Section 5 - List Objects

1. List Object Basics

In [1]:
```
x = [ 1, 2, 3, 4, 5]
x
```

Out[1]:
```
[1, 2, 3, 4, 5]
```

In [2]:
```
type(x)
```

Out[2]:
```
list
```

In [3]:
```
len(x)
```

Out[3]:
```
5
```

In [4]:
```
x[0]
```

Out[4]:
```
1
```

In [5]:
```
x[4]
```

Out[5]:
```
5
```

In [6]:
```
x[10]
```

```
---------------------------------------------------------------------
---------
IndexError                                Traceback (most recent c
all last)
<ipython-input-6-7644c71c757c> in <module>()
----> 1 x[10]

IndexError: list index out of range
```

In [7]:
```
x = x + x
x
```

Out[7]:
```
[1, 2, 3, 4, 5, 1, 2, 3, 4, 5]
```

In [8]:
```
len(x)
```

Out[8]:
```
10
```

In [9]:
```
x[-1]
```

Out[9]: 5

In [10]:
```
x[-4]
```

Out[10]: 2

In [11]:
```
x
```

Out[11]: [1, 2, 3, 4, 5, 1, 2, 3, 4, 5]

In [12]:
```
x[0] = 10
```

In [13]:
```
x
```

Out[13]: [10, 2, 3, 4, 5, 1, 2, 3, 4, 5]

In [14]:
```
x
```

Out[14]: [10, 2, 3, 4, 5, 1, 2, 3, 4, 5]

In [15]:
```
x[0:4]
```

Out[15]: [10, 2, 3, 4]

In [16]:
```
x[4:8]
```

Out[16]: [5, 1, 2, 3]

In [17]:
```
y = x[4:8]
```

In [18]:
```
type(y)
```

Out[18]: list

In [19]:
```
y
```

Out[19]: [5, 1, 2, 3]

In [20]:
```
a = [ 1, 2, 3, 4]
```

In [21]:
```
a = a + [ a[3] + 1 ]
a
```

Out[21]: [1, 2, 3, 4, 5]

In [22]:
```
a = []
```

In [23]:
```
a
```

Out[23]:
```
[]
```

In [24]:
```
type(a), len(a)
```

Out[24]:
```
(list, 0)
```

In [25]:
```
x = [1, 2, 3, 4 ]
a, b, c, d = x
```

In [26]:
```
a, b, c, d
```

Out[26]:
```
(1, 2, 3, 4)
```

In [27]:
```
a
```

Out[27]:
```
1
```

In [28]:
```
# range( start, stop, step )
```

In [31]:
```
x = range(10)  # start = 0, step = 1
x
x1 = list(x)
x1
```

Out[31]:
```
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
```

In [32]:
```
x = range(5, 20 )  # step = 1
x
x1 = list(x)
x1
```

Out[32]:
```
[5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19]
```

In [33]:
```
x = range(5, 20, 3 )
x
x1 = list(x)
x1
```

Out[33]:
```
[5, 8, 11, 14, 17]
```

2. List Methods

In [45]:
```
x = [ 1, 2, 3, 4 ]
x
```

Out[45]:
```
[1, 2, 3, 4]
```

In [47]:
```
x.append(5)
x
```

Out[47]: [1, 2, 3, 4, 5]

In [48]:
```
x.insert(3, 33 )
x
```

Out[48]: [1, 2, 3, 33, 4, 5]

In [49]:
```
x.remove(33)
x
```

Out[49]: [1, 2, 3, 4, 5]

In [50]:
```
x.append(5)
```

In [51]:
```
x
```

Out[51]: [1, 2, 3, 4, 5, 5]

In [52]:
```
x.remove(5)
x
```

Out[52]: [1, 2, 3, 4, 5]

In [54]:
```
x.pop(0)
x
```

Out[54]: [2, 3, 4, 5]

In [55]:
```
x = x + x
```

In [56]:
```
x
```

Out[56]: [2, 3, 4, 5, 2, 3, 4, 5]

In [57]:
```
x.pop(5)
x
```

Out[57]: [2, 3, 4, 5, 2, 4, 5]

In [58]:
```
x.clear()
```

In [59]:
```
x
```

Out[59]: []

In [62]:
```
x = [ 1, 2, 3]
x
```

Out[62]: [1, 2, 3]

In [63]:
```
x = []
x
```

Out[63]: []

In [64]:
```
x = range(10)
x
```

Out[64]: range(0, 10)

In [65]:
```
x1 = list(x)
x1
```

Out[65]: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]

In [66]:
```
x1.index(5)
```

Out[66]: 5

In [67]:
```
x1 = x1 + x1
```

In [68]:
```
x1
```

Out[68]: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9]

In [69]:
```
x1.index(9)
```

Out[69]: 9

In [70]:
```
x1.count(9)
```

Out[70]: 2

In [71]:
```
x1.append(8)
```

In [72]:
```
x1.count(8)
```

Out[72]: 3

In [73]:
```
x1
```

Out[73]: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 8]

In [74]:
```
x1.sort()
```

In [75]:
```
x1
```

Out[75]: [0, 0, 1, 1, 2, 2, 3, 3, 4, 4, 5, 5, 6, 6, 7, 7, 8, 8, 8, 9, 9]

In [76]:
```
n = ["Gary", "Alex", "Tom", "jane", "Kris", "John"]
n
```

Out[76]: ['Gary', 'Alex', 'Tom', 'jane', 'Kris', 'John']

In [77]:
```
n.sort()
```

In [78]:
```
n
```

Out[78]: ['Alex', 'Gary', 'John', 'Kris', 'Tom', 'jane']

In [79]:
```
x1.reverse()
x1
```

Out[79]: [9, 9, 8, 8, 8, 7, 7, 6, 6, 5, 5, 4, 4, 3, 3, 2, 2, 1, 1, 0, 0]

In [80]:
```
n.reverse()
n
```

Out[80]: ['jane', 'Tom', 'Kris', 'John', 'Gary', 'Alex']

In [82]:
```
x = [ 10, 33, 11, 10, 17, 99, 45 ]
x
```

Out[82]: [10, 33, 11, 10, 17, 99, 45]

In [83]:
```
x.reverse()
x
```

Out[83]: [45, 99, 17, 10, 11, 33, 10]

In [84]:
```
x1
```

Out[84]: [9, 9, 8, 8, 8, 7, 7, 6, 6, 5, 5, 4, 4, 3, 3, 2, 2, 1, 1, 0, 0]

In [85]:
```
del x1[1]
x1
```

Out[85]: [9, 8, 8, 8, 7, 7, 6, 6, 5, 5, 4, 4, 3, 3, 2, 2, 1, 1, 0, 0]

In [86]:
```
del x1[0:7]
x1
```

Out[86]: [6, 5, 5, 4, 4, 3, 3, 2, 2, 1, 1, 0, 0]

3. Lists as Stacks and Queues

In [87]:
```
S = []
```

In [88]:
```
type(S)
```

Out[88]:
```
list
```

In [89]:
```
len(S)
```

Out[89]:
```
0
```

In [91]:
```
S.append(10)
S
```

Out[91]:
```
[10]
```

In [92]:
```
S.append(20)
S
```

Out[92]:
```
[10, 20]
```

In [93]:
```
S.append(30)
S.append(40)
S.append(50)
```

In [94]:
```
S
```

Out[94]:
```
[10, 20, 30, 40, 50]
```

In [95]:
```
S.pop()
S
```

Out[95]:
```
[10, 20, 30, 40]
```

In [96]:
```
S.append(500)
S
```

Out[96]:
```
[10, 20, 30, 40, 500]
```

In [97]:
```
S.pop()
S
```

Out[97]:
```
[10, 20, 30, 40]
```

In [98]:
```
Q = []
Q
```

Out[98]: []

In [99]:
```
Q.append(10)
Q.append(20)
Q
```

Out[99]: [10, 20]

In [100]:
```
Q.append(30)
Q.append(40)
Q.append(50)
Q
```

Out[100]: [10, 20, 30, 40, 50]

In [101]:
```
Q.pop(0)
Q
```

Out[101]: [20, 30, 40, 50]

In [102]:
```
Q.pop(0)
Q
```

Out[102]: [30, 40, 50]

4. Some List Comprehensions

In [103]:
```
sq = []
for x in range(10):
    sq.append( x ** 2 )
sq
```

Out[103]: [0, 1, 4, 9, 16, 25, 36, 49, 64, 81]

In [104]:
```
sq2 = [ x ** 2 for x in range(10)]
sq2
```

Out[104]: [0, 1, 4, 9, 16, 25, 36, 49, 64, 81]

In [105]:
```
sqEven = [ x for x in sq2 if x % 2 == 0 ]
sqEven
```

Out[105]: [0, 4, 16, 36, 64]

In [106]:
```
sqOdd = [ x for x in sq2 if x % 2 != 0 ]
sqOdd
```

Out[106]: [1, 9, 25, 49, 81]

In [107]:
```
p = []
for x in range(15):
    p.append( 2 ** x )
p
```

Out[107]: [1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024, 2048, 4096, 8192, 16384]

In [108]:
```
2 ** 14
```

Out[108]: 16384

In [109]:
```
p2 = [ 2 ** x for x in range(15)]
p2
```

Out[109]: [1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024, 2048, 4096, 8192, 16384]

In [111]:
```
a = [ 1, 2, 3]
b = [ 4, 5, 1]
c = []

for x in a:
    for y in b:
        if( x != y ):
            c.append( (x,y))
c
```

Out[111]: [(1, 4), (1, 5), (2, 4), (2, 5), (2, 1), (3, 4), (3, 5), (3, 1)]

In [112]:
```
c2 = [(x,y) for x in a for y in b if x != y ]
c2
```

Out[112]: [(1, 4), (1, 5), (2, 4), (2, 5), (2, 1), (3, 4), (3, 5), (3, 1)]

In [ ]:
```

```