# Python
# at the speed of light

July 24, 2018

schupbach@fnal.gov

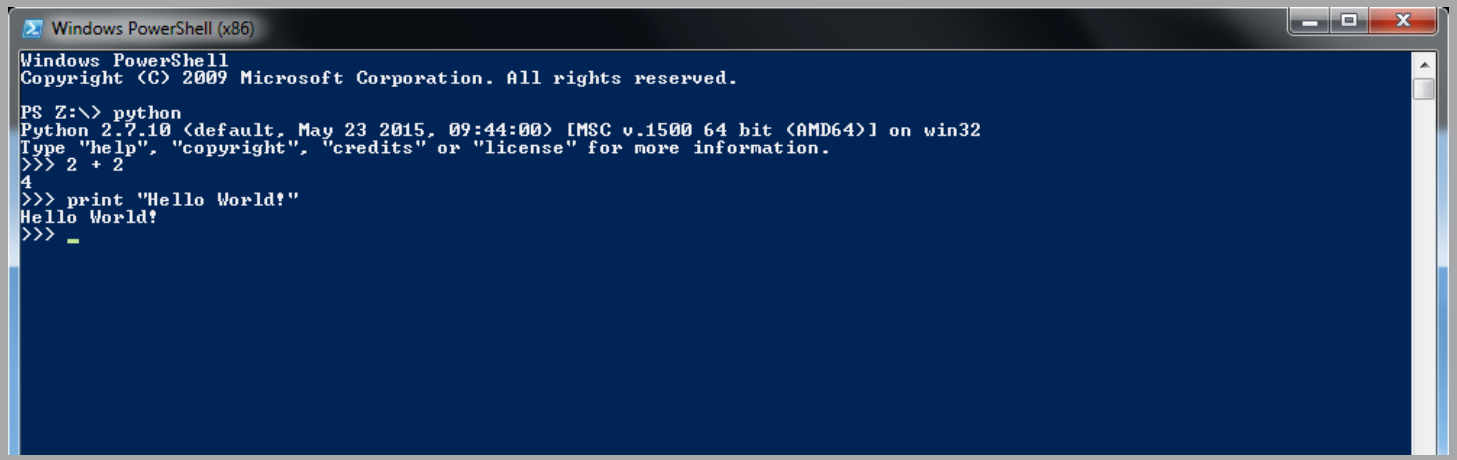
# Ground level

Python27 is a well supported high-level programming language to be used on small and large scales; from scripting to the Grid computer.

- interpreted, he said she said
  - (python) code $\rightarrow$ interpreter $\rightarrow$ bytecode $\rightarrow$ VM $\rightarrow$ bare metal
  - (C) code $\rightarrow$ asm $\rightarrow$ bare metal
- readable, efficient use of whitespace
  - no semi colons, curly braces
- multiparadigm
  - supports procedural, functional, and class based techniques
- yes, it's slow, but fast enough for most purposes, depends on what you are doing

# Post install, now what

-the interpreter




I need more libraries,

- use PIP, the python package manager
- C:\> pip install mylib

```
import mylib
from mylib import myfunc as f
```

# OK but

-what about source files

Within hello.py

```
import sys
print "Hello %s!" % (sys.argv[1])
```

Running the following from the terminal or cmd

```
C:\> python hello.py Tyler
Hello Tyler!
```

# the basics

## -data types

```
# I'm a comment

x = 42                          # integer
y = 13.0                        # float
name = "Tyler"                  # string

d = {"name" : name}             # dictionary
d['name'] = "Tyler"

a = [1,2,3,4]                   # homogeneous list
a[2] = 3

b = [1.0,2,{'i':3},"4"] # heterogeneous list
b[1] = 2
```

# the basics

-contd.

```
# import the numpy library
import numpy as np
a = np.array([2,3,4])

# just the array stuff
from numpy import array

# functions
def square(x):
    """square x"""
    return x*x

y = square(4)
```

To view the docstring for a function: print square.__doc__

# the basics

-contd.

```
# control
if i == 0:
    print "yep"
else:
    print "nope"

i = 0                       # the interpreter
while True:                 # will run out of memory...
    i = i + 1               # at some point

for i in range(5):          # print each elem in
    print i                 # the list [0,1,2,3,4]
```

# the basics

-contd.

## Read a file

```
lines = []
with open("file.txt", 'r') as f:
    line = f.readline()
    while line:
        lines.append(line)
        line = f.readline()
```

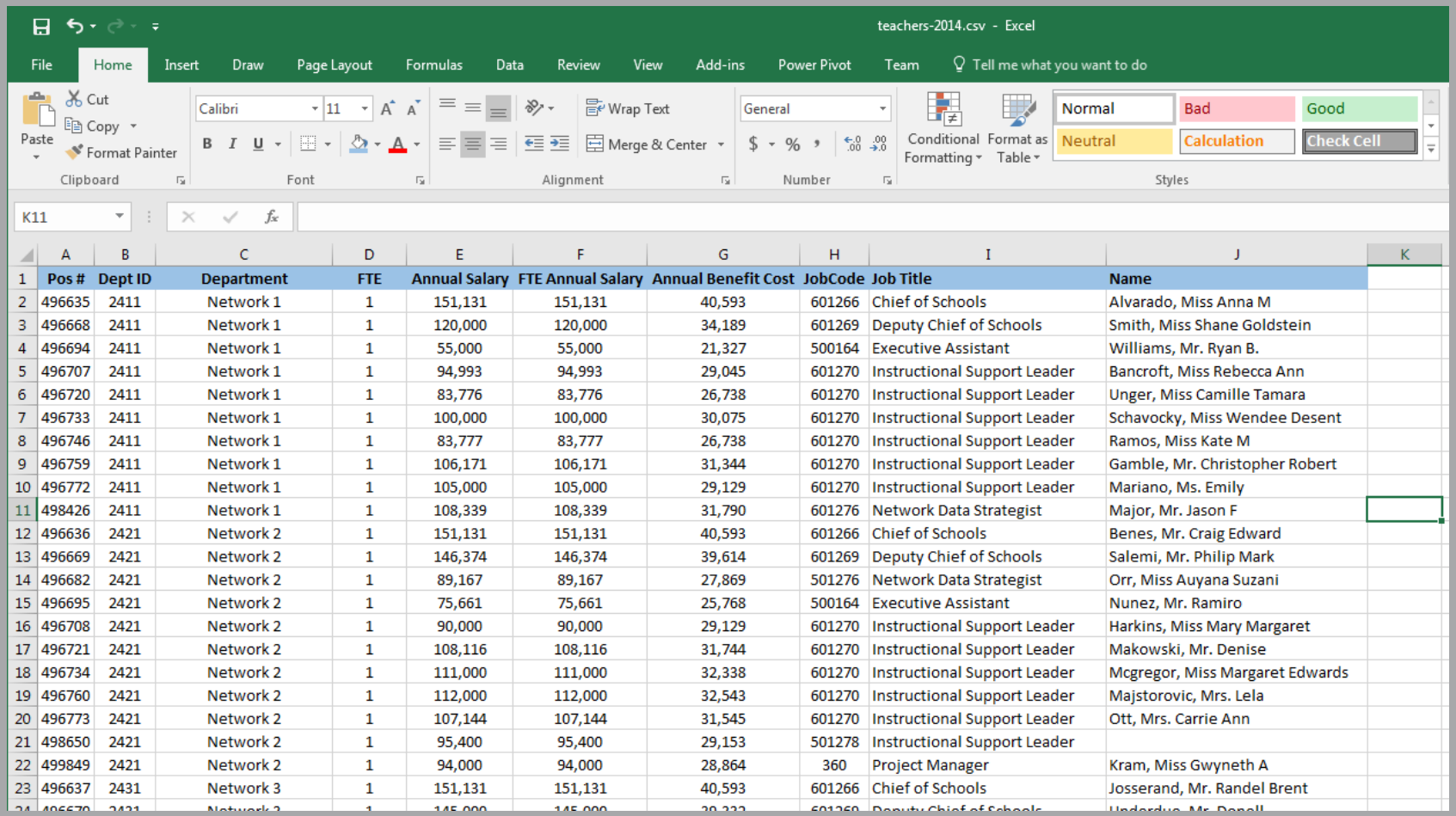
## Write to file

```
lines = ["hello\n","world\n"]
with open("file.txt", 'w') as f:
    for line in lines:
        f.write(line)
```

# code tour

## -Operating on Excel/CSV files

Let's start with the following example: ≈40k rows.




| | A | B | C | D | E | F | G | H | I | J | K |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | Pos # | Dept ID | Department | FTE | Annual Salary | FTE Annual Salary | Annual Benefit Cost | JobCode | Job Title | Name | |
| 2 | 496635 | 2411 | Network 1 | 1 | 151,131 | 151,131 | 40,593 | 601266 | Chief of Schools | Alvarado, Miss Anna M | |
| 3 | 496668 | 2411 | Network 1 | 1 | 120,000 | 120,000 | 34,189 | 601269 | Deputy Chief of Schools | Smith, Miss Shane Goldstein | |
| 4 | 496694 | 2411 | Network 1 | 1 | 55,000 | 55,000 | 21,327 | 500164 | Executive Assistant | Williams, Mr. Ryan B. | |
| 5 | 496707 | 2411 | Network 1 | 1 | 94,993 | 94,993 | 29,045 | 601270 | Instructional Support Leader | Bancroft, Miss Rebecca Ann | |
| 6 | 496720 | 2411 | Network 1 | 1 | 83,776 | 83,776 | 26,738 | 601270 | Instructional Support Leader | Unger, Miss Camille Tamara | |
| 7 | 496733 | 2411 | Network 1 | 1 | 100,000 | 100,000 | 30,075 | 601270 | Instructional Support Leader | Schavocky, Miss Wendee Desent | |
| 8 | 496746 | 2411 | Network 1 | 1 | 83,777 | 83,777 | 26,738 | 601270 | Instructional Support Leader | Ramos, Miss Kate M | |
| 9 | 496759 | 2411 | Network 1 | 1 | 106,171 | 106,171 | 31,344 | 601270 | Instructional Support Leader | Gamble, Mr. Christopher Robert | |
| 10 | 496772 | 2411 | Network 1 | 1 | 105,000 | 105,000 | 29,129 | 601270 | Instructional Support Leader | Mariano, Ms. Emily | |
| 11 | 498426 | 2411 | Network 1 | 1 | 108,339 | 108,339 | 31,790 | 601276 | Network Data Strategist | Major, Mr. Jason F | |
| 12 | 496636 | 2421 | Network 2 | 1 | 151,131 | 151,131 | 40,593 | 601266 | Chief of Schools | Benes, Mr. Craig Edward | |
| 13 | 496669 | 2421 | Network 2 | 1 | 146,374 | 146,374 | 39,614 | 601269 | Deputy Chief of Schools | Salemi, Mr. Philip Mark | |
| 14 | 496682 | 2421 | Network 2 | 1 | 89,167 | 89,167 | 27,869 | 501276 | Network Data Strategist | Orr, Miss Auyana Suzani | |
| 15 | 496695 | 2421 | Network 2 | 1 | 75,661 | 75,661 | 25,768 | 500164 | Executive Assistant | Nunez, Mr. Ramiro | |
| 16 | 496708 | 2421 | Network 2 | 1 | 90,000 | 90,000 | 29,129 | 601270 | Instructional Support Leader | Harkins, Miss Mary Margaret | |
| 17 | 496721 | 2421 | Network 2 | 1 | 108,116 | 108,116 | 31,744 | 601270 | Instructional Support Leader | Makowski, Mr. Denise | |
| 18 | 496734 | 2421 | Network 2 | 1 | 111,000 | 111,000 | 32,338 | 601270 | Instructional Support Leader | Mcgregor, Miss Margaret Edwards | |
| 19 | 496760 | 2421 | Network 2 | 1 | 112,000 | 112,000 | 32,543 | 601270 | Instructional Support Leader | Majstorovic, Mrs. Lela | |
| 20 | 496773 | 2421 | Network 2 | 1 | 107,144 | 107,144 | 31,545 | 601270 | Instructional Support Leader | Ott, Mrs. Carrie Ann | |
| 21 | 498650 | 2421 | Network 2 | 1 | 95,400 | 95,400 | 29,153 | 501278 | Instructional Support Leader | | |
| 22 | 499849 | 2421 | Network 2 | 1 | 94,000 | 94,000 | 28,864 | 360 | Project Manager | Kram, Miss Gwyneth A | |
| 23 | 496637 | 2431 | Network 3 | 1 | 151,131 | 151,131 | 40,593 | 601266 | Chief of Schools | Josserand, Mr. Randel Brent | |

# code tour

-contd.

Using data provided by Chicago Public Schools would like to know the following

- the number of employees in each titled position
- avg annual salary for each position

# code tour

-contd.

## Bring the data into the python env

```
1       # we need a library for working with csv files
2       import csv
3
4       # our data files
5       files = ['teachers-2014.csv', 'teachers-2015.csv']
6
7       # a list of dictionaries with keys and values
8       for file in files:
9           edus = []
10
11          # open the file, read the rows, and put them in a list
12          with open(file, 'r') as f:
13              reader = csv.DictReader(f)
14              for row in reader:
15                  teachers.append(row)
16
17          # process the edus list
18          # print the results
19
20      # teachers[0] = {'Pos #':'496635', 'Dept ID':'2411', ...}
```

# code tour

-contd.

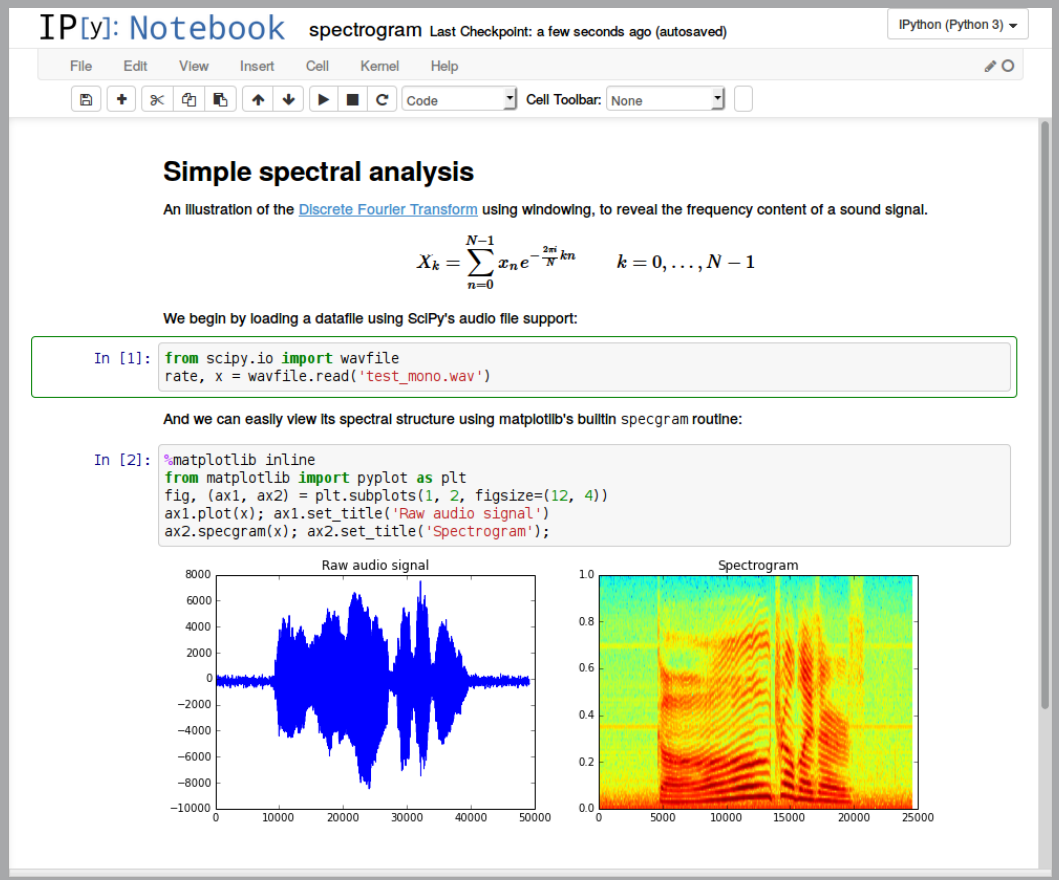
## Process the list of dictionaries

```
1       # a list of Regular Teachers
2       rts = filter(lambda t: float(t['FTE']) > 0.5 and t['Job Title'] == "Regular Teacher", edus)
3
4       # a list of salaries for the the Regular Teachers
5       sals = map(lambda t: float(t['FTE Annual Salary'].replace(',','')), rts)
6
7       # functionally
8       avg_fas = reduce(lambda x,y: x+y, sals) / len(sals)
9
10      # procedurally
11      avg_fas = 0
12      for s in sals:
13          avg_fas += s
14      avg_fas = avg_fas / len(sals)
```

```
1       # the results
2       print "%s %-64s (%5d): %6.2f" %(file,"Regular Teacher",len(sals),avg_fas)
3       > teachers-2014.csv Regular Teacher                (12881): 74840.72
4       > teachers-2015.csv Regular Teacher                (12872): 74792.72
```

# IDE's

Jupyter notebooks make for a convienient way to investigate and analyze data. Uses localhost and the browser as an interface to the interpreter.






Fermilab

# IDE's

-contd.

I like Visual Studio Code, free, lots of plugins, no browser per se, similar features.

teachers.py

```python
from operator import itemgetter
import csv

files = ['teachers-2014.csv', 'teachers-2015.csv', 'teachers-2016.csv']

for file in files:
    teachers = []
    titles = set()

    with open(file, 'r') as f:
        reader = csv.DictReader(f)
        for row in reader:
            teachers.append(row)
            if row['Job Title'] not in titles:
                titles.add(row['Job Title'])

    rt = filter(lambda t: float(t['FTE']) > 0.5 and t['Job Title'] == "Regular Teacher", teachers)
    sals = map(lambda t: float(t['FTE Annual Salary'].replace(',','')), rt)
    avg_fas = reduce(lambda x,y: x+y, sals) / len(sals)
    print "%s %-64s (%5d): %6.2f" %(file,"Regular Teacher",len(sals),avg_fas)

    pp = filter(lambda t: t['FTE'] > 0.5 and t['Job Title'] == "Principal", teachers)
    sals = map(lambda t: float(t['FTE Annual Salary'].replace(',','')), pp)
    avg_fas = reduce(lambda x,y: x+y, sals) / len(sals)
    print "%s %-64s (%5d): %6.2f\n" %(file,"Principal",len(sals),avg_fas)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: cmd

```
C:\Users\schupbach\Documents\code\python\teachers>python teachers.py
teachers-2014.csv Regular Teacher                                (12881): 74840.72
teachers-2014.csv Principal                                      (  442): 134400.69

teachers-2015.csv Regular Teacher                                (12872): 74792.72
teachers-2015.csv Principal                                      (  436): 133858.85

teachers-2016.csv Regular Teacher                                (12710): 73628.62
teachers-2016.csv Principal                                      (  430): 133245.20
```

# Resources

Some online resoures to help bring you up to speed. Start at the source.

- https://www.python.org/
- https://docs.python.org/2/index.html
- https://learnxinyminutes.com/docs/python/
- https://learnpythonthehardway.org/

Thanks.