Basics of Programming

L08: Dictionaries

Mar 2020

Dr. Ram P Rustagi
Professor, CSE Dept
KRP, KSGI
rprustagi@ksit.edu.in

Resources and Acknowledgements

- A first course in programming
  - https://introcs.cs.princeton.edu/python/20functions/
- Python for everybody
  - https://www.py4e.com

Collections

- Collection:
  - Multiple values together in a single entity
  - Multiple values are represented by a variable
  - Allows checking for existing of a value
  - Allows adding new value, removing existing value
  - Allows modification/update of existing values
- Collections studies so far:
  - Lists: values are accessed by index
    - vowels = [‘A’, ‘E’, ‘I’, ‘O’, ‘U’]
- New collection type: Dictionary
  - Values are stored with their own label
  - Values can be accessed using the label
  - Similar to typical dictionary access

Dictionaries

- A very powerful data collection
- Enables fast database like operations
- Known by different name in other languages
  - Perl/PHP: Associative arrays
  - Java: Hashmap, Map, Properties
  - C#/.Net: Property bag
- Lists implies ordering of elements
- Dictionary does not imply any order
  - Elements are accessed by key or tag or label

Example

- Consider owners Mantri Tranquil apartments

```
owner = {}
owner['I-205'] = "Ram Rustagi"
owner['D-1001'] = "S Sundar"
owner['D-306'] = 'Prasanna Neelavar'
print(owner)
```

```
{'I-205': 'Ram Rustagi', 'D-1001':
'S Sundar', 'D-306': 'Prasanna
Neelavar'}
```

- The print output need not be in same order.

Dictionary Literals

- Creating a dictionary requires use of curly braces and `key:value` pairs.

```
owner = {'I-205': 'Ram Rustagi',
         'D-1001': 'S Sundar',
         'D-306' : 'P Neelavar'}
daysinmonth = {'Jan':31, 'Feb':28,
   'Mar':31, 'Apr':30, 'May':31, 'Jun':
   30, 'Jul':31, 'Aug': 31,
   'Sep':30, 'Oct':31, 'Nov':30, 'Dec':31}
```

- Key of a dictionary can be a number as well
  - Key has to be unique
  - Can't associate multiple values with a key

```
monthofdays = {31:'Jan Mar May Jul Aug
Oct Dec', 30:'Apr Jun Sep Nov',
28:'Feb', 29:'Feb'}
```

Application: Count of words in

- Given text input, count how many times a given word appears.

```
text='we are here to learn python
programming. python is easy to learn. C++
programming language is hard.'
count={} # empty dictionary
for word in text.split():
   if word[-1] = '.':
      word.remove('.')
   if word in count.keys():
      count[word] = count[word] + 1
   else:
      count[word] = 1
print(count)
```

Dictionary Errors

- Accessing an non-existing key gives error
- For example, in previous exampe
  - `print(owner['I-105'])`

```
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
KeyError: 'I-105'
```

- So, if not sure of key's existence, check for it

```
if 'I-105' in owner: # defaults for keys
if 'I-105' in owner.keys():
```

get() method: key's default value

- getting a default value for non-existing key
  `count.get(key,default)`
- The word count program can be written as

```
count={} # empty dictionary
for word in text.split():
   if word[-1] = '.':
      word.remove('.')
   currcnt = count.get(word,0)
   count[word] = currcnt + 1

print(count)
```

Iterating over a Dictionary

- For loop can be used iterate over dictionary
  - both **keys** and **values** and together as **items**

```
for key in count:
  print(key, count[key])

for value in count.values():
  print(value)

for key,value in count.items():
  print(key, value)
```

Copying Dictionary

- Assigning dictionary variable to another variable just gives the new name. Both refers to same dictionary.
- For example

```
square={1:1, 2:4, 3:9, 4:16}
newsq = square
newsq[5] = 25
print(square[5])
```

- To make a explicit copy, use the copy method

```
square={1:1, 2:4, 3:9, 4:16}
newsq = square.copy
newsq[5] = 25
print(square[5]) # gives error
```

Other Dictionary Methods

- Removing all elements of dictionary
  - clear()
- for example

```
square={1:1, 2:4, 3:9, 4:16}
square.clear()
print(square)
```

Exercises

- Ex01a: Write a program to create dictionary
  - Input rollnumber, name on terminal
  - add to dictionary student with key as roll number.
  - print the student dictionary.

- Ex01b: Find the roll number
  - Read the roll number, names from a file. Each line of file conains two entries (roll number, name)
  - Build a dictionary.
  - Input a name on terminal, and display its roll number.
    - Hint: Iterate over all items and check if the name exists,.

Exercise 2

- Dictionary of dictionaries:
- For each of your subect in a class, build a dictionary of your marks.
  - Key would be subject, and values will be list having 2 values (internal marks, and external marks).
  - for example, for class 11,
    - dictionary will have 5 entries (CBSE), 6 entries(PU)
  - For each year study, class becomes the key, and dictionary of subject marks become values.

```
class11 = {'M':[20,79], 'P':[18,78],
'C': [19,76], 'E':[20,75],'C':[20,80]}
class12 = {'M':[20,80], 'P':[17,77],
'C': [20,79], 'E':[16,76],'C':[20,80]}
marks={11: class11, 12:class12}
```

Classes in Future

- Your online school classes
- Your competitive coaching classes
- Possible suggestions for continuing
  - twice a week same time or other time
  - thrice a week, same time or other time
  - once a week on Thu/Fri/Sat?

Questions