inclass demo

May 3, 2017

```
In [1]: 2+3
```

```
Out[1]: 5
```

```
In [2]: print "Hello world"
```

```
Hello world
```

```
In [3]: string1 = "Hello world"
```

```
In [4]: string2 = "sez Victor"
```

```
In [7]: string3 = string1+"! ("+string2+")"
```

```
In [8]: string3
```

```
Out[8]: 'Hello world! (sez Victor)'
```

```
In [10]: arr = [1,2,5,7,12]
```

```
In [11]: print arr[1]
```

```
2
```

```
In [13]: arr.append(23)
```

```
In [14]: arr
```

```
Out[14]: [1, 2, 5, 7, 12, 23, 23]
```

```
In [15]: def fact(n):
             if n==0:
                 return 1
             else:
                 print "Evaluating:",n
                 return n*fact(n-1)
```

```
In [16]: fact(5)
```

```
Evaluating: 5
Evaluating: 4
Evaluating: 3
Evaluating: 2
Evaluating: 1


Out[16]: 120

In [18]: for i in [3,5,12,37]:
             print 2*i

6
10
24
74


In [19]: evens = [ 2*i for i in range(2,10)]

In [20]: evens

Out[20]: [4, 6, 8, 10, 12, 14, 16, 18]

In [21]: 7%3

Out[21]: 1

In [23]: if 7%3==0:
             print "it divides"
         else:
             print "it doesnot"

it doesnot


In [33]: class point():
             def __init__(self,x,y):
                 self.x = x; self.y = y
             def render(self):
                 print "x:",self.x,", y:",self.y
             def length(self):
                 return math.sqrt(self.x*self.x+self.y*self.y)

In [34]: p = point(1,2)

In [35]: p.render()

x: 1 , y: 2
```

```
In [38]: print p.length()

2.2360679775


In [40]: import math

In [41]: string="the quick brown fox jumps"

In [42]: import re

In [45]: print re.search("a",string)

None


In [47]: print "foo"+1




        TypeErrorTraceback (most recent call last)

        <ipython-input-47-f23511c396b2> in <module>()
    ----> 1 print "foo"+1


        TypeError: cannot concatenate 'str' and 'int' objects


In [48]: string.split()

Out[48]: ['the', 'quick', 'brown', 'fox', 'jumps']

In [49]: for word in string.split():
             print word

the
quick
brown
fox
jumps


In [51]: re.sub("o","OH!",string)

Out[51]: 'the quick brOH!wn fOH!x jumps'

In [52]: import os

In [53]: os.listdir(".")
```

```
Out[53]: ['Untitled.ipynb',
          '.ipynb_checkpoints',
          'Untitled2.ipynb',
          'foo.txt',
          'Untitled1.ipynb',
          'Trying out python stuff.ipynb',
          'ispdemo.txt',
          'data']
```

```
In [55]: with open("ispdemo.txt") as file:
             for line in file:
                 line = line.strip()
                 print line
```

```
the quick
brown fox jumps
over the lazy dog
```

```
In [56]: len( ["a","b","c"])
```

```
Out[56]: 3
```

```
In [58]: os.cwd()
```

```
        AttributeErrorTraceback (most recent call last)

        <ipython-input-58-a5cadddc454d> in <module>()
    ----> 1 os.cwd()


        AttributeError: 'module' object has no attribute 'cwd'
```

```
In [60]: for p,d,f in os.walk("."):
             print "At path %s directories: %s files: %s" % (p,str(d),str(f))
             for file in f:
                 if re.search(r'txt$',file):
                     print "textfile: %s" % file
```

```
At path . directories: ['.ipynb_checkpoints', 'data'] files: ['Untitled.ipynb', 'Un
At path ./.ipynb_checkpoints directories: [] files: ['Untitled2-checkpoint.ipynb',
At path ./data directories: ['.ipynb_checkpoints'] files: ['numbers.txt']
At path ./data/.ipynb_checkpoints directories: [] files: []
```

```
In [68]: class segment():
             def __init__(self,lo,hi):
                 self.lo = lo; self.hi = hi
             def __iter__(self):
                 for i in range(self.lo,self.hi):
                     print "I am going to yield %d" % i
                     yield i
                     print "I did it with %d" % i
```

```
In [69]: s = segment(7,12)
```

```
In [70]: for i in s:
             print 3*i
```

```
I am going to yield 7
21
I did it with 7
I am going to yield 8
24
I did it with 8
I am going to yield 9
27
I did it with 9
I am going to yield 10
30
I did it with 10
I am going to yield 11
33
I did it with 11
```

```
In [ ]:
```