Intro to Python- Mr. Powell

Use this to check your answers or help you complete your work.

## Answers to Exercises for Chapter 5

## Exercises 5.1

1. a. 7
2. 7
3. 3
4. [5, 3]
5. False
6. [5, 3, 7, 2, 10, 5]
7. (5, 3, 7)
8. a. data[0] = -data[0]

b. data.append(10)

c. data.insert(2, 22)

d. data.pop(1)

e. data.extend(newData)

f. if 7 in data:

position = data.index(7)

else:

position = -1

g. data.sort()

1. A mutator method modifies the internal state of an object. A mutator method returns the value **None** because its purpose is just to modify the state of an object.
2. sum = 0

for item in data:

sum += data

1. for item in data:

if item != 0:

result.append(item)

1. for index in range(len(data)):

data[index] = abs(data[index])

1. Aliasing occurs when two or more variables reference the same object. When this object is a large data structure, such as a list or dictionary, changes to the structure through one variable reference will have an impact on the other references. These side effects might not be anticipated by programmers who are using the other references. To protect against a side effect, each variable should refer to a copy of the object that has the same internal structure. The cost of this maneuver is the extra memory needed for the copies of the object.
2. Two objects are structurally equivalent if they are distinct objects and their contents are equal. Object identity involves just one object and possibly several references to it.

## Exercises 5.2

1. A parameter requires the function’s caller to pass data to the function for its use when it is called. The **return** statement allows the caller to receive a value computed by the function. Both are mechanisms for sharing information between a function and its caller.
2. def even(x):

return x % 2 == 0

1. def odd(x):

return not even(x)

1. def sum(low, high):

result = 0

for number in range(low, high + 1):

result += number

return result

1. A **main** function provides an entry point for a program and structures its top-level tasks.

## Exercises 5.3

1. Three examples of dictionary-like objects are encyclopedias, dictionaries, and cookbooks.
2. a. 35

b. None

c. 2

d. ['b', 'a']

e. [20, 35]

f. 20

g. {'a':35}

1. a. data["b"] = -data["b"]

b. data["c"] = 40

c. data.pop("b", None)

d. keys = list(data.keys())

keys.sort()

for key in keys:

print(key)