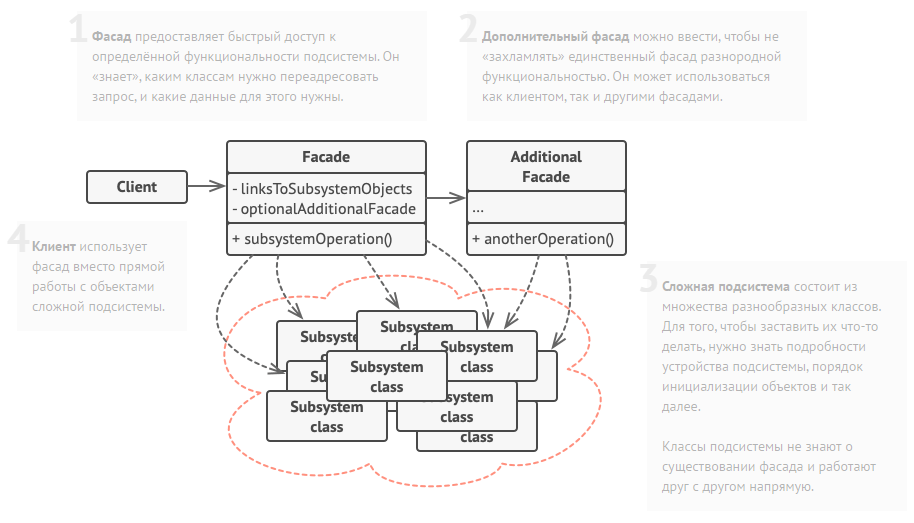
<https://refactoring.guru/ru/design-patterns/facade>

**Фасад** — это структурный паттерн проектирования, который предоставляет простой интерфейс к сложной системе классов, библиотеке или фреймворку.



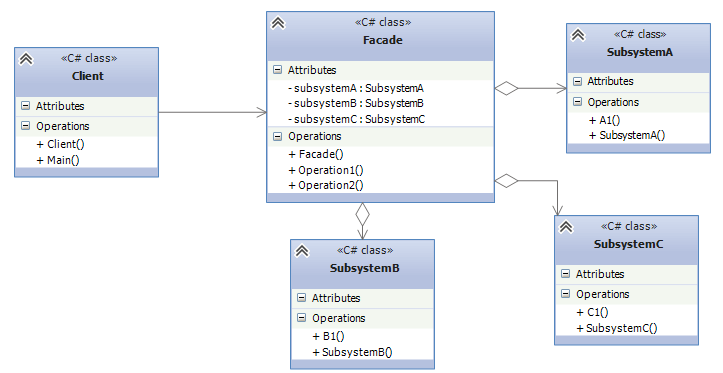
Фасад делегирует запросы клиентов соответствующим объектам внутри подсистемы. Фасад также отвечает за управление их жизненным циклом (но не обезательно). Все это защищает клиента от нежелательной сложности подсистемы.

Подсистема может принимать запросы либо от фасада, либо от клиента напрямую. В любом случае, для Подсистемы Фасад – это еще один клиент, и он не является частью Подсистемы.

<https://metanit.com/sharp/patterns/4.3.php>

Когда использовать фасад?

* Когда имеется сложная система, и необходимо упростить с ней работу. Фасад позволит определить одну точку взаимодействия между клиентом и системой.
* Когда надо уменьшить количество зависимостей между клиентом и сложной системой. Фасадные объекты позволяют отделить, изолировать компоненты системы от клиента и развивать и работать с ними независимо.
* Когда нужно определить подсистемы компонентов в сложной системе. Создание фасадов для компонентов каждой отдельной подсистемы позволит упростить взаимодействие между ними и повысить их независимость друг от друга.



<https://www.dofactory.com/net/facade-design-pattern>