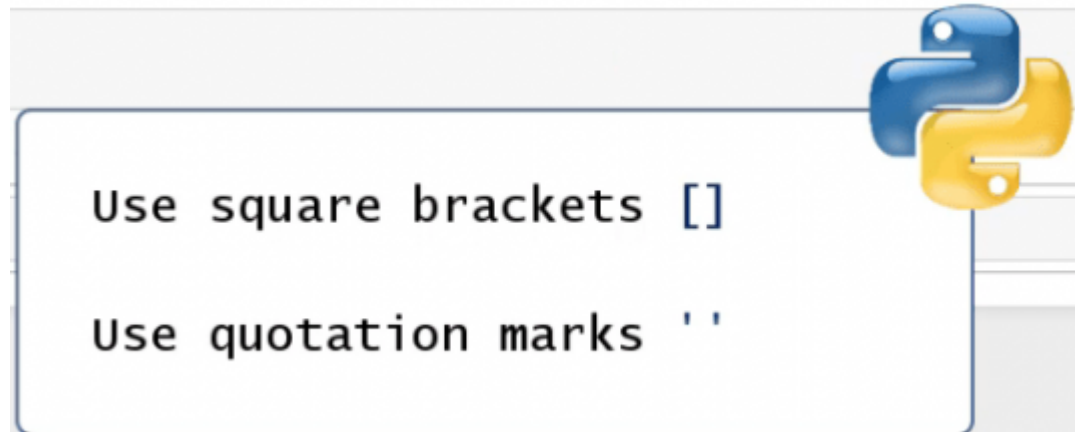


What is a List?

A list is a type of sequence of data points such as floats, integers, or strings.



```
In [1]: Names = ['Python', "Anaconda", 'Jupyter Notebook', 'MySQL']
print(Names)

['Python', 'Anaconda', 'Jupyter Notebook', 'MySQL']
```

Extract Elements

Like how we extract a letter from a string, we use the same technique here.



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

```
Out[3]: 4
```

```
In [2]: Names[2]
```

```
Out[2]: 'Jupyter Notebook'
```

```
In [4]: Names[2][5]
```

```
Out[4]: 'e'
```

```
In [5]: print("We are using:", Names[0])
```

```
We are using: Python
```

Another Way to Extract Elements

A string can be a combination of spaces and In addition, there is a way to get to the last element from a list in **Python** – start counting from the end towards the beginning. Then, we'd need the minus sign before the digit and we should not be thinking that we begin enumerating from 0 again!:

```
In [6]: Names[-1]
```

```
Out[6]: 'MySQL'
```

Replacing items

```
In [7]: Names[-1] = 'Oracle'
Names
```

```
Out[7]: ['Python', 'Anaconda', 'Jupyter Notebook', 'Oracle']
```

Deleting items

```
In [8]: del Names[3]
Names
```

```
Out[8]: ['Python', 'Anaconda', 'Jupyter Notebook']
```

```
In [9]: Names.remove('Jupyter Notebook')
Names
```

```
Out[9]: ['Python', 'Anaconda']
```

```
In [10]: len(Names)
```

```
Out[10]: 2
```

.append()

```
In [11]: Names.append("Jupyter Notebook")
Names
```

```
Out[11]: ['Python', 'Anaconda', "Jupyter Notebook"]
```

.extend()

Merging Multiple Lists

```
In [12]: Names1 = ["mySQL", "Oracle"]
```

```
In [13]: Names.extend(Names1)
Names
```

```
Out[13]: ['Python', 'Anaconda', "Jupyter Notebook", 'mySQL', 'Oracle']
```

```
In [14]: len(Names)
```

```
Out[14]: 5
```

.insert()

```
In [15]: del Names[3:3]
```

```
Out[15]: ['Python', 'Anaconda', 'mySQL', 'Oracle']
```

```
In [16]: Names.insert(2, "Jupyter Notebook")  
print(Names)
```

```
['Python', 'Anaconda', 'Jupyter Notebook', 'mySQL', 'Oracle']
```

pop()

```
In [17]: Names.pop()
```

```
Out[17]: 'Oracle'
```

```
In [18]: print(Names)
```

```
['Python', 'Anaconda', 'Jupyter Notebook', 'mySQL']
```

```
In [19]: Names.append("mySQL")  
Names
```

```
Out[19]: ['Python', 'Anaconda', 'Jupyter Notebook', 'mySQL', 'mySQL']
```

```
In [20]: Names.pop(3)
```

```
Out[20]: 'mySQL'
```

```
In [21]: print(Names)
```

```
['Python', 'Anaconda', 'Jupyter Notebook', 'mySQL']
```

```
In [22]: Names.insert(1, "Anaconda")  
Names
```

```
Out[22]: ['Python', 'Anaconda', 'Anaconda', 'Jupyter Notebook', 'mySQL']
```

Slicing a List

```
In [23]: Names[1:4:1]
```

```
Out[23]: ['Anaconda', 'Anaconda', 'Jupyter Notebook']
```

```
In [24]: Names[1:4:2]
```

```
Out[24]: ['Anaconda', 'Jupyter Notebook']
```

```
In [25]: Names[1:4]
```

```
Out[25]: ['Anaconda', 'Anaconda', 'Jupyter Notebook']
```

```
In [26]: Names[:4]
```

```
Out[26]: ['Python', 'Anaconda', 'Anaconda', 'Jupyter Notebook']
```

```
In [27]: Names[2:]
```

```
Out[27]: ['Anaconda', 'Jupyter Notebook', 'mySQL']
```

```
In [28]: Names[:3] # Names[0:3]
```

```
Out[28]: ['Python', 'Anaconda', 'Anaconda']
```

```
In [29]: Names[3:5]
```

```
Out[29]: ['Jupyter Notebook', 'mySQL']
```

```
In [30]: Names[3:]
```

```
Out[30]: ['Jupyter Notebook', 'mySQL']
```

```
In [31]: Names
```

```
Out[31]: ['Python', 'Anaconda', 'Anaconda', 'Jupyter Notebook', 'mySQL']
```

```
In [33]: Names.pop(1)
```

```
Out[33]: 'Anaconda'
```

```
In [34]: Names[::-1]
```

```
Out[34]: ['mySQL', 'Jupyter Notebook', 'Anaconda', 'Python']
```

```
In [35]: Names[3:0:-1]
```

```
Out[35]: ['mySQL', 'Jupyter Notebook', 'Anaconda']
```

```
In [36]: Names[3::-1]
```

```
Out[36]: ['mySQL', 'Jupyter Notebook', 'Anaconda', 'Python']
```

```
In [37]: Names[0:3:-1]
```

```
Out[37]: []
```

```
In [38]: Names[::-2]
```

```
Out[38]: ['mySQL', 'Anaconda']
```

.index()

```
In [39]: Names.index("mySQL")
```

```
Out[39]: 3
```

```
In [40]: "mySQL" in Names
```

```
Out[40]: True
```

.sort()

`.sort()` sorts objects of list



```
In [41]: Names.sort()  
Names
```

```
Out[41]: ['Anaconda', 'Jupyter Notebook', 'Python', 'mySQL']
```

```
In [42]: Names.sort(reverse=True)  
Names
```

```
Out[42]: ['mySQL', 'Python', 'Jupyter Notebook', 'Anaconda']
```

```
In [43]: print(Names * 2)
```

```
['mySQL', 'Python', 'Jupyter Notebook', 'Anaconda', 'mySQL', 'Python', 'Jupyter No  
tebook', 'Anaconda']
```

```
In [1]: ages = [23, 16, 14, 28, 19, 11, 38]
```

```
In [3]: youngest = min(ages)  
oldest = max(ages)  
total_years = sum(ages)  
count = len(ages)  
  
print(f"Youngest: {youngest}")  
print(f"Eldest: {oldest}")  
print(f"Sum of Ages: {total_years}")  
print(f"Count: {count}")  
print(f"Avg.: {total_years/count}")
```

```
Youngest: 11  
Eldest: 38  
Sum of Ages: 149  
Count: 7  
Avg.: 21.285714285714285
```

```
In [4]: ages.sort()  
ages
```

```
Out[4]: [11, 14, 16, 19, 23, 28, 38]
```

```
In [5]: ages.sort(reverse=True)  
ages
```

```
Out[5]: [38, 28, 23, 19, 16, 14, 11]
```

a List of Lists

```
In [6]: Names1 = ['Python', 'Anaconda']  
Names1
```

```
Out[6]: ['Python', 'Anaconda']
```

```
In [7]: Names2 = ['Jupyter Notebook', 'mySQL', 'Oracle']
Names2
```

```
Out[7]: ['Jupyter Notebook', 'mySQL', 'Oracle']
```

```
In [8]: Names3 = [Names1, Names2]
Names3
```

```
Out[8]: [['Python', 'Anaconda'], ['Jupyter Notebook', 'mySQL', 'Oracle']]
```

```

      0      1      2
0 ['Python', 'Anaconda']
1 ['Jupyter Notebook', 'mySQL', 'Oracle']
```

```
In [17]: len(Names3)
```

```
Out[17]: 2
```

```
In [18]: len(Names3[1])
```

```
Out[18]: 3
```

```
In [9]: Names3[1]
```

```
Out[9]: ['Jupyter Notebook', 'mySQL', 'Oracle']
```

```
In [10]: Names3[1][0]
```

```
Out[10]: 'Jupyter Notebook'
```

```
In [19]: len(Names3[0])
```

```
Out[19]: 2
```

```
In [11]: Names3[0][1]
```

```
Out[11]: 'Anaconda'
```

```
Row
Col 0 1 2
0   [8, 4, 2],
1   [5, 7, 9]
```

```
In [12]: a = [[8, 4, 2], [5, 7, 9]]
```

```
In [13]: a[1][2]
```

```
Out[13]: 9
```

```
In [14]: a[0]
```

```
Out[14]: [8, 4, 2]
```

In [15]: `len(a)`

Out[15]: 2

In [16]: `len(a[0])`

Out[16]: 3

In []: