In [1]:

```
"""
1a. Modify the code below to count 'down' in increments of -1 (5 points)
"""
for word in reversed(range(0, 100)):
    print (word)
```

```
99
98
97
96
95
94
93
92
91
90
89
88
87
86
85
84
83
82
81
80
79
78
77
76
75
74
73
72
71
70
69
68
67
66
65
64
63
62
61
60
59
58
57
56
55
54
53
52
51
50
49
48
```

```
47
46
45
44
43
42
41
40
39
38
37
36
35
34
33
32
31
30
29
28
27
26
25
24
23
22
21
20
19
18
17
16
15
14
13
12
11
10
9
8
7
6
5
4
3
2
1
0
```

In [2]:

```
"""
1b. Change the code above to count down in increments of -0.5 (5 points)
"""
import numpy as np

for word in np.arange(99, -0.5, -0.5):
    print (word)
```

```
99.0
98.5
```

98.0
97.5
97.0
96.5
96.0
95.5
95.0
94.5
94.0
93.5
93.0
92.5
92.0
91.5
91.0
90.5
90.0
89.5
89.0
88.5
88.0
87.5
87.0
86.5
86.0
85.5
85.0
84.5
84.0
83.5
83.0
82.5
82.0
81.5
81.0
80.5
80.0
79.5
79.0
78.5
78.0
77.5
77.0
76.5
76.0
75.5
75.0
74.5
74.0
73.5
73.0
72.5
72.0
71.5
71.0
70.5
70.0
69.5
69.0
68.5

68.0
67.5
67.0
66.5
66.0
65.5
65.0
64.5
64.0
63.5
63.0
62.5
62.0
61.5
61.0
60.5
60.0
59.5
59.0
58.5
58.0
57.5
57.0
56.5
56.0
55.5
55.0
54.5
54.0
53.5
53.0
52.5
52.0
51.5
51.0
50.5
50.0
49.5
49.0
48.5
48.0
47.5
47.0
46.5
46.0
45.5
45.0
44.5
44.0
43.5
43.0
42.5
42.0
41.5
41.0
40.5
40.0
39.5
39.0
38.5

38.0
37.5
37.0
36.5
36.0
35.5
35.0
34.5
34.0
33.5
33.0
32.5
32.0
31.5
31.0
30.5
30.0
29.5
29.0
28.5
28.0
27.5
27.0
26.5
26.0
25.5
25.0
24.5
24.0
23.5
23.0
22.5
22.0
21.5
21.0
20.5
20.0
19.5
19.0
18.5
18.0
17.5
17.0
16.5
16.0
15.5
15.0
14.5
14.0
13.5
13.0
12.5
12.0
11.5
11.0
10.5
10.0
9.5
9.0
8.5

```
8.0
7.5
7.0
6.5
6.0
5.5
5.0
4.5
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0
```

In [3]:

```
"""
2. Using list comprehension (HINT: []) populate an array whose individual word
character lengths are >= 4 (10 points)
"""
the_sentence = 'the student went above and beyond the call of duty on the homewo

my_ar = list()
#for word in the_sentence.split():
#    if len(word) >= 4:
#        my_ar.append(word)

my_ar = [word for word in the_sentence.split() if len(word) >= 4]
my_ar
```

Out[3]:

```
['student', 'went', 'above', 'beyond', 'call', 'duty', 'homework']
```

In [4]:

```
"""
3. Fix the errors in the code below and replace the word dog with turtle (10 poi
"""
import re
the_sentence = 'the orange cat jumped over the dog, but the orange cat landed on

my_ar = re.sub('dog', 'turtle', the_sentence)

my_ar
```

Out[4]:

```
'the orange cat jumped over the turtle, but the orange cat landed on another ora
nge cat'
```

In [5]:

```
"""
4. Remove all special characters from the sentence except for @ (10 points)
"""
the_sentence = 'woah!!! the @student really^ #impressed me, and& so did ??you!!!

the_sentence = re.sub('[^A-Za-z@!]+', " ", the_sentence)
the_sentence
```

Out[5]:

```
'woah!!! the @student really impressed me and so did you!!!'
```

In [6]:
```
"""
5. create a sentence, each word seperated by one space, out of the following arr
and replace any special characters with a no space, except '!', with '' (10 poin
"""

the_ar = ['woah!!!','the','@student',' really^','# impressed','me,','and&','so',

sentence = " ".join(the_ar)
sentence = re.sub('[^A-Za-z!]+', " ", sentence)
sentence
```

Out[6]:
```
'woah!!! the student really impressed me and so did you!!!'
```

In [7]:
```
"""
6. Create a program that loops 10 times and for each loop generate a random inte
and print 'even' if the number is even and 'odd' if the number is odd (HINT: imp
if the number is odd (10 points)
"""
import random

for x in range(1, 11):
    int = random.randint(0, 10)
    if int % 2 == 0:
        print(str(int) + ' even')
    else:
        print(str(int) + ' odd')
```

```
4 even
5 odd
9 odd
3 odd
8 even
5 odd
8 even
9 odd
1 odd
7 odd
```

In [8]:
```
"""
7. Create a program that counts the length of each word in an arbitrary sentence
each word and count pair in a pandas dataframe where one column is the word and
word length (number of characters) (10 points)
"""
import pandas as pd

def word_length(sentence):
    df = pd.DataFrame(columns = ["word", "length"])
    for word in sentence.split():
        row = pd.DataFrame({"word": [word], "length":[len(word)]})
        df = pd.concat([df, row], ignore_index = True)
    return df

my_sentence = "Men always remember love because of romance only"
word_length(my_sentence)
```

Out[8]:

	word	length

	word	length
0	Men	3
1	always	6
2	remember	8
3	love	4
4	because	7
5	of	2
6	romance	7
7	only	4

In [9]:
```
"""
8. Replace the word cat with mouse (10 points)
"""
the_sentence = 'the cat jumped over the dog, but the cat landed on another cat!'

import regex as re
re.sub('cat', 'mouse', the_sentence)
```

Out[9]:
```
'the mouse jumped over the dog, but the mouse landed on another mouse!'
```

In [10]:
```
"""
9. Cleanse the following sentence by removing all special characters
except when the hyphen joins to two words and exclamation points (10 points)
"""
the_sentence = 'The impact*of data-driven$^%&marketing approaches!!'

re.sub("[^A-Za-z!/-]+", " ", the_sentence)
```

Out[10]:
```
'The impact of data-driven marketing approaches!!'
```

In [11]:
```
"""
10. Write a python program that inputs the sentence below and returns a dictiona
that has each unique character as a key and count of that character as the value
(10 points)
"""
the_sentence = 'Everything flows, and nothing abides, everything gives way, and

def word_dict(sentence):
    the_dict = dict()
    character = ([*sentence])
    for x in set(character):
        the_dict[x] = character.count(x)
    return the_dict

word_dict(the_sentence)
```

Out[11]:
```
{'x': 1,
 'y': 4,
 'b': 1,
 'w': 2,
```

```
's': 5,
'l': 1,
'E': 1,
'v': 3,
'a': 5,
'g': 5,
'f': 2,
't': 5,
'i': 7,
'd': 4,
',': 3,
'r': 2,
'n': 8,
'h': 4,
' ': 11,
'o': 3,
'e': 6}
```