nahrání šablony pomocí API. např. <https://pdfgeneratorapi.com/>



1) Reportlab

```
from reportlab.lib.pagesizes import letter
from reportlab.pdfgen import canvas

canvas = canvas.Canvas("form.pdf", pagesize=letter)
canvas.setLineWidth(.3)
canvas.setFont('Helvetica', 12)

canvas.drawString(30, 750, 'OFFICIAL COMMUNIQUE')
canvas.drawString(30, 735, 'OF ACME INDUSTRIES')
canvas.drawString(500, 750, "12/12/2010")
canvas.line(480, 747, 580, 747)

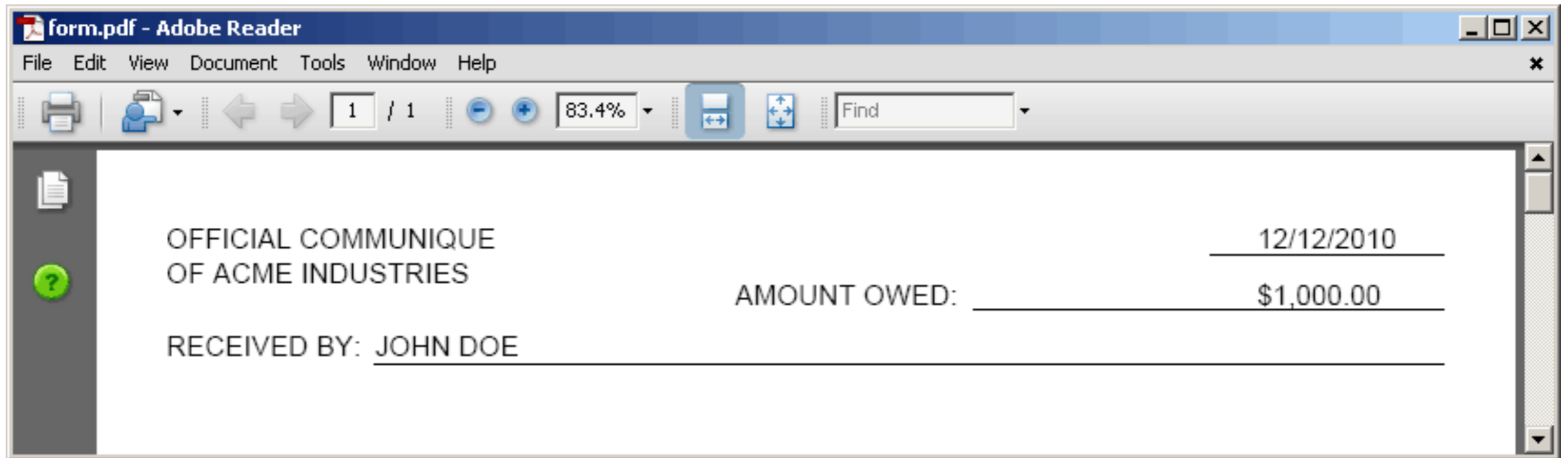
canvas.drawString(275, 725, 'AMOUNT OWED:')
canvas.drawString(500, 725, "$1,000.00")
canvas.line(378, 723, 580, 723)

canvas.drawString(30, 703, 'RECEIVED BY:')
canvas.line(120, 700, 580, 700)
canvas.drawString(120, 703, "JOHN DOE")

canvas.save()
```



1) Reportlab



2) Pomocí vlastních šablon

1. Vytvoření vlastní šablony (markup languages):

- HTML (CSS), TeX, Markdown, XML, Scribe

2. Úprava šablony o své proměnné (render) pomocí Jinja2, Django, Flask

3. Konverze upravené šablony do .pdf

- tex - pdflatex - command pdflatex v příkazové řádce funguje pouze když je správně nainstalován MikTex nebo MacTex, což může být problematické instalační balíček může mít velkou velikost
- html - WeasyPrint, xhtml2pdf, pdfkit



2) Pomocí vlastních šablon HTML konverze

1. WeasyPrint:

- dobře udržovaný projekt, obsáhlá dokumentace v které lze najít zajímavé tipy
- potřeba doinstalovat balíčky do počítače
 - MacOS - brew install python3 **cairo pango gdk-pixbuf libffi**
 - Fedora - sudo yum install redhat-rpm-config python-devel python-pip python-setuptools python-wheel python-cffi **libffi-devel cairo pango gdk-pixbuf2**

2. xhtml2pdf

- není až tak udržováno (not so update but stable version exist)
- pro použití jsem neinstalovat nic do PC
- nepovedla se mi konverze nejspíš vyžaduje jednoduchý html soubor bez tříd apod.
 - ValueError: invalid literal for int() with base 10: '100%'
 - AttributeError: module 'cgi' has no attribute 'escape' - python 3.8 problem - Issue nahlášeno je třeba překopírovat soubor z github
 - Nakonec si neporadil s graficky složitějšími css styly (např. zaoblené hrany v tabulce)

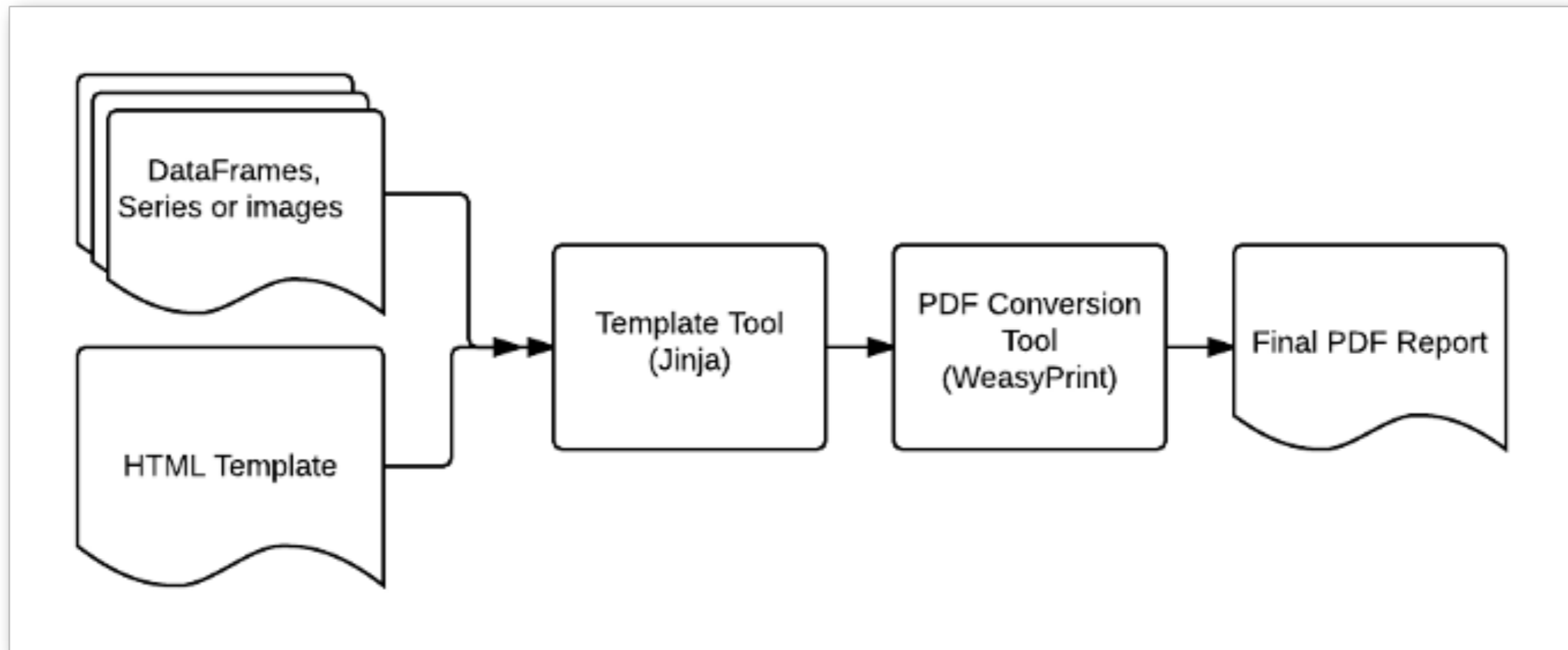
3. pdfkit

- nezkoušel jsem
- nutná instalace wkhtmltopdf do PC



2) Pomocí vlastních šablon

Schéma průběhu vytvoření



<https://pbpython.com/pdf-reports.html>



Ukázka vlastního kódu

ukázka kódu - načítání šablony, render a vytváření pdf

- https://github.com/nappex/invoice_generator



Další možnosti

- .doc files - <https://github.com/mikemaccana/python-docx>
- .net framework - <https://ironpython.net/>



Zdroje

LaTeX to PDF

- <http://eosrei.net/articles/2015/11/latex-templates-python-and-jinja2-generate-pdfs>
- <https://miller-blog.com/latex-with-jinja2/>

HTML to PDF

- <https://pbpython.com/pdf-reports.html>
- <https://philfreo.com/blog/render-a-pdf-from-html-using-xhtml2pdf-and-mimerender-in-flask/>
- <https://www.smashingmagazine.com/2015/01/designing-for-print-with-css/>
- <https://github.com/oreillymedia/HTMLBook/blob/master/stylesheets/pdf/pdf.css>

Google script to PDF

- <https://www.codementor.io/@garethdwyer/create-pdf-files-from-templates-with-python-and-google-scripts-p63ka1vb>

