OOP essentials

From [Mosh Design Patterns](https://www.youtube.com/watch?v=NU_1StN5Tkk) video

Encapsulations:

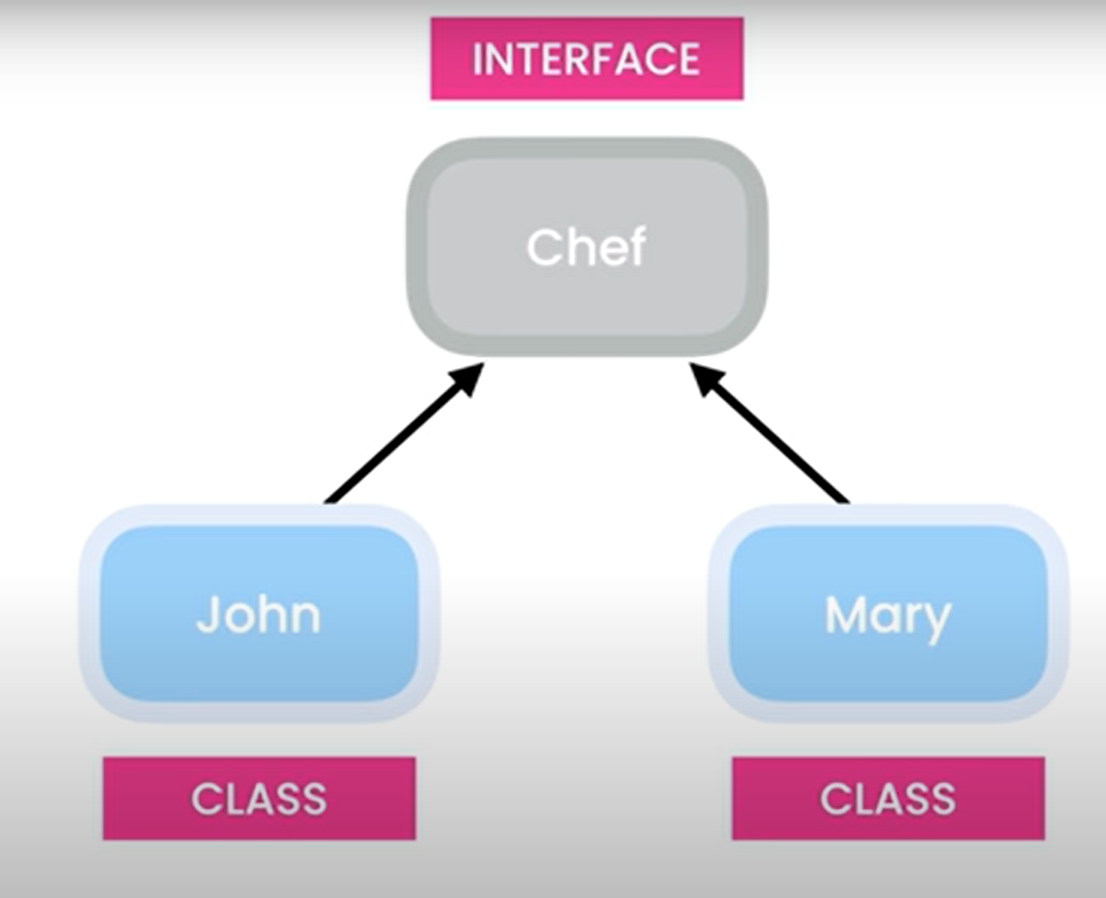
Bundling data and make the properties private into the class and just use method to set/get them.

Coupling:

We should not have related couple classes. Why? Because we want our code to be extensible.

For solving this issue, we use interfaces + dependency injection methods (?)

Interfaces:



A contract that shows capabilities of a class

Abstraction:

Reduce complexity by hiding unnecessary details. e.g., a remote controller doesn’t show circuits. It’s only buttons and has easy-to-use interface.

Polymorphism:

Poly = many, morph = form

We maybe need to implement different forms of a function. Imagine we have a UI controller that Textbox and Checkbox are inheriting from that. So we have a “draw()” method in both Textbox and Checkbox. So we define “draw()” method as an abstract method in UI controller class and we define the body of the method in Textbox and Checkbox separately.

UML (Unified modeling Language):

* This is more complicated than the 1st session UML. This is for showing classes.

