Soil Mechanics and Geotechnical Analysis
Programming with Python

Dr Daniel Barreto

d.barreto@napier.ac.uk

May 19th, 2025

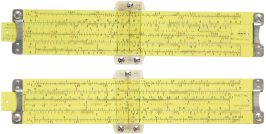
Programme

Day	08:00-09:30	09:45-11:15	13:00-14:30	14:45-16:15
19/05/25	Introduction	Programming	Phase Rel.	Tutorial
20/05/25	Classification	Tutorial	LAB	LAB
21/05/25	Effective Str.	Tutorial		
22/05/25	Seepage	Tutorial	LAB	LAB
23/05/25	Str. Incr.	Tutorial		
26/05/25	Settl.	Tutorial	Settl.	Tutorial
27/05/25	Tutorial	Shear Str.	LAB	LAB
28/05/25	Shear Str.	Tutorial		
29/05/25	Footings	Tutorial	Piles	Piles
30/05/25	Tutorial	Review		

Overview

1. Why programming?
2. Introduction to python
3. More on Jupyter notebooks
4. Useful links

Why do you need to learn how to program?



We live in the digital age, slide rulers (and arguably all pocket calculators) are a thing of the past

Why python?

The python that is relevant to us

You should have installed the biggest snake (Anaconda)

If you have not done this, then it is time to do it now. It is easy to do, but it may take a few minutes depending on your computer's performance.

Go to https://www.anaconda.com/, click on "Free download" at the top, skip registration, download distribution, and follow the instructions.

Then initiate the Anaconda Navigator, then launch a Jupyter Notebook and we can start exploring!

Python - comments

A line starting with # is a comment. It is then ignored by the computer. Comments are often used to describe your program and what it is doing.

```
>>> # This is a comment
>>> # ... and this one too!
```

Try it with the Jupyter Notebook!

Python - use as an old calculator

Write the operation, prese Shift+Enter and see the result!

```
>>> 2+2
>>> 4
```

Try subtraction (-), product(*), division(/), and power(**)
Also try

```
>>> print(2+2)
    4
```

Note that the result is output differently. You should use () to give priority unless you use the BODMAS rule (do you know it?)

Python - variables

Use specific, case-sensitive names!

```
>>> radius = 1.0
>>> print(radius)
    1
```

You can also do calculations using variables:

```
>>> rectangleLength = 2.0
>>> rectangleHeight = 3.0
>>> rectangleArea = rectangleLength*rectangleHeight
>>> print(rectangleArea)
    6.0
```

Try with different dimensions... No need to re-write!

Python - variable types

Try the following:

```
>>> height = 1.0
    length = 2
    units = ‘square metres’
    area = height*length
    yes = True
    type(height)
>>> float
```

Use the type() function for ‘height’, ‘length’, ‘units’, ‘area’ and ‘yes’

Python - lists

Create a list of heights for 5 of your friends. For example:

```
>>> height1 = 1.75
>>> height2 = 1.75
>>> height3 = 1.78
>>> height4 = 1.54
>>> height1 = 1.65
```

A bit inconvenient. Try this instead:

```
>>> heights = [1.75, 1.75, 1.78, 1.54, 1,65]
>>> print(heights)
    [1.75, 1.75, 1.78, 1.54, 1,65]
```

Python - lists

Lists can contain any variable type, and also contain different types

```
>>> friends = ["Zhongxuan", 1.54, "Lizhong", 1.65]
>>> print(friends)
    ["Zhongxuan", 1.54, "Lizhong", 1.65]
>>> type(friends)
```

‘list’ is another variable type. Now try this:

```
>>> friends = [["Yi", 1.73],
               ["Lizhong", 1.65]]
>>> print(friends)
    [["Yi", 1.73], ["Lizhong", 1.65]]
```

Python - subsetting and slicing lists

What do these lines/commands do?

```
>>> friends = ["Zhongxuan", 1.54, "Lizhong", 1.65]
>>> friends[3]
>>> friends[-1]
>>> friend[1:3]
```

Note:

Indexing in python starts at 0, not 1.

Index -1 takes you to the end of the list.

Slicing [start : end] is [inclusive : exclusive].

Python - changing lists

List elements can be changed...

```
>>> friends = ["Zhongxuan", 1.54, "Lizhong", 1.65]
>>> friends[1] = 1.53
>>> print(friends[1])
```

You can add/remove elements to lists. Try it!

```
>>> friends = ["Zhongxuan", 1.54, "Lizhong", 1.65]
>>> friends_ext = friends + ["Yi", 1.50]
>>> print(friends_ext)
>>> del friends_ext[2]
```

Python - functions

List elements can be changed...

```
>>> friends = ["Zhongxuan", 1.54, "Lizhong", 1.65]
>>> friends[1] = 1.53
>>> print(friends[1])
```

We used type() before. That is an intrinsic function, just like max() and min()! Create a named list of numbers and use these three functions.

Add a number to the list using listname.append(number). append is a method. Methods call functions on objects.

More on Jupyter notebooks

We will learn (a lot more about python as we progress with the course. For the moment we can return to Jupyter notebooks to:

- Add explanatory text (not comments with #) using Markdown language
- Add cells
- Save and download the notebook

After saving and dowloading your notebook, open it again in Ananconda/Jupyter

... and start using Github. Visit the course repository to see what this is about!

https://github.com/dbarretog/shnu-sm

Useful links

For installation and access:

(Anaconda) https://www.anaconda.com/
(Github) https://github.com/

For reference and documentation:

(Python) https://docs.python.org/3/
(Markdown) https://www.markdownguide.org/

Before you go for lunch...

Are there any questions?