



# Lab from Last Time



Dictionary: object that stores a collection of data

- Each element consists of a key and a value

- Often referred to as mapping of key to value
- Key must be an immutable object (cannot be changed!)
- To retrieve a specific value, use the key associated with it
- Format for creating a dictionary

dictionary =

{key1:val1, key2:val2}

# **Retrieving a Value from a Dictionary**

· Elements in dictionary are unsorted

- General format for retrieving value from dictionary: dictionary[key]
  - If key in the dictionary, associated value is returned, otherwise, KeyError exception is raised
- Test whether a key is in a dictionary using the in and not in operators
  - Helps prevent KeyError exceptions

## **Retrieving a Value from a Dictionary**

def main():

print(phonebook)
print("Katie's phone number is: ", phonebook['Katie'])

main()

### Program Output {'Chris': '555-1111', 'JoAnne': '555-2222', 'Katie': '555-3333'} Katie's phone number is: 555-3333

# Adding Elements to an Existing Dictionary

- Dictionaries are mutable objects
- To add a new key-value pair:
  - dictionary[key] = value
  - If key exists in the dictionary, the value associated with it will be changed

# Adding Elements to an Existing Dictionary

def main():

print("Before Add:", phonebook)
phonebook['Andy'] = '555-0123'
print("After Add:", phonebook)

main()

### **Program Output**

Before Add: {<sup>1</sup>JoAnne': '555-2222', 'Chris': '555-1111', 'Katie': '555-3333'}

After Add: {'JoAnne': '555-2222', 'Chris': '555-1111', 'Katie': '555-3333', 'Andy': '555-0123'}



Deleting Elements From an Existing Dictionary

def main():

```
phonebook = {'Chris': '555-1111', 'Katie': '555-3333'
    'JoAnne': '555-2222'}
print("Before Delete:", phonebook)
del phonebook['JoAnne']
print("After Delete:", phonebook)
```

main()

### **Program Output**

Before Delete: {'Katie': '555-3333', 'JoAnne': '555-2222', 'Chris': '555-1111'} After Delete: {'Katie': '555-3333', 'Chris': '555-1111'}

# Using a Dictionary as a Color Map

Dictionaries map one value to another

We could map numbers to color names instead of names to phone #s, or we could map numbers to images.

### Example:

Rhodes College

colors = {0: 'white', 1: 'black', 2: 'pink', 3: 'purple', 4: 'red', 5: 'blue', 6: 'green', 7: 'orange', 8: 'yellow'}

images = {0: 'blank.gif', 1: 'a.gif', 2: 'b.gif', 3: 'c.gif', 4: 'd.gif', 5: 'e.gif', 6: 'f.gif', 7: 'g.gif', 8: 'h.gif'}

Rhodes\_College

Getting the Number of Elements and Mixing Data Types

- Len function: used to obtain number of elements in a dictionary
- Keys must be immutable objects, but associated values can be any type of object
  - One dictionary can include keys of several different immutable types
- · Values stored in a single dictionary can be of different types



12

# Creating an Empty Dictionary and Using for Loop to Iterate Over a Dictionary

• To create an empty dictionary:

- Use { }
- Use built-in function dict ()
- Elements can be added to the dictionary as program executes
- Use a for loop to iterate over a dictionary
  - General format: for key in dictionary:

Phodee	College
EVINUARS	Concyc

Rhodes College

13

# Some Dictionary Methods get method: gets a value associated with specified key from the dictionary. format: dictionary.get(key, default) default is returned if key is not found Alternative to [] operator Cannot raise KeyError exception Asys method: returns all the dictionaries keys as a sequence format: dictionary.keys()

# Some Dictionary Methods (cont'd.)

- <u>values method</u>: returns all the dictionaries values as a sequence
  - Format: dictionary.values()
  - Use a for loop to iterate over the values



Dictionary	Methods
------------	---------

Method	Description
clear	Clears the contents of a dictionary.
get	Gets the value associated with a specified key. If the key is not found, the method does not raise an exception. Instead, it returns a default value.
items	Returns all the keys in a dictionary and their associated values as a sequence of tuples.
keys	Returns all the keys in a dictionary as a sequence of tuples.
рор	Returns the value associated with a specified key and removes that key-value pair from the dictionary. If the key is not found, the method returns a default value.
popitem	Returns a randomly selected key-value pair as a tuple from the dictionary and removes that key-value pair from the dictionary.
values	Returns all the values in the dictionary as a sequence of tuples.





