Notes:

Object-oriented programming (OOP) is a programming model that emphasizes organizing code around *objects*, rather than purely on *procedures and logic*.

Classes: a blueprint of an object with properties and methods

Objects: an individual instant of a class

Constructors: functions in classes that are automatically executed when a new object

Getters: retrieve values of private properties and enhance encapsulation by controlling access to data

Setters: modify values of private properties. A controlled way to update the internal state.

Fields are variables declared within a class or object that holds data.

* Why would you want to write you code in a object-oriented programming language?
  + It’s easy to organize.
  + Good for big projects since you can add more parts without it falling apart.
  + It’s safe and secure.
  + You can talk about objects and what they do like youre talking about everyday stuff, I’ts easier to understand.
* What's the difference between a class and an object?
  + A class is a general template/blueprint that defines how an object behaves. An object is a specific instance of a class.
  + For example: if theres a class named “ice cream” there could be objects like “strawberry sundae” and “rolled Nutella ice cream”. Or if theres a class named “person” there could be objects like “John” or “Lisa”.