

Python Crash Course Review (Part II)

1. Lists and other sequences

```
from math import pi
[] # empty list
a = [-7, 1., 'you', 3] # list (mutable)
B = [[1, 2, 3], pi, 2] # nested list
c = (1,2) # tuple (immutable)
d = 1,2 # tuple
e = 'numerical differential equations' # string
```

Change the first element of `a` to `a[0] = 10` and try to change the first element of `c` to `c[0]=10`.

```
print(a, c, d)
print('a+B: ',a+B) # concatenating two lists
```

Write two lists `i=[1,2,3]`, `j=[5]` and concatenate `i` and `j`.

1.a) Basics Indexing/Slicing (in Python list/array indices start with 0, in Matlab with 1)

```
print(e[0])
print(B, "\n", B[0][2]) # indexing a nested list
```

```
v = list(range(10))
print(v, "\n", v[-1], "\n", v[0:2], "\n", v[2:])
```

Create a new list `n = [0,3,6,9]` from `v` by specifying a suitable set of indices `n=v[start:stop:step]`.

<https://stackoverflow.com/questions/509211/understanding-slice-notation>

1.b) Indexing n dimensional arrays

```
import numpy as np
A = np.array([[0,1,2],[3,4,5]])
print("A =\n",A)
print(A.shape)
```

```
print(A[... ,0])
print(A[:,0])
print(A[:2,0])
print(A[-2:,0])
print(A[0:,0])
```

Find 3 (or 5) ways to create the array $A[0,:]=[0,1,2]$ from A.

<https://docs.scipy.org/doc/numpy-1.13.0/reference/arrays.indexing.html> .

1.c Assemble an nd-array from nested lists of blocks

```
B = np.array([[6,7,8],
              [9,10,11]])
C = np.block([[A],
              [B]])
print(C)
```

1.c Extract a submatrix using indexing arrays

```
i = np.array([0,1])
j = np.array([0,1])
C = np.ix_(i, j)
print(C)
```

```
(array([[0],
        [1]]), array([[0, 1]]))
```

Find the 'scipy.docs.org' reference describing the 'numpy.ix_' function.

2. Functions

2.a) Defining a function with two arguments which returns $f(x,y)=\sqrt{x}+\text{e}^y$. (Remember code block indentation)

```
def f(x, y):
    from math import sqrt, exp    # 4 empty spaces
    return sqrt(x)+exp(y)
```

```
print(f(0,0), f(1,1))
```

2.b) Always add a docstring to your python program.

```
def g(x=0, y=None):          # initialized parameter
    ''' demo function '''   # docstring
    from math import sqrt, exp
    #z = 2                   # local variable z
    return z, sqrt(x)+exp(y)
```

2.c) What is the scope of a variable (local/global)?

```
z = 4                        # global variable z
print(g(0,0))
print(z)
```

Uncomment z=2 in the function g.

```
help(g)
```

2.d) A python script file that calls its function, main scope

<https://docs.python.org/3/library/main.html>

```
def greet():
    print("Hello World")

if __name__ == "__main__":
    greet()
```

3. PEPs and more in-depth tutorials

The [Zen of Python](#) and [Style Guide for Python Code](#)

<https://docs.python-guide.org/>

<https://docs.python-guide.org/intro/learning/>

<http://www.davekuhlman.org/pythonbook01.html>