

DS2000

10/11 - Fri :)

Admin

- HW3 due 9pm
- HW4 out!, due 10/19 9pm
- Mon holiday :)
- no liney OH

Agenda

1. writing functions
2. organizing code
3. Python

0. Review

- calling functions

(ex) `result = spam(3, "eggs")`

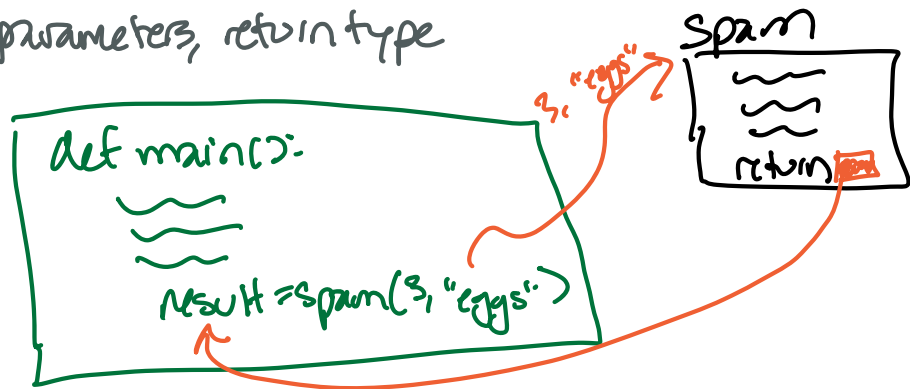
- How do I know I'm calling?
`functionname()`
- Name of function?
`spam`
- parameters?
2 params, int and string
- does spam return a value?
yes (probably)!

1. Writing Functions

What do we know about fncs:

- Fnc returns one thing, or nothing!
- Fnc has one job!
- need: name, parameters, return type

result = spam(~)
- if spam returns nothing
result = None



Why write functions?

- Sometimes Naming code blocks
- gets code out of the way
- reuse code that we write once
- same functions in multiple .py files
- makes code more flexible

When writing a function...

- put params, return, descr in block comment

```
'''  
params: int, string  
return: one int  
desc: some calc about spam  
'''
```

- to write a function, start with def, then name, then params

give params

names, like variables

```
def spam(myint, mystr):
```

- in function, use return to give back a value

```
~~~~~  
return value
```

Terms

- ↳ function prints $\textcircled{A}$ returns, not both.

2. organizing code

↳ we define functions, they sit there
we call functions, they execute

today's data ~ hunk's locations

- ↳ Count dunks in a city

↳ maybe also plot

CSR file ~ many lines!

- one line per location

- lat, long, city/state, address

- ↳ data separated by commas

→ process are live ^{at} a time

① row (string) $\rightarrow$ list w/ good data types

- params: one string, lat pos, long position
- return: list w/ lat, long as floats

② find if local

params: list for one location
name of city
city/state pos

Retum: bodleazn

$\{lat, long, "w", "w"\}$ return: body
city/state address

"Boston" in "Boston, MA" - True