CFG Advanced Python Course - Session 2
Basics of Python, Part 2

Today we will be making use of an online exercise book to get comfortable with a few Python concepts. Treat this session as a lab, where you learn by doing.

Our goal today is to get through as many of these exercises as possible. We will be linking to some external exercises from a book called Learn Python The Hard Way.

Let's review what we learned last week!

Printing
print "Hello, CFG!"

Basic Maths calculations
print 5 + 6
print 5 ** 5

Int and Float type
print 1 / 3
print 1 / 3.0

Variables
my_name = "CFG"

Printing variables
...and how to print them! We have seen multiple ways to do so:
print "Hello, {}".format(my_name)
print "Hello, {0}".format(my_name)
print "Hello, {name}".format(name = my_name)

All these statements are equivalent.

If you remember we discussed about another way to do it:
print "Hello, %s" % my_name

It's not the prefered way to print variables, but we have to tell you about it because the exercises in Learn Python The Hard Way use this syntax.

You could easily translate one to the other:
print 'Hello %s' % my_name
Is equivalent to:
print 'Hello {}'.format(my_name)

You will not always encounter %s but also %d, %r ...

It's no-brainer to translate it:
%s => {}
%r => {}
%d => {}

Of course these letters after the percentage sign mean something, it's a way to convert the variable from a specific type:
"s" for string
"d" for decimal
etc...

More on the official Python doc.

Task: Asking Questions

Let's move on to writing some code that takes some input and does something with it! Work your way through the following exercises:

- Asking Questions
    - Work through the examples, study drills & have a look at the common student questions
- Prompting People
    - Work through the examples but skip study drills & common student questions
- Prompting and Passing

Task: Variables & Functions

Let's focus on functions; what are they, how do they work and what can you use them for.

- Names, Variables, Code, Functions
    - Work through the examples but skip study drills & common student questions
- Functions and Variables
    - Work through the examples but skip study drills & common student questions
- Functions can return something
    - Work through the examples, study drills & have a look at the common student questions

Task: Logic

In Python we have the following terms (characters and phrases) for determining if something is "True" or "False." Logic on a computer is all about seeing if some combination of these characters and some variables is True at that point in the program.

- and
- or
- not

- != (not equal)
- == (equal)
- >= (greater-than-equal)
- <= (less-than-equal)
- True
- False

1. Pair up with someone
2. Create a new Python file on one of your laptops
3. For each of the following Python expressions, try to agree what value you think they evaluate to. Check if you are right by printing the result of each expression from within your Python file:
    - not False
    - not True

    - True or False
    - True or True
    - False or True
    - False or False

    - True and False
    - True and True
    - False and True
    - False and False

    - not (True or False)
    - not (True or True)
    - not (False or True)
    - not (False or False)

    - not (True and False)
    - not (True and True)
    - not (False and True)
    - not (False and False)

    - 1 != 0
    - 1 != 1
    - 0 != 1
    - 0 != 0

    - 1 == 0
    - 1 == 1
    - 0 == 1
    - 0 == 0

    - 21 >= 42

- 21 <= 42
- 42 > 42
- 42 >= 42

- "test" == "test"
- "test" != "test
- "test" == "random"
- "test" != "random"

You can do some more Boolean practice if you'd like to now!

Task: Decision-making (if, else, elif)

- What If
  - Work through the examples but skip study drills & common student questions
- Else and If
  - Work through the examples, study drills & have a look at the common student questions
- Making Decisions
  - Work through the examples, study drills & have a look at the common student questions

Task: Looping, lists and accessing lists

- Loops and Lists
  - Work through the examples, study drills & have a look at the common student questions
- Accessing Elements of Lists
  - Work through the examples but skip study drills

Programming Challenge

If you finish Exercise 34 and want to give yourself a bit of a challenge applying what you've learnt so far, send us an e-mail at **twitter-uk-cfg@twitter.com** and let us know you are ready for the challenge. We will send you a problem that you should try and solve on your own.