Functional Programming with Python

https://github.com/treehouse/python_fp_meetup

Functional Programming

Computation is the evaluation of functions

Programming is done with expressions

Functions are first-class citizens of the language

Functions shouldn’t have side-effects

```
def reverse_list(lst):
    lst.reverse()
    return lst
```

eww

Functions should be limited in scope and functionality

```
def mutate_list(lst):
    lst = lst[::-1]
    lst[0], lst[-1] = lst[-1], lst[0]
    lst.insert(0, ‘Hi there’)
    if isinstance(lst[0], str):
        lst[0] = lst[0][::-1]
    elif isinstance(lst[0], (float, int):
        lst[0] = lst[0] // 2
    return lst
```

ewwwwwwwwwwww

```
def reverse_list(lst):
    lst = lst[::-1]
def swap_ends(lst):
    lst[0], lst[-1] = lst[-1], lst[0]
def greeting()
    return ‘Hi there’
def do_my_stuff(lst):
    reverse_list(lst)
    swap_ends(lst)
    lst.insert(0, greeting())
```

Better in scope, but...

Chainable Functions

It’s hard to chain functions if they mutate items in a higher scope

Use return a lot

Like, more than that. **A lot**

And, if you didn’t guess, **you don’t want to return mutations**

Python gives us some chaining-friendly functions to start with

map(function, iterable)

Applies the function to each member of the iterable

filter(function or None, iterable)

Applies the function to each member of the iterable and returns each member where the function evaluated to True

If None is provided, returns only truthy members of the iterable

functools.reduce(function, iterable)

Returns a single item calculated by applying the function to sets of members from the iterable

Enough talk!

this slide left intentionally blank

Functions to know and love

max(*args): returns the largest item

min(*args): returns the smallest item

any(*args): returns True if any item is truthy

all(*args): returns True if all items are truthy

reversed(iterable): returns a reversed copy of the iterable

sorted(iterable): returns a naturally-sorted copy of the iterable

Questions?

Time for more?

Lambdas

Anonymous, temporary functions

List comprehensions

Statements that return a list. Can be used most places you use **filter()** or **map()**.

Dictionary comprehensions

Statements that return a dictionary.

Want help?

PyLadies PDX

meetup.com/PyLadies-PDX/

Monday Python Flying Circus

meetup.com/pdxpython/