**USING DICTIONARIES**

**Using Dictionaries**

Now that we know how to create a dictionary, we can start using already created dictionaries to solve problems.

In this lesson, you’ll learn how to:

* Use a key to get a value from a dictionary
* Check for existence of keys
* Iterate through keys and values in dictionaries

**Get A Key**

Once you have a dictionary, you can access the values in it by providing the key. For example, let’s imagine we have a dictionary that maps buildings to their heights, in meters:

building\_heights = {"Burj Khalifa": 828, "Shanghai Tower": 632, "Abraj Al Bait": 601, "Ping An": 599, "Lotte World Tower": 554.5, "One World Trade": 541.3}

Then we can access the data in it like this:

>>> building\_heights["Burj Khalifa"]  
828  
>>> building\_heights["Ping An"]  
599

**Instructions**

**1.**

We have provided a dictionary that maps the elements of astrology to the zodiac signs. Print out the list of zodiac signs associated with the "earth" element.

Checkpoint 2 Passed

**2.**

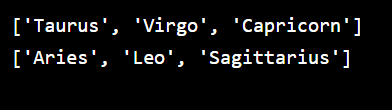
Print out the list of the "fire" signs.

**script.py**

zodiac\_elements = {"water": ["Cancer", "Scorpio", "Pisces"], "fire": ["Aries", "Leo", "Sagittarius"], "earth": ["Taurus", "Virgo", "Capricorn"], "air":["Gemini", "Libra", "Aquarius"]}

print(zodiac\_elements["earth"])

print(zodiac\_elements["fire"])

****

**Get an Invalid Key**

Let’s say we have our dictionary of building heights from the last exercise:

building\_heights = {"Burj Khalifa": 828, "Shanghai Tower": 632, "Abraj Al Bait": 601, "Ping An": 599, "Lotte World Tower": 554.5, "One World Trade": 541.3}

What if we wanted to know the height of the Landmark 81 in Ho Chi Minh City? We could try:

print(building\_heights["Landmark 81"])

But "Landmark 81" does not exist as a key in the building\_heights dictionary! So this will throw a KeyError:

KeyError: 'Landmark 81'

One way to avoid this error is to first check if the key exists in the dictionary:

key\_to\_check = "Landmark 81"  
  
if key\_to\_check in building\_heights:  
  print(building\_heights["Landmark 81"])

This will not throw an error, because key\_to\_check in building\_heights will return False, and so we never try to access the key.

**Instructions**

**1.**

Review the code in the editor and predict what the output will be. Run the code to see if you are correct!

Checkpoint 2 Passed

**2.**

Because "energy" is not a key in zodiac\_elements, a KeyError is thrown in the terminal!

Using an if statement, check if "energy" is a key in zodiac\_elements. Nest the existing print() statement within the if statement so that it will only execute if "energy" is a key.

Run your code again. This time, there should be no errors in the terminal!

Checkpoint 3 Passed

**3.**

Add the key "energy" to the zodiac\_elements. It should map to a value of "Not a Zodiac element". Run the code. Since "energy" is now a key, its value prints to the terminal!

**script.py**

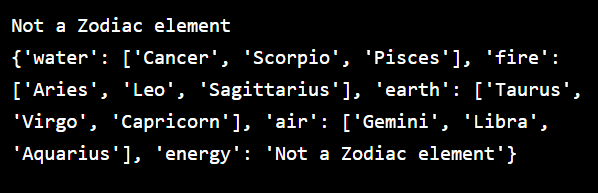
zodiac\_elements = {"water": ["Cancer", "Scorpio", "Pisces"], "fire": ["Aries", "Leo", "Sagittarius"], "earth": ["Taurus", "Virgo", "Capricorn"], "air":["Gemini", "Libra", "Aquarius"]}

zodiac\_elements["energy"] = "Not a Zodiac element"

if "energy" in zodiac\_elements:

  print(zodiac\_elements["energy"])

print(zodiac\_elements)

****