Python Lists

```
In [2]:  # Ordered Collection
         # Indexed
         # Mutable
         # Duplicates Elements Allow
         # Compound Data Type
         # Reference Type

In [3]:  values = [[3, 6, 7], [2, 4, 6]]

In [4]:  value = values[0]

In [5]:  value
Out[5]:  [3, 6, 7]

In [ ]:

In [6]:  import keyword

In [7]:  keyword.iskeyword('lambda')
Out[7]:  True

In [ ]:

In [8]:  values = [[3, 6, 7], [2, 4, 6]]  # 2D List or Matrix

In [9]:  sorted(values, key=lambda value: value[2])
Out[9]:  [[2, 4, 6], [3, 6, 7]]

In [ ]:

In [10]: users = ["usman", "hassan", "ali"]

In [11]: users.sort(key=lambda user: len(user), reverse=True)  # 6, 5, 3

In [12]: users
Out[12]: ['hassan', 'usman', 'ali']

In [ ]:

In [54]: # ord('a')

In [55]: # chr(65)

In [ ]:

In [13]: # 3D List

In [14]: data = [
             [
               [2, 5, 5],
               [1, 6, 7]
             ],
             [
               [1, 5, 1],
               [4, 6, 4]
             ]
         ]

In [15]: data[-1][0][-1] = 50  # update

In [16]: data[1]
Out[16]: [[1, 5, 50], [4, 6, 4]]

In [ ]:

In [17]: x = [3, 6, 8]

In [18]: x + [3]
Out[18]: [3, 6, 8, 3]

In [19]: x * 3
Out[19]: [3, 6, 8, 3, 6, 8, 3, 6, 8]

In [20]: x
Out[20]: [3, 6, 8]

In [21]: 3 in x
Out[21]: True

In [ ]:

In [22]: x = [2, 5, 7, 8, 6, 7, 1]

In [23]: # list.remove()
         # list.pop()
         # list.clear()
         # del

In [24]: # del x

In [25]: # x

In [26]: del x[:]

In [29]: # del x[0::2]

In [30]: # x

In [31]: # x.clear()

In [33]: # x.pop(2)

In [34]: # x.remove(7)

In [35]: # x

In [ ]:
```

Python Array, Range, Tuple, Dict, Set, Frozenset

Python Array

```
In [36]: from array import array

In [37]: type(array)
Out[37]: type

In [39]: # array?

In [40]: data_1 = array('b', [2, 5, 7, 3])  # -128 to 127

In [41]: x = data_1

In [42]: x[1] = 50

In [43]: x
Out[43]: array('b', [2, 50, 7, 3])

In [44]: data_1
Out[44]: array('b', [2, 50, 7, 3])

In [ ]:

In [46]: data_2 = array('b', [2, 5, 7, 3])

In [47]: data_2.append(2)

In [48]: data_2
Out[48]: array('b', [2, 5, 7, 3, 2])

In [ ]:
```

Python Range

```
In [49]: x = range(5)

In [50]: set(x)
Out[50]: {0, 1, 2, 3, 4}

In [51]: list(x)
Out[51]: [0, 1, 2, 3, 4]

In [52]: tuple(x)
Out[52]: (0, 1, 2, 3, 4)

In [53]: x = range(1, 11, 2)

In [54]: list(x)
Out[54]: [1, 3, 5, 7, 9]

In [55]: x = range(-10, -5)

In [56]: list(x)
Out[56]: [-10, -9, -8, -7, -6]

In [193…  # 10 to 1

In [58]: list(range(10, 0, -1))
Out[58]: [10, 9, 8, 7, 6, 5, 4, 3, 2, 1]

In [ ]:
```

Python Tuple

```
In [59]: t1 = (4,)

In [60]: type(t1)
Out[60]: tuple

In [61]: t2 = tuple((2, 4, 5))

In [62]: type(t2)
Out[62]: tuple

In [63]: t3 = 3, 5, 6   # tuple packing

In [64]: a, b, c = t3   # tuple unpacking

In [65]: *a, b = t3

In [66]: a
Out[66]: [3, 5]

In [67]: b
Out[67]: 6

In [ ]:

In [68]: t4 = (2, 5, 7)

In [69]: # Indexing
         # Slicing

In [70]: # t4[-1] = 50   # Error

In [71]: # Remove X
         # Update X
         # Append X

In [72]: # index()
         # count()

In [73]: t5 = (3, 6, 7)

In [74]: t5 + (2, 6)
Out[74]: (3, 6, 7, 2, 6)

In [75]: t5 * 2
Out[75]: (3, 6, 7, 3, 6, 7)

In [76]: t5
Out[76]: (3, 6, 7)

In [77]: 6 in t5
Out[77]: True

In [ ]:

In [78]: t6 = ((2, 4), (2, 8), (2, 9))

In [79]: a, b = t6[-1]

In [80]: # del t6[0]

In [81]: t7 = (3, 5.7, "xyz", [5, 8, 9])

In [82]: t7
Out[82]: (3, 5.7, 'xyz', [5, 8, 9])

In [83]: t7[-1].append(10)

In [84]: t7
Out[84]: (3, 5.7, 'xyz', [5, 8, 9, 10])

In [85]: t7[2].upper()
Out[85]: 'XYZ'

In [86]: t7
Out[86]: (3, 5.7, 'xyz', [5, 8, 9, 10])

In [87]: t7[-1].sort(reverse=True)

In [88]: t7
Out[88]: (3, 5.7, 'xyz', [10, 9, 8, 5])

In [ ]:

In [89]: t1 = (2, 5, 8)

In [90]: # t2 = t1

In [ ]:

In [91]: t8 = (2, 7, 1, 2, 8)

In [92]: t8.index(7)
Out[92]: 1

In [ ]:

In [93]: # print(3, 4, 6) #

In [94]: # str.format?

In [95]: # "value 1: {}, value 2: {}".format(5)

In [ ]:

In [96]: t9 = (1, 6, 8, 9)

In [97]: # t9[5]

In [ ]:

In [98]: t = 3,

In [99]: type(t)
Out[99]: tuple

In [ ]:
```

Python Sets

```
In [100…  s1 = set()

In [101…  type(s1)
Out[101…  set

In [102…  s2 = {5}

In [103…  type(s2)
Out[103…  set

In [104…  s3 = {2, 6, 7, 8, 1, 6}

In [105…  print(s3)
          {1, 2, 6, 7, 8}

In [ ]:

In [106…  users = ["ali", "usman", "ali"]

In [107…  list(set(users))
Out[107…  ['usman', 'ali']

In [108…  data_users = {"ali", "usman", "ali"}

In [109…  data_users.add("anas")

In [110…  data_users
Out[110…  {'ali', 'anas', 'usman'}

In [111…  # data_users[0]

In [112…  new_users = list(data_users)

In [113…  new_users[-1] = "xyz"

In [ ]:

In [114…  # union        |
          # intersection &
          # difference   -
          # s.d          ^

In [115…  s1 = {3, 7, 1, 5, 2}

In [116…  s2 = {8, 4, 3, 7, 6}

In [117…  s1 - s2
Out[117…  {1, 2, 5}

In [118…  s1.difference(s2)
Out[118…  {1, 2, 5}

In [ ]:

In [119…  s1 ^ s2
Out[119…  {1, 2, 4, 5, 6, 8}

In [120…  s1.symmetric_difference(s2)
Out[120…  {1, 2, 4, 5, 6, 8}

In [ ]:

In [121…  s1.intersection(s2)
Out[121…  {3, 7}

In [122…  s1 & s2
Out[122…  {3, 7}

In [ ]:

In [123…  s1 | s2
Out[123…  {1, 2, 3, 4, 5, 6, 7, 8}

In [124…  s1.union(s2)
Out[124…  {1, 2, 3, 4, 5, 6, 7, 8}

In [125…  s2.union(s1)
Out[125…  {1, 2, 3, 4, 5, 6, 7, 8}

In [ ]:

In [126…  s1 = {2, 6, 7}

In [127…  s1.update({1, 7, 9})

In [128…  s1
Out[128…  {1, 2, 6, 7, 9}

In [ ]:

In [130…  s1.pop()
Out[130…  1

In [131…  s1
Out[131…  {2, 6, 7, 9}

In [ ]:

In [132…  s1 = {2, 6, 7, 9}

In [133…  s1
Out[133…  {2, 6, 7, 9}

In [136…  # s1.remove(10)

In [137…  # s1

In [138…  # s1 * 2

In [139…  # {2, 5} * {1, 4}  # Error

In [ ]:
```

Python Frozensets

```
In [140…  fs1 = frozenset({3, 7, 8, 7, 1, 9})

In [141…  fs1
Out[141…  frozenset({1, 3, 7, 8, 9})

In [142…  fs1.union({3, 2, 4})
Out[142…  frozenset({1, 2, 3, 4, 7, 8, 9})

In [ ]:
```

Happy Learning :)