**Difference Between Inheritance and Composition**

**Inheritance**- Inheritance is a mechanism that allows us to inherit all the properties from another class. The class from which the properties and functionalities are utilized is called the **parent class (**also called as **Base Class)**. The class which uses the properties from another class is called as **Child Class (**also known as **Derived class)**. Inheritance is also called an **Is-A Relation**.

**Composition-** by using the class name or by creating the object we can access the members of one class inside another class. It enables creating complex types by combining objects of different classes. It means that a class Composite can contain an object of another class Component. This type of relationship is known as **Has-A Relation**.

**List Append and List Extend**

* **Effect**: .append() adds a single element to the end of the list while .extend() can add multiple individual elements to the end of the list.
* **Argument**: .append() takes a single element as argument while .extend() takes an iterable as argument (list, tuple, dictionaries, sets, strings).

**Trying to Pop elements more than its size list** => error raised “index out of bound”

**Package in python**

a Python package can contains several module. In simpler terms a package is folder that contains various modules as files.

**How to Write Package**

* Create a folder named mypckg.
* Inside this folder create an empty Python file i.e. \_\_init\_\_.py
* Then create two modules mod1 and mod2 in this folder.

\_\_init\_\_.py helps the Python interpreter to recognise the folder as package. It also specifies the resources to be imported from the modules. If the \_\_init\_\_.py is empty this means that all the functions of the modules will be imported.