



16.1. Introduction: Sorting with Sort and Sorted

When we first introduced lists, we noted the existence of a method `sort`. When invoked on a list, the order of items in the list is changed. If no optional parameters are specified, the items are arranged in whatever the natural ordering is for the item type. For example, if the items are all integers, then smaller numbers go earlier in the list. If the items are all strings, they are arranged in alphabetic order.

Save & Run

Original - 1 of 1

Show in CodeLens

```
1 L1 = [1, 7, 4, -2, 3]
2 L2 = ["Cherry", "Apple", "Blueberry"]
3
4 L1.sort()
5 print(L1)
6 L2.sort()
7 print(L2)
8
```

```
[-2, 1, 3, 4, 7]
['Apple', 'Blueberry', 'Cherry']
```

Activity: 1 -- ActiveCode (ac18_1_1)

Note that the `sort` method does **not** return a sorted version of the list. In fact, it returns the value `None`. But the list itself has been modified. This kind of operation that works by having a *side effect* on the list can be quite confusing.

In this course, we will generally use an alternative way of sorting, the function `sorted` rather than the method `sort`. Because it is a function rather than a method, it is invoked on a list by passing the list as a parameter inside the parentheses, rather than putting the list before the period. More importantly, `sorted` does not change the original list. Instead, it returns a new list.

Save & Run

Original - 1 of 1

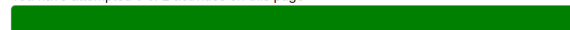
Show in CodeLens

```
1 L2 = ["Cherry", "Apple", "Blueberry"]
2
3 L3 = sorted(L2)
4 print(L3)
5 print(sorted(L2))
6 print(L2) # unchanged
7
8 print("----")
9
10 L2.sort()
11 print(L2)
12 print(L2.sort()) #return value is None
13
```

```
['Apple', 'Blueberry', 'Cherry']
['Apple', 'Blueberry', 'Cherry']
['Cherry', 'Apple', 'Blueberry']
----
['Apple', 'Blueberry', 'Cherry']
None
```

Activity: 2 -- ActiveCode (ac18_1_2)

You have attempted 3 of 2 activities on this page



16. Sorting">

16.2. Optional reverse parameter">

ing">

✓ Completed. Well Done!

16.2. Optional reverse parameter">Next Section - 16.2. Optional reverse parameter