

String Methods

This is a list of common string methods. In it, S is the string to which the method is applied, and s and t are other strings.

Operation	Description
<code>S.capitalize()</code>	If the first character of S is a letter, capitalize it
<code>S.count(s)</code>	Count the number of times s occurs in S
<code>S.endswith(s)</code>	True if S ends with s ; False otherwise
<code>S.find(s)</code>	Return the index of the first occurrence of s in S ; -1 if s not in S
<code>S.index(s)</code>	Return the index of the first occurrence of s in S ; <code>ValueError</code> exception if s not in S
<code>S.isalnum()</code>	True if S contains only alphanumerics (letters and digits); False otherwise
<code>S.isalpha()</code>	True if S contains only alphabetic (letters); False otherwise
<code>S.isdigit()</code>	True if S contains only digits; False otherwise
<code>S.islower()</code>	True if all letters in S are lower case; False otherwise
<code>S.isspace()</code>	True if S contains only white space; False otherwise
<code>S.isupper()</code>	True if all letters in S are upper case; False otherwise
<code>S.lower()</code>	Change all upper case letters in S to lower case
<code>S.lstrip()</code>	Delete all leading white space from S and return the result
<code>S.replace(s, t)</code>	Replace all occurrences of s with t in S
<code>S.rfind(s)</code>	Return the index of the last occurrence of s in S ; -1 if s not in S
<code>S.rindex(s)</code>	Return the index of the last occurrence of s in S ; <code>ValueError</code> exception if s not in S
<code>S.rstrip()</code>	Delete all trailing white space from S
<code>S.strip()</code>	Delete all leading and trailing white space from S
<code>S.swapcase()</code>	Change all upper case letters in S to lower case and all lower case letters to upper case
<code>S.title()</code>	Capitalize each word in S
<code>S.upper()</code>	Change all lower case letters in S to upper case

List Methods

This is a list of list methods. In it, L is the list to which the method is applied, M is a list, x is an element to be added to, looked for, or removed from, a list, and i is an index of a list element.

Operation	Description
<code>L.append(x)</code>	Append element x to L
<code>L.count(x)</code>	Count the number of times x occurs in L
<code>L.extend(M)</code>	Extend L by adding the elements of M at the end
<code>L.index(x)</code>	Return the index of the first occurrence of x in L ; <code>ValueError</code> exception if x not in L
<code>L.insert(i, x)</code>	Insert x at position i in L
<code>L.pop()</code>	Remove and return the last element of L
<code>L.pop(i)</code>	Remove and return the element of L at position i ; <code>IndexError</code> exception if i out of range
<code>L.remove(x)</code>	Remove the first occurrence of x from L ; <code>ValueError</code> exception if x not in L
<code>L.reverse()</code>	Reverse L in place (does <i>not</i> make a copy)
<code>L.sort()</code>	Sort L in place (does <i>not</i> make a copy)