Working with Lists

Recall that a variable can hold multiple values in the form of a list. The values are separated by commas and wrapped in square brackets.

Lists have ***methods*** (built-in functions) that can be called using dot notation. For example, to add a new element to the end of a list, we can use the **append** method.

Python code	Shell
`rolls = [4, 6, 6, 2, 6]`	
`len(rolls)`	5
`print(rolls[5])`	Index Error
`rolls.append(1)`	
`print(rolls)`	[4,6,6,2,6,1]
`print(rolls[5])`	1
`lucky.append(1)`	'lucky' not defined
`lucky = []`	
`print(lucky[0])`	Index Error
`lucky.append(5)`	
`print(lucky)`	[5]
`print(lucky[0])`	5
`rolls.count(6)`	3
`rolls.remove(6)`	
`print(rolls)`	[4,6,2,6,1]

1. What is the result of calling the append method on a list?

Adds the argument within append( ) to the list.

2. What must be defined prior to using a method like append?

The list itself, like rolls or lucky.

3. Explain why two lines caused an IndexError.

The first error was because there was no index 5, only 0-4. The second error was because the list used was empty, meaning no index, therefore, no element at index 0.

4. What is the result of calling the remove method on a list?

Removes the first occurance of the item being removed

5. Give one example of a list method that requires an argument and one that does not.

Needs argument: rolls.count(6)

No argument: len(rolls)

6. Describe the syntax similarities and differences between using a list method like append and Python built-in functions like `print`.

Both require something to be passed within their parentheses. However, append() requires the list in context to be typed first with a period separating them, whereas print can stand on its own.

7. Complete the function below (two lines are missing). It should prompt the user for numbers and build a list by adding one number at a time to the end of the list. The loop terminates when the user inputs the number 0.

```
def input_numbers():
    x = 1

    numbers = []
    x = int ( input ("Enter the first number: "))
    while x != 0:

        numbers.append(x)
        x = int ( input ("Enter the next number: "))

    return numbers
```