



CLI tools in Python + interesting/useful packages

~ Workshop ~

Martin Pavlásek
RedHat, DevConf, 2016-02-05

Real life 😞



Real life 😞

```
{ "gifts": { "socks": { "S": 2, "M": 5,
"L": 1 }, "toys": [ "Yellow duck",
"Wooden bricks" ] }, "TODO": [ "bake
Vanocka", "pack gifts", "tidy own
desk" ] }
```

Better life 😊

```
$ pytube \  
    'https://www.youtube.com/watch?  
v=jrmZIgVoQw4 '
```

Better life ☺

```
$ python -m json.tool ugly.json
```

```
{
  "TODO": [
    "bake Vanocka",
    "pack gifts",
    "tidy own desk"
  ],
  "gifts": {
    "socks": {
      "L": 1,
      "M": 2,
      "S": 5
    },
    "toys": [
      "Yellow duck",
      "Wooden bricks"
    ]
  }
}
```

Preparation ...

What can you expect?

- demo of some existings handy (often CLI) tools
- as-short-as-necessary introduction to **pip**, **PyPI** and **virtualenv**
- it's not necessary know the language

What **do not** expect:

- course like "Programming in Python"

Try it on your own!

Everything for this workshop is available in:

`http://github.com/mpavlase/python-talk`

Prepare workplace:

```
cd python-talk  
./prepare-env.sh
```


PyPI/pip

PyPI = Python Package Index repository of software for Python

```
$ sudo pip install odpdown
```

target: /usr/lib/python2.7/site-packages

... or

```
$ pip install --user odpdown
```

target: ~/lib/python2.7/site-packages

... or

virtualenv

```
$ sudo dnf install python-virtualenv
```

```
$ mkdir -p ~/sandbox
```

```
$ cd ~/sandbox
```

```
$ virtualenv venv
```

```
$ source venv/bin/activate
```

```
$ pip install odpdwn
```

target:

`~/sandbox/venv/usr/lib/python2.7/site-packages`

Demo time!



Overview

- File manipulation
- Text/image transformation
- Python
- Data generators
- Presentation
- System utilities
- Internet / multimedia
- Terminal



File manipulation

zipfile

Basic operations with ZIP archives. [pypi]

```
$ python -m zipfile -c my.zip file [...]  
$ python -m zipfile -e my.zip dest-folder  
$ python -m zipfile -l my.zip  
$ python -m zipfile -t my.zip
```

in Python:

```
from zipfile import ZipFile  
with ZipFile('my.zip', 'w') as myzip:  
    myzip.write('content.txt')
```

filecmp

Compare two directories. [builtin]

```
python -m filecmp dir1 dir2
```

in Python:

```
from filecmp import dircmp  
diff = dircmp('a', 'b')  
diff.report()
```

Text/image transformation

json.tool

Reformat JSON (and validate as well). [builtin]

```
$ python -m json.tool < ugly.json $  
python -m json.tool ugly nice
```

in Python:

```
import json  
j = { '1': 25, 'key': ['arr'] }  
print json.dumps(j, indent=4)
```

encodings.rot_13

En/decoding stream oneliner. [builtin]

```
$ echo "abcd" | python -m  
encodings.rot_13  
$ echo "nopq" | python -m  
encodings.rot_13
```

in Python:

```
from encodings import rot_13  
rot_13.rot13(open('file.in'),  
open('file.out', 'w'))
```

markdown

Convert Markdown into HTML. [pypi]

```
$ python -m markdown example.md
```

in Python:

```
from markdown import markdown
print markdown("# I'm a H1!")
```

markdownify

Convert HTML into Markdown. [pypi]
in Python:

```
from markdownify import markdownify
print markdownify('Be <b>bold</b>')
```

oneliner

similar to `n1`, but much more versatile. (pypi)

```
$ python -m oneliner -ne \  
'"(%s): %s" % (NR, _.upper())' \  
example.py
```

pygments

Make your code colorful! [pypi]

```
$ pygmentize < example.sh
```

```
$ pygmentize -O full,style=monokai
```

```
-o demo.html example.py
```

imgdiff

Display visual difference of two pictures. [pypi]

```
$ imgdiff -H --opacity 30 \  
-o diff.png a.png b.png
```

Python

pdb

Interactive text-based debugger. [pypi]
in Python:

```
import pdb; pdb.set_trace()
```

notebook

Web-based editor(IDE) and more. [pypi]

```
$ python -m notebook
```

Data generators

randomdata.datagen

Generates CSV according to spec file. [pypi]

```
$ echo spec.conf < \  
"id, IntID  
age, Int, {'maximum': 100}"  
$ python -m randomdata.datagen \  
-l 10
```

fake-factory

Generates fake userdata (name, email, ...) [pypi]

```
$ faker -r 5 name
```

in Python:

```
from faker import Faker
fake = Faker()
for _ in range(5):
    print fake.name()
```



Presentation

pydown

Make HTML presentation from Markdown. [pypi]

```
$ cat > slides.md <<EOF
```

```
!SLIDE
```

```
# pydown
```

```
!SLIDE
```

```
## heading text
```

```
EOF
```

```
$ pydown slides.md html
```

```
# entry point: html/index.html
```

odpdown

Create presentation from Markdown. [pypi]

```
$ odpdown --content-master content \  
--break-master chapter \  
input.md template.odp output.odp
```


playitagainsam

Record and replay terminal session. [pypi]

```
$ pias record session.pias  
$ pias play --auto-type 50 \  
--auto-waypoint 500 session.pias
```

Note: there exists HTML/js player as well.

System utilities

copypaste

Platform independent copy + paste. [pypi]

```
$ date | copy $ paste
```

pyinotify

Linux filesystem events monitoring. [pypi]

```
$ pyinotify -v .
```

Internet / multimedia

SimpleHTTPServer

Share current folder as HTTP page. [builtin]

```
$ python -m SimpleHTTPServer
```

telnetlib

telnet client. [builtin]

```
$ python -m telnetlib url port
```

pytube

Download videos from YouTube. [pypi]

```
$ pytube \
  --extension mp4 \
  --resolution 720p \
  'https://www.youtube.com/watch?
v=jrmZIgVoQw4'
```


status_server

HTTP status code on demand. [pypi]

```
$ status_server
```

```
$ curl -i
```

```
http://localhost:8888/404/200
```

signal

Simple image gallery generator. [pypi]

```
$ signal init  
# will process ./pictures folder  
$ signal build
```

Run webserver with the gallery:

```
$ signal serve
```



Terminal

pychalk

Print color text into terminal. [pypi]
in Python:

```
import chalk  
chalk.green("Hello world!")
```

rad

Console highlighter. [pypi]

```
$ rad -n # new rule
```

```
$ cat some.log | rad log
```

ConwayCPU

Game of Life (cellular automata), changes according of CPU usage. [pypi]

```
$ ConwayCPU.py
```

Disclaimer

"During work on this topic wasn't hurt even killed any zoologist or snake."

Python ...



Python ...



Thank You!

Resources

- <https://twitter.com/pypi>
- <https://pymotw.com/2/>
- https://groups.google.com/d/topic/comp.lang.python/_QLuMFfSLKk/discussion
- <http://pythonwise.blogspot.cz/2015/01/python-m.html>
- ... and pypi pages to each showed package