

Guided Exercises

Luis Pedro Coelho

Programming for Scientists

September 24, 2012



Goals for this hour



- A quiz
- Do a few exercises.
- Play around.
- You can work alone, in pairs, in triples,...

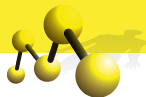
Consider the following code:

```
class Point2(object):  
    def __init__(self, x, y):  
        self.x = x  
        self.y = y  
  
    def dist2(self):  
        return self.x**2 + self.y**2
```

```
p = Point2(2, 2)  
print p.dist2()  
p.y = 0  
print p.dist2()  
print p.x  
print p.y
```

This code prints four numbers. What are they:

- a 8, 8, 2, 0
- b 8, 8, 2, 2
- c 8, 4, 2, 0
- d 8, 4, 2, 2



Write a piece of code that writes the string "Hello World" to a file called "hello".

What happens if you use your code above when a file called hello already exists?

What does the following code do?

```
from random import choice  
print choice(range(30))
```

For this, you might want to read the Python documentation.

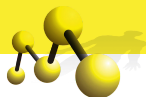
You will sometimes see the following programming idiom:

```
import numpy as np
mystery = np.uint32(-1)
```

Remember:

- uint32 is short for **unsigned integer of 32 bits**.
- Unsigned means that it should be interpreted as a positive number.
- So, mystery cannot have the value -1 !

What is the value of mystery? Why would we be interested in this particular value? (Hint: think of its bit representation).



What is a text file?

- ① A file with an extension TXT
(for example file.txt)
- ② A file whose content can be interpreted as printable characters.
- ③ A Word file.
- ④ A file with text in a human language (like English).

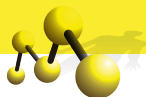
Factorial



Write a factorial function.

$$N! = N \cdot (N - 1) \cdot (N - 2) \cdots 1$$

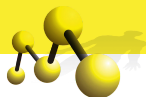
Factorial



Write a factorial function.

$$N! = N \cdot (N - 1) \cdot (N - 2) \cdots 1$$

$$N! = \begin{cases} 1 & N = 0 \\ N \cdot (N - 1)! & \text{ow.} \end{cases}$$

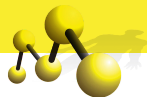


Write a factorial function.

$$N! = N \cdot (N - 1) \cdot (N - 2) \cdots 1$$

$$N! = \begin{cases} 1 & N = 0 \\ N \cdot (N - 1)! & \text{ow.} \end{cases}$$

```
def factorial(n):  
    assert n >= 0, \  
        'factorial is only for n >= 0'  
    if n == 0: return 1  
    return n * factorial(n-1)
```



Write a Fibonacci function

$$F_0 = 1$$

$$F_1 = 1$$

$$F_{n+2} = F_{n+1} + F_n$$