Python Introduction: Indexing

The `range` function lets us build a list of numbers.

In [12]:

```
list(range(10, 20))
```

Out[12]:

```
[10, 11, 12, 13, 14, 15, 16, 17, 18, 19]
```

Notice anything funny?

Python uses this convention everywhere.

In [13]:

```
a = list(range(10, 20))
type(a)
```

Out[13]:

```
list
```

Let's talk about indexing.

Indexing in Python starts at 0.

In [14]:

```
a[0]
```

Out[14]:

```
10
```

And goes from there.

In [15]:

```
a[1]
```

Out[15]:

```
11
```

In [16]:

```
a[2]
```

Out[16]:

```
12
```

What do negative numbers do?

In [6]:

```
a[-1]
```

Out[6]:

```
19
```

In [17]:

```
a[-2]
```

Out[17]:

```
18
```

You can get a sub-list by *slicing*.

In [19]:

```
a[3:7]
```

Out[19]:

```
[13, 14, 15, 16]
```

Start and end are optional.

In [20]:

```
a[3:]
```

Out[20]:

```
[13, 14, 15, 16, 17, 18, 19]
```

In [21]:

```
a[:3]
```

Out[21]:

```
[10, 11, 12]
```

Again, notice how the end entry is not included:

In [23]:

```
print(a[:3])
print(a[3])
```

```
[10, 11, 12]
13
```

Slicing works on any collection type! (`list`, `tuple`, `str`, `numpy` array)

In [24]:

```
a = "CS357"
a[-3:]
```

Out[24]:

```
'357'
```