

Python, a practitioner's perspective

Alex Morega
@mgax
<http://grep.ro>

The plan

opinionated hand-waving
lots of code
discuss

Filesystem

pathlib

```
from pathlib import Path
```

```
p = Path('.')  
hello = p / 'hello.txt'
```

```
with hello.open('wb') as f:  
    f.write("hi!")
```

```
hello.unlink()
```

Web framework

Django

Flask

uWSGI

```
import flask
app = flask.Flask(__name__)

@app.route('/')
def hello():
    if flask.request.method == 'POST':
        return ('searching for %s'
                % flask.request.form['q'])
    return ('<form method=post>'
            '<input name=q></form>')

app.run()
```

Database

SQLAlchemy
alembic

```
import flask
from flask.ext.sqlalchemy import SQLAlchemy

app = flask.Flask(__name__)
db = SQLAlchemy(app)

class User(db.Model):
    id = db.Column(db.Integer, primary_key=True)
    name = db.Column(db.String)
    email = db.Column(db.String)

with app.app_context():
    db.create_all()
    u = User(name="Joe", email='joe@example.com')
    db.session.add(u)
    db.session.commit()

    u.name = "Joe Smith"
    db.session.commit()

print [u.name for u in User.query.all()]
```



```

def get_voting_similarity_list(self):
    other_vote = aliased(Vote)
    my_vote = aliased(Vote)
    similarity_cte = (
        db.session.query(
            other_vote.mandate_id,
            func.count(other_vote.id).label('count'),
        )
        .join(
            my_vote,
            (my_vote.choice == other_vote.choice) &
            (my_vote.voting_session_id == other_vote.voting_session_id),
        )
        .filter(my_vote.mandate == self.mandate)
        .group_by(other_vote.mandate_id)
        .cte()
    )
    similarity_query = (
        db.session.query(
            Person,
            similarity_cte.c.count,
            MpGroup.short_name,
        )
        .join(Person.mandates)
        .filter_by(year=2012)
        .join(similarity_cte, similarity_cte.c.mandate_id == Mandate.id)
        .join(Mandate.group_memberships)
        .filter(MpGroupMembership.interval.contains(date.today()))
        .join(MpGroupMembership.mp_group)
    )
    return similarity_query.all()

```

```
revision = '2b7972a5d85'  
down_revision = '36e9cd2a91c'
```

```
from alembic import op  
import sqlalchemy as sa
```

```
def upgrade():  
    op.add_column('user',  
                  sa.Column('created', sa.DateTime))  
    op.execute("update user set created = now()")  
    op.alter_column('user', 'created', nullable=False)
```

```
def downgrade():  
    op.drop_column('user', 'created')
```

Concurrency

greenlets

threads?

~~twisted~~

~~yield~~

Job queue

rq

Celery

```
# my_module.py  
import requests  
def count_words_at_url(url):  
    resp = requests.get(url)  
    return len(resp.text.split())
```

```
# start worker:  
#$ rqworker
```

```
# other.py  
import rq, redis  
from my_module import count_words_at_url  
q = rq.Queue(connection=redis.Redis())  
result = q.enqueue(count_words_at_url,  
                    'http://ropython.org')
```

Error monitoring

Sentry

```
import raven
```

```
SENTRY_DSN = 'http://...@example.com/1'  
client = raven.Client(SENTRY_DSN)
```

```
try:
```

```
    1 / 0
```

```
except:
```

```
    client.captureException()
```



Stream

Settings

Search query or event ID



sqlalchemy.sql.expression in _boolean_compare
ArgumentError: Only '=', '!=', 'is_()', 'isnot()' operators can be used with None/True/False



users 1 25 days ago root



The system first saw this event — 2014-09-19

Add Note

Newer Event

Event at Sept. 22, 2014, 8:12 a.m. [36.8 KB]
ID: 850fe421d1964133b924ae69816324dd

Older Event

Tags

Exception

Request

Additional Data

Tags

level = error logger = root server_name = firenze browser = Microsoft Internet Explorer 5.0

os = Windows url = http://parlament.openpolitics.ro/export/membri_grupuri.csv

Exception (most recent call last)

Raw

ArgumentError Root Cause

Only '=', '!=', 'is_()', 'isnot()' operators can be used with None/True/False

Aggregate

Tags

Similar Events

AGGREGATE DETAILS

Status: Unresolved

First Seen: a month ago

Last Seen: 25 days ago

This event is not publicly visible. [Change](#)

ACTIONS

[Mute Event](#)

[Remove Event Data](#)

BROWSER

3 [Microsoft Internet Explorer 5.0](#)

LEVEL

3 [error](#)

LOGGER

3 [root](#)

OS

3 [Windows](#)

SERVER NAME

3 [firenze](#)

URL

3 http://parlament.openpolitics.ro/export/membri_grupuri.csv

mptracker/app.py ? in patch_request

```
1460.         return self.view_functions[rule.endpoint](**req.view_args)
```

mptracker/website/pages.py ? in wrapper

```
25.         return func(*args, **kwargs)
```

mptracker/website/pages.py ? in export_group_membership

```
639.         'nume': row['name'],
640.         'inceput': row['start'].isoformat(),
641.         'sfarsit': '' if row['end'] is None else row['end'].isoformat(),
642.         'partid': row['group'],
643.     }
644.     for row in dal.get_group_membership(interval=interval, end=end)
645. ]
646.
647.     return csv_response(
648.         csv_lines(['nume', 'inceput', 'sfarsit', 'partid'], membership_list),
649.     )
```

'end' False

'initial' datetime.date(2012, 12, 19)

'interval'

Index

Value

0

datetime.date(2014, 9, 22)

1

None

[Expand Context \(3 hidden\)](#)

mptracker/website/pages.py ? in <listcomp>

```
638.     {
```

mptracker/website/dal.py ? in get_group_membership

```
1400.         .filter(func.upper(Mandate.interval) >= q_upper)
```

sqlalchemy/sql/operators.py ? in __ge__

Request

[Replay this request](#)

URL: http://parlament.openpolitics.ro/export/membri_grupuri.csv

Method: GET

Query: migrare=curent

Headers:

Accept	text/html
Accept-Encoding	identity
Cache-Control	no-cache, no-cache

[Expand Context \(5 hidden\)](#)

Environment:

REMOTE_ADDR	127.0.0.1
SERVER_NAME	localhost
SERVER_PORT	42710

Additional Data

sys.argv	Index	Value
	0	'/var/local/mptracker/venv/bin/waitress-serve'
	1	'--port'
	2	'42710'

[Expand Context \(3 hidden\)](#)

Automated testing

~~nose, unittest~~

pytest

mock

tox

travis

```
def func(x):  
    return x + 1
```

```
def test_answer():  
    assert func(3) == 5
```

```
# run tests:  
#$ py.test
```

```
>>> from mock import Mock
```

```
>>> m = Mock(foo='bar')
```

```
>>> m.foo
```

```
bar
```

```
>>> m.a.b.c = 13
```

```
>>> print m.a.b.c
```

```
13
```

```
>>> m.count.return_value = 'hello'
```

```
>>> m.count()
```

```
'hello'
```

```
# mycode.py
import requests
def count_words(url):
    try: return len(requests.get(url).text.split())
    except: return None

# test_mycode.py
from mock import patch
from mycode import count_words
def test_is_daytime():
    with patch('mycode.requests') as requests:
        requests.get.return_value.text = "hello world"
        assert count_words('http://ex.com/foo') == 2

        requests.get.side_effect = ValueError
        assert count_words('http://ex.com/bar') is None

    assert requests.get.mock_calls == [
        call('http://ex.com/foo'),
        call('http://ex.com/bar'),
    ]
```

```
;; tox.ini  
[tox]  
envlist = py26, py27, py33, pypy  
[testenv]  
deps =  
    pytest  
commands = py.test
```

```
## travis.yml  
language: python  
python: ["2.7", "2.6", "3.3", "pypy"]  
install:  
    - pip install pytest  
    - pip install --editable .  
script: py.test
```

HTML forms

WTForms


```
from wtforms import Form, StringField, \
    PasswordField, validators

class RegistrationForm(Form):
    name = StringField('Name', [validators.Length(min=4)])
    email = StringField('Email', [validators.Email()])
    password = PasswordField('Password', [
        validators.InputRequired(),
        validators.EqualTo('confirm',
            message='Passwords must match'),
    ])
    confirm = PasswordField('Repeat Password')

def register(request):
    form = RegistrationForm(request.POST)
    if request.method == 'POST' and form.validate():
        user = User()
        form.populate_obj(user)
        user.save()
        redirect('register')
    return render_response('register.html', form=form)
```

```
<form method="post">
  {% for field in form %}
    <div>
      <label for="{{ field.id }}">
        {{ field.label }}
      </label>

      {% for err in field.errors %}
        <p>{{ err }}</p>
      {% endfor %}

      {{ field }}
    </div>
  {% endfor %}
</form>
```

HTTP client

~~urllib, urllib2~~
requests

```
>>> r = requests.get('https://api.github.com/
user', auth=('user', 'pass'))

>>> r.status_code
200

>>> r.headers['content-type']
'application/json; charset=utf8'

>>> r.encoding
'utf-8'

>>> r.text
u'{"type": "User" ...'

>>> r.json()
{u'private_gists': 419, u'total_private_repos':
77, ...}
```

Parsing HTML, XML

lxml

cssselect

pyquery

```
from lxml import html
```

```
doc = html.fromstring("""  
<html>  
    <p>hello <a href="#">world</a>!</p>  
</html>  
""")
```

```
print [p.text_content()  
        for p in doc.cssselect('p')]  
## ['hello world!']
```

```
from lxml import etree
from lxml.builder import E

page = \
    E.html(
        E.head(
            E.title("This is a sample document"),
        ),
        E.body(
            E.h1("Hello!"),
            E.p("Paragraph with ", E.b("bold"), " text!"),
            E.p("Another paragraph, with a", "\n",
                E.a("link", href="http://python.org"), "."),
            E.p("Reserved characters: <spam&egg>."),
            etree.XML("<p>Finally, embedded XHTML</p>"),
        )
    )

print etree.tostring(page, pretty_print=True)
```

Science

numpy, scipy
matplotlib, bokeh
ipython notebook

Simple spectral analysis

An illustration of the [Discrete Fourier Transform](#)

$$X_k = \sum_{n=0}^{N-1} x_n e^{-\frac{2\pi i}{N} kn} \quad k = 0, \dots, N-1$$

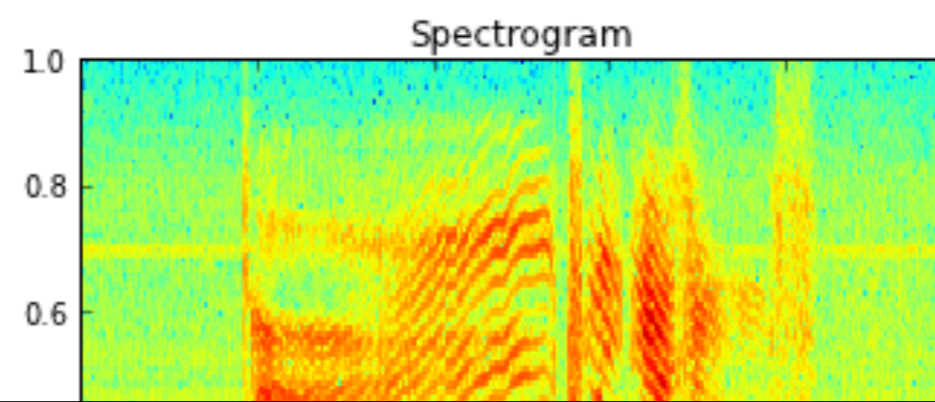
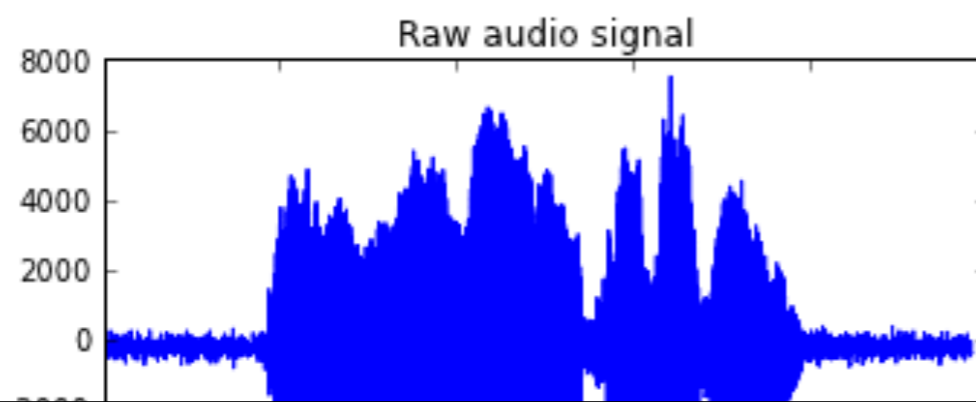
using windowing, to reveal the frequency content of a sound signal.

We begin by loading a datafile using SciPy's audio file support:

```
In [1]: from scipy.io import wavfile
rate, x = wavfile.read('test_mono.wav')
```

And we can easily view its spectral structure using matplotlib's builtin `specgram` routine:

```
In [2]: fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(12, 4))
ax1.plot(x); ax1.set_title('Raw audio signal')
ax2.specgram(x); ax2.set_title('Spectrogram');
```



Documentation

sphinx
rtd.org

Games

PyGame

```
import random
import pygame
```

```
BLACK = (0, 0, 0)
WHITE = (255, 255, 255)
```

```
(WIDTH, HEIGHT) = (400, 300)
FPS = 60
```

```
def main():
    # ...
```

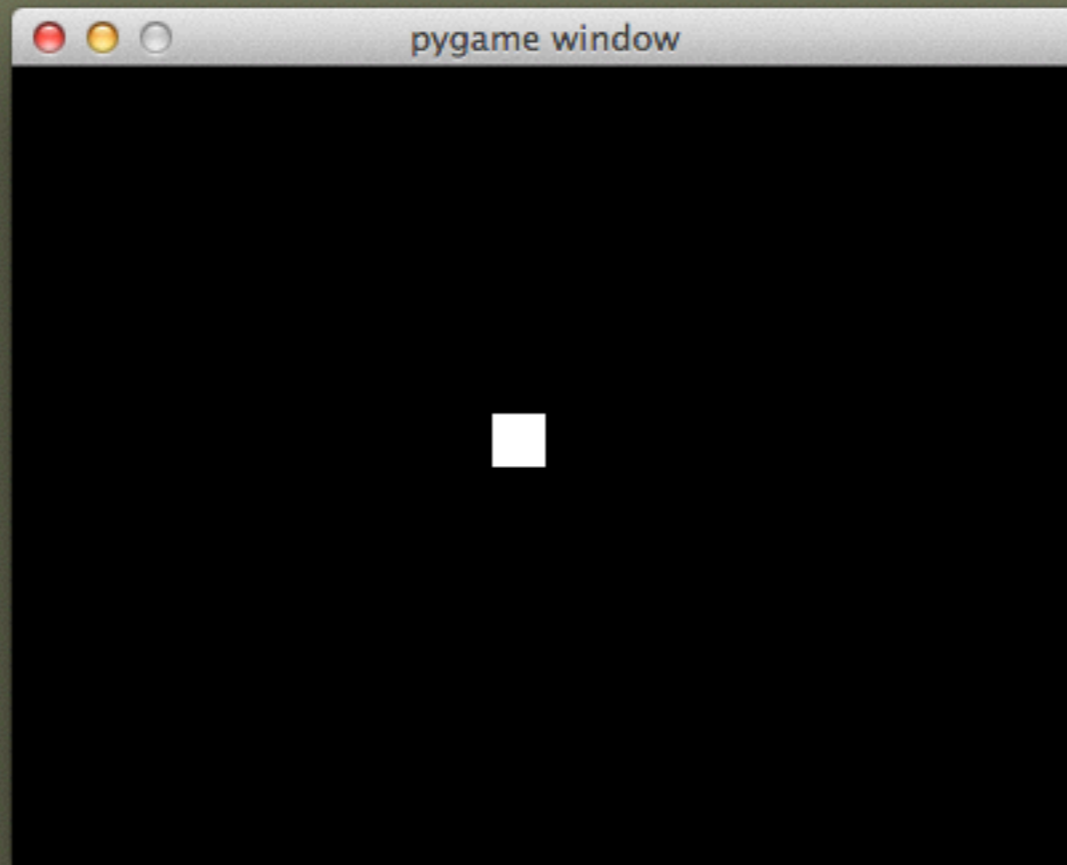
```
pygame.init()
main()
pygame.quit()
```

```
def main():
    screen = pygame.display.set_mode((WIDTH, HEIGHT))
    pygame.time.set_timer(pygame.USEREVENT, 1000 / FPS)

    v = [5, 5]
    (vx, vy) = (5, 5)
    thing = pygame.Rect(0, 0, 20, 20)
    thing.center = (WIDTH / 2, HEIGHT / 2)

    while True:
        event = pygame.event.wait()
        if event.type == pygame.USEREVENT:
            (x, y) = thing.center
            thing.center = (x + v[0], y + v[1])
            if not screen.get_rect().contains(thing):
                v[0] = -v[0]; v[1] = -v[1]
            screen.fill(BLACK)
            screen.fill(WHITE, thing)
            pygame.display.flip()

        if event.type == pygame.KEYUP:
            if event.key == pygame.K_ESCAPE:
                return
            v[random.choice([0, 1])] *= -1
```



Python 3

shared codebase
library support

```
# _compat.py
```

```
import sys
```

```
PY2 = sys.version_info[0] == 2
```

```
if PY2:
```

```
    text_type = unicode
```

```
    string_types = (str, unicode)
```

```
    integer_types = (int, long)
```

```
    from itertools import imap
```

```
    range_type = xrange
```

```
    iteritems = lambda d: d.iteritems()
```

```
    exec('def reraise(tp, value, tb=None):\n raise tp, value, tb')
```

```
else:
```

```
    text_type = str
```

```
    string_types = (str, )
```

```
    integer_types = (int, )
```

```
    imap = map
```

```
    range_type = range
```

```
    iteritems = lambda d: iter(d.items())
```

```
    def reraise(tp, value, tb=None):
```

```
        if value.__traceback__ is not tb:
```

```
            raise value.with_traceback(tb)
```

```
        raise value
```


PyPy

!?

thanks!
~~ @mgax