Python

List comprehensions - what are they? They are useful!

List Comprehensions

Python supports a concept called "List Comprehensions". Imagine you want to create a list of square numbers from the list of numbers from 0 to 9. You would type:

```
>>> S = []
>>> for x in range(10):
...     S.append(x**2)

>>> print(S)
[0, 1, 4, 9, 16, 25, 36, 49, 64, 81]
```

Saving on lines of code

List Comprehensions allow you to do it on **one line**:

```
>>> S = [x**2 for x in range(10)]
>>> print(S)
[0, 1, 4, 9, 16, 25, 36, 49, 64, 81]
```

These can be used to construct lists in a natural and easy way.

It gets better - include conditions

Imagine our previous example - but you only want to include values in the list where the result is an even number:

```
>>> S = []
>>> for x in range(10):
...     res = x**2
...     if res % 2 == 0:
...         S.append(res)
>>> print(S)
[0, 4, 16, 36, 64]
```

Can be simplified to…

All one line

```
>>> S =
[x**2 for x in range(10) if x**2 % 2 == 0]

>>> print(S)
[0, 4, 16, 36, 64]
```

See more info at:
https://www.python.org/dev/peps/pep-0202/#examples