1. In Python, lists should normally be used when we want to store several instances of the "same" type of object; lists of strings or lists of numbers; most often, lists of objects we've defined ourselves. Lists should always be used when we want to store items in some kind of order. Often, this is the order in which they were inserted, but they can also be sorted by some criteria.As we saw in the case study from the previous chapter, lists are also very useful when we need to modify the contents: insert to or delete from an arbitrary location of the list, or update a value within the list.
2. Like dictionaries, lists are objects too, and they have several methods that can be invoked upon them. Here are some common ones:

* The append(element) method adds an element to the end of the list
* The insert(index, element) method inserts an item at a specific position
* The count(element) method tells us how many times an element appears in the list
* The index()method tells us the index of an item in the list, raising an exception if it can't find it
* The find()method does the same thing, but returns -1 instead of raising an exception for missing items
* The reverse() method does exactly what it says—turns the list around
* The sort() method has some rather intricate object-oriented behaviors, which we'll cover now