Lesson 6

Revisiting lists

- List can contain any type of elements, for example:
    - list_of_ints = [1,2,3,4]
    - list_of_floats = [1.0,5.5,3.4]
    - list_of_strs = ['Hello', 'Hi', 'Howdyy']
    - mixed_list = ['Hello', 1, 1.5]

Iterating over lists/strings

- There are several ways to iterate over elements in the list, those are a common 3 ways:

In [1]:

```
## first way
number_list = [1,2,3,4]

for element in number_list:
    print(element)
```

```
1
2
3
4
```

In [3]:

```
number_list = [1,2,3,4]
index = 0
while index < len(number_list):
    print(number_list[index])
    index += 1
```

```
1
2
3
4
```

In [2]:

```
for element in range(6):
    print(element)
```

```
0
1
2
3
4
5
```

In [3]:

```
number_list = ['hello','hi','check','world']
size = len(number_list)
for index in range(size):
    print('index', index)
    print('element',number_list[index])
```

```
('index', 0)
('element', 'hello')
('index', 1)
('element', 'hi')
('index', 2)
('element', 'check')
('index', 3)
('element', 'world')
```

In [5]:

```
help(range)
```

```
Help on built-in function range in module __builtin__:

range(...)
    range(stop) -> list of integers
    range(start, stop[, step]) -> list of integers

    Return a list containing an arithmetic progression of integers.
    range(i, j) returns [i, i+1, i+2, ..., j-1]; start (!) defaults
to 0.
    When step is given, it specifies the increment (or decrement).
    For example, range(4) returns [0, 1, 2, 3].  The end point is om
itted!
    These are exactly the valid indices for a list of 4 elements.
```

In [7]:

```
for index in range(10,1,-2):
    print(index)
```

```
10
8
6
4
2
```

Exercise 1

Write a function that takes two lists as inputs and returns a list that sums the two input lists (element wise). Assume two input lists are of the same size Example:

- List 1: [1,2,3] - List 2 = [5,6,7], output = [6, 8, 10] so it sums 1st element from 1st list with 1st element and 2nd element with 2nd element, and so on

In [1]:

```
### simple version
list1 = [1,2,3]
list2 = [5,6,7]

element1 = list1[0] + list2[0]
element2 = list1[1] + list2[1]
element3 = list1[2] + list2[2]

print(element1, element2, element3)
```

```
(6, 8, 10)
```

In [2]:

```
### program version
list1 = [1,2,3]
list2 = [5,6,7]

result_list = []

index = 0

while index < len(list1):
    sum_element = list1[index]+ list2[index]
    result_list = result_list + [sum_element]
    index = index + 1

print(result_list)
```

```
[6, 8, 10]
```

In [3]:

```
### final solution
def sum_elements(list1, list2):
    result_list = []
    index = 0
    while index < len(list1):
        sum_element = list1[index]+ list2[index]
        result_list = result_list + [sum_element]
        index = index + 1
    return result_list

x = [1,2,3,4]
y = [5,6,7,8]

output = sum_elements(x,y)
print 'Output' , output
```

```
Output [6, 8, 10, 12]
```

Nested lists

- List can contain as elements a list, we call this **nested list**

In [ ]:

```
list_of_lists = [
                    ['Assignment 1', 5],
                    ['Assignment 2', 6],
                    ['Assignment 3', 11]
                ]
```

In [ ]:

```

```

In [40]:

```
list_of_lists = [
                    ['hello', 'check', 'blah'],
                    [100,500000],
                    [1.5, 3.5, 'mmm']
                ]
print(list_of_lists)
```

```
[['hello', 'check', 'blah'], [100, 500000], [1.5, 3.5, 'mmm']]
```

Accessing Nested list elements

In [44]:

```
list_of_lists[0]
```

Out[44]:

```
['hello', 'check', 'blah']
```

In [45]:

```
list_of_lists[0][0]
```

Out[45]:

```
'hello'
```

In [48]:

```
list_of_lists[2][2]
```

Out[48]:

```
'mmm'
```

In [33]:

```
len(list_of_lists)
```

Out[33]:

```
3
```

In [35]:

```
x = list_of_lists[0]
x
```

Out[35]:

```
['Assignment 1', 5]
```

In [36]:

```
x[0]
```

Out[36]:

```
'Assignment 1'
```

In [39]:

```
list_of_lists[0][0]
```

Out[39]:

```
'Assignment 1'
```

In [22]:

```
len(list_of_lists[0])
```

Out[22]:

```
2
```

Exercise 2

Given a list of soccer match scores for team x , where each element in the list is a two element list. The first element is the score of team x and the second element is the score of the other team.

Write a function to calculate the overall score of the team (win is 3 points, draw is 1 point and lose is 0 points)

Example

match_scores = [[3,2], [4,1], [1,1], [0,1], [0,2]]

overall score = 3+3+1 = 7

Hint We will use the function we did last lesson

```
In [24]:
```

```
def calculate_score(score_list):
    total_score = 0
    for game in score_list:
        if game == 'W':
            total_score = total_score +3
        elif game == 'D':
            total_score = total_score + 1
    return total_score

man_united_results = ['W', 'D', 'W', 'L', 'W', 'D', 'D', 'D']
score = calculate_score(man_united_results)
```

Exercise 3

Write a function that given a list of student grades , each element is a list of size 2, where first element is the assignment name and the second element is the grade, calculate the average grade for student

Example:

```
grades_list = [['Assignment 1', 5], ['Assignment 2', 10], ['Assignment 3',
 15], ['Assignment 4', 5] ]
output = 8.75
```

Exercise 4

write a function that given a list of lists , returns the maximum number, Assume all numbers in the list are positive. Do not use **max** function.

Example:

```
numbers_list = [[1,2,3], [5,6,10,1], [4,3,2]]
output = 10
```

Exercise 5

write a program that given a list of lists , outputs the sum of numbers in the list. Do not use **sum** function

Example:

```
numbers_list = [[1,2,3], [5,6,10,1], [4,3,2]]
output = 37
```