

Python

Loop

Class IX

Lab 14

**Lab Objectives:**

* While loops
* For Loops

Loop

In computer **programming**, a **loop** is a sequence of instructions that is continually repeated until a certain condition is reached.

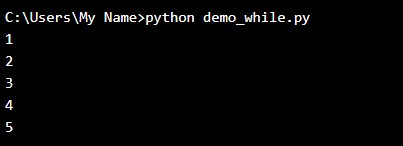
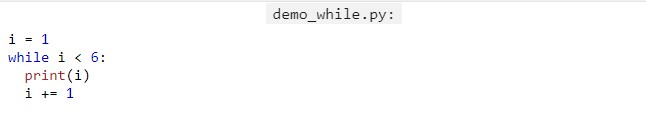
Python Loops

Python has two primitive loop commands:

* while loops
* for loops

The while Loop

With the while loop we can execute a set of statements as long as a condition is true.



Note: remember to increment i, or else the loop will continue forever.

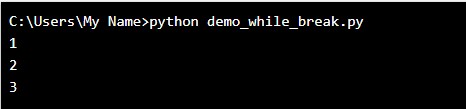
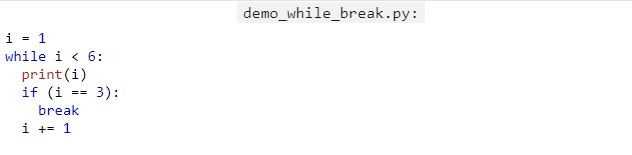
The while loop requires relevant variables to be ready, in this example we need to define an indexing variable, i, which we set to 1.

The break Statement

With the break statement we can stop the loop even if the while condition is true:

Example

Exit the loop when i is 3:

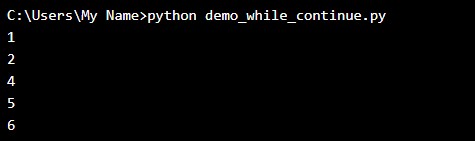
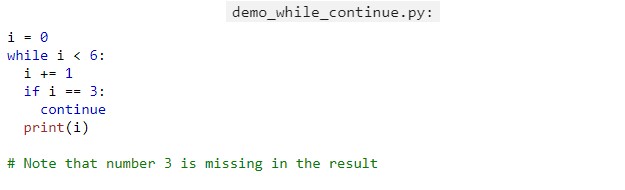


The continue Statement

With the continue statement we can stop the current iteration, and continue with the next:

Example

Continue to the next iteration if i is 3:



Python For Loops

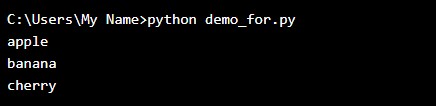
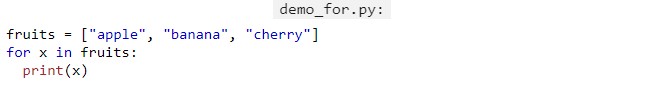
A for loop is used for iterating over a sequence (that is either a list, a tuple, a dictionary, a set, or a string).

This is less like the for keyword in other programming language, and works more like an iterator method as found in other object-orientated programming languages.

With the for loop we can execute a set of statements, once for each item in a list, tuple, set etc.

Example

Print each fruit in a fruit list:



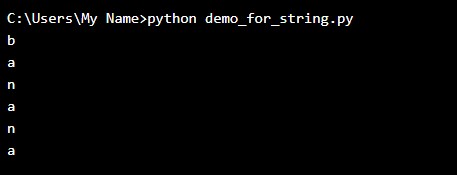
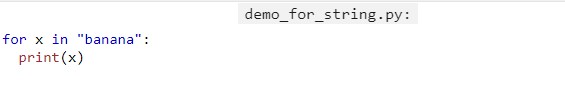
The for loop does not require an indexing variable to set beforehand.

Looping Through a String

Even strings are iterable objects, they contain a sequence of characters:

Example

Loop through the letters in the word "banana":

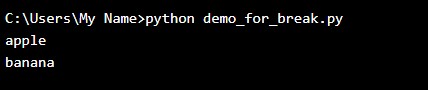
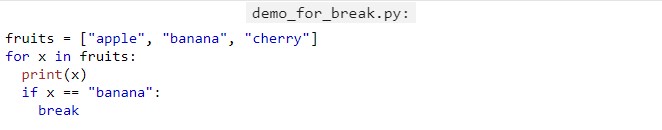


The break Statement

With the break statement we can stop the loop before it has looped through all the items:

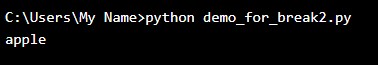
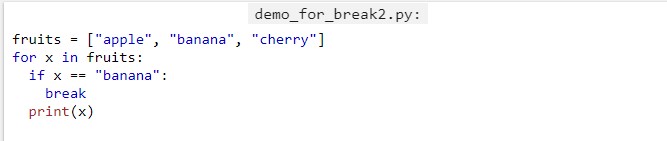
Example

Exit the loop when x is "banana":



Example

Exit the loop when x is "banana", but this time the break comes before the print:

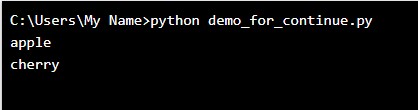
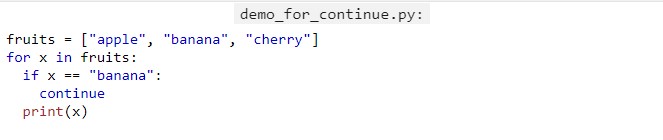


The continue Statement

With the continue statement we can stop the current iteration of the loop, and continue with the next:

### **Example**

Do not print banana:



The range() Function

To loop through a set of code a specified number of times, we can use the range() function,

The range() function returns a sequence of numbers, starting from 0 by default, and increments by 1 (by default), and ends at a specified number.

Example

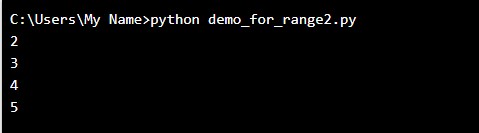
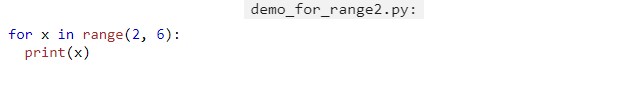
Using the range() function:

Note that, range(6) is not the values of 0 to 6, but the values 0 to 5.

The range() function defaults to 0 as a starting value, however it is possible to specify the starting value by adding a parameter: range(2, 6), which means values from 2 to 6 (but not including 6):

Example

Using the start parameter:



The range() function defaults to increment the sequence by 1, however it is possible to specify the increment value by adding a third parameter: range(2, 30, 3):

Example

Increment the sequence with 3 (default is 1):

