Data Bootcamp: Python Fundamentals 1 Practice

Revised: February 13, 2017

1. Construct a dictionary whose keys are the integers 1, 2 and 3 and whose values are the same numbers as words: one, two, three. How would you get the word associated with the key 2?

2. What does the built-in function `zip()` do? Start with the lists

   ```
   l1 = [1, 2, 3]
   l2 = ['one', 'two', 'three']
   ```

   - Construct the list of tuples `[(1, 'one'), (2, 'two'), (3, 'three')]`.
   - Construct a dictionary whose keys are the integers 1, 2 and 3 and whose values are the same numbers as words: one, two, three.

3. Take the variables `name1` and `name2`, both of them strings. Write a program using `if ... else` that prints the name that comes first in alphabetical order. Test your program with

   ```
   name1 = 'Dave'
   name2 = 'Glenn'
   ```

   Could you write a function for this task?

4. Take the list `stuff = ['cat', 3.7, 5, 'dog']`.
   - Write a program that tells us the type of each elements of `stuff` (value and type)
   - Write a program that prints only the elements of `stuff` that are strings.

5. Write a program that given two integer values `a` and `b` returns `True` if either one is 6, or if their sum or difference is 6. *Hint:* you might want to use the built-in function `abs()`. Test your program with
   - `a=6`, `b=4` (should get `True`)
   - `a=4`, `b=5` (should get `False`)
   - `a=1`, `b=5` (should get `True`)

   Could you write a function for this task?

6. Write a program that given two integers `a` and `b` returns their sum. However, sums in the range $[10, 19]$ are forbidden, so in that case just return 20. Test your program with
   - `a=3`, `b=4` (should get 7)
   - `a=9`, `b=4` (should get 20)
   - `a=10`, `b=11` (should get 21)

Could you write a function for this task?

7. Take the list of bond yields `y = [.01, .02, .03]` for maturities of one, two and three years. Write a program that multiplies all elements with 100, i.e. returns the list

   ```
   y_perc = [1.0, 2.0, 3.0]
   ```

   Can you do it with list comprehension? And with a function?

8. Using only core `Python` functionality define a list that contains
   - all integers from 0 to `n` (test it with the example: `n=5`)
   - all integers from `n1` to `n2` (test it with the example: `n1=4` and `n2=8`)

   Could you define a function that can do both? (*Hint:* use default values)

---

9. Suppose that you have two strings

   ```
   s1 = "abc def ghi"
   s2 = "def ghi abc"
   ```

   - How would you check if the two strings contain the same characters? (irrespective of their order)
   - How would you check if the two strings contain the same words?

10. Take the list `x = [1, 4, 5, 6, 4, 2, 7, 12, 4, 23, 4]`. Using only loops and/or conditionals create a list with all the indices `i` for which `x[i] == 4`.