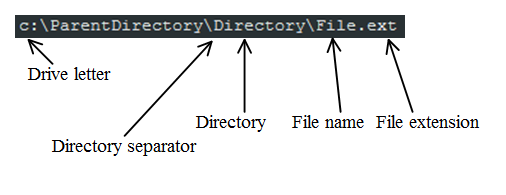
Generic Vocabulary:

|  |  |  |
| --- | --- | --- |
| Absolute file path | The sequence of directories in which a file is located, beginning with the drive letter and a colon, and ending with the file extension | C:\Python\scripts\script.py |
| Relative file path | A file path that begins with the active directory, instead of a drive letter. May begin with a ‘.’ | \scripts\script.py (active directory is C:\Python) |
| Active directory | The directory the program is running out of (the save location of the .py file). Python uses the ‘.’ character to represent the active directory. | os.path.abspath(‘.’) (returns the absolute path of the active directory) |
| File extension | The string after the last ‘.’ in a file path that indicates the file type to the operating system. | .py |
| Character | A single symbol (such as a number or letter) | A, 1 |
| Implicit | An action that a program performs that is not specifically declared by a statement in a program. | **1** is implicitly converted to a Boolean by the **if** statement |
| Explicit | An action performed by a program that is declared by a statement in the program. | **1** is explicitly converted to a Boolean by the **bool** statement, before being evaluated by the **if** statement. |
| Builtin | A function that is available in Python without importing any modules | abs() |
| Module | A single Python file that is imported into a Python script | os |
| Package | A group of modules that are imported into a Python script. For most purposes (except creating them), these may be treated as modules |  |
| Operator | A symbol or set of symbols used to modify data | +, -, not |
| Iteration | Going through each value of an iterable and doing something to or with each value in the iterable | [1, 2, 3, 4] is being iterated. The operation performed is printing each value. |
| Iterable | A data type that may be iterated | List, string, tuple, dictionary |
| Index | The number, starting at 0, assigned to a value in an iterable . Negative indexes can be used to start at the end of an iterable (-1 is the last value, -2 is second to last) | **a[0]** returns the first value in list **a** |
| Floor round | Rounding a float down to the nearest integer less than the value. the int() function performs this. Negative numbers are considered greater as they increase | **int(2.7)=2**  **int(-2.7)=-2** |
| Ceiling round | Rounding a float up to the nearest integer greater than the value. math.ceil() performs this. Negatives are considered lesser as they increase | **math.ceil(2.2)=3**  **math.ceil(-2.2)=-2** |
| Comment | A part of the program that is not executed. Indicated with a ‘#’ in front of the comment | **#This will never be executed, so no syntax error arises** |
| Argument | A value supplied to a function. Generally interchangeable with parameter | **range(1, 3, 5)**  **1, 3, and five are arguments** |
| Parameter | A value required by a function. | **def f(a, b, c)**  **A, b, and c are parameters** |

Anatomy of a file path:



Data Types

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Type | Abbreviation/function name | Explanation | Example | Operators usable (not an exclusive list) |
| Boolean | bool | A True/False value. | True, False | and, or, not |
| Integer | int | An integer. | 1 | +,-,\*,/,\*\* |
| Float | float | Any number. Integers will appear with ‘.0’ on the end | 5.125 | see int |
| Long | long | An integer too large to be stored in an `int` variable, or with an L on the end | 97[several hundred digits here]9290L | see int |
| String | str | A sequence of charecters surrounded by single or double quotes (‘/”) | ‘name’ | +,\* |
| Multi-line string | none | A string spanning multiple lines, indicated with three double or single quotes (‘’’/”””) | “””This  is  a  string””” | +,\* |
| Raw string | typing a lower case ‘r’ before the opening quote of a string | A string which may contain unescaped backslashes(see escape characters below). | r’\<-that is an unescaped backslash’ | +,\* |
| List | list | A list of variables, denoted with brackets. May be changed after creating | [1, ‘a’, False] | +,\* |
| Tuple | tuple | A list of variables denoted with parenthesis. **May not be changed after creation** | (1, ‘a’, False) | none |
| Dictionary | dict | A list of variables indexed with keys instead of numbers. Uses curly brackets. | {key:value} | + |
| Generator | none | A generator object is similar to a list, except it has no indexes and can only be used once. This is because the values in it are generated **during** iteration, rather than before iteration. This makes it faster and uses less memory than other iterables. | (i for i in [1, 2, 3, 4]) will return a generator object that yields 1, 2, 3, 4 upon iteration | none |

Operators

|  |  |  |
| --- | --- | --- |
| Operator | Description | Example |
| + | Adds numbers. Can also be used to add lists and dictionaries | ‘1’+’a’=’1a’ |
| - | Subtracts numbers |  |
| \* | Multiplies numbers. Can also multiply lists and strings by integers. |  |
| / | Divides numbers  Python 2 performs division of integers by taking the floor round of the quotient. Avoid this by making sure that at least one of the values being divided is a float, or adding “**from \_\_future\_\_ import division”** to your imports. | 2/5=0  2.0/5.0=0.4 |
| \*\* | An exponent | 2\*\*3=8  4\*\*0.5=2 |
| not | Inverses a Boolean value | not False = True |
| and | Returns True if all the Booleans in the operation are true. Otherwise returns False | True and False = False  False and False = False  True and True = True |
| or | Returns True if at least one of the Booleans in the statement are true | True or False = True  False or False = False  True or True = True |
| == | Returns true if the values compared are equivalent | 1==1.0 is True  1== 3 is False |
| != | Returns true if the values compared are not equivalent | 1!=1 is False  1 !=3 is True |

Common built-ins

|  |  |  |
| --- | --- | --- |
| raw\_input(string=’’) | Gets user input in the form of a string  Note: put ‘\n’ at the end of the query to have the user input appear on a new line |  |
| range(start=0, stop, step=1) | Returns a list of the values starting at 0 (or the specified start) and ending 1 step interval below the stop value. xrange() works the same way but produces a generator object. |  |

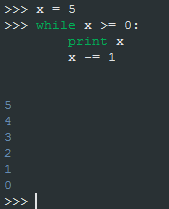
Statements

Statements are commands that do not require parenthesis

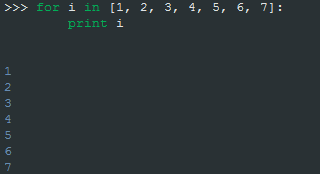
|  |  |  |
| --- | --- | --- |
| Statement | Description | Example |
| print | Prints the variable to the console, implicitly converting to a string |  |
| if/else | if the supplied Boolean is true, run code. |  |
| elif | If the previous if statement is false, and the elif statement is true, run code |  |
| return | Found in a function. Ends the function and sets the returned value to a variable. | If the function is run as a(), the return value will be discarded |
| yield | Only in functions. Temporarily exits the function until the set variable is iterated over. A function with yield will produce generator objects |  |
| Try/except/else | Attempts the code under try. If an error is raised, executes the code under except. If no error is raised, executes the code under else. Note: you do not need else for the statement to run | The list [1, 2, 3] has no index 5, so this raises an IndexError. the except does not list any specific errors, so it catches all the errors    This time, except catches a specific error and saves it to the variable ‘e’  If you want to write the error to the variable, but do not want to list all possible exceptions, you can use Exception:    Multiple exceptions use tuples:    If an exception other than the one(s) listed occurs, it will raise the error |

Loops

While loops will perform a snippet of code while a condition remains true.

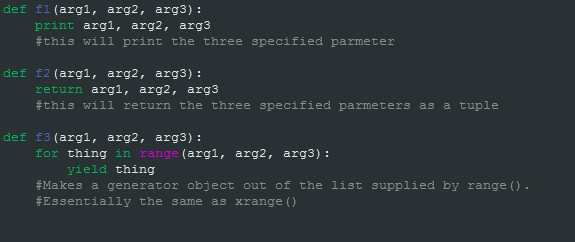


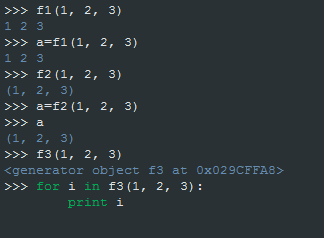
For loops will iterate over an iterable, and perform a snippet of code for each item in the iterable.



Functions

Functions are defined using def. They can contain any number of arguments, and return or yield statements. Note that the Python console will print returned values. If this were executed in a script, the returned values would not be printed.





Useful code snippets:

a, b = b, a

#switches A and B

a=(os.path.join(root, filename)

for root, \_, filenames in os.walk(path)

for filename in filenames)

#Sets variable A to a generator object of the full path names of the files found by walking the variable path.

os.path.abspath(‘.’)

#returns path of active directory