**Dictionaries and Sets**

* Introduction

6.1 Q1. Which of the following statements is false?

a. Lists and tuples are built-in sequence collections.

b. Dictionaries, strings and sets are built-in non-sequence collections.

c. A dictionary is an unordered collection which stores key–value pairs that map immutable keys to values, just as a conventional dictionary maps words to definitions.

d. A set is an unordered collection of unique immutable elements.

Answer: b. Dictionaries, strings and sets are built-in non-sequence collections. Actually, Dictionaries and sets are built-in non-sequence collections. Strings are built-in sequence collections.

* Dictionaries

6.2 Q1. Which of the following statements is false?

a. A dictionary associates keys with values. Each key maps to a specific value.

b. A dictionary’s keys must be mutable (such as strings, numbers or tuples) and unique (that is, no duplicates).

c. Multiple keys can have the same value, such as two different inventory codes that have the same quantity in stock.

d. All of the above are true.

Answer: b. A dictionary’s keys must be mutable (such as strings, numbers or tuples) and unique (that is, no duplicates). Actually, a dictionary’s keys must be *immutable* (such as strings, numbers or tuples) and unique (that is, no duplicates).

* Creating a Dictionary

6.2 Q2. Which of the following statements is false?

a. You can create a dictionary by enclosing in curly braces, {}, a comma-separated list of key–value pairs, each of the form key: value.

b. You can create an empty dictionary with {}.

c. The following code creates a dictionary with the country-name keys 'Finland', 'South Africa' and 'Nepal' and their corresponding Internet country code values 'fi', 'za' and 'np':

country\_codes = {'Finland', 'fi', 'South Africa', 'za',   
 'Nepal', 'np'}

d. When you output a dictionary, its comma-separated list of key–value pairs is always enclosed in curly braces.

Answer: c. The following code creates a dictionary with the country-name keys 'Finland', 'South Africa' and 'Nepal' and their corresponding Internet country code values 'fi', 'za' and 'np':

country\_codes = {'Finland', 'fi', 'South Africa', 'za',   
 'Nepal', 'np'}

Actually, c. The following code creates a dictionary with the country-name keys 'Finland', 'South Africa' and 'Nepal' and their corresponding Internet country code values 'fi', 'za' and 'np':

country\_codes = {'Finland': 'fi', 'South Africa': 'za',  
 'Nepal': 'np'}

6.2 Q3: Which of the following statements is false?

a. The built-in function len returns the number of key–value pairs in a dictionary.

b. You can use a dictionary as a condition to determine if it’s empty—a non-empty diction-ary evaluates to True.

c. An empty dictionary evaluates to False.

d. Method clean deletes the dictionary’s key–value pairs:

Answer: d. Method clean deletes the dictionary’s key–value pairs. Actually, method clear deletes the dictionary’s key–value pairs.

* Iterating through a Dictionary

6.2 Q4. Which of the following statements is false?

a. The following dictionary maps month-name strings to int values representing the numbers of days in the corresponding month:

days\_per\_month = {'January': 31, 'February': 28, 'March': 31}

b. Multiple keys in a dictionary can have the same value.

c. Dictionaries are unordered.

d. The following for statement iterates through dictionary days\_per\_month’s key–value pairs. Dictionary method items returns each key–value pair as a tuple, which is unpacked into the target variables month and days:

for month, days in days\_per\_month.items():  
 print(f'{month} has {days} days')

Answer: b. Multiple keys in a dictionary can have the same key. Actually, multiple values in a dictionary can have the same value. Keys must be unique.

* Basic Dictionary Operations

6.2 Q5. Which of the following statements is false?

a. The following code creates the dictionary roman\_numerals, which maps roman numerals to their integer equivalents (the value for 'X' is intentionally wrong):

roman\_numerals = {'I': 1, 'II': 2, 'III': 3, 'V': 5, 'X': 100}

b. The following code gets the value associated with the key 'V':

roman\_numerals['V']

c. You can update a key’s associated value in an assignment statement, which we do here to replace the incorrect value associated with the key 'X' in Part (a):

roman\_numerals['X'] = 10

d. All of the above statements are true.

Answer: d. All of the above statements are true.

6.2 Q6: Which of the following statements is false?

a. Assigning a value to a nonexistent key inserts the key–value pair in the dictionary.

b. String keys are case insensitive.

c. Assigning a value to a nonexistent key could be a logic error.

d. All of the above statements are true.

Answer: b. String keys are case insensitive. Actually, string keys are **case sensitive**.

6.2 Q7: Which of the following statements is false?

a. You can delete a key–value pair from a dictionary with the del statement

b. You can remove a key–value pair with the dictionary method pop, which returns the value for the removed key.

c. The one argument version of dictionary method get, normally returns its argument’s corresponding value—if that key is not found, get raises a KeyError.

d. Operators in and not in can determine whether a dictionary contains a specified key.

Answer: c. The one argument version of dictionary method get, normally returns its argument’s corresponding value—if that key is not found, get raises a KeyError. Actually, if the key is not found, get returns None, rather than raising an exception.

* Dictionary Methods keys and values

6.2 Q8. Which of the following statements is false?

a. Dictionary methods keys and values can be used to iterate through only a dictionary’s keys or values, respectively.

b. Dictionary methods items, keys and values each return a view of a dictionary’s data.

c. When you iterate over a view, it “sees” a copy of the dictionary’s current contents.

d. You should not modify a dictionary while iterating through a view. According to the Python Standard Library documentation, either you’ll get a RuntimeError or the loop might not process all of the view’s values.

Answer: c. When you iterate over a view, it “sees” a copy of the dictionary’s current contents. Actually, when you iterate over a view, it “sees” the dictionary’s current contents—it does *not* have its own copy of the data.

* Dictionary Comparisons

No questions.

* Example: Dictionary of Student Grades

6.2 Q9. Which of the following statements is false?

a. The comparison operators == and != can be used to determine whether two dictionaries have identical or different contents.

b. An equals (==) comparison evaluates to True if both dictionaries have the same key–value pairs, regardless of the order in which those pairs were added to each dictionary.

c. In the following dictionary, the keys are lists of three integer student grades and the values are strings representing the students’ names.

grade\_book = {   
 'Susan': [92, 85, 100],   
 'Eduardo': [83, 95, 79],  
 'Azizi': [91, 89, 82],   
 'Pantipa': [97, 91, 92]   
}

d. All of the above statements are true.

Answer: c. In the following dictionary, the keys are lists of three integer student grades and the values are strings representing the students’ names:

grade\_book = {   
 'Susan': [92, 85, 100],   
 'Eduardo': [83, 95, 79],  
 'Azizi': [91, 89, 82],   
 'Pantipa': [97, 91, 92]   
}

Actually, the keys are the student name strings and the values are the lists of three integer grades per student.

* Example: Word Counts

6.2 Q10. Which of the following statements is false?

a. The following code builds a dictionary to count the number of occurrences of each word in a string named text—the dictionary’s keys will be the unique words, and its values will be integer counts of how many times each word appears in text:

word\_counts = {}  
  
for word in text.split():  
 if word in word\_counts:   
 word\_counts[word] += 1   
 else:  
 word\_counts[word] = 1

b. Breaking a string into words is known as tokenizing the string.

c. Python automatically concatenates strings separated by whitespace in parentheses.

d. All of the above statements are true.

Answer: d. All of the above statements are *true*.

6.2 Q11: Which of the following statements about the following code is false?

word\_counts = {}  
  
for word in text.split():  
 if word in word\_counts:   
 word\_counts[word] += 1   
 else:  
 word\_counts[word] = 1

a. The expression text.split() tokenizes text by calling string method split, which separates the words using the method’s delimiter string argument—if you do not provide an argument, split uses a space.

b. Method split returns a list of tokens (that is, the words in text).

c. The expression word\_counts[word] += 1 inserts a new key-value pair in the dictionary.

d. All of the above statements are true.

Answer: c. The expression word\_counts[word] += 1 inserts a new key-value pair in the dictionary. Actually, this expression adds 1 to the value of an existing key-value pair in the dictionary.

* Dictionary Method update

6.2 Q12. Which of the following statements is false?

a. The following update call receives a dictionary key–value pair to insert or update:

country\_codes.update({'South Africa': 'za'})

b. Method update can convert keyword arguments into key–value pairs to insert—the following call converts the parameter name Australia into the string key 'Australia' and associates the incorrect value 'ar' with that key:

country\_codes.update(Australia='ar')

c. The snippet in Part (b) provided an incorrect country code for Australia. The following code corrects this by using a keyword argument to update the value associated with 'Australia':

country\_codes.update(Australia='au')

d. All of the above statements are true.

Answer: d. All of the above statements are *true*.

* Dictionary Comprehensions

6.2 Q13. Which of the following statements is false?

a. Dictionary comprehensions provide a convenient notation for quickly generating dic-tionaries, often by mapping one dictionary to another.

b. The following code, which has a dictionary with unique values, reverses the key–value pairs:

months = {'January': 1, 'February': 2, 'March': 3}  
months2 = {number: name for name, number in months.items()}

c. If the values in Part (b) were not unique, then the code would generate a ValueError.

d. All of the above statements are true.

Answer: c. If the values in part b) were not unique, then the code would generate a ValueError. Actually, it’s not an error. The non-unique values would become non-unique keys. That’s not allowed, but it’s not a problem because each non-unique key–value pair would simply update that key’s value in the dictionary to the new value. Thus, the new dictionary would have fewer key-value pairs.

* Sets

6.3 Q1. Which of the following statements is false?

a. A set is an unordered collection of unique values.

b. Sets may contain only immutable objects, like strings, ints, floats and tuples that contain only immutable elements.

c. Sets are iterable, so they are sequences and they support indexing and slicing with square brackets, [].

d. Dictionaries do not support slicing.

Answer: c. Sets are iterable, so they are sequences and they support indexing and slicing with square brackets, []. Actually, though sets are iterable, they are *not* sequences and do *not* support indexing and slicing.

6.3 Q2. Which of the following statements is false?

a. The following code creates a set of strings named colors:

colors = {'red', 'orange', 'yellow', 'green', 'red', 'blue'}

b. Duplicate elimination is automatic when creating a set.

c. Sets are ordered, so you can write code that depends on the order of their elements.

d. All of the above statements are true.

Answer: c. Sets are ordered, so you can write code that depends on the order of their elements. Actually, sets are unordered, so you should not write code that depends on the order of their elements.

6.3 Q3. Which of the following statements is false?

a. You can determine the number of items in a set with the built-in len function.

b. You can check whether a set contains a particular value using the in and not in operators.

c. Sets are not iterable, so you cannot process each set element with a for loop.

d. Sets are unordered, so there’s no significance to the iteration order.

Answer: c. Sets are not iterable, so you cannot process each set element with a for loop. Actually, sets *are* iterable, so you *can* process each set element with a for loop.

6.3 Q4. Which of the following would be displayed where we wrote ??? by Out[3]?

In [1]: numbers = list(range(10)) + list(range(5))  
  
In [2]: numbers  
Out[2]: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 0, 1, 2, 3, 4]  
  
In [3]: set(numbers)  
Out[3]: ???

a. [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]

b. (0, 1, 2, 3, 4, 5, 6, 7, 8, 9)

c. {0, 1, 2, 3, 4, 5, 6, 7, 8, 9}

d. {[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]}

Answer: c. {0, 1, 2, 3, 4, 5, 6, 7, 8, 9}

6.3 Q5. Which of the following statements is false?

a. Sets are immutable, so sets can have other sets as elements.

b. A frozenset is an immutable set—it cannot be modified after you create it, so a set can contain frozensets as elements.

c. The built-in function frozenset creates a frozenset from any iterable.

d. All of the above statements are true.

Answer: a. Sets are immutable, so sets can have other sets as elements. Actually, sets are mutable—you can add and remove elements, but set elements must be immutable. Therefore, a set cannot have other sets as elements.

* **Comparing Sets**

6.2 Q6. Which of the following statements is false?

a. The <= operator tests whether the set to its left is an improper subset of the one to its right—that is, all the elements in the left operand are in the right operand, and the sets might be equal.

b. You may also check for an improper subset with the set method issubset.

c. You may check for an improper superset with the set method issuperset.

d. The argument to issubset or issuperset must be a set.

Answer: d. The argument to issubset or issuperset must be a set. Actually, the argument to issubset or issuperset can be *any* iterable. When either of these methods receives a non-set iterable argument, it first converts the iterable to a set, then performs the operation.

* **Mathematical Set Operations**

6.3 Q7. Which of the following statements is false?

a. The union of two sets is a set consisting of all the unique elements from both sets.

b. You can calculate the union with the | operator or with the set type’s union method.

c. The operands of the binary set operators, like |, must both be any iterable.

d. The corresponding set methods may receive any iterable object as an argument—we passed a list.

Answer: c. The operands of the binary set operators, like |, must both be any iterable. Actually, the operands of the binary set operators, like |, must both be sets.

6.3 Q8: Which of the following statements is false?

a. You can calculate the intersection with the & operator or with the set type’s intersection method.

b. The difference between two sets is a set consisting of the elements in the left operand that are not in the right operand.

c. You can calculate the difference with the - operator or with the set type’s difference method.

d. All of the above statements are true.

Answer: d. All of the above statements are *true*.

6.3 Q9 Two sets are disjoint if they do not have any common elements. You can determine this with the set type’s isdisjoint method. What values are actually displayed where we’ve inserted ??? in Out[1] and Out[2]:

In [1]: {1, 3, 5}.isdisjoint({2, 4, 6})  
Out[1]: ???  
  
In [2]: {1, 3, 5}.isdisjoint({4, 6, 1})  
Out[2]: ???

a. False, False

b. False, True

c. True, False

d. True, True

Answer: c. True, False

* Mutable Set Operators and Methods

6.3 Q10: Which of the following statements is false?

a. Like operator |, union augmented assignment |= performs a set union operation, but |= doesn’t modify its left operand.

b. The set type’s update method performs a union operation modifying the set on which it’s called—the argument can be any iterable.

c. Other mutable set methods are intersection augmented assignment &=, difference augmented assignment -= and symmetric difference augmented assignment ^=. Their corresponding methods with iterable arguments are intersection\_update, difference\_update and symmetric\_difference\_update.

d. All of the above statements are true.

Answer: a. Like operator |, union augmented assignment |= performs a set union operation, but |= doesn’t modify its left operand. Actually, |= does modify its left operand to store the union of the sets.

6.3 Q11: Which of the following statements is false?

a. Set method add inserts its argument if the argument is not already in the set; otherwise, the set remains unchanged.

b. Set method remove removes its argument from a set—a KeyError occurs if the value is not in the set.

c. Method discard also removes its argument from a set but does not cause an exception if the value is not in the set.

d. You can remove the first element of a set with pop.

Answer: d. You can remove the first element of a set with pop Actually, you can remove an *arbitrary* set element and return it with pop—sets are unordered, so you do not know which element will be returned.

* Set Comprehensions

No questions.

* Intro to Data Science: Dynamic Visualizations

6.4 Q1. Which of the following statements is false?

a. If the random module’s randrange function indeed produces integers at random, then every number in the specified range has an equal probability (or likelihood) of being chosen each time the function is called.

b. For a six-sided die, each value 1 through 6 should occur about one-sixth of the time, so the probability of any one of these values occurring is 1/6th or about 16.667%.

c. The more die rolls we attempt, the closer each die value’s percentage of the total rolls gets to 16.667%. This is a manifestation of the Moore’s Law.

d. Each of the above statements is true.

Answer: c. The more die rolls we attempt, the closer each die value’s percentage of the total rolls gets to 16.667%. This is a manifestation of the Moore’s Law. Actually, this is a manifestation of the Law of Large Numbers.

* How Dynamic Visualization Works

6.4 Q2. Which of the following statements is false?

a. Matplotlib’s FuncAnimation drives a frame-by-frame animation.

b. Each animation frame specifies everything that should change during one plot update. Stringing together many of these updates over time creates the animation effect.

c. FuncAnimation decides what each frame displays.

d. Generally, displaying more frames-per-second yields smoother animation.

Answer: c. FuncAnimation decides what each frame displays. Actually, you decide what each frame displays with a function you define and pass as an argument when you create a FuncAnimation.

6.4 Q3. Which of the following statements is false?

a. For each animation frame, FuncAnimation calls a function that you define to specify how to change the plot.

b. Displaying animation frames to the screen is relatively fast compared to the die rolls, which occur at the computer’s CPU speeds.

c If we roll only one die per animation frame, we won’t be able to run a large number of rolls in a reasonable amount of time. You can increase the execution speed of the simulation by rolling the die more times per animation frame.

d. All of the above statements are true.

Answer: b. Displaying animation frames to the screen is relatively fast compared to the die rolls, which occur at the computer’s CPU speeds. Actually, displaying animation frames to the screen is a relatively slow *input/output-bound* operation compared to the die rolls, which occur at the computer’s super fast CPU speeds.

* Implementing a Dynamic Visualization

6.4 Q4. Which of the following statements about FuncAnimation is false?

a. FuncAnimation has two required arguments—the Figure object in which to display the animation, and a function to call once per animation frame.

b. The argument interval is the number of milliseconds between animation frames (the default is 200).

c. The argument fargs (short for “function arguments”) is a tuple of other arguments to pass to the function you specify in FuncAnimation’s second argument.

d. Each of the above statements is true.

Answer: d. Each of the above statements is *true*.