New Stuff: list comprehensions

List comprehensions

- effecient syntax for manipulating information in an existing list and assigning to a new list
- Syntax: for each element in an existing list: do some operation and assign the output to a new list

In [ ]:

```
# long way to square a list of numbers using a 'for loop'
list_num = list(range(0,6))

# initialize a list of length(list_num) to store the squred values
square_num = []

# for loop to square each value and to fill up our square_num list
for num in list_num:
  square_num.append(num**2)

print(square_num)
```

```
[0, 1, 4, 9, 16, 25]
```

The list comprehension way to do the same thing in more compact notation

- syntax: new_list = [expression for each_item in old_list]
- note the brackets [ ] - can remember because this is how you define a list (and here you are defining a new list to store the output)

In [ ]:

```
list_num = list(range(0,6))

square_num = [num**2 for num in list_num]

print(square_num)
```

```
[0, 1, 4, 9, 16, 25]
```

Evaluate an equation over a set of numbers (like y = mx + b)

In [ ]:

```
# x range over whic to eval the equation
x_vals = range(0,10)
m = .5
b = 7

# here it is!
y = [((m*x)+b) for x in x_vals]

print(y)
```

```
[7.0, 7.5, 8.0, 8.5, 9.0, 9.5, 10.0, 10.5, 11.0, 11.5]
```

Another slightly more complex example

In [ ]:

```
# eval the sum of squared differences between f1 and f1**2
some_function = range(-10,10)

# compute sum of squares...
y = [(f-(f**2))**2 for f in some_function]

# then take the sum
print(sum(y))
```

```
43336
```

Using methods in a list comprehension

In [ ]:

```
names = ['sabina', 'lorraine', 'madison']

title_name = [name.upper() for name in names]

# print caps of first letter of each name
print(title_name)
```

```
['SABINA', 'LORRAINE', 'MADISON']
```

Another example, but operate on only one character in each string in the list

In [ ]:

```
names = ['jolene', 'rui', 'rio']

first_letter = [name[0].upper() for name in names]

# print caps of first letter of each name
print(first_letter)
```

```
['J', 'R', 'R']
```

Combine list comprehension with a if...statement

- for an example, filter a list of bike manufacturers and find all that start with a 'c'so that you can show just some brands to a user...
- first do it the traditional way...
- then with a list comprehension

In [ ]:

```
# traditional way...

# start with a long
all_bikes = ['cannondale', 'trek', 'specialized', 'giant', 'canyon', 'cervelo']

# initialize empty list to store the output
c_bikes = []

for name in all_bikes:
  if name[0]=='c':
    c_bikes.append(name)

print(c_bikes)
```

```
['cannondale', 'canyon', 'cervelo']
```

In [ ]:

```
# list comprehension way
c_bikes = [name for name in all_bikes if "c" in name[0]]
print(c_bikes)
```

```
['cannondale', 'canyon', 'cervelo']
```