

COMPUTATIONAL PHYSICS WITH PYTHON

FIRST STEPS: SHELLS, SCRIPTS AND NOTEBOOKS

Tobias Micklitz, Linneu Holanda, Carsten Hensel

OUTLINE

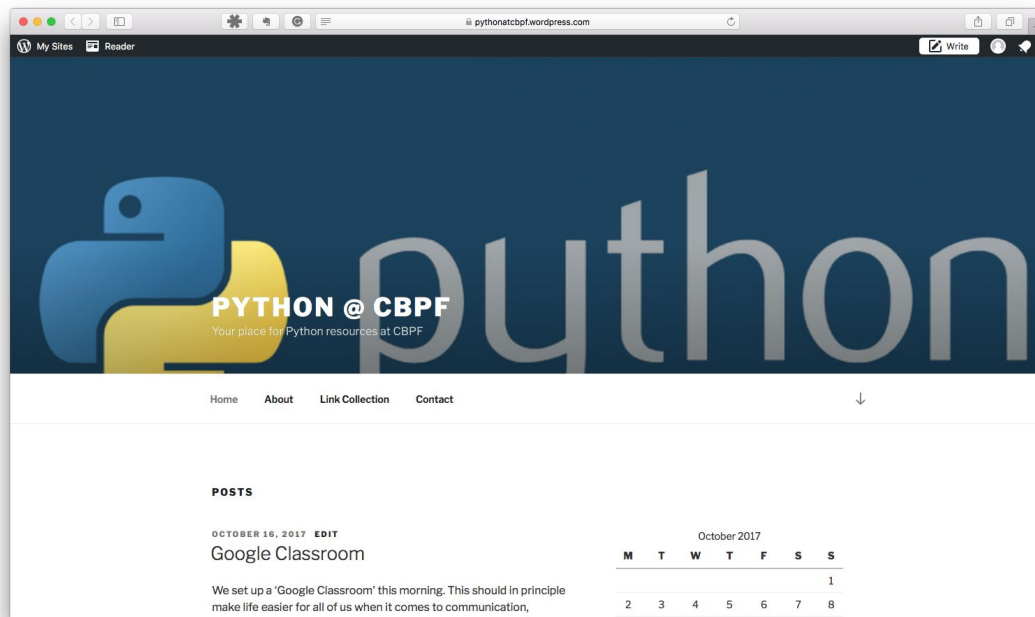
- Organisational Issues
 - Website
 - Google Classroom
- Python - First Steps
 - The Python Shell
 - Python Scripts
 - Jupyter Python Notebooks



ORGANISATIONAL ISSUES

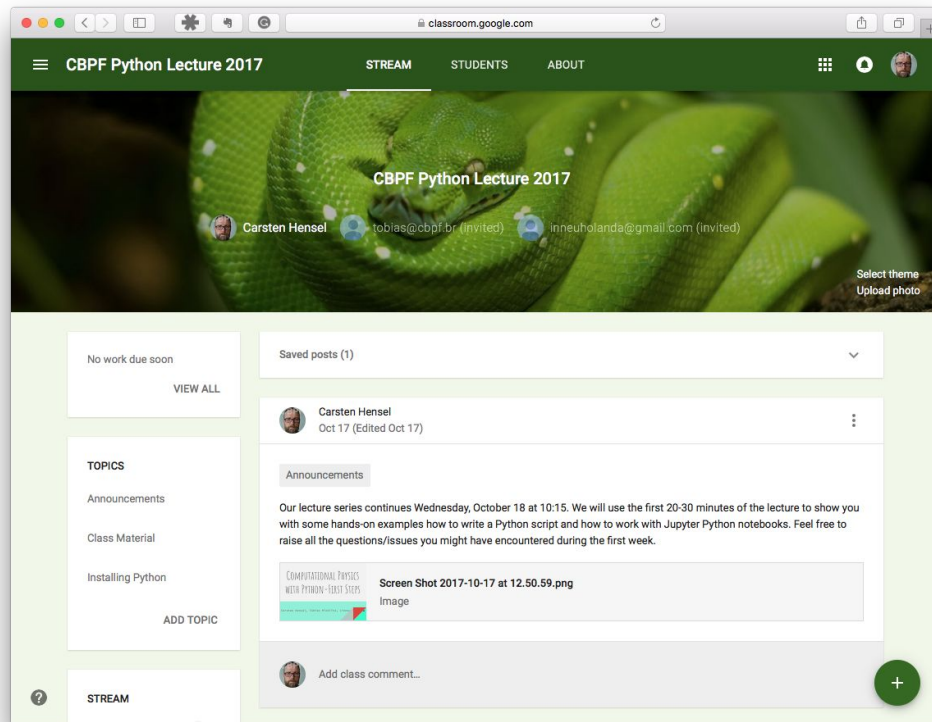
WEBSITE

- pythonatcbpf.wordpress.com
- Central place for everything Python at CBPF.
- All the lecture material (slides, notebooks, etc.)
- Content Python related that doesn't fit into the lecture.



GOOGLE CLASSROOM

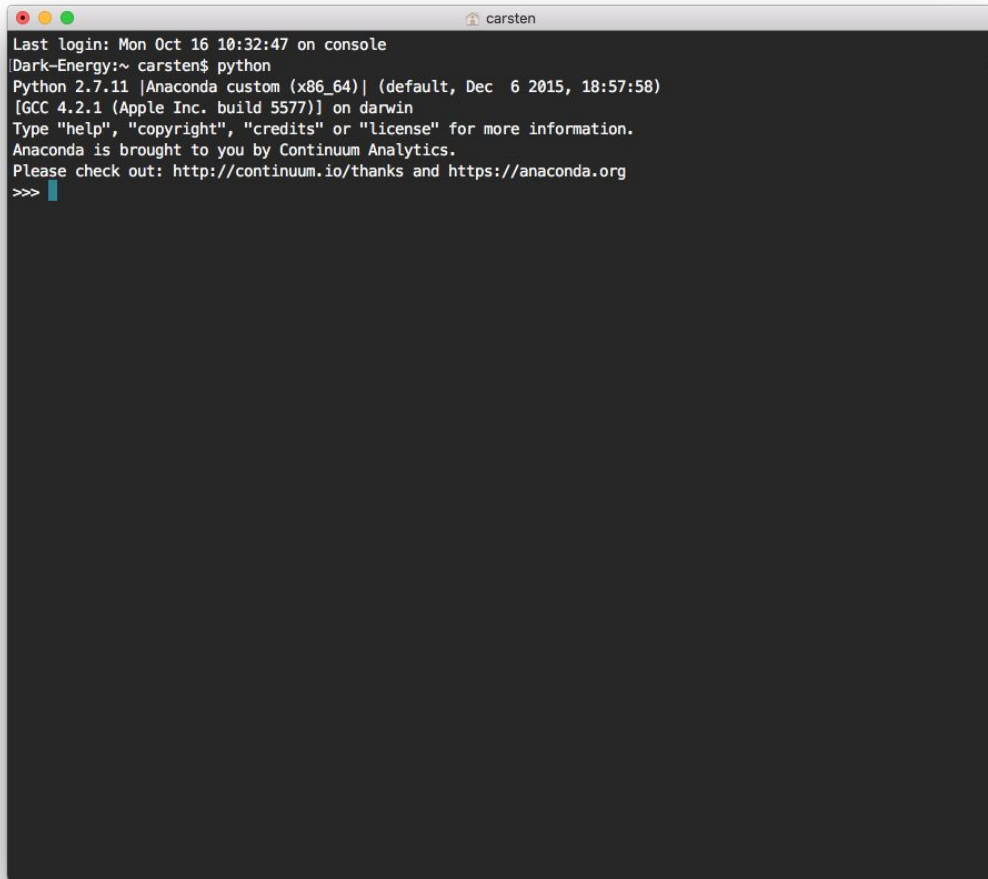
- Virtual Classroom
- Everyone should've received an invite. **If not, please let me know!**
- Everything lecture related.
 - announcements
 - communication
 - Q&A (?)
 - slides and notebooks
 - assignments (?)



PYTHON - FIRST STEPS

THE PYTHON SHELL

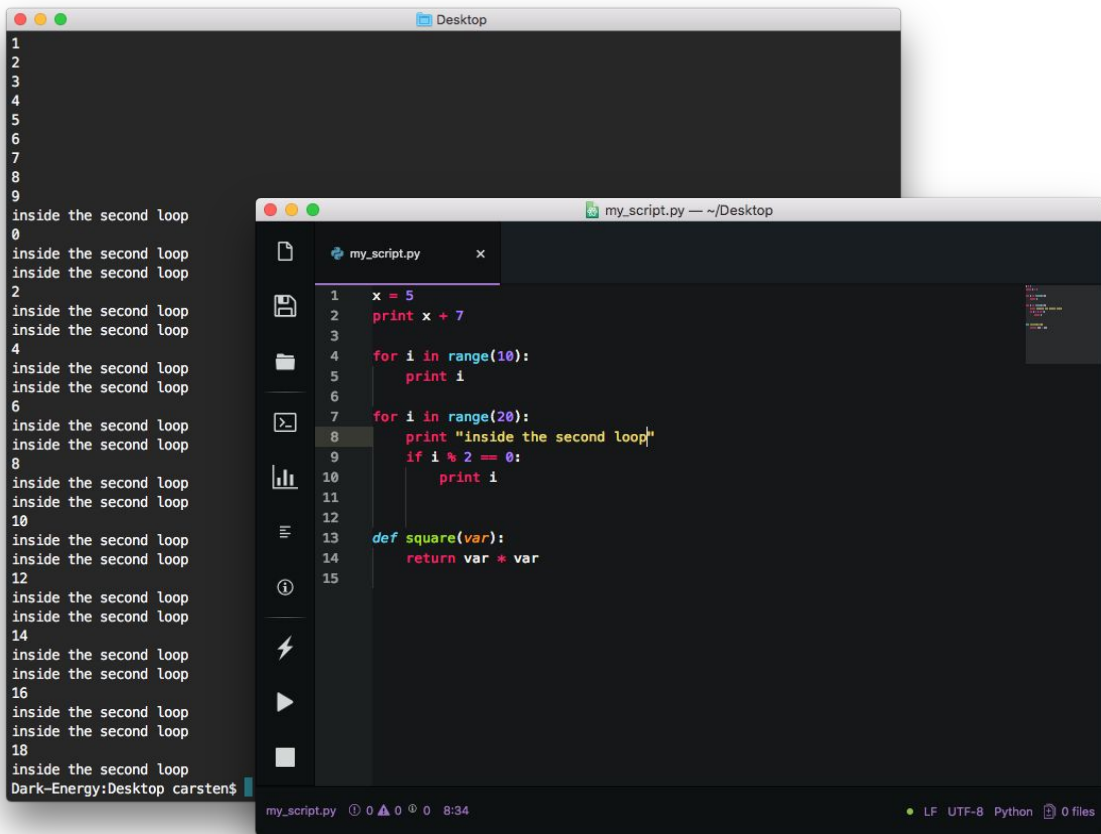
- From within a terminal window just type **python**
- Each line is one command.
- Use **dir** to investigate modules.
- Use **help** to learn about modules and functions.
- To exit hit **Ctrl-d**
- Not the best way to write longer programs.



```
Last login: Mon Oct 16 10:32:47 on console
Dark-Energy:~ carsten$ python
Python 2.7.11 |Anaconda custom (x86_64)| (default, Dec 6 2015, 18:57:58)
[GCC 4.2.1 (Apple Inc. build 5577)] on darwin
Type "help", "copyright", "credits" or "license" for more information.
Anaconda is brought to you by Continuum Analytics.
Please check out: http://continuum.io/thanks and https://anaconda.org
>>> 
```

WRITING A PYTHON SCRIPT

- The Python shell has a very limited use case.
- Every time you would like execute more than just a few commands you put your code into a file.
- Open your favorite text editor ([Emacs](#), [Atom](#), [Vim](#), [Light Table](#),...)
- And start typing...
- Execute your script from a terminal with
`python my_script.py`



The image shows two overlapping windows on a desktop. The background window is a terminal titled 'Desktop' with a dark background. It contains a list of numbers 1 through 9, followed by the text 'inside the second loop' repeated 18 times. The prompt at the bottom is 'Dark-Energy:Desktop carsten\$'. The foreground window is a code editor titled 'my_script.py -- ~/Desktop'. It shows a Python script with the following code:

```
1 x = 5
2 print x + 7
3
4 for i in range(10):
5     print i
6
7 for i in range(20):
8     print "inside the second loop"
9     if i % 2 == 0:
10        print i
11
12
13 def square(var):
14     return var * var
15
```

The code editor has a sidebar on the left with icons for file operations and a status bar at the bottom showing 'my_script.py', '0', 'UTF-8', 'Python', and '0 files'.

JUPYTER NOTEBOOKS

- Scripts have one big disadvantage: they lack interactivity.
- Especially when developing new code, producing a plot, calculating a number,... instant feedback would be great.
- **Jupyter Notebooks** to the help!



HOW TO USE JUPYTER?

- What is Jupyter?
 - Formerly known as IPython
 - Interactive programming environment
 - Not only available for Python (C++, Perl, Julia, R, ...)
 - Something between the Python shell and a Python script.
 - Allows to mix source code and commentary (-> makes for a great logbook)
- Jupyter comes pre-installed with Anaconda
- Jupyter runs in your browser (Chrome, Safari, Firefox, ...)
- How to start a notebook?
 - Open a terminal and type **jupyter notebook**
 - A new tab should open in your browser.

TYPICAL JUPYTER WORKFLOW

- Cells
- Adding text/code
- Code execution (**shift-return**)
- Clearing the output
- Adding and moving cells
- Export code to script

The screenshot displays the Jupyter Notebook interface in a web browser. The title bar shows 'localhost' and the notebook title is 'My Python Notebook'. The top menu bar includes 'File', 'Edit', 'View', 'Insert', 'Cell', 'Kernel', and 'Help'. The right side of the top bar shows 'Trusted', 'Python 2', and a 'Logout' button. The main area contains three cells:

- Code Cell (In [1]):** Contains a Python code snippet:

```
# This is a code cell.  
for i in range(10):  
    print i
```

The output shows the numbers 0 through 9, each on a new line.
- Text Cell:** Contains the text: 'This is a text cell. I can use markdown syntax to make the text look **pretty**.' Below this, it says 'Another text cell this time with LaTeX equations:' followed by the LaTeX equation
$$\int_0^{\infty} \frac{1}{x^2} dx$$
.
- Code Cell (In [2]):** Contains a Python function definition:

```
# more code  
import math  
def pythagoras(a, b):  
    return math.sqrt(a * a + b * b)
```
- Code Cell (In [3]):** Contains a single line of code:

```
print pythagoras(3,4)
```

The output shows the value 5.0.
- Empty Code Cell (In []):** A new, empty code cell is at the bottom, ready for input.

SUMMARY - PART 1

SUMMARY

- Let's try to utilize the communication tools introduced.
- There are various ways to work with Python.
 - Python shell - the quick calculator
 - Python script - for anything larger
 - Jupyter Notebook - an interactive programming environment
- Pick and choose according to your needs.