Cmpe 150 Lab 7: Dictionaries

Dictionaries

- They are similar to lists, yet we want to store several pairs as (key, value)

- For example, the name of the people and the number of books they have.

Python Dictionaries

- empty_dict = {}

- non_empty_dict = {'Ali': 15, 'Hasan': 13, 'Osman': 15}

Python Dictionaries (Cont.)

- The type of key and value can be anything, string and int combination is only an example.

- my_key in my_dict to learn if the given item exists as a key in the dictionary.

Access or Change the Value of a Key

- print(my_dict[my_key]) -> Be careful since it gives an error if the key is not in the dictionary, so using “in” before it might be better.

- my_dict[my_key] = new_val -> If my_key is not in the dictionary, it will define it.

Delete a Pair from the Dictionary

- my_dict.pop(key)
- Before: my_dict = {1: 2, 2: 4, 3: 6}
- After my_dict.pop(2) -> {1: 2, 3:6}

Keys and Values

- my_keys_list = list(my_dict.keys())
- my_values_list = list(my_dict.values())

Items

- Returns all the existing information as a list of tuples (key, value)

- my_items = my_dict.items()

Items (Cont.)

- Returns all the existing information as a list of tuples (key, value)

- We can use sorted function to do interesting stuff.

Using a Loop Over Dictionaries

- for key in my_dict:

  print(key, my_dict[key])

- for value in my_dict.values():

  print(value)

Using a Loop Over Dictionaries (Cont.)

- Using items function is also possible.

- for k, v in my_dict.items():

  print(k, v)

Thanks

Any questions?

References

1. https://www.w3schools.com/python/python_dictionaries.asp