Lecture 22: Introduction to Computer Programming Course - CS1010

DEPARTMENT OF COMPUTER SCIENCE | 04/15/2019

Rensselaer

Announcements

- Homework 10 was posted on Saturday
- Due on April 23 (Tuesday)
- Last Class on April 25: Will be a review class

Goals for Today

- **Dictionaries (Part 2):**
- Nesting with Dictionaries
- Dictionary Methods
- Merging Dictionaries
- Iterating over dictionaries

Nesting with Dictionaries

- Python dictionaries are extremely powerful in terms of flexibility.
- You can nest objects within a dictionary
- More specifically you can have a dictionary nested inside a dictionary
- Accessing values then depends on key nesting
- Check in Spyder

Dictionary Methods

- A dictionary method is a function provided by the dictionary type (dict) that operates on a specific dictionary object.
- Using Dictionary methods we can perform useful operations like:
  - adding or removing elements,
  - obtaining all the keys or values in the dictionary,
  - merging dictionaries, etc.
- Let’s check a few methods in Spyder

Methods	Description
copy()	They copy() method returns a shallow copy of the dictionary.
clear()	The clear() method removes all items from the dictionary.
pop()	Removes and returns an element from a dictionary having the given key.
popitem()	Removes the arbitrary key-value pair from the dictionary and returns it as tuple.
get()	It is a conventional method to access a value for a key.
values()	returns a list of all the values available in a given dictionary.
str()	Produces a printable string representation of a dictionary.
update()	Adds dictionary dict2’s key-values pairs to dict
setdefault()	Set dict[key]=default if key is not already in dict
keys()	Returns list of dictionary dict’s keys
items()	Returns a list of dict’s (key, value) tuple pairs
type()	Returns the type of the passed variable.

Merging

- We can concatenating dictionaries, like we did with lists
- There is something similar for dictionaries:
- The update method
- update() merges the keys and values of one dictionary into another, overwriting values of the same key

Iteration

- Can iterate over dictionaries just like any other iterable.
- No method is required
- for key in d:
- 	print key

Iteration continued

But it's possible to use the method keys():

```
for key in d.keys():
        print key
```

The method values() is a convenient way for iterating directly over the values:

```
for val in d.values():
        print val
```

The above loop is of course equivalent to the following one:

```
for key in d:
        print d[key]
```

Dictionary from Lists

- Can convert lists to dictionary
- Let’s check examples

Problem 1

- Write a Python script to merge two Python dictionaries.
- d1 = {'a': 100, 'b': 200}
- d2 = {'x': 300, 'y': 200}

Problem 2

- Find the second largest value from a Python dictionary.

Problem 3

- Given a dictionary with employee names and salaries, find the total salary of all employees.

Problem 4

- Filtering selective keys based on certain criteria.
- Method 1: List Comprehensions
- Method 2: Sets

Counter

- A Counter is a container that keeps track of how many times equivalent values are added.
- It can be used to implement the same algorithms for which bag or multiset data structures are commonly used in other languages.

Problem 5

- Element Occurrence in dictionary of list values
- Count the occurrences of test elements in list
- Method 1 : Using dictionary comprehension + sum()
- Method 2 : Using collections.Counter()

Next lecture

- Problems on Dictionaries
- In Class Exercise