

**Dictionaries** A python dictionary contains keys and values. Sometimes they're called key-value pairs. Often they're called hash tables. By any name they organize information by association. So the values can be located by use of the corresponding key. A list, on the other hand, organizes its elements by position. So you could easily find the third item in a list. A dictionary doesn't have a third item. It does, actually, but you don't access it that way. You would access the value by knowing the corresponding key you gave it. The keys and values in a python dictionary can be of any type, not just integers and strings. You could even have a dictionary that contains values which are lists or other dictionaries.

A python dictionary, like a regular dictionary makes a great look up table. So if you wanted to store the cities of some states you could create a dictionary like this: `cities = {'CA': 'San Francisco', 'MI': 'Detroit', 'FL': 'Jacksonville'}`

Notice that the dictionary uses the {} characters (sometimes called brackets). A list uses the [] characters. I don't know what they are called - square brackets?!

You can access a state's city as: `city = cities['CA']`. Now the variable called city will contain the value 'San Francisco'. You can add a city as: `cities['OR'] = 'Portland'`.

If you want to print a sorted list of the cities for each state in your program you would do something like this: Note that I converted the dictionary to a list so I could sort it because you can't sort a dictionary.

```
theKeys = list(amounts.keys()) theKeys.sort() for key in theKeys: print(key,amounts[key])
```

There are more examples in the book and online. Here is a link if you want more information: <http://learnpythonthehardway.org/book/ex39.html>.

So, let's use the dictionary in a program.

1. Create a program that will prompt the user to enter an amount. When the user enters the amount, the program will respond with the text spelling out of the amount. For example, the user enters '10', the program responds with 'ten'.
  2. If the user enters an amount that the program does not have stored then the program shall prompt the user to enter the text. So if the user enters '201' and the program doesn't know that value then the program will prompt the user to type in the text, 'two-hundred one'.
  3. Once the user types in the text, the program shall store the text in a dictionary. This will enable the program to quickly locate the value the next time the user enters it.
-