Sets and Dictionaries

Exercises

Week 7

Specify two ways in which a Set varies from a List.

Answer: Immutability and Order of elements

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Write a Python statement that uses the set() constructor to produce the same Set as the

following -

languages = { "C++", "Java", "C#", "PHP", "JavaScript" }

Answer:

languages = set(("C++", "Java", "C#", "PHP", "JavaScript")))

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Is a Set mutable or immutable?

Answer: You can't change the content of a set directly using the assignment operator,

you can still modify a set using methods

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Why does a Set not support indexing and slicing type operations?

Answer: Because the order of elements in a set is undefined and cannot be guaranteed we cannot use

indexing in it and about slicing sets in python are not sequence types.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Why is a frozenset() different from a regular set?

Answer: When you require an immutable set that can be used as a dictionary key, frozensets are

beneficial, whereas sets are handy when you need a changeable set that can have elements added,

deleted, or updated.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

How many elements would exist in the following set?

names = set("John", "Eric", "Terry", "Michael", "Graham", "Terry")

Answer: Five since Terry is repeated twice and in set you cant add same value twice.

And how many elements would exist in this set?

vowels = set("aeiou")

Answer: Five

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

What is the name given to the following type of expression which can be used to

programmatically populate a set?

chars = {chr(n) for n in range(32, 128)}

Answer: The name given to this type of expression is a set comprehension.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

What operator can be used to calculate the intersection (common elements) between two

sets?

Answer: "&" this intersection operator can be used to calculate the intersection between two sets.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

What operator can be used to calculate the difference between two sets?

Answer: "-" this difference operator can be used to calculate difference between the two sets.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

What would be the result of each of the following expressions?

{ "x", "y", "z" } < { "z" , "u", "t", "y", "w", "x" }

Answer: True

{ "x", "y", "z" } < { "z", "y", "x" }

Answer: False

{ "x", "y", "z" } <= { "y", "z", "x" }

Answer: True

{ "x" } > { "x" }

Answer: False

{ "x", "y" } > { "x" }

Answer: True

{ "x", "y" } == { "y", "x" }

Answer: True

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Write a Python statement that uses a method to perform the equivalent of the following

operation -

languages = languages | { "Python" }

Answer: To add an element on existing set you can also use add method.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Do the elements which are placed into a set always remain in the same position?

Answer: Yes

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Is the following operation a mutator or an accessor?

languages &= oo\_languages

Answer: Mutator

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

What term is often used to refer to each pair of elements stored within a dictionary?

Answer: keys and values

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Is it possible for a dictionary to have more than one key with the same value?

Answer: Yes it is possible for a dictionary to have more than one key with the same value.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Is it possible for a dictionary to have the same value appear more than once?

Answer: Yes, it is possible for a dictionary to have the same value appear more than once

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Is a Dictionary mutable or immutable?

Answer: Dictionary is a mutable data type.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Are the key values within a dictionary mutable or immutable?

Answer: They are immutable within a dictionary.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

How many elements exist in the following dictionary?

stock = {"apple":10, "banana":15, "orange":11}

Answer: Three

And, what is the data-type of the keys?

Answer: In above dict the data type of keys are string.

And, what output would be displayed by executing the following statement -

print(stock["banana"])

Answer: It will print 15 because the value assigned to it is 15

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Write a Python statement that uses the dictionary() constructor to produce the same

dictionary as the following -

lang\_gen = { "Java":3, "Assembly":2, "Machine Code":1 }

Answer: lang\_gen = dict([("Java", 3), ("Assembly", 2), ("Machine Code", 1)])

Now write a simple expression that tests whether the word "Assembly" is a member of the

dictionary.

Answer: print("Assembly" in lang\_gen)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Write some Python code that uses a for statement to iterate over a dictionary called

module\_stats and print only its values (i.e. do not output any keys) -

Answer:

module\_stats = {"Math": 95, "English": 88, "Physics": 92}

for value in module\_stats.values():

print(value)

Now write another loop which prints the only the keys -

Answer:

module\_stats = {"Math": 95, "English": 88, "Physics": 92}

for key in module\_stats.keys():

print(key)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Is it possible to construct a dictionary using a comprehension style expression, as

supported by lists and sets?

Answer: Yes, it is possible to construct a dictionary using a comprehension style expression.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

When a Dictionary type value is being passed as an argument to a function, what characters

can be used as a prefix to force the dictionary to be unpacked prior to the call being made?

Answer: The \*\* character can be used as a prefix to force the dictionary to be unpacked prior to

the call being made.