* **Composition VS Inheritance**

| **Basis of Comparison** | **Composition** | **Inheritance** |
| --- | --- | --- |
| Relationship | It is a "has-a" kind of relationship. | It's a case of "is-a." relationship |
| Functionality | We can evaluate the functioning of the classes that we are using without having to be concerned with determining whether or not they are the parent or the child classes. | It is not possible to test a child class without first testing its parent class. |
| Usability | Because of composition, it is possible to reuse the code even in the final classes. | Inheritance cannot be used to extend the functionality of the final class. |

**list.append()**

Python’s append() function inserts a single element into an existing list. The element will be added to the end of the old list rather than being returned to a new list. Adds its argument as a single element to the end of a list. The length of the list increases by one.

The append() method in the programming language Python adds an item to a list that already exists

**list.extend()**

Iterates over its argument and adding each element to the list and extending the list. The length of the list increases by a number of elements in its argument.

The extend() method adds each of the iterable element which is supplied as a parameter to the end of the original list

**Packages in Python**

A package is basically a directory with Python files and a file with the name \_\_init\_\_.py. This means that every directory inside of the Python path, which contains a file named \_\_init\_\_.py, will be treated as a package by Python. It's possible to put several modules into a Package.

Packages are a way of structuring Python’s module namespace by using "dotted module names". A.B stands for a submodule named B in a package named A. Two different packages like P1 and P2 can both have modules with the same name, let's say A, for example. The submodule A of the package P1 and the submodule A of the package P2 can be totally different. A package is imported like a "normal" module

When you import a module or a package, the object created by Python is always of type module.When you import a package, only the methods and the classes in the \_\_init\_\_.py file of that package are directly visible.

**Popping elements from a list**

The pop() method removes the item at the given index from the list and returns the removed item.

## *pop() parameters*

* The pop() method takes a single argument (index).
* The argument passed to the method is optional. If not passed, the default index -1 is passed as an argument (index of the last item).
* If the index passed to the method is not in range, **it throws IndexError**: pop index out of range exception.