Introduction to Computation for the Humanities and Social Sciences

CS 3

Chris Tanner

Lecture 11

Dictionary Practice and Project

Lecture 11

- Functions
- Dictionaries

Functions

- Functions are akin to chapters in books. It’s okay if you don’t know all the chapters ahead of time.

- Only create functions once you have **at least** a good, general idea of how all your code will work.

- Then, just wrap sections of code into functions.

- First focus on getting all the code to work, even if that means putting all your code in a single **main**() function.

Dictionaries

- Remember, you can't access the values directly. Only access a value for a specific key.
- Akin to a mailbox. Can't just say "give me mail" (aka **value**).
- Need to tell it the mailbox (i.e., **key**) whose mail (i.e., **value**) you want.
- The value can be any data structure:
  - single-value (e.g., integer value, String value…) or
  - a **`list`** of stuff, etc.

Example of Updating a key’s value

```
# counts how many times each word occurs in a file
word_counts = {}
input_file = open(filename, ‘r’)
words = input_file.read().split(“ “)
for current_word in words:

    if current_word in words.keys(): # update the count
       word_counts[current_word] = word_counts[current_word] + 1

    else: # initialize the count
       word_counts[current_word] = 1
```

Dictionaries

Iterate through the Dictionary’s Keys

```
# iterate through the keys
for word in word_counts.keys():
   print(word + “ appears “ + str(word_counts[word]) + “ times”)
```

value