Introduction to Computer Programming

Section 8

Lecture 25

Dictionaries
What are Python Dictionaries?

A dictionary is a map of values to values.

• Example: Birthday map

  ‘Mom’    ->  ‘5/20’
  ‘Greta’  ->  ‘3/19’
  ‘Tom’    ->  ‘9/30’

• You can look birthdays up very easily BY NAME
  – (Lists require knowing the position, or index)
Keys and Values

birthdays = { 'Mom' : '5/20',
             'Greta' : '3/19',
             'Tom' : '9/30' }

• The first item in a pair is called the **key**
• The second item in a pair is called the **value**
• You can retrieve a value by specifying the key:

```python
>>> birthdays['Greta']
'3/19'
```
What are the keys and values?

birthdays = { 'Mom': '5/20',
             'Greta': '3/19',
             'Tom': '9/30' }


Adding another dictionary entry

birthdays =  { ‘Mom’ : ‘5/20’,
               ‘Greta’ : ‘3/19’,
               ‘Tom’ : ‘9/30’ } 

>>>birthdays[‘Jethro’] = ‘11/2’

>>>birthdays

Python Dictionary Syntax

>>> birthdays = { }
>>> birthdays[‘Mom’] = ‘5/20’
>>> birthdays[‘Greta’] = ‘3/19’
>>> birthdays[‘Tom’] = ‘9/30’

>>> birthdays[‘Mom’]
‘5/20’
Facts about Dictionaries

• Dictionaries are “unordered”
  – If you print the dictionary, the order of the items is not predictable, and it doesn’t matter!

• There can only be one entry for each key.

• Lists cannot be keys
  – Strings and numbers can be keys

• Values can be anything
  – Strings, numbers, lists, other dictionaries
Example: Keys are unique

birthdays = { 'Mom': '5/20',
              'Greta': '3/19',
              'Tom': '9/30' }

>>>birthdays['Mom'] = '11/2'

>>>birthdays
{'Mom': '11/12', 'Greta': '3/19',
 'Tom': '9/30'}
Out-of-bounds Checking with the \texttt{in} operator

```python
>>> birthdays = {'Tom': '9/30', 'Mom': '5/20', 'Greta': '3/19'}
>>> birthdays['Fred']
Traceback (most recent call last):
  File "<pyshell#17>", line 1, in <module>
    birthdays['Fred']
KeyError: 'Fred'

if 'Fred' in birthdays :
    fb = birthdays['Fred']
```
When to use Lists vs Dictionaries?

• Use Lists for integer-indexed maps.
  – Month number (1-12) to month name

```
```

• Use Dictionaries for string-indexed maps.
  – Month name to month number

```
{‘January’:1, ‘February’:2, ‘March’:3}
```