Lists in Python

- the basics -

Nick Szirbik

Sept. 2020

Week 2 of the DAPOM course (2)

First: Data structures

- All advanced programming languages offer various **built-in** data structures or **constructs** that allow their development by the programmer

- Typical data structures are: **indexed arrays** of various dimensionality, labelled records, linked lists, trees, heaps, hash tables, graphs, networks

- Python has four built-in data structures (generically named **collections**): tuples, lists, sets, and dictionaries.

- We recommend to study them using tutorials from **`w3Schools.com`**

But before collections: the **string** data type

- To some extent, a string is a data structure,

because it is in fact an **array** of unicode characters

- It can be use as **indexed**, like an array

(or mathematical vector). For example:


```
digits = str('1234567890ABCDEF')
print(digits)
whatDigit = digits[0]
andThis = digits[9]
for count, digit in enumerate(digits):
    print("the", count, "nth digit in the string is", digit)

# more at https://www.w3schools.com/python/python_strings.asp
```

```
1234567890ABCDEF
the 0 nth digit in the string is 1
the 1 nth digit in the string is 2
the 2 nth digit in the string is 3
the 3 nth digit in the string is 4
the 4 nth digit in the string is 5
the 5 nth digit in the string is 6
the 6 nth digit in the string is 7
the 7 nth digit in the string is 8
the 8 nth digit in the string is 9
the 9 nth digit in the string is 0
the 10 nth digit in the string is A
the 11 nth digit in the string is B
the 12 nth digit in the string is C
the 13 nth digit in the string is D
the 14 nth digit in the string is E
the 15 nth digit in the string is F
```

A list is a collection which is ordered and changeable.
In Python lists are written with square brackets.

- An example of a list of strings only:

```
fruit_list =
["apple", "banana", "cherry", "orange", "kiwi", "melon", "mango"]
print(fruit_list[2:5]) # display only from the third to the sixth element
```


- How to check the length of a list – use the len() function:

```
print(len(fruit_list))
```


- Insert an item (in the second position) – use the insert() function

```
fruit_list.insert(1, "pear")
print(fruit_list)
```

- Delete an item – use the remove() function:

```
fruit_list.remove("banana“)
print(fruit_list)
```

Lists – very important for the next practical

- They can be **nested** also (lists of lists is possible!)

https://www.w3schools.com/python/python_lists.asp

To remember:

1. The most important method for you: **append()**

2. Copying lists (and other collections) is done always with **copy()**


```
# some code example with a matrix (as list of lists)
###
matrix = list(([[1,2,3],[4,5,6],[7,8,9]]))
total_of_matrix_elements = sum([sum (row) for row in
matrix])
print(total_of_matrix_elements)
```