



Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

More Types

Flow Control

File I/O

Plotting

Assessed
Programming
Task

Technological Infrastructures for GIS

More Python

Magnus Hagdorn

School of GeoSciences
University of Edinburgh

October 8, 2019



THE UNIVERSITY
of EDINBURGH



Last Time...

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

More Types

Flow Control

File I/O

Plotting

Assessed
Programming
Task

- introduced programming environment consisting of an editor, terminals and ipython shell
- operators
- variables
- scalars int (), float (), strings
- functions



This Time...

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

More Types

Flow Control

File I/O

Plotting

Assessed
Programming
Task

we will talk about

- objects
- container objects
- flow control
- file I/O
- plotting



So what is Object Oriented Programming?

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects
Introduction
Objects

More Types

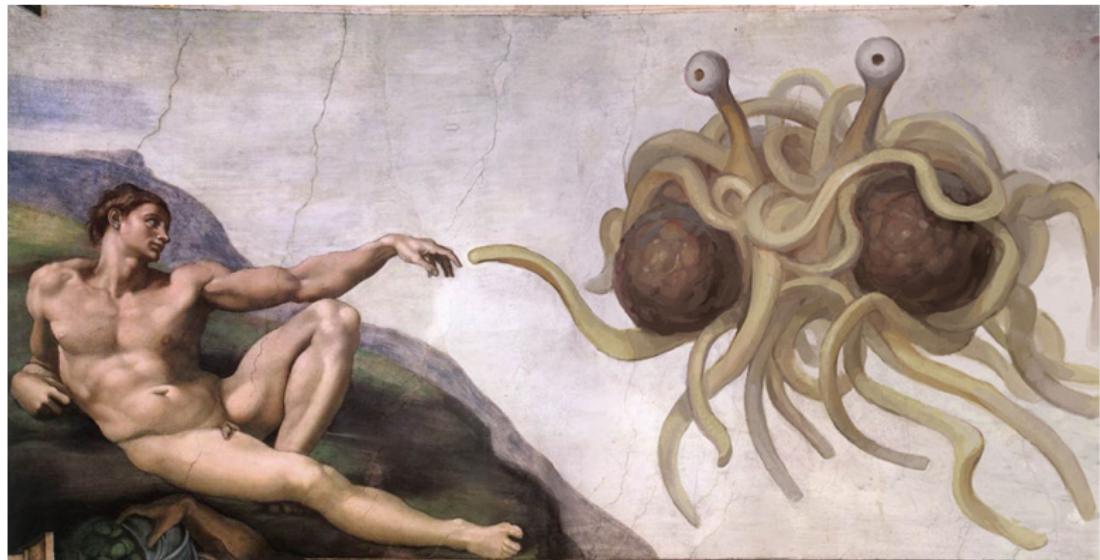
Flow Control

File I/O

Plotting

Assessed
Programming
Task

In the beginning there was spaghetti





So what is Object Oriented Programming?

Procedural Programming

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects
Introduction
Objects

More Types

Flow Control

File I/O

Plotting

Assessed
Programming
Task

then after some consideration came **Procedural Programming:**



CC soycamo



So what is Object Oriented Programming?

Procedural Programming

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

Introduction
Objects

More Types

Flow Control

File I/O

Plotting

Assessed
Programming
Task

- computational steps are grouped together in units called *subroutines*, *functions* or *methods*
- a procedural program is just a list of tasks to be done
- data is passed around via *arguments*, some functions may *return* values
- functions may have *side effects*
- quite often *global data* is used



So what is Object Oriented Programming?

Modular Programming

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

Introduction
Objects

More Types

Flow Control

File I/O

Plotting

Assessed
Programming
Task

then after some more consideration came **Modular**
Programming:



CC NASA's Marshall Space Flight Center



So what is Object Oriented Programming?

Modular Programming

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

Introduction
Objects

More Types

Flow Control

File I/O

Plotting

Assessed
Programming
Task

- functionality is separated into independent modules
- define *public interfaces*
- can have *private data and functions*



So what is Object Oriented Programming?

Object Oriented Programming

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects
Introduction
Objects

More Types

Flow Control

File I/O

Plotting

Assessed
Programming
Task

with a little bit of further thought - [Object Oriented Programming](#):



CC val'sphotos



So what is Object Oriented Programming?

Object Oriented Programming

- *objects* combine both data fields and associated procedures
- OO programs are collections of objects that interact with each other
- typically data access is abstracted using accessors
- *open recursion* - a special variable that can be used to invoke another method of the same object
- *inheritance* - a child class inherits *attributes* and *methods* from a base class
- *subtype polymorphism* - methods of the supertype can operate on elements of the subtype
- *dynamic dispatch* - methods are looked up at run time



So what is Object Oriented Programming?

Functional Programming

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects
Introduction
Objects

More Types

Flow Control

File I/O

Plotting

Assessed
Programming
Task

there is also **Functional Programming...**



CC magnus.hagdorn



So what is Object Oriented Programming?

Functional Programming

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

Introduction
Objects

More Types

Flow Control

File I/O

Plotting

Assessed
Programming
Task

- avoids changing state and mutable data
- function results only depend on the function arguments
- no side effects



Objects and Object Oriented Programming

Objects

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

Introduction
Objects

More Types

Flow Control

File I/O

Plotting

Assessed
Programming
Task

Real-world objects share two characteristics:

state the particular properties/condition that the object has/is in at a specific time

behaviour the potential ways in which the object can function



Objects and Object Oriented Programming

Objects

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

Introduction
Objects

More Types

Flow Control

File I/O

Plotting

Assessed
Programming
Task



CC magnus.hagdorn

Thinking about objects

- 1 What is the state of my bike?
- 2 What are the possible ways it can function? — its behaviours



Objects and Object Oriented Programming

Objects

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects
Introduction
Objects

More Types

Flow Control

File I/O

Plotting

Assessed
Programming
Task



CC magnus.hagdorn

My bike

- has a colour
- has drop handlebars
- has disk breaks
- has wheels of a particular size
- is locked
- is in the 5th gear
- has a location



Objects and Object Oriented Programming

Objects

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

Introduction
Objects

More Types

Flow Control

File I/O

Plotting

Assessed
Programming
Task



CC magnus.hagdorn

I can

- unlock my bike
- apply breaks
- change gears
- peddle harder
- attach a pannier



Objects and Object Oriented Programming

Objects

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

Introduction
Objects

More Types

Flow Control

File I/O

Plotting

Assessed
Programming
Task

Software objects

- are conceptually similar to real objects
- also consist of *state* — the attributes
- and related *behaviour* — the methods



Objects and Object Oriented Programming

Objects

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

Introduction
Objects

More Types

Flow Control

File I/O

Plotting

Assessed
Programming
Task

in python

- methods and attributes are accessed using a .
- methods end with (), attributes don't

```
aCmplx = 1 + 2j
```

```
print(aCmplx.real, aCmplx.imag)
```

```
print(aCmplx.conjugate())
```

```
# more on format strings
```

```
print("Complex value {cmplx} has"
```

```
    " real {cmplx.real}"
```

```
    " and imaginary {cmplx.imag}"
```

```
    " parts".format(cmplx=aCmplx))
```



Containers

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

More Types

Containers
Arrays

Flow Control

File I/O

Plotting

Assessed
Programming
Task

Containers are objects that collect a number of objects. There are

- **tuples** fixed size, immutable
- **lists** variable size, mutable
- **dictionaries** a mapping from an immutable object to a mutable object

Other containers are available from for example the [collections](#) module. You can also write your own.



Containers

lists

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

More Types

Containers
Arrays

Flow Control

File I/O

Plotting

Assessed
Programming
Task

- lists are created using
`aList = []`
- you can append to a list
`aList.append('Hello')`
- have indices 0, 1, 2, 3, ...
- get the number of items
`len(aList)`
- items are accessed
`aList[0]`
- items are deleted
`del aList[0]`
- you can also remove the first occurrence of a value using
`aList.remove('Hello')`



Containers

tuples

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

More Types
Containers
Arrays

Flow Control

File I/O

Plotting

Assessed
Programming
Task

- create a tuple
`aTuple = (1, 'hello', 'world')`
- indexing is similar to lists



Containers

Slicing

slicing allows you to extract a subset of items from a list or tuple

- access a single item: `container[i]`
- access a range of items: `container[i:j]`
- negative indices i and j are interpreted as $n+i$ and $n+j$ where n is the size of the container, ie the index is counted from the right
- get all items starting at index i : `container[i:]`
- get all items up to index j (excluding it): `container[:j]`
- specify a step size k : `container[i:j:k]`
- count down when k is negative: `container[::-1]`



Containers

dictionaries

- create a dictionary
`aDict = {}`
- add some items
`aDict['first'] = 'hello \world'`
`aDict[10] = 1.0`
- get the keys
`print(aDict.keys())`
- access an item
`aDict['first']`
- delete an item
`del aDict['first']`



Containers

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

More Types
Containers
Arrays

Flow Control

File I/O

Plotting

Assessed
Programming
Task

Exercise

- create some lists and tuples
- try adding, modifying and deleting items
- try out the slicing
- create a dictionary with the names and ages of the people at your desk (you can also make them up). Modify one of the values, delete an item.



Arrays

arrays vs lists

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

More Types
Containers
Arrays

Flow Control

File I/O

Plotting

Assessed
Programming
Task

- List
 - one-dimensional
 - can dynamically grow/shrink
 - collection of any kind of python object (including lists)
 - + concatenates two lists
- arrays
 - multi-dimensional
 - fixed size
 - array elements have to be all of the same type (including python objects)
 - element-wise operations
 - universal functions, e.g. sine, square,...
- both
 - (unlike fortran or matlab) zero based indexing
 - slicing [start:stop:step]



Arrays

Array Creation

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

More Types
Containers
Arrays

Flow Control

File I/O

Plotting

Assessed
Programming
Task

```
import numpy
```

```
a = numpy.array ([1,2,3,4,5])
```

```
a = numpy.array ([1,2,3,4,5], dtype=float)
```

```
b = numpy.array ([[1,2],[3,4]])
```

```
c = numpy.zeros([10,20])
```

```
d = numpy.arange(0,10)+1
```

```
d = numpy.linspace(0,10,21)
```

```
e = numpy.logspace(1,10,10)
```

check the numpy documentation on [array creation](#)



Arrays

Array Attributes and Methods

- `ndarray.shape` tuple of array dimensions
- `ndarray.ndim` the number of dimensions
- `ndarray.nbytes` total amount of memory consumed by the array
- `ndarray.dtype` the data type of the array
- `ndarray.T` the transpose of the array
- `ndarray.reshape(shape)` return an array containing the same data with a new shape
- `ndarray.tolist()` return the array as a (possibly nested) list

ndarrays have many more **attributes** and **methods**.



Arrays

Array Indexing and Slicing

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

More Types
Containers
Arrays

Flow Control

File I/O

Plotting

Assessed
Programming
Task

Similar to indexing and slicing of lists and tuples only more dimensions:

- $x[i:j:k]$ where i is the start index, j the stop index and k the step. The index j is never reached.
- negative i and j are interpreted as $n+i$ and $n+j$ where n is the size of the dimension, ie the index is counted from the right
- negative k means reversing the order
- you can omit any of the i , j and/or k . They are by default $i=0$, $j=n$ and $k=1$
- multiple dimensions are separated by a comma, eg
 $x[i_1:j_1:k_1, i_2:j_2:k_2]$



Arrays

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

More Types

Containers
Arrays

Flow Control

File I/O

Plotting

Assessed
Programming
Task

a_{11}	a_{12}	a_{13}	a_{14}
a_{21}	a_{22}	a_{23}	a_{24}
a_{31}	a_{32}	a_{33}	a_{34}

- a 3×4 matrix $\mathbf{A} = a_{ij}$
- computer memory is 1D and linear
- need to decide how the elements are laid out in memory



Arrays

column-major ordering

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

More Types
Containers
Arrays

Flow Control

File I/O

Plotting

Assessed
Programming
Task

a_{11}	a_{12}	a_{13}	a_{14}
a_{21}	a_{22}	a_{23}	a_{24}
a_{31}	a_{32}	a_{33}	a_{34}

- consecutive elements of a column are next to each other
- Fortran, matlab
- first index varies fastest



Arrays

row-major ordering

a_{11}	a_{12}	a_{13}	a_{14}
a_{21}	a_{22}	a_{23}	a_{24}
a_{31}	a_{32}	a_{33}	a_{34}

- consecutive elements of a row are next to each other
- C, python
- last index varies fastest



Arrays

linearisation

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

More Types
Containers
Arrays

Flow Control

File I/O

Plotting

Assessed
Programming
Task

a_{11}	a_{12}	a_{13}	a_{14}
a_{21}	a_{22}	a_{23}	a_{24}
a_{31}	a_{32}	a_{33}	a_{34}

Assuming we have a $M \times N$ 2D array A with M rows and N columns and elements a_{ij} we can compute the 1D index using python row-major ordering

$$k = iN + j$$

where $i = 0, \dots, M - 1$ and $j = 0, \dots, N - 1$.

Given a linear index k we can compute the 2D indices i, j

$$i = \lfloor k/N \rfloor$$

$$j = k \bmod N$$



Arrays

Exercise

- create some arrays, both 1D and multi-dimensional arrays,
eg

$$\begin{bmatrix} 1 & 2 & 3 & 4 & 5 \end{bmatrix}$$
$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$$

- you can use the reshape method to turn a 1D array into a multi-dimenstional array
- investigate the various array attributes
- do some operations on arrays, like adding two arrays
- play with slicing
- figure out with row/column major means



Flow Control

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

More Types

Flow Control

Loops

if-then-else

while loops

File I/O

Plotting

Assessed

Programming

Task

control which statements are executed

- conditional branches
- loops
- subroutines
- program halts



Flow Control

Loops

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...
Objects

More Types

Flow Control
Loops
if-then-else
while loops

File I/O

Plotting

Assessed
Programming
Task

Sometimes you would like to get a sequence of numbers, $0, 1, 2, 3, \dots, n$. You can use the `range` type to produce the sequence of numbers.

```
for i in range(10):  
    print(i)
```

- `i` is the loop variable or loop counter
- at each iteration of the loop it assumes the next value



Flow Control Loops

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

More Types

Flow Control

Loops

if-then-else

while loops

File I/O

Plotting

Assessed
Programming
Task

Containers are also *iterable*, ie you can visit each item of a container in turn:

```
aList = ['Edinburgh', 'Glasgow', 'Aberdeen']  
for x in aList:  
    print(x, len(x))
```

- x is the loop variable or loop counter
- at each iteration of the loop it assumes one value in the container



Flow Control Loops

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...
Objects

More Types

Flow Control
Loops
if-then-else
while loops

File I/O

Plotting

Assessed
Programming
Task

when iterating over dictionaries, the loop variable assumes the keys

```
aDict = { 'Edinburgh':488050,  
          'Glasgow':1209143,  
          'Aberdeen':196670}  
for k in aDict:  
    print(k,aDict[k])
```



Flow Control

Exercise

- write a hello-world program that writes "Hello, World!" ten times
- create some lists and dictionaries
- loop over them
- what happens when you loop over dictionaries?
- try combining loops with slices
- what happens when you loop over string?



Flow Control

if–then–else

- used to branch executing depending on some boolean condition
- syntax

```
if condition1:  
    do_something  
elif condition2:  
    do_something_else  
else:  
    do_something_completely_different
```
- you need at least one if condition, you can have one or more elif conditions, you can have one else condition.



Flow Control

logical expressions

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...
Objects

More Types

Flow Control
Loops
if-then-else
while loops

File I/O

Plotting

Assessed
Programming
Task

- evaluates to either True or False
- comparison operators: ==, <, <=, >, >=, !=
- logical operators: and, or, not
- membership operators: in, not in
- identity operators: is, is not



Flow Control

if–then–else

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

More Types

Flow Control

Loops

if–then–else

while loops

File I/O

Plotting

Assessed

Programming

Task

```
if a>0:  
    print('a is greater than 0')  
elif a<0:  
    print('a is less than 0')  
else:  
    print('a is zero')
```



Flow Control

while loops

- execute the loop body while some condition is true
- syntax
 - while** condition :
 do_something
- you can jump out of the loop using **break**
- you can jump to the next iteration of the loop using **continue**



Flow Control

while loops

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

More Types

Flow Control

Loops

if-then-else

while loops

File I/O

Plotting

Assessed
Programming
Task

you can simulate a for-loop:

```
for i in range(n):  
    do_something
```

```
i = 0
```

```
while i < n:  
    do_something  
    i = i+1
```



Flow Control

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

More Types

Flow Control

Loops

if-then-else

while loops

File I/O

Plotting

Assessed
Programming
Task

Exercise

- use a while loop to write "Hello, World!" ten times
- use an endless loop (where the loop condition is always True) to write "Hello, World!" ten times
- use a while loop to print even numbers between -1 and 11



Flow Control

Solution 0 — the obvious

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

More Types

Flow Control

Loops

if–then–else

while loops

File I/O

Plotting

Assessed

Programming

Task

```
for i in range(0,11,2):  
    print(i)
```



Flow Control

Solution 1 — using a while loop

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

More Types

Flow Control

Loops

if-then-else

while loops

File I/O

Plotting

Assessed

Programming

Task

```
i=0
while i < 11:
    print(i)
    i = i + 2
```



Flow Control

Solution 2 — doing extra work

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

More Types

Flow Control

Loops

if-then-else

while loops

File I/O

Plotting

Assessed

Programming

Task

```
i=0
while i < 11:
    if i%2 == 0:
        print(i)
    i = i + 1
```



Flow Control

Solution 3 — test for odd numbers

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

More Types

Flow Control

Loops

if-then-else

while loops

File I/O

Plotting

Assessed

Programming

Task

```
i=-1
while i < 11:
    i = i + 1
    if i%2 == 1:
        continue
    print(i)
```



Flow Control

Solution 4 — beware the endless loop

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

More Types

Flow Control

Loops

if-then-else

while loops

File I/O

Plotting

Assessed
Programming
Task

```
i=0
while True:
    if i >10:
        break
    if i%2 == 0:
        print(i)
    i = i + 1
```



File I/O

Inevitably you will want to read input data from a file or write results to a file. Before you can do any operations on a file you need to [open](#) it.

```
aFile = open(fname,mode)
```

where

- `fname` is a string containing the name (including path) of the file to be opened and
- `mode` is an operational string that specifies how the file should be opened. It can be
 - 'r' open for reading (default)
 - 'w' open for writing, truncating the file first
 - 'x' open for exclusive writing, failing if it already exists
 - 'a' open for writing, appending to the end of the file if it already exists
 - '+' open a disk file for updating (reading and writing)



File I/O

Once you have a file object you can use [various methods](#) to read/write

- `aFile.read()` read the entire file into a string
- `aFile.write(msg)` write the string `msg` to the file
- `aFile.readline()` read a *single* line from the file
- `aFile.readlines()` get an iterator that you can use to read a file line by line
- `aFile.seek()` to jump to a particular location
- `aFile.close()` to close the file



File I/O

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

More Types

Flow Control

File I/O

Plotting

Assessed
Programming
Task

```
data = []
inFile = open('some_file', 'r')
for line in inFile.readlines():
    data.append(line)
inFile.close()
print(data)
```



File I/O

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...
Objects

More Types
Flow Control

File I/O

Plotting

Assessed
Programming
Task

Exercise

- open the file scatter.data
- read the two columns into two lists
- how would you convert the data to floats?



Plotting using Matplotlib

[Matplotlib](#) is a 2D plotting library which produces publication quality figures.

- basic interface similar to matlab
- full pythonic [OO interface](#)
- extra toolkits, including [3D plotting](#) and maps
- output to jpeg, pdf, svg, etc
- embed in GUIs or ipython

The [gallery](#) is a great place to start looking what it can do



Plotting using Matplotlib

xy plots

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

More Types

Flow Control

File I/O

Plotting

Assessed
Programming
Task

```
import numpy
from matplotlib import pyplot

x=numpy.random.normal(size=(500,))
y=numpy.random.normal(size=(500,))

pyplot.plot(x,y, 'o')
pyplot.ylabel('some_random_numbers')
pyplot.xlabel('some_more_random_numbers')
pyplot.title('some_random_points')
pyplot.show()
```



Plotting using Matplotlib

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

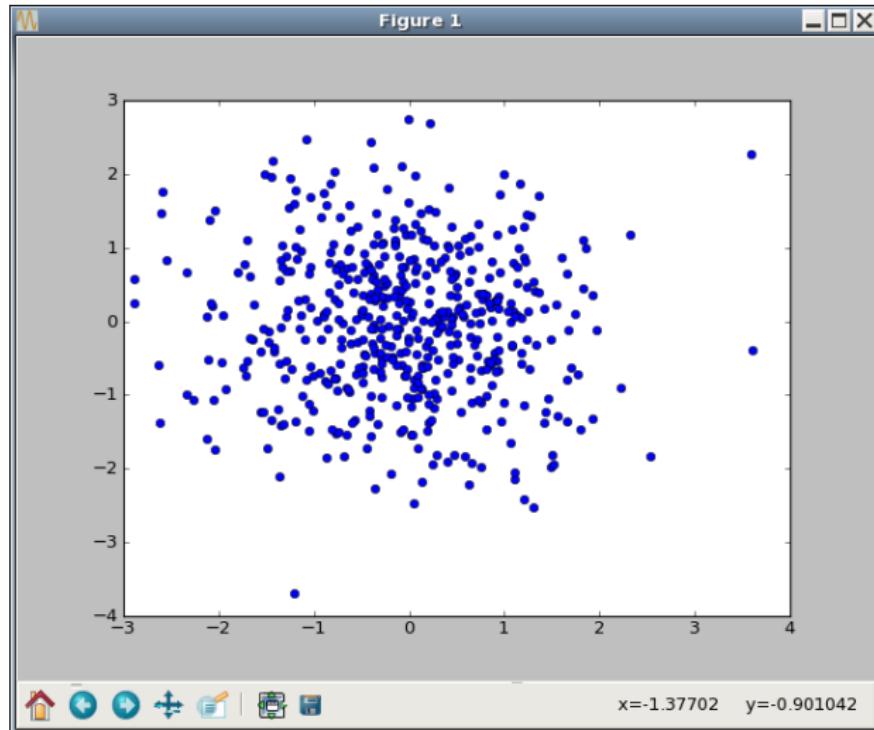
More Types

Flow Control

File I/O

Plotting

Assessed
Programming
Task





Plotting using Matplotlib

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

More Types

Flow Control

File I/O

Plotting

Assessed

Programming
Task

Exercise

- plot the data you read in the last example
- play with various styles, labels, etc
- have a look at the matplotlib gallery

<http://matplotlib.org/gallery.html>



More Python plotting timeseries

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

More Types

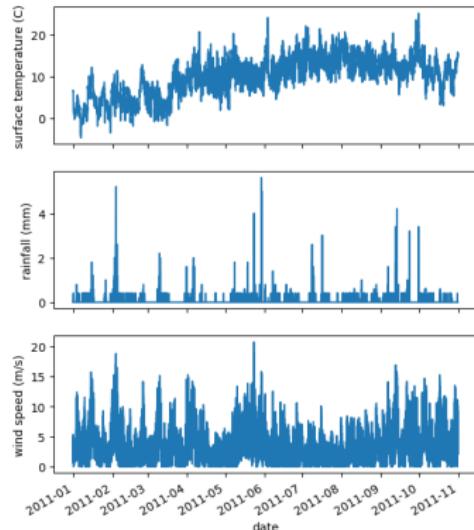
Flow Control

File I/O

Plotting

Assessed
Programming
Task

```
mate-terminal
File Edit View Search Terminal Tabs Help
mate-terminal X mate-terminal X
date-time,atmospheric pressure (mBar),rainfall (mm),wind speed (m/s),wind direct-
ion (degrees),surface temperature (C),relative humidity (%),solar flux (Kw/m2),s
tevenson screen temperature (C),battery (V)
2011/01/01 00:01,1022,0,4.104,239.1,6.593,85.8,0,-9999,-9999
2011/01/01 00:02,1022,0,4.167,248.5,6.591,85.9,0,-9999,-9999
2011/01/01 00:03,1022,0,3.854,250.9,6.582,85.9,0,-9999,-9999
2011/01/01 00:04,1022,0,4.104,239.1,6.593,85.8,0,-9999,-9999
2011/01/01 00:05,1022,0,3.917,248.8,6.591,85.9,0,-9999,-9999
2011/01/01 00:06,1022,0,3.688,244.1,6.580,85.9,0,-9999,-9999
2011/01/01 00:07,1022,0,4.521,241.6,6.58,85.9,0,-9999,-9999
2011/01/01 00:08,1022,0,4.063,246.4,6.588,85.8,0,-9999,-9999
2011/01/01 00:09,1022,0,4.394,246.2,6.588,85.8,0,-9999,-9999
2011/01/01 00:10,1022,0,3.313,249.6,6.577,85.8,0,-9999,-9999
2011/01/01 00:11,1022,0,3.458,247.6,6.587,85.8,0,-9999,-9999
2011/01/01 00:12,1022,0,3.146,239.1,6.582,85.8,0,-9999,-9999
2011/01/01 00:13,1022,0,3.729,244.7,6.555,85.9,0,-9999,-9999
2011/01/01 00:15,1022,0,3.917,244.7,6.555,85.9,0,-9999,-9999
2011/01/01 00:16,1022,0,4.249,5.6,552,85.9,0,-9999,-9999
2011/01/01 00:17,1022,0,3.125,248.3,6.555,85.8,0,-9999,-9999
2011/01/01 00:18,1022,0,2.979,252.6,6.55,85.8,0,-9999,-9999
2011/01/01 00:19,1022,0,3.252,5.6,555,85.8,0,-9999,-9999
2011/01/01 00:20,1022,0,2.667,253.4,6.522,85.8,0,-9999,-9999
JCHB_2011.csv
```





More Python

plotting timeseries

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

More Types

Flow Control

File I/O

Plotting

Assessed
Programming
Task

- 1 write a function that reads the data set (`JCMB_2011.csv`)
- 2 select a suitable data structure (maybe a dictionary)
- 3 handle the header line
- 4 you can convert the first column to a datetime object which can be plotted nicely, see <https://docs.python.org/3.6/library/datetime.html>
- 5 you can handle missing data using masked arrays, see <https://numpy.org/doc/1.16/reference/maskedarray.html>
- 6 plot some fields, eg surface temperature, rainfall and wind speed

More Python

Assessed Programming Task – Task 1

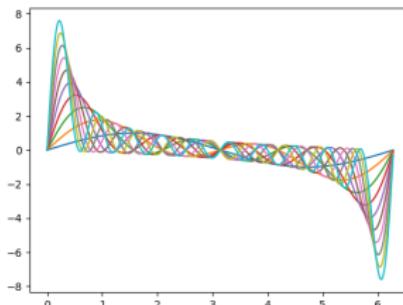
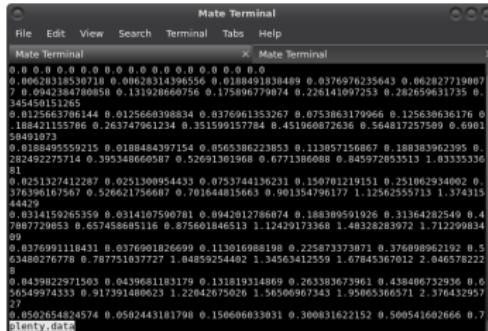
Technological
Infrastructures
for GIS

Magnus
Hagdorn

Objects

File I/O

Assessed Programming Task



Task 1

- 1 write a function that reads the data set (plenty.data)
 - 2 the data should be converted to float
 - 3 handle any number of columns
 - 4 plot all data
 - 5 comment your code



More Python

Assessed Programming Task – Task 2

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

More Types

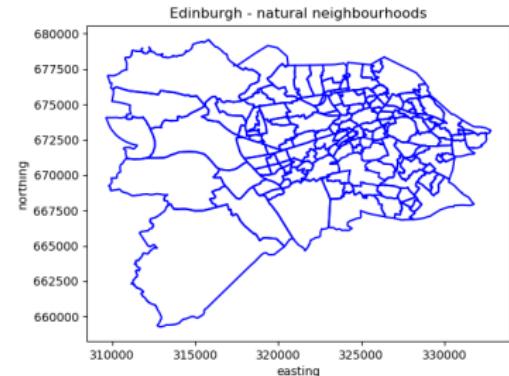
Flow Control

File I/O

Plotting

Assessed
Programming
Task

```
Terminal
File Edit View Search Terminal Help
# Natural neighbourhoods
# http://data.edinburghcouncilmaps.info/datasets/4082b44746eb4da8b5935be2d3a0018
5.27
Brunstane/Gilberstoun
((331222, 621000000003, 672917, 15310000000)
(331336, 094600000004, 672911, 78160000003)
(331488, 494900000003, 672932, 4191999994)
(331676, 614000000006, 672964, 9629999995)
(331708, 7516999999, 672982, 4254999999)
(331813, 3679999998, 673006, 80870000009)
(331821, 55399999954, 672997, 9003999997)
(331831, 1991999997, 673006, 8398999995)
(331854, 31709999964, 672978, 34380000008)
(331919, 9841, 672931, 1251999997)
(331968, 33330000000, 672887, 30600999997)
(332824, 6942999996, 672812, 57870000004)
(332851, 449300000044, 672857, 07760000004)
(332151, 1787999999, 672944, 7237999998)
(332214, 6503999997, 672973, 4368999992)
(332258, 47759999987, 672930, 98990000004)
(332389, 84960000003, 672952, 77690000009)
(332421, 31800000001, 672955, 36790000004)
(332421, 07000000002, 672988, 5428999998)
natural_neighbourhoods.dat
```



- projection is OSGB 1936/British National Grid
- data set downloaded from City of Edinburgh Council Mapping Portal
- data is licensed under Open Government License v3.0



More Python

Assessed Programming Task – Task 2

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

More Types

Flow Control

File I/O

Plotting

Assessed
Programming
Task

- start with parsing a single line of data
'(331222.6210000003, 672917.1531000007)',
you should get two floats
- read one polygon including header



More Python

Assessed Programming Task – Task 2

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

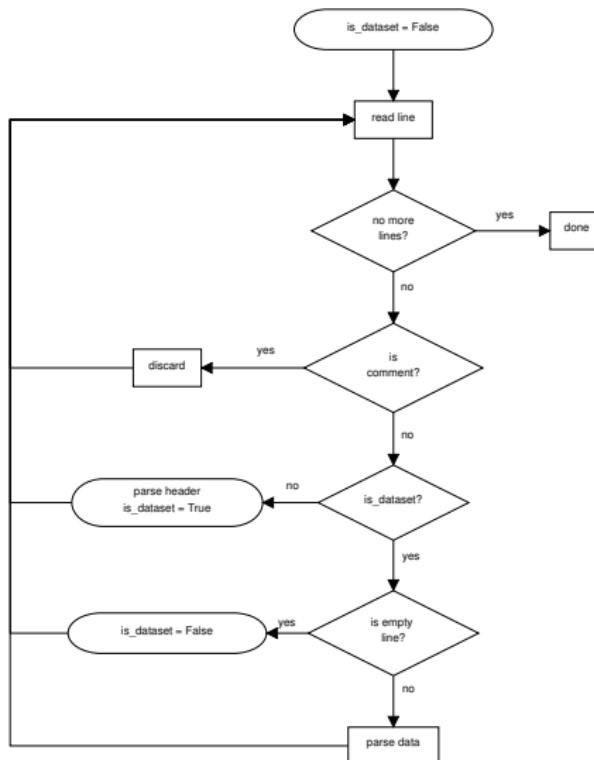
More Types

Flow Control

File I/O

Plotting

Assessed
Programming
Task





More Python

Assessed Programming Task – Task 2

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

More Types

Flow Control

File I/O

Plotting

Assessed
Programming
Task

- 1 read single polygon data set `poly.dat` to figure out how to handle coordinates (splitting and slicing the strings will be useful)
- 2 write a function that reads the data set (`natural_neighbourhoods.dat`)
- 3 select a suitable data structure (maybe a dictionary)
- 4 handle comments and the header lines
- 5 plot the polygons
- 6 add labels to plot
- 7 comment your code
- 8 have a look at the [cartopy](#) and produce a nice map



More Python

Assessed Programming Task

Technological
Infrastructures
for GIS

Magnus
Hagdorn

Last Time...

Objects

More Types

Flow Control

File I/O

Plotting

Assessed
Programming
Task

intermediate deadline — noon Tuesday Week 5

Please submit a python file solving task 1. We will provide some limited feedback. We'd just like to check you are on the right track.

both task 1 and 2 due noon Tuesday Week 8

Submit two python files, one solving task 1, the other task 2 to learn.