

Order Of Precedence

1. Parenthesis ()
2. Function Call f(args..)
3. Indexing []
4. Exponents **
5. Unary plus, Unary minus +x, -x
6. Multiplication, division, floor division, remainder *, /, //, %
7. Addition and subtraction +, -
8. Identity operators, membership operators (in, not in, is, is not)
9. Equality Operators !=, ==
10. Comparison operators <, <=, >, >=
11. Boolians (NOT, AND, OR)

Files, Lists, Loops, selections

fo=open()	open file under (name, w/r)
fo.write(str(list[i]) + "\n or ,")	
fo.read()	reads rows of text in file
fo.readlines	reads columns in file
close()	closes file
Index[0:0:0]	Start, stop, stride
while i > var:	make sure to add :
Single outcome decision structure	if ...
Double outcome decision structure	if, else
Chained decision structure	if, elif, elif, else
Nested decision structure	if, (indent) if, else, (outdent) else

Definitions

\n	New line
\t	Tab line
Chained IF structure	a set of conditions that allows for only one block of code to run: if...elif...elif...else
Concat-enation	the process by which strings are joined together (uses an overloaded + operator)
Implicit data type	variable = 5, variable = 5.0
Explicit Data Type	variable = int(5.0), variable = string(5)

Definitions (cont)

De Morgans Laws	not (x and y) == (not x) or (not y) not (x or y) == (not x) and (not y)
Definite Loop	a loop that repeats a specified number of times; also called a counted loop
Indefinite Loop	loop that repeats an unspecified number of time based on the condition (conditional loop)
Infinite Loop	a loop that never stops. Usually the result of an error in the internal change/read
Sentinel Loop	continues to process data until reaching a special value that signals the end.
function (Void and Value-- Returning)	def nameOfFunction (optional parameters, default parameters, etc) : # code as needed # that is indented # return optional value
len()	a value returning function that indicates how long an object is (ie how many characters in a string or how many items in a list)
max()	prints largest ASCII value
min()	prints lowest ASCII value
sorted()	sorts in ASCII order
sum()	In Python, a value returning function that adds up all the numerical items in a list.n Excel, a value returning function that alls up all the numerical values in a range.
value-ret-urning function	a function that does its processes to produce a result that is returned to the main program
void function	a function that performs processes, but does not create/return a result

Definitions (cont)

scope a description of the location within code and of a time when that code is being executed, where/when the computer can access/know about that code

Methods

var.lower() turns var lowercase

var.upper() turns var uppercase

var.swapcase swaps the case of letters (upper to lower, vice versa)

var.capitalize capitalizes first letter

list.append(x) a void method that is part of all lists that will add x to the end of that list

list.index(x) a value returning method that is part of all list objects, that produces the location (index number) of the value x

list.count(x) a value returning method that all lists have that will produce the number of times x occurs in the list

list.remove(x) a void list object method that deletes the first instance of the value x from the list

list.insert(index, x) a void method that is part of all list objects that inserts the value x at index

list.sort() a void list object method, one that takes the list and sorts in ascending order

list.pop(x) a value returning list method that produces the value of the item at x. If x is left blank, the default is -1, the last item's value

logic Error an error in the code that does NOT prevent the code from running, but stems from an incorrect solution that produces unintended or undesired output or other behavior

Methods (cont)

Index Error A type of run time error (only happens when the program runs) when the code is asked to work with an index location that is larger than the length of the list or string

Name Error a Run-time error in the code where a named thing (like a variable or a function) is invoked and/or used before it is defined.

run-time Error an error that occurs only when the program is run - usually due to input/output and data type conflicts

syntax error an error in the code that prevents it from running where the code does not follow the rules of spelling, punctuation and/or grammar

Type Error A kind of Run-time error where the variable types do not match the function's or mathematical operator's requirements.