Python's range() Parameters

The range() function has two sets of parameters, as follows:

range(stop)

* stop: Number of integers (whole numbers) to generate, starting from zero. eg. range(3) == [0, 1, 2].

range([start], stop[, step])

* start: Starting number of the sequence.
* stop: Generate numbers up to, but not including this number.
* step: Difference between each number in the sequence.

Note that:

* All parameters must be integers.
* All parameters can be positive or negative.
* range() (and Python in general) is 0-index based, meaning list indexes start at 0, not 1. eg. The syntax to access the first element of a list is mylist[0]. Therefore the last integer generated by range() is up to, but not including, stop. For example range(0, 5) generates integers from 0 up to, but not including, 5.

Python's range() Function Examples

Simple Usage

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32 | >>> # One parameter  >>> for i in range(5):  ...     print(i)  ...  0  1  2  3  4  >>> # Two parameters  >>> for i in range(3, 6):  ...     print(i)  ...  3  4  5  >>> # Three parameters  >>> for i in range(4, 10, 2):  ...     print(i)  ...  4  6  8  >>> # Going backwards  >>> for i in range(0, -10, -2):  ...     print(i)  ...  0  -2  -4  -6  -8 |

Iterating Lists

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10 | >>> my\_list = ['one', 'two', 'three', 'four', 'five']  >>> my\_list\_len = len(my\_list)  >>> for i in range(0, my\_list\_len):  ...     print(my\_list[i])  ...  one  two  three  four  five |

**Questions**

1. Define a function max() that takes two numbers as arguments and returns the largest of them. Use the if-then-else construct available in Python.
2. Define a function max\_of\_three() that takes three numbers as arguments and returns the largest of them.
3. Define a function that computes the length of a given list or string.
4. Write a function that takes a character (i.e. a string of length 1) and returns True if it is a vowel, False otherwise.
5. Write a function translate() that will translate a text into "rövarspråket" (Swedish for "robber's language"). That is, double every consonant and place an occurrence of "o" in between. For example, translate("this is fun")should return the string "tothohisos isos fofunon".
6. Define a function sum() and a function multiply() that sums and multiplies (respectively) all the numbers in a list of numbers. For example, sum([1, 2, 3, 4]) should return 10, and multiply([1, 2, 3, 4]) should return 24.
7. Define a function reverse() that computes the reversal of a string. For example, reverse("I am testing") should return the string "gnitset ma I".
8. Define a function is\_palindrome() that recognizes palindromes (i.e. words that look the same written backwards). For example, is\_palindrome("radar") should return True.
9. Define a *procedure* histogram() that takes a list of integers and prints a histogram to the screen. For example,histogram([4, 9, 7]) should print the following:

\*\*\*\*

\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*