

# Térinformatikai programozás Pythonban



Siki Zoltán  
BME Geod  
[siki.zoltan@epito.bme.hu](mailto:siki.zoltan@epito.bme.hu)

# Térinformatikai programozás Pythonban



Siki Zoltán  
BME Geod  
[siki.zoltan@epito.bme.hu](mailto:siki.zoltan@epito.bme.hu)

# Python alapelvek

A szép jobb mint a csúnya

A közvetlen jobb mint a közvetett.

Az egyszerű jobb mint a komplex

A komplex jobb mint a komplikált .

A sima jobb mint az egymásba ágyazott.

A ritka jobb mint a sűrű.

Az olvashatóság számít.

A speciális esetek nem elég speciálisak,  
hogy a szabályokat megsértsék.

Bár a praktikusság veri a tisztaságot.

A hibákat nem szabad csendben elengedni.

Kivéve, ha szándékos.

...

```
>>> import this
```

# Python alapelvek

A szép jobb mint a csúnya

A közvetlen jobb mint a közvetett.

Az egyszerű jobb mint a komplex

A programot embereknek kell írni, hogy el tudják olvasni, mellékesen a gépeknek, hogy végrehajthassák.

Abelson & Sussman

Az olvashatóság számít.

A speciális esetek nem elég speciálisak, hogy a szabályokat megsértsék.

Bár a praktikusság veri a tisztaságot.

A hibákat nem szabad csendben elengedni.

Kivéve, ha szándékos.

...

```
>>> import this
```

# Miért?

- Kiforrott (1989-től), Python 2 – 2000, Python 3 - 2008
- Objektum orientált és funkcionális programozás is
- Platform független (Linux/Win/OSX)
- Dinamikus (szkript)
- Gyors (előfordító .pyc – byte code)
- Könnyen tanulható
- Interaktív és programozott használat is
- Nyílt forráskódú
- Számptalan bővítő modul, könyvtár elérhető  
GDAL/OGR, Shapely, Proj.4, Cython
- Sok program használja  
QGIS, GRASS, R, MapServer/MapScript, PyWPS, ...



Guido van Rossum

# Mire?

- Egyszerű problémák gyors interaktív megoldása
- Kötegelte feldolgozás, pl. 5000 shape fájl átalakítása
- Meglévő programok bővítése (pl. QGIS modul)
- Önálló GUI-val rendelkező program készítése

Python szkriptek könnyebben hordozhatók az operációs rendszerek között mint a burok szkriptek vagy .bat fájlok (pl. GRASS 7)

# Mivel?

- Térinformatikai adatok olvasása  
GDAL/OGR (142 raszter és 84 vektor formátum)
- Vetületi transzformáció  
pyproj (Proj.4)
- Vektoros számítások, elemzések  
Shapely (GEOS)
- Adatbázis kapcsolat, adat elemzés (adat struktúrák)  
sqlalchemy, pandas
- Tudományos számítások (mátrixok, egyenletek)  
numpy
- 2D rajzolás (grafikonok)  
Matplotlib
- GUI – PyQt vagy wxPython vagy TkIntern

# Miben?

- QGIS
  - GRASS 7
  - MapServer – MapScript Python
  - PostgreSQL/PostGIS – PL/Python
  - Marble
  - pyCSW
  - pyWPS
- 
- GIMP
  - Libre Office/Open Office
  - ...





# Mikor és hol?



# QGIS Python programozás

## PyQGIS

- Python konzol – interaktív használat
- ScriptRunner modul – egyszerű szkriptek
- Projekt esemény szkriptek – megnyitás, mentés, lezárás
- Feldolgozás keretrendszer – szkriptek összekapcsolása
- QGIS inicializáló szkript – QGIS indításakor automatikusan
- Saját függvények kifejezésekhez  
(mező kalkulátor, kifejezéssel szelektálás)
- Műveletek
- Önálló alkalmazás készítése

# Néhány gyöngyszem

Két változó tartalmának felcserélése

```
temp = a;  
a = b;  
c = temp;
```

```
a, b = b, a
```

100-ig a páratlan számok négyzetének összege

```
s = 0  
for (i = 1; i < 100; i++)  
    if (i % 2)  
        s += i * i;
```

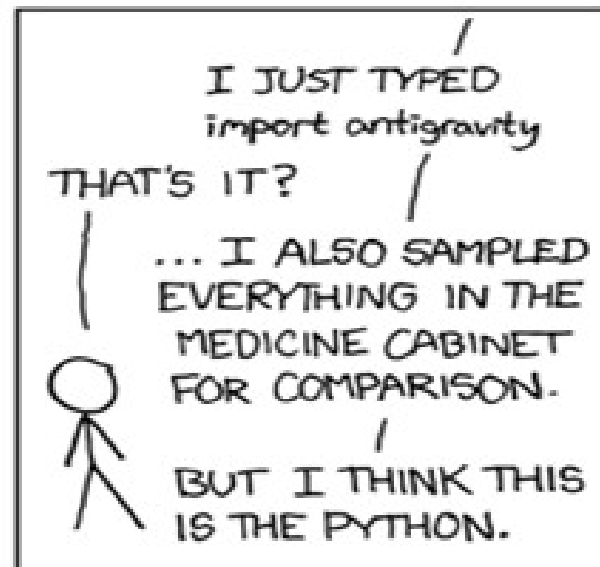
```
sum ([ i ** 2 for i in range(100) if i % 2])
```

vagy

```
sum ([ i ** 2 for i in range(1,100,2)])
```

Hány elem van egy shape fájlban?

```
from osgeo import ogr  
shp = ogr.Open("megye.shp")  
layer = shp.GetLayer(0)  
print layer.GetFeatureCount()
```



# Menjünk Pythonban programozni...