Functions and Methods of Sequential Data Types

Prof.Dr. Bahadır AKTUĞ
BME362 Introduction To Python

**Compiled from sources given in the references.*

Functions for Sequential Data Types

- There are various functions that could be used for sequential data types including the "lists".
- In this respect, the functions in this course are equally applicable to other sequential data types as well.
- We have already seen that any element of a list can be accessed and modified through indexing.
- Since lists are, in fact, objects (like everything else in Python), they have object functions (methods) which directly operate on the list or sequential data type functions which take list as arguments.

Sequential Data Type Functions

- Sequential data type functions are those which take sequential data types as arguments.
- Since these are functions, they "return values".
- For the case of "lists", several examples are given below.

Function	Description
len(list)	returns the total number of elements
max(list)	returns the largest element
min(list)	returns the smallest element
list(tuple/string)	converts a tuple to a list and return that list

Methods (object functions)

- Methods are functions of the specific object and does not require the object itself as an argument.

Function	Description
list.append(obj)	appends another object to the list
list.count(obj)	returns the number of a specific element in the sequential data type
list.extend(seq)	extends the sequential data type with another sequential data type given as "seq"
list.index(obj)	return the order/index of a specific element in a sequential data type
list.insert(index, obj)	inserts an element into a specific position in the sequential data type
list.pop()	returns the last element and removes it from the sequential data type
list.remove(obj)	removes a specific element ("obj") from the sequential data type

max(list)/min(list)/list(tuple)

```
>>> c = ["Ankara","Izmir","Istanbul","Zonguldak"]
>>> max(c)
'Zonguldak'
>>> min(c)
'Ankara'

>>> a = [ 1, 2, 3]
>>> max(a)
3

>>> list(min(c))
['A', 'n', 'k', 'a', 'r', 'a']
>>> list((5,8,2))
[5, 8, 2]
```

del/append/count

```
>>> c = ['Ankara', 'Izmir', 'Istanbul', 'Zonguldak']
>>> del c[0]
>>> c
['Izmir', 'Istanbul', 'Zonguldak']

>>> c.append("Bursa")
>>> c
['Izmir', 'Istanbul', 'Zonguldak', 'Bursa']
>>> c.append("Bursa")
>>> c
['Izmir', 'Istanbul', 'Zonguldak', 'Bursa', 'Bursa']
>>> c.count("Bursa")
2
>>> c.count("Ankara")
0
>>> c.count("Zonguldak")
```

extend/sort/reverse

```
>>> c
['Izmir', 'Istanbul', 'Zonguldak', 'Bursa', 'Bursa']
>>> d = [ 'Samsun','Erzurum']
>>> c.extend(d)
>>> c
['Izmir', 'Istanbul', 'Zonguldak', 'Bursa', 'Bursa', 'Samsun', 'Erzurum']

>>> c.sort()
>>> c
['Bursa', 'Bursa', 'Erzurum', 'Istanbul', 'Izmir', 'Samsun', 'Zonguldak']
>>> c.reverse()
>>> c
['Zonguldak', 'Samsun', 'Izmir', 'Istanbul', 'Erzurum', 'Bursa', 'Bursa']
>>> c.sort()
>>> c
['Bursa', 'Bursa', 'Erzurum', 'Istanbul', 'Izmir', 'Samsun', 'Zonguldak']
>>> c.sort(reverse=True)
>>> c
['Zonguldak', 'Samsun', 'Izmir', 'Istanbul', 'Erzurum', 'Bursa', 'Bursa']
```

References

1 Wentworth, P., Elkner, J., Downey, A.B., Meyers, C. (2014). How to Think Like a Computer Scientist: Learning with Python (3nd edition).
2 Pilgrim, M. (2014). Dive into Python 3 by. Free online version: DiveIntoPython3.org ISBN: 978-1430224150.
3 Summerfield, M. (2014) Programming in Python 3 2nd ed (PIP3) : - Addison Wesley ISBN: 0-321-68056-1.
4 Summerfield, M. (2014) Programming in Python 3 2nd ed (PIP3) : - Addison Wesley ISBN: 0-321-68056-1.
5 Jones E, Oliphant E, Peterson P, et al. SciPy: Open Source Scientific Tools for Python, 2001-, http://www.scipy.org/.
6 Millman, K.J., Aivazis, M. (2011). Python for Scientists and Engineers, Computing in Science & Engineering, 13, 9-12.
7 John D. Hunter (2007). Matplotlib: A 2D Graphics Environment, Computing in Science & Engineering, 9, 90-95.
8 Travis E. Oliphant (2007). Python for Scientific Computing, Computing in Science & Engineering, 9, 10-20.
9 Goodrich, M.T., Tamassia, R., Goldwasser, M.H. (2013). Data Structures and Algorithms in Python, Wiley.
10 http://www.diveintopython.net/
11 https://docs.python.org/3/tutorial/
12 http://www.python-course.eu
13 https://developers.google.com/edu/python/
14 http://learnpythonthehardway.org/book/