Python Methods: String, List, Dictionary, Set, Tuple

 **String Methods**:

* Examples: .upper(), .lower(), .replace(), .split(), .join()

 **List Methods**:

* Examples: .append(), .remove(), .pop(), .extend(), .sort()

 **Dictionary Methods**:

* Examples: .get(), .keys(), .values(), .items(), .update()

 **Set Methods**:

* Examples: .add(), .remove(), .union(), .intersection(), .difference()

 **Tuple**:

* Explanation: Tuples are immutable, no modification methods.

# String Methods

String methods perform operations on strings. Strings are immutable, meaning these methods do not modify the original string but return a new one.

|  |  |
| --- | --- |
| **Method** | **Description** |
| [capitalize()](https://www.w3schools.com/python/ref_string_capitalize.asp) | Converts the first character to upper case |
| [casefold()](https://www.w3schools.com/python/ref_string_casefold.asp) | Converts string into lower case |
| [center()](https://www.w3schools.com/python/ref_string_center.asp) | Returns a centered string |
| [count()](https://www.w3schools.com/python/ref_string_count.asp) | Returns the number of times a specified value occurs in a string |
| [encode()](https://www.w3schools.com/python/ref_string_encode.asp) | Returns an encoded version of the string |
| [endswith()](https://www.w3schools.com/python/ref_string_endswith.asp) | Returns true if the string ends with the specified value |
| [expandtabs()](https://www.w3schools.com/python/ref_string_expandtabs.asp) | Sets the tab size of the string |
| [find()](https://www.w3schools.com/python/ref_string_find.asp) | Searches the string for a specified value and returns the position of where it was found |
| [format()](https://www.w3schools.com/python/ref_string_format.asp) | Formats specified values in a string |
| format\_map() | Formats specified values in a string |
| [index()](https://www.w3schools.com/python/ref_string_index.asp) | Searches the string for a specified value and returns the position of where it was found |
| [isalnum()](https://www.w3schools.com/python/ref_string_isalnum.asp) | Returns True if all characters in the string are alphanumeric |
| [isalpha()](https://www.w3schools.com/python/ref_string_isalpha.asp) | Returns True if all characters in the string are in the alphabet |
| [isascii()](https://www.w3schools.com/python/ref_string_isascii.asp) | Returns True if all characters in the string are ascii characters |
| [isdecimal()](https://www.w3schools.com/python/ref_string_isdecimal.asp) | Returns True if all characters in the string are decimals |
| [isdigit()](https://www.w3schools.com/python/ref_string_isdigit.asp) | Returns True if all characters in the string are digits |
| [isidentifier()](https://www.w3schools.com/python/ref_string_isidentifier.asp) | Returns True if the string is an identifier |
| [islower()](https://www.w3schools.com/python/ref_string_islower.asp) | Returns True if all characters in the string are lower case |
| [isnumeric()](https://www.w3schools.com/python/ref_string_isnumeric.asp) | Returns True if all characters in the string are numeric |
| [isprintable()](https://www.w3schools.com/python/ref_string_isprintable.asp) | Returns True if all characters in the string are printable |
| [isspace()](https://www.w3schools.com/python/ref_string_isspace.asp) | Returns True if all characters in the string are whitespaces |
| [istitle()](https://www.w3schools.com/python/ref_string_istitle.asp) | Returns True if the string follows the rules of a title |
| [isupper()](https://www.w3schools.com/python/ref_string_isupper.asp) | Returns True if all characters in the string are upper case |
| [join()](https://www.w3schools.com/python/ref_string_join.asp) | Converts the elements of an iterable into a string |
| [ljust()](https://www.w3schools.com/python/ref_string_ljust.asp) | Returns a left justified version of the string |
| [lower()](https://www.w3schools.com/python/ref_string_lower.asp) | Converts a string into lower case |
| [lstrip()](https://www.w3schools.com/python/ref_string_lstrip.asp) | Returns a left trim version of the string |
| [maketrans()](https://www.w3schools.com/python/ref_string_maketrans.asp) | Returns a translation table to be used in translations |
| [partition()](https://www.w3schools.com/python/ref_string_partition.asp) | Returns a tuple where the string is parted into three parts |
| [replace()](https://www.w3schools.com/python/ref_string_replace.asp) | Returns a string where a specified value is replaced with a specified value |
| [rfind()](https://www.w3schools.com/python/ref_string_rfind.asp) | Searches the string for a specified value and returns the last position of where it was found |
| [rindex()](https://www.w3schools.com/python/ref_string_rindex.asp) | Searches the string for a specified value and returns the last position of where it was found |
| [rjust()](https://www.w3schools.com/python/ref_string_rjust.asp) | Returns a right justified version of the string |
| [rpartition()](https://www.w3schools.com/python/ref_string_rpartition.asp) | Returns a tuple where the string is parted into three parts |
| [rsplit()](https://www.w3schools.com/python/ref_string_rsplit.asp) | Splits the string at the specified separator, and returns a list |
| [rstrip()](https://www.w3schools.com/python/ref_string_rstrip.asp) | Returns a right trim version of the string |
| [split()](https://www.w3schools.com/python/ref_string_split.asp) | Splits the string at the specified separator, and returns a list |
| [splitlines()](https://www.w3schools.com/python/ref_string_splitlines.asp) | Splits the string at line breaks and returns a list |
| [startswith()](https://www.w3schools.com/python/ref_string_startswith.asp) | Returns true if the string starts with the specified value |
| [strip()](https://www.w3schools.com/python/ref_string_strip.asp) | Returns a trimmed version of the string |
| [swapcase()](https://www.w3schools.com/python/ref_string_swapcase.asp) | Swaps cases, lower case becomes upper case and vice versa |
| [title()](https://www.w3schools.com/python/ref_string_title.asp) | Converts the first character of each word to upper case |
| [translate()](https://www.w3schools.com/python/ref_string_translate.asp) | Returns a translated string |
| [upper()](https://www.w3schools.com/python/ref_string_upper.asp) | Converts a string into upper case |
| [zfill()](https://www.w3schools.com/python/ref_string_zfill.asp) | Fills the string with a specified number of 0 values at the beginning |

Examples of String Methods:

s = " Hello, World! "  
  
# Lowercase  
print(s.lower()) # Output: " hello, world! "  
  
# Uppercase  
print(s.upper()) # Output: " HELLO, WORLD! "  
  
# Strip  
print(s.strip()) # Output: "Hello, World!"  
  
# Replace  
print(s.replace("Hello", "Hi")) # Output: " Hi, World! "  
  
# Split  
print(s.split(",")) # Output: [' Hello', ' World! ']  
  
# Join  
words = ["Python", "is", "awesome"]  
print(" ".join(words)) # Output: "Python is awesome"  
  
# Find  
print(s.find("World")) # Output: 10  
  
# Startswith  
print(s.startswith(" He")) # Output: True  
  
# Endswith  
print(s.endswith("! ")) # Output: True

# List Methods

List methods allow modification of a list. Lists are mutable, meaning they can be modified directly.

|  |  |
| --- | --- |
| **Method** | **Description** |
| [append()](https://www.w3schools.com/python/ref_list_append.asp) | Adds an element at the end of the list |
| [clear()](https://www.w3schools.com/python/ref_list_clear.asp) | Removes all the elements from the list |
| [copy()](https://www.w3schools.com/python/ref_list_copy.asp) | Returns a copy of the list |
| [count()](https://www.w3schools.com/python/ref_list_count.asp) | Returns the number of elements with the specified value |
| [extend()](https://www.w3schools.com/python/ref_list_extend.asp) | Add the elements of a list (or any iterable), to the end of the current list |
| [index()](https://www.w3schools.com/python/ref_list_index.asp) | Returns the index of the first element with the specified value |
| [insert()](https://www.w3schools.com/python/ref_list_insert.asp) | Adds an element at the specified position |
| [pop()](https://www.w3schools.com/python/ref_list_pop.asp) | Removes the element at the specified position |
| [remove()](https://www.w3schools.com/python/ref_list_remove.asp) | Removes the first item with the specified value |
| [reverse()](https://www.w3schools.com/python/ref_list_reverse.asp) | Reverses the order of the list |
| [sort()](https://www.w3schools.com/python/ref_list_sort.asp) | Sorts the list |

Examples of List Methods:

lst = [1, 2, 3]  
  
# Append  
lst.append(4)  
print(lst) # Output: [1, 2, 3, 4]  
  
# Extend  
lst.extend([5, 6])  
print(lst) # Output: [1, 2, 3, 4, 5, 6]  
  
# Insert  
lst.insert(2, 99)  
print(lst) # Output: [1, 2, 99, 3, 4, 5, 6]  
  
# Remove  
lst.remove(99)  
print(lst) # Output: [1, 2, 3, 4, 5, 6]  
  
# Pop  
popped = lst.pop(2)  
print(popped) # Output: 3  
print(lst) # Output: [1, 2, 4, 5, 6]  
  
# Sort  
lst.sort()  
print(lst) # Output: [1, 2, 4, 5, 6]  
  
# Reverse  
lst.reverse()  
print(lst) # Output: [6, 5, 4, 2, 1]

# Dictionary Methods

Dictionary methods are used for operations related to key-value pairs. Dictionaries are mutable.

|  |  |
| --- | --- |
| **Method** | **Description** |
| [clear()](https://www.w3schools.com/python/ref_dictionary_clear.asp) | Removes all the elements from the dictionary |
| [copy()](https://www.w3schools.com/python/ref_dictionary_copy.asp) | Returns a copy of the dictionary |
| [fromkeys()](https://www.w3schools.com/python/ref_dictionary_fromkeys.asp) | Returns a dictionary with the specified keys and value |
| [get()](https://www.w3schools.com/python/ref_dictionary_get.asp) | Returns the value of the specified key |
| [items()](https://www.w3schools.com/python/ref_dictionary_items.asp) | Returns a list containing a tuple for each key value pair |
| [keys()](https://www.w3schools.com/python/ref_dictionary_keys.asp) | Returns a list containing the dictionary's keys |
| [pop()](https://www.w3schools.com/python/ref_dictionary_pop.asp) | Removes the element with the specified key |
| [popitem()](https://www.w3schools.com/python/ref_dictionary_popitem.asp) | Removes the last inserted key-value pair |
| [setdefault()](https://www.w3schools.com/python/ref_dictionary_setdefault.asp) | Returns the value of the specified key. If the key does not exist: insert the key, with the specified value |
| [update()](https://www.w3schools.com/python/ref_dictionary_update.asp) | Updates the dictionary with the specified key-value pairs |
| [values()](https://www.w3schools.com/python/ref_dictionary_values.asp) | Returns a list of all the values in the dictionary |

Examples of Dictionary Methods:

d = {"a": 1, "b": 2, "c": 3}  
  
# Keys  
print(d.keys()) # Output: dict\_keys(['a', 'b', 'c'])  
  
# Values  
print(d.values()) # Output: dict\_values([1, 2, 3])  
  
# Items  
print(d.items()) # Output: dict\_items([('a', 1), ('b', 2), ('c', 3)])  
  
# Get  
print(d.get("b")) # Output: 2  
  
# Update  
d.update({"d": 4})  
print(d) # Output: {'a': 1, 'b': 2, 'c': 3, 'd': 4}  
  
# Pop  
popped = d.pop("c")  
print(popped) # Output: 3  
print(d) # Output: {'a': 1, 'b': 2, 'd': 4}

# Set Methods

Set methods are for working with unordered collections of unique elements. Sets are mutable.

|  |  |  |
| --- | --- | --- |
| **Method** | **Shortcut** | **Description** |
| [add()](https://www.w3schools.com/python/ref_set_add.asp) |  | Adds an element to the set |
| [clear()](https://www.w3schools.com/python/ref_set_clear.asp) |  | Removes all the elements from the set |
| [copy()](https://www.w3schools.com/python/ref_set_copy.asp) |  | Returns a copy of the set |
| [difference()](https://www.w3schools.com/python/ref_set_difference.asp) | [-](https://www.w3schools.com/python/ref_set_difference.asp) | Returns a set containing the difference between two or more sets |
| [difference\_update()](https://www.w3schools.com/python/ref_set_difference_update.asp) | [-=](https://www.w3schools.com/python/ref_set_difference_update.asp) | Removes the items in this set that are also included in another, specified set |
| [discard()](https://www.w3schools.com/python/ref_set_discard.asp) |  | Remove the specified item |
| [intersection()](https://www.w3schools.com/python/ref_set_intersection.asp) | [&](https://www.w3schools.com/python/ref_set_intersection.asp) | Returns a set, that is the intersection of two other sets |
| [intersection\_update()](https://www.w3schools.com/python/ref_set_intersection_update.asp) | [&=](https://www.w3schools.com/python/ref_set_intersection_update.asp) | Removes the items in this set that are not present in other, specified set(s) |
| [isdisjoint()](https://www.w3schools.com/python/ref_set_isdisjoint.asp) |  | Returns whether two sets have a intersection or not |
| [issubset()](https://www.w3schools.com/python/ref_set_issubset.asp) | [<=](https://www.w3schools.com/python/ref_set_issubset.asp) | Returns whether another set contains this set or not |
|  | [<](https://www.w3schools.com/python/ref_set_issubset.asp) | Returns whether all items in this set is present in other, specified set(s) |
| [issuperset()](https://www.w3schools.com/python/ref_set_issuperset.asp) | [>=](https://www.w3schools.com/python/ref_set_issuperset.asp) | Returns whether this set contains another set or not |
|  | [>](https://www.w3schools.com/python/ref_set_issuperset.asp) | Returns whether all items in other, specified set(s) is present in this set |
| [pop()](https://www.w3schools.com/python/ref_set_pop.asp) |  | Removes an element from the set |
| [remove()](https://www.w3schools.com/python/ref_set_remove.asp) |  | Removes the specified element |
| [symmetric\_difference()](https://www.w3schools.com/python/ref_set_symmetric_difference.asp) | [^](https://www.w3schools.com/python/ref_set_symmetric_difference.asp) | Returns a set with the symmetric differences of two sets |
| [symmetric\_difference\_update()](https://www.w3schools.com/python/ref_set_symmetric_difference_update.asp) | [^=](https://www.w3schools.com/python/ref_set_symmetric_difference_update.asp) | Inserts the symmetric differences from this set and another |
| [union()](https://www.w3schools.com/python/ref_set_union.asp) | [|](https://www.w3schools.com/python/ref_set_union.asp) | Return a set containing the union of sets |
| [update()](https://www.w3schools.com/python/ref_set_update.asp) | [|=](https://www.w3schools.com/python/ref_set_update.asp) | Update the set with the union of this set and others |

Examples of Set Methods:

s = {1, 2, 3}  
  
# Add  
s.add(4)  
print(s) # Output: {1, 2, 3, 4}  
  
# Remove  
s.remove(2)  
print(s) # Output: {1, 3, 4}  
  
# Discard (no error if element doesn't exist)  
s.discard(5)  
print(s) # Output: {1, 3, 4}  
  
# Union  
s2 = {3, 4, 5}  
print(s.union(s2)) # Output: {1, 3, 4, 5}  
  
# Intersection  
print(s.intersection(s2)) # Output: {3, 4}  
  
# Difference  
print(s.difference(s2)) # Output: {1}

# Tuple Methods

Tuple methods are limited since tuples are immutable. However, they allow counting occurrences and finding indices.

Examples of Tuple Methods:  
t = (1, 2, 2, 3)  
  
# Count  
print(t.count(2)) # Output: 2  
  
# Index  
print(t.index(3)) # Output: 3