Functions

Python for Ecologists

Jon Flaishans, Tom Purucker, Tao Hong, Marcia Snyder

Ecological Society of America Workshop
Minneapolis, MN

purucker.tom@gmail.com

August 3, 2013

Functions

- A function takes one or more arguments and returns something
- Can return any type of data structure, including None
- Basic organization of a function:
- ```
  def some_function(arg1, arg2):
      #some statements
      return answer
  ```
  ```

whitespace and naming

- Whitespace usage is important
  - Indent consistently with 2 or 4 spaces
  - Use spaces instead of tabs, -tt
- Naming conventions
  - lower case words, cannot start with a number
  - use verbs that describe what the function does
  - underscore_between_words

```
def some_function(arg1, arg2):
    #some statements
    return answer
```

Example Function

- Argument names can be informative, helping documentation (e.g., growth_rate instead of arg1)

```
def add_two_numbers(num1, num2):
  return num1 + num2

add_two_numbers(6,5)
```

docstrings

- Tripe quoted comment at beginning of functions is accessible as a dunder doc (.__doc__) or help()

```
def add_two_numbers(num1, num2):
    """ this function adds two numbers
    together"""
    return num1 + num2

help(add_two_numbers)
```

Types of Arguments

- Required arguments do not have a default
- Keyword arguments can have a default value (and are optional)
- Keyword arguments are differentiated by setting equal to a value in the function argument list

```
def double_it(required1, keyword2=2):
  return required1 * keyword2

double_it(6)
double_it(6,3) #actually triples it
```