<http://code.tutsplus.com/tutorials/design-patterns-the-adapter-pattern--cms-22262>

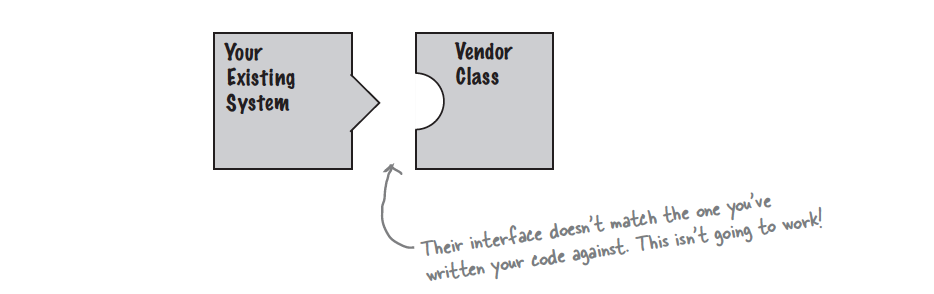
<http://www.smsnica.com/reveal.js/index.html#/7/4>

Definition :

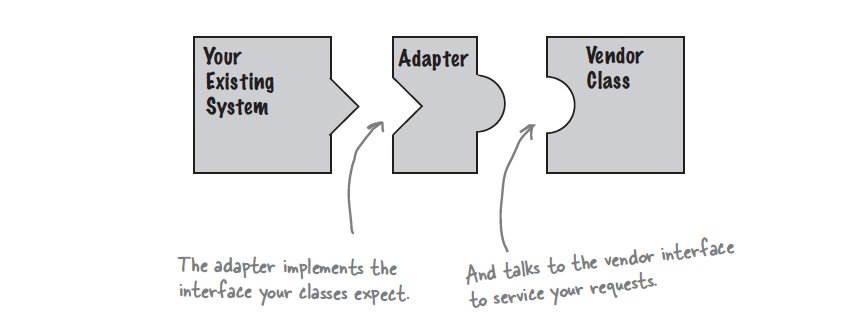
Adapter design pattern can be used when your code is dependent on some external API, or any other class that is prone to change frequently. This pattern falls under the category of "structural patterns" because it teaches us how our code and our classes should be structured in order to manage and/or extend them easily.

When we wrap some objects with a different purpose: to make their interface look like something they’re not.

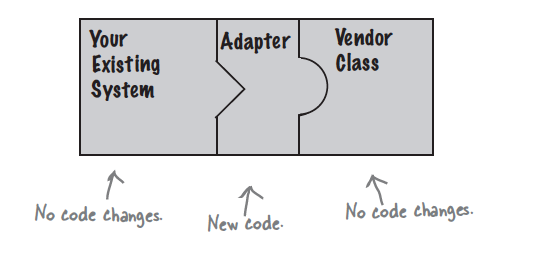
1. Your existing systems some vendor
2. Now you have new vendor class (interface of new vendor is differently than the last vendor)



1. You don’t want to change your code as well as new vendor class code
2. You can write a class that adapts the new vendors interface into the one you are expecting.



1. The adapter acts as middleman by receiving requests from the client and converting them into request that make sense on the vendor class.



1. Two kind of Adapter Class(subclass) and Object (uses Composition)