PythonTip 02 - List Slicing

February 17, 2021

1 List Slicing

List slicing is a way to get not just one element of a list, but a whole portion.

[1]:
```
L = ["a", "b", "c", "d", "e", "f"]
```

`L[a:b]` means the portion of the list from index `a` (inclusive) to index `b` (exclusive).

[2]:
```
L[2:5]
```

[2]:
```
['c', 'd', 'e']
```

If you leave out `a`, it starts from the beginning of the list. If you leave out `b`, it goes to the end.

[3]:
```
L[:4]
```

[3]:
```
['a', 'b', 'c', 'd']
```

[4]:
```
L[1:]
```

[4]:
```
['b', 'c', 'd', 'e', 'f']
```

[6]:
```
L[:] # a copy of the list!
```

[6]:
```
['a', 'b', 'c', 'd', 'e', 'f']
```

You can use a third piece `L[a:b:c]`, and `c` means how much to go up by each time.

[9]:
```
L[1:5:2]
```

[9]:
```
['b', 'd']
```

[10]:
```
L[::2]
```

[10]:
```
['a', 'c', 'e']
```

[11]:
```
L[::-1]
```

[11]:
```
['f', 'e', 'd', 'c', 'b', 'a']
```

Lastly, you can use negative indexing too. For example, to get the last 3 elements of a list:

[13]:
```
L[-3:]
```

[13]:
```
['d', 'e', 'f']
```

To get all except the last element:

[14]:
```
L[:-1]
```

[14]:
```
['a', 'b', 'c', 'd', 'e']
```