Learning Quiz 20: Sets

Due Oct 30 at 1pm **Points** 5 **Questions** 5 **Available** until Dec 4 at 11:59pm **Time Limit** None
Allowed Attempts Unlimited

Instructions

Prior to completing this quiz, be sure to read:

- Section 6.2: Sets (p. 177-181)

Please also go over Practice Problem 6.6 in the textbook (solutions at the end of the chapter) before attempting this quiz.

This quiz was created for learning purposes. You may attempt this quiz as many times as you would like. The highest score *prior to the deadline* will count towards the final course grade. No late submissions will be accepted.

Take the Quiz Again

Attempt History

	Attempt	Time	Score
LATEST	Attempt 1	9 minutes	4 out of 5

Score for this attempt: **4** out of 5
Submitted Nov 1 at 1:03pm
This attempt took 9 minutes.

Question 1

1 / 1 pts

Another built-in Python container type is the **set**. Sets are used to store an unordered collection of items.

To create a set, use curly-brackets (and not semi-colons):

```
myset = {1,2,3,4}
```

Fill in the blanks with one special character per blank so that thisset is a set.

thisset = { 'apple', 'orange', 'banana', 'pear' }

Answer 1:

Correct!

{

Answer 2:

Correct!

```
}
```

Question 2

1 / 1 pts

Note that both dictionaries and sets use curly-brackets. What if we want to create an empty set or dictionary?

We can create an empty set or dictionary by using set() and dict(), respectively.

```
myset = set()
mydict = dict()
```

What class does thisvar belong to in the code below? Hint: use type() to see the class type.

```
thisvar = {}
```

- set
- Correct! dictionary
- tuple
- list

Question 3

0 / 1 pts

In sets, we cannot have duplicates. If we accidentally create a set with duplicates, the duplicate(s) will be ignored (and you'll only have one copy).

```
myset = {'red', 'black', 'blue', 'white', 'purple', 'green', 'black', 'yellow'}
print(myset)  ## 'black' only appears once even though we had two of them
```

How many items are in the following set (theset)? (Hint, you can use len(theset) to help you determine the answer)

```
theset = {23, 24, 25, 27, 23, 26, 24, 24, 27, 21, 28, 29, 30, 30}
```

You Answered: 7

Correct Answers: 9 (with margin: 0)

Question 4

1 / 1 pts

Table 6.3 on page 180 of the textbook contains a list of operators for sets.

```
set1 = seta & setb     ## intersection
set2 = seta | setb     ## union
set3 = seta - setb     ## what seta has, but setb doesn't
set4 = seta ^ setb     ## what is only in one set and not the other
```

Consider the following sets *friends* and *lions*.

```
friends = {'Simba', 'Pumbaa', 'Timone'}
lions = {'Mufasa', 'Simba', 'Nala'}
```

Match the code with the resulting set. (Note, set order doesn't matter because sets do not have order!)

Correct! **friends & lions** {'Simba'}

Correct! **friends | lions** {'Simba', 'Timone', 'Mufasa', 'l

Correct! **friends - lions** {'Timone', 'Pumbaa'}

Correct! **lions - friends** {'Mufasa', 'Nala'}

Correct! **friends ^ lions** {'Timone', 'Pumbaa', 'Mufasa'

Other Incorrect Match Options:

- {'Simba', 'Pumbaa', 'Timone'}
- {'Mufasa', 'Simba', 'Nala'}

Question 5

1 / 1 pts

Some commonly used methods for sets include add(), remove() and clear(). clear() removes everything from the set.

Consider the following code:

```
lions = {'Mufasa', 'Simba', 'Nala'}
lions.remove('Mufasa')
lions.add('Scar')
lions.add('Simba')
lions.remove('Simba')
```

What is stored in lions at the end of the code above?

{'Mufasa', 'Simba', 'Nala'}

{'Mufasa', 'Scar', 'Nala'}

{'Scar', 'Simba', 'Nala'}

Correct!

{'Nala', 'Scar'}

{'Scar', 'Simba', 'Nala', 'Simba'}

Quiz Score: **4** out of 5