Four pillars of object oriented programming

The four pillars of programing are abstraction, Encapsulation, Inheritance and Polymorphism.

Abstraction is about finding similarities between the components of your code and creating a common code that can be used over and over, hiding the most of the implementation using simpler codes. Whether a function or a any type of code written this way makes if easier to maintain and make changes to a project.

Encapsulation is another pillar that brings an idea of (capsuling)making parts of your codes inaccessible based on what needs to accessed.

Inheritance like the other pillars is focused on simplifying coding hustle by allowing one part of our code (child) to inherit properties alike from other part (parent) without having to rewrite similar properties and features.

Polymorphism finally helps the ability of applying a code in multiple forms. Lets say you have a child that inherits properties from another parent, but if the are some things to the child you would customize the parent coding properties and that inheriting and yet customizing it would be the application of the polymorphism pillar.

freecodcamp.org

Relationship of a Class and an object

Object takes it properties and behaviors from what is outlined in the class, let say if an object is to have a ‘name’ and a ‘barking’ properties the would be outlined in the respecting class of the object thus a class acts as the framework for the object.

https://www.ncl.ucar.edu/

Exceptions

Exception are runtime error that are different from syntax errors. You can have a code that have completely free syntax error but have an exception or the runtime error which means you wouldn't even know you have an error until the exception of your code, Using the the try, catch and finally method is the best way to catch runtime errors. The try, catch and finally method helps us to run our code even with the presence of a runtime error and. Cannot be used to control a syntax error.

www.tutorialspoint.com