

Python, ruumiandmed ja ruumiandmeteenused

ESTGISi suveülikool 23.08.2014, Ale-Sepa

Tõnis Kärdi (KEMIT)
@tkardi
tonis.kardi@kemit.ee

Python 101

- QGIS ja pyqgis

http://docs.qgis.org/testing/en/docs/pyqgis_developer_cookbook/

- ArcGIS ja arcpy (arcgisscripting)

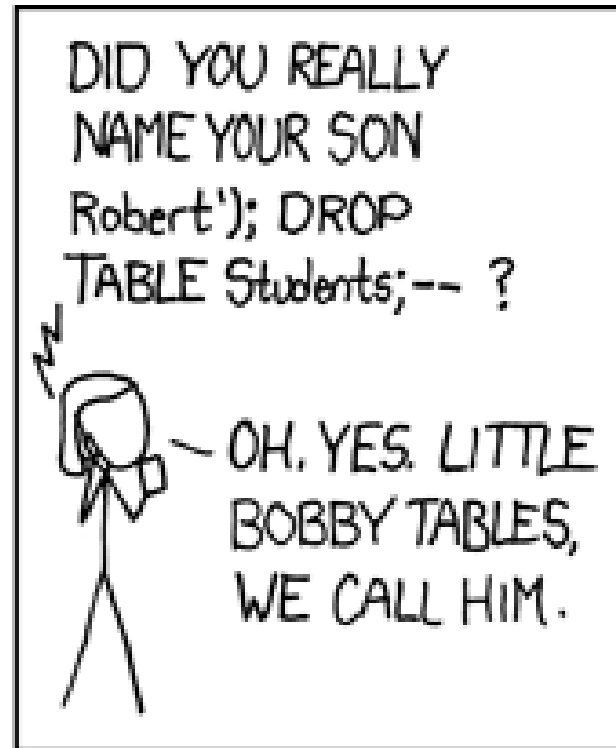
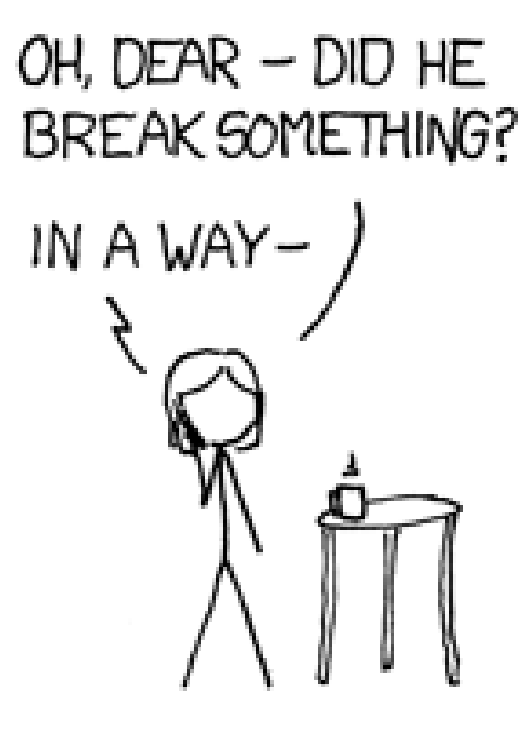
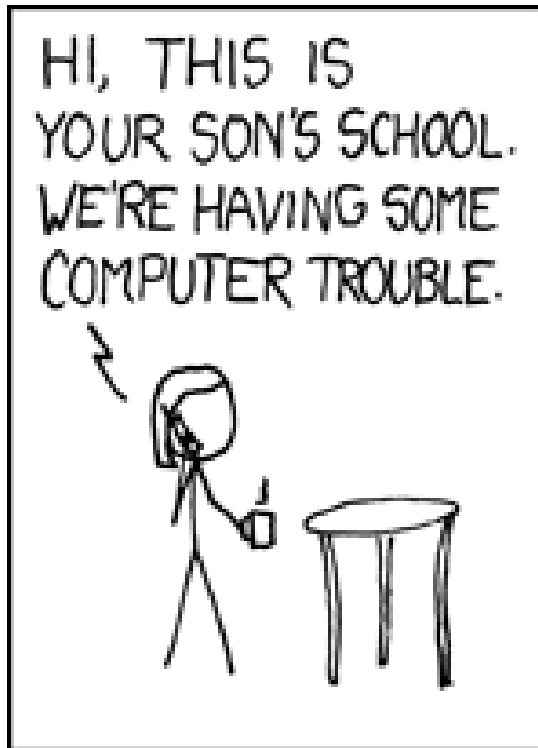
http://resources.arcgis.com/en/help/main/10.2/index.html#/What_is_ArcPy/000v000000v7000000/

- arcpy. ArcSDESQLExecute:

```
sql = "select OBJECTID,{0} from {1} where {0} = {2}".format( col, tbl, bad_val)
```

Ei! Mitte mingil juhul!

Python 101



Python 101

- <http://www.stavros.io/tutorials/python/> (10min tutorial)
- <http://learnpythonthehardway.org/>
- <https://docs.python.org/>
- Loe teistekirjutatud koodi (github, bitbucket, pypi, ...)
- „G**gle teab ja oskab kõike“ – lihtsalt oska õigeid asju küsida
- Parim algus on mingi praktiline vajadus...

Mõned kasulikud moodulid

requests – HTTP for humans

<http://docs.python-requests.org/en/latest/>

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

psycopg2 - PostgreSQL adapter for Python

<http://initd.org/psycopg/docs/>

```
>>> import psycopg2
>>> conn = psycopg2.connect("dbname=test user=postgres")
>>> cur = conn.cursor()
>>> cur.execute("CREATE TABLE test (id serial PRIMARY KEY, num integer, data varchar)")

# Pass data to fill a query placeholders and let Psycopg perform
# the correct conversion (no more SQL injections!)
>>> cur.execute("INSERT INTO test (num, data) VALUES (%s, %s)",
...             (100, "abc'def"))

# Query the database and obtain data as Python objects
>>> cur.execute("SELECT * FROM test")
>>> cur.fetchone()
(1, 100, "abc'def")

# Make the changes to the database persistent
>>> conn.commit()

# Close communication with the database
>>> cur.close()
>>> conn.close()
```

fiona - OGR's neat, nimble, no-nonsense API for Python

<http://toblerity.org/fiona/>

```
>>> import fiona
>>> with fiona.open('docs/data/test_uk.shp') as source:
...     for f in source.filter(bbox=(-5.0, 55.0, 0.0, 60.0)):
...         # do something
...
>>>
```


shapely - set-theoretic analysis and manipulation of planar features

<http://toblerity.org/shapely/manual.html>

```
>>> from shapely.geometry import Point
>>> patch = Point(0.0, 0.0).buffer(10.0)
>>> patch
<shapely.geometry.polygon.Polygon object at 0x...>
>>> patch.area
313.6548490545938
>>> patch.__geo_interface__
{"type": "Polygon", "coordinates": [[[10.0, 0.0], [9.95184726672197, -0.980171403295605], [9.807852804032306, -1.9509032201612808], [9.56940335732209, -2.902846772544621], [9.23879532511287, -3.826834 ...
```

Jne, jne, jne...

- numpy, scipy, pandas, geopandas – „scientific computing“
- lxml – xml/html lugemine ja kirjutamine
- pyproj – koordinaatide teisendus proj.4 peal.
- paramiko – SSH2 protokoll
- Django, Flask, Tornado, Pyramid, ... – veebiarenduse raamistikud
- wxPython, tkInter, PyQt, ... – Desktop GUI
- Kivy – „cross-platform apps“ (s.h iOS + Android)
- osgeo – GDAL bindings
- matplotlib – 2D plotting
- networkx – komplekssete võrgustike loomine/manipuleerimine
- epydoc – html/text/latex/pdf dokumentatsiooni genereerimiseks Pythoni koodist
- ...

...and now for something completely different

Täna!

Materjalid: <https://bitbucket.org/tkardi/estgis>

Tõnis Kärdi (KEMIT)
@tkardi
tonis.kardi@kemit.ee