

Objectives

- More on Dictionaries
- Exceptions

Review

- What is a dictionary in Python?
 - What does it represent?
 - What are its rules?
 - What is the syntax?
 - What are some operations we can perform on dictionaries?

Problem

years_dictionary.py

- Part 1:
 - Given a file of the form
 - <lastname> <classyear>
 - Goal: I want to quickly find out what a student's class is
 - How do we want to model the data?
 - What is the key? What is the value?
 - How to display the mapping in a pretty way?
 - What order is the data printed in?
- Part 2:
 - Prompt user for the last name of the student
 - Display the student's graduation year

Nov 13, 2017

Sprenkle - CSCI111 Part 3: Repeat Part 2

3

Algorithm to Problem

- Create an empty dictionary
- Read in the file line by line
 - Split the line
 - From the split, get the last name and the year
 - Add a mapping of the last name to the year in the dictionary
 - (accumulate the data in the dictionary)
- Process the data in the dictionary, e.g.,
 - Display it, in sorted order
 - Get user input to get answers

Nov 13, 2017

Sprenkle - CSCI111

4

Problem

- Keep track of the *number* of students of each year
 - How do we want to model the data?
 - What is the key? What is the value?

Could we solve this using a list?

years_dictionary2.py

Nov 13, 2017

Sprenkle - CSCI111

5

Analyzing years_dictionary2.py

- Anything useful/general that we could put in a function?

Nov 13, 2017

Sprenkle - CSCI111

6

Equivalent Solutions

```
if key not in dictionary :  
    dictionary[key] = 1  
else:  
    value = dictionary[key] + 1  
    dictionary[key] = value
```

```
if key not in dictionary :  
    dictionary[key] = 1  
else:  
    dictionary[key] += 1
```

Discussion

- Compare lists and dictionaries
 - What are their properties?
 - How are they similar?
 - How are they different?
 - When do you use one or the other?

Why Dictionaries?

- An alternative way to store data
- Allow fast lookup of data
 - Requires keys, unique keys
 - Data may not have a natural mapping

Pros	Cons
Fast lookup (much faster than looking through a list if a lot of elements)	Requires a lot of space, Requires unique keys

Nov 13, 2017

Sprenkle - CSCI111

9

Lists vs. Dictionaries

Lists	Dictionaries
<i>integer positions</i> (0, ...) to any type of value	Map <i>immutable keys</i> (int, float, string) to any type of value
Ordered	Unordered
Slower to find a value (in)	Fast to find a value (use key)
Fast to print in order	Slower to print in order (by key)
Only as big as you make it (ish)	Takes up a lot of space (so can add elements in the middle)

Nov 13, 2017

Sprenkle - CSCI111

10

EXCEPTION HANDLING

Nov 13, 2017

Sprenkle - CSCI111

11

Handling Exceptions

- Using try/except statements

- Syntax:

```
try:
    <body>
except [<errorType>] :
    <handler>
```

Optional: use this to handle specific error types appropriately

- Example:

```
try:
    age = eval(input("Enter your age: "))
    currentyear = int(input("Enter the current year: "))
except:
    print("ERROR: Your input was not in the correct form.")
    print("Enter integers for your age and the current year")
    sys.exit()
```

Nov 13, 2017

Sprenkle - CSCI111 [yearborn.py](#)

12

Handling Exceptions

- Other types of exceptions
 - File exceptions:
 - File doesn't exist
 - Don't have permission to read/write file

Nov 13, 2017

Sprenkle - CSCI111

`file_handle.py` 13

Looking Ahead

- Lab 9 tomorrow
 - Lab prep
- Broader Issue - Friday

Nov 13, 2017

Sprenkle - CSCI111

14