

Q1.

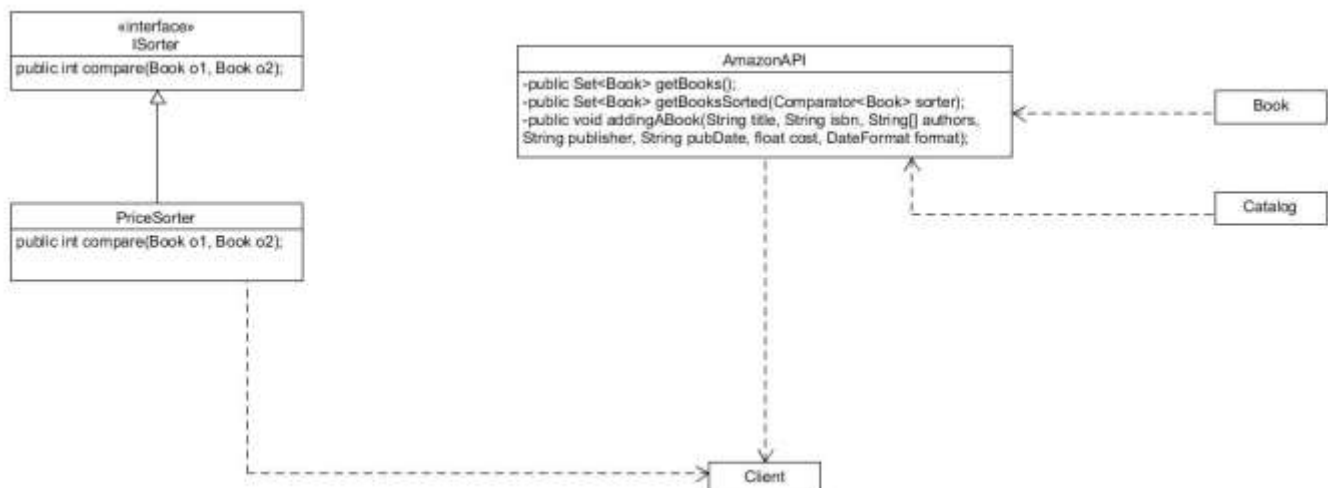
In the available source code in src-amazon the patterns that are used is the façade pattern which is because the in AmazonAPI class keeps everything from the client and create everything for them and you can just call the methods in the AmazonAPI. Another pattern used is in Catalog which is programming to an interface when the Set interface is used to declare the catalog.

Q2.

Problem with this solution the client can only use the sorting algorithms that the AmazonAPI has given and the user is not able to make a custom sorting algorithm that they want. When adding a book in the API you have a lot of repeated code which is unnecessary and can be simplified by creating a method that hands creating and adding books and just pass in the information.

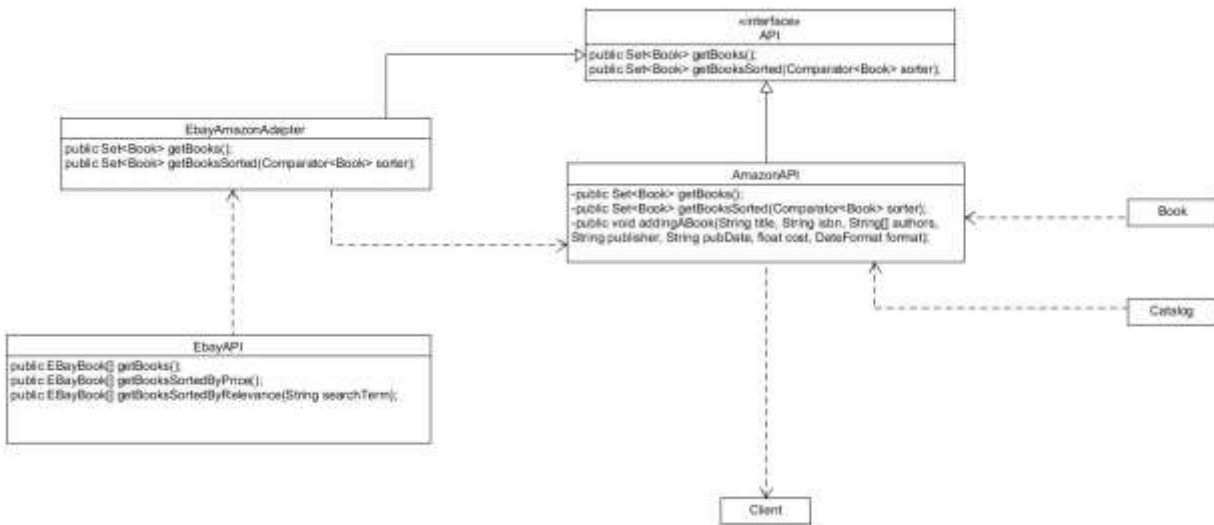
Feature 1 Design

For feature one I made the AmazonAPI class clean by creating a method to handle adding books to the catalog and also made it so the client can sort the books anyway that they want them by adding in a comparator. To complete this feature I used the template pattern. The user will have to supply their own sorter if they wish to have it sorted otherwise it is just FIFO style.



Feature 2 Design

For feature two I had to change the AmazonAPI and not the EbayAPI and make it so I am able to add all the EBay API into the Amazon API and display them both at the same time. To do this is needed to use the adapter pattern. I ended up adapting the EBay's books to be in the format of Amazon's books.



Feature 3 Design

For feature three I needed to create two different types of memberships, gold and diamond. The diamond member could see both EBay and Amazon books while the gold could only see Amazon books. To complete this feature I used the proxy pattern.

