Due to my background in computer science, I have observed hierarchical structures in the way developers write code. Specifically, object oriented programming (OOP) has long been used across the coding world. Java, for example, is most well known for these design patterns, failing to function without them. OOP is inherently a hierarchical structure, where a final phase (the end product), is composed of many smaller levels of varying size and complexity. In total, there are 5 levels to this structure: methods/fields, objects, classes, abstract classes, and base classes. At the smallest level there are methods and fields. These make up objects, and objects can call, reference, and modify these depending on their access modifiers. This level makes up the actual data being stored, and the next level up organizes that data (objects). Objects provide a framework for how fields and methods should be interacted with. They are implementations of classes, which define how the object should behave, and how it should be able to interact with other objects in the structure (i.e., comparisons, operations, etc.). Classes, being a level above objects, also make use of methods and fields, defining how the objects it creates should interact with them through the use of parameter definitions. Above classes, there are abstract classes. This level contains a rough outline of the class' behavior. It may define base implementation of the methods in a class, or it may simply provide a commonality between classes. Finally, the base classes are at the highest level, providing a framework for abstract classes and fully defining the OOP architecture. OOP's architecture can be thought of as a tree, where the root is the base class and the methods and fields are the branches. Abstract classes can be found at the base of all the branches, and classes can be found at splitting points in the tree branch, followed by objects. This defines strict implementation rules and restrictions, where data from one branch cannot access data from another, but the tree as a whole can perform all operations defined by the branches. It is for this reason that OOP can be thought of as a hierarchical structure.

All information presented is my own, and no current sources were used (all prior knowledge).