

HackerFrogs Afterschool

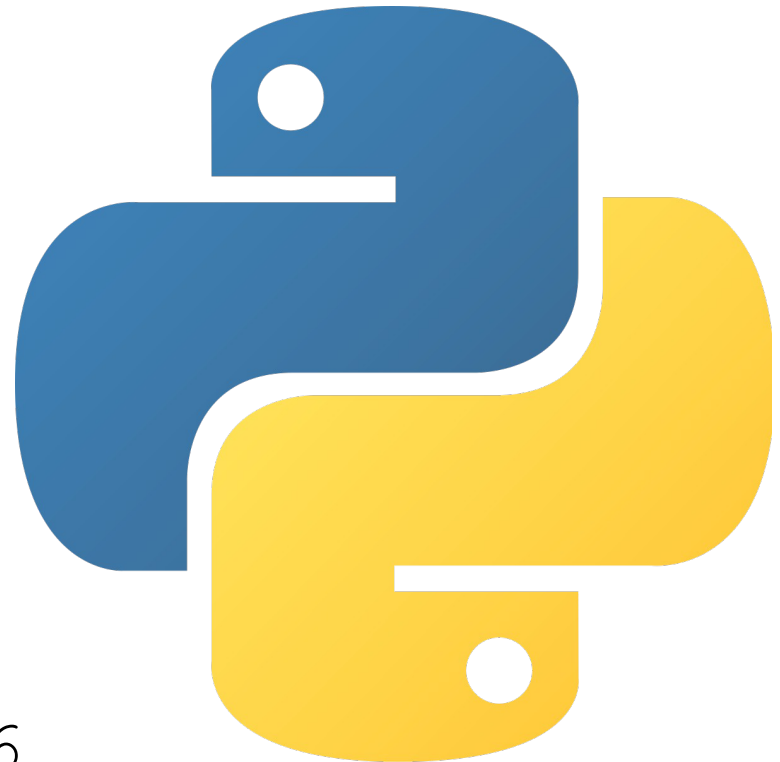
Python Programming Basics: Part 7

Class:
Programming (Python)

Workshop Number:
AS-PRO-PY-07

Document Version:
1.2

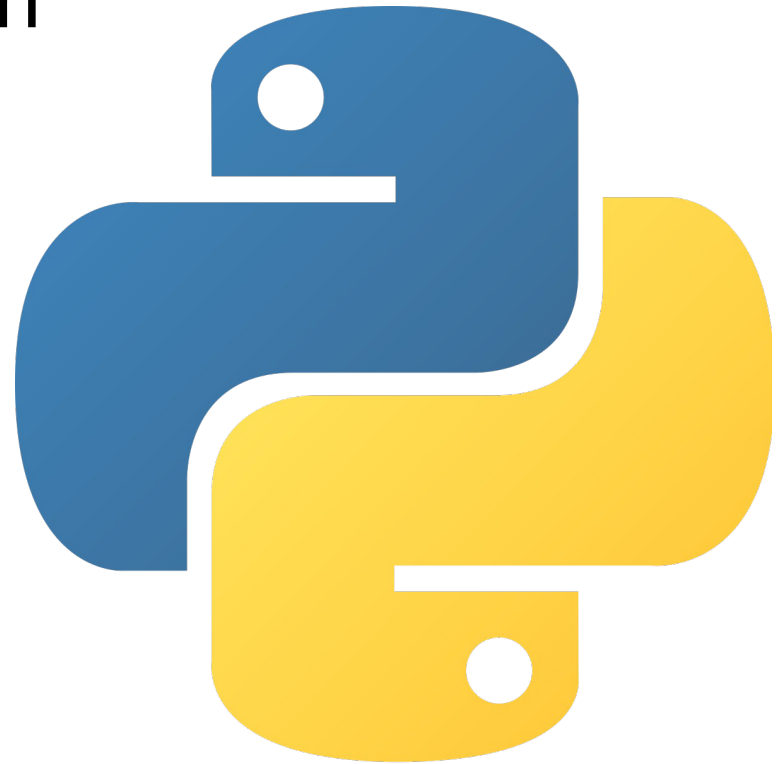
Special Requirements:
Completion of AS-PRO-PY-06



What We Learned Before

This workshop is the seventh class for intro Python programming.

During our last workshop, we learned about a few programming concepts through Python, including the following:



Classes & Objects

```
print(type(1), type('word'))
```

```
<class 'int'> <class 'str'>
```

Python is an object-oriented programming language, and as such all data in Python code are considered objects.

Classes & Objects

```
print(type(1), type('word'))
```

```
<class 'int'> <class 'str'>
```

All objects belong to a class, and an object's class indicates which built-in functions, methods, and variables it has.

This Workshop's Topics

New topics for this session:

- Dictionaries

Dictionaries

```
country_capitals = {"England": "London", "Canada": "Ottawa"}
```

Dictionaries in Python are similar to other container objects, such as lists, except that data stored in dictionaries are pairs of keys / values, as opposed to individual items.

Dictionaries

```
country_capitals = {"England":  
    "London", "Canada": "Ottawa"}
```

Dictionaries can be identified by **a pair of curly braces**, within which are pairs of **keys / values**, each pair separated by **commas**. The key and value are themselves separated by a **colon**.

Dictionaries

```
country_capitals = {  
    "England": "London",  
    "Canada": "Ottawa"  
}
```

Dictionaries can also be initialized in the format above, which is easier-to-read.

Iterating Over Dictionaries

```
country_capitals = {"England": "London", "Canada": "Ottawa"}  
  
for i, j in country_capitals.items():  
    print(i, j)
```

```
England London  
Canada Ottawa
```

Items in a dictionary can be iterated over using a for loop, but we must use the **items** method, which renders the dictionary as a list with tuples as the content, which hold the key / value pairs.

Adding Keys / Values

```
friend_city = {'Carl': 'Toronto', 'Rachel': 'New York'}  
friend_city['Bobby'] = 'Texas'  
print(friend_city)
```

```
{'Carl': 'Toronto', 'Rachel':  
'New York', 'Bobby': 'Texas'}
```

To add entries to an existing dictionary, simply define the dictionary's index with the name of the key, then supply the value on the other side of the equals sign, as shown above.

Removing Keys / Values

```
friend_city = {'Carl': 'Toronto', 'Rachel': 'New York'}  
del friend_city['Carl']  
print(friend_city)
```

```
{'Rachel': 'New York'}
```

To remove entries from a dictionary, there are two ways. The first way is to use the **del** keyword along with the dictionary key as the index name.

Removing Keys / Values

```
friend_city = {'Carl': 'Toronto', 'Rachel': 'New York'}  
friend_city.pop('Rachel')  
print(friend_city)
```

```
{'Carl': 'Toronto'}
```

The other way of removing entries from dictionaries is to the **pop** dictionary method, as illustrated above.

Dictionary Keys are Immutable

```
friend_city = {'Carl': 'Toronto', 'Rachel': 'New York'}  
friend_city['Carl'] = 'Kelly'  
print(friend_city)
```

```
{'Carl': 'Kelly', 'Rachel': 'New York'}
```

The names of keys cannot be changed after they've been created. One the key's value can be changed.

Dictionaries Exercise

Let's practice using Python dictionaries at the following URL:

<https://learnpython.org/en/Dictionaries>

Workshop Review Exercise

Before we finish, let's write a program which features many of the concepts we learned in this workshop.

A game developer wants to write a program that keeps track of a game character's statistics, as well as function that adjusts one of those statistics.

Dictionaries Quiz - Q1

Which of the following statements about Python dictionaries is true?

- A) A dictionary is a mutable, ordered collection of items.
- B) A dictionary can have duplicate keys.
- C) Dictionary keys must be immutable and unique.
- D) Dictionary values must be unique.

Dictionaries Quiz - Q1

Which of the following statements about Python dictionaries is true?

- A) A dictionary is a mutable, ordered collection of items.
- B) A dictionary can have duplicate keys.
- C) Dictionary keys must be immutable and unique.
- D) Dictionary values must be unique.

Dictionaries Quiz - Q2

Which of the following methods could be used to delete a key-value pair from a dictionary?

- A) the del keyword
- B) the pop method
- C) A + B
- D) None of the above

Dictionaries Quiz - Q2

Which of the following methods could be used to delete a key-value pair from a dictionary?

- A) the del keyword
- B) the pop method
- C) A + B
- D) None of the above

Workshop Review Exercise

The program should have a function named “Damage” which updates the dictionary's Health key, decreasing it by 20 when the function is called.

The Damage function should be called, then the dictionary should be printed to the console, showing the newly updated Health key.

Summary



Let's review the programming concepts we learned in this workshop:

Dictionaries

```
country_capitals = {"England": "London", "Canada": "Ottawa"}
```

Dictionaries in Python are similar to other container objects, such as lists, except that data stored in dictionaries are pairs of keys / values, as opposed to individual items.

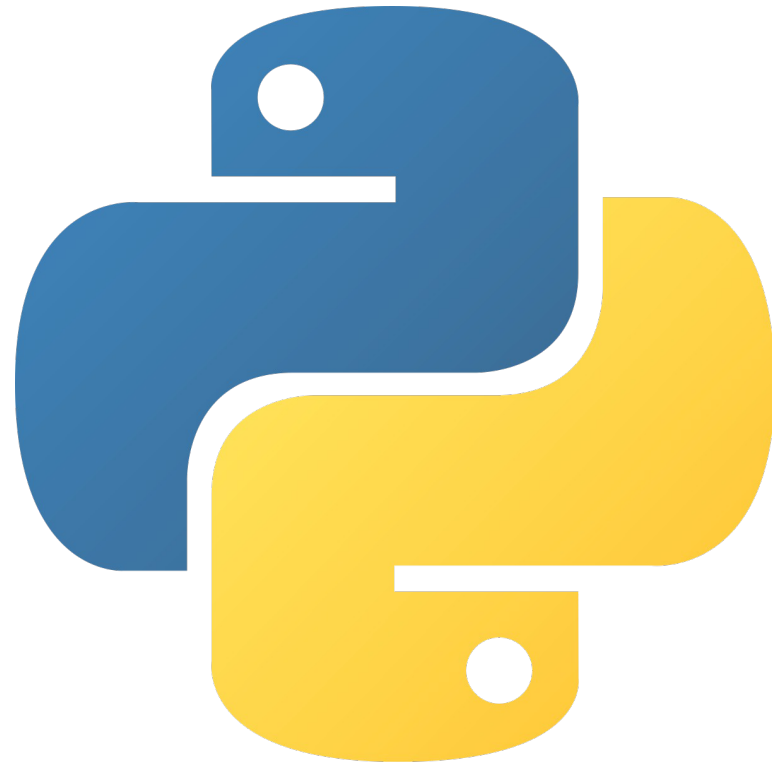
Dictionaries

```
john_vital_stats = {  
    "height": 170,  
    "weight": 90.2,  
    "age": 25,  
    "gender": "male"  
}
```

Dictionaries are useful for recording different key / value pairs that all pertain to one subject, or one common value across a number of subjects.

What's Next?

This is the end of the HackerFrogs AfterSchool intro to Python programming classes, but now that we know the basics we can start learning a lot more about coding in this extremely popular programming language!



Extra Credit

Looking for more study material on this workshop's topics?

See this video's description for links to supplemental documents and exercises!



Until Next Time, HackerFrogs!

