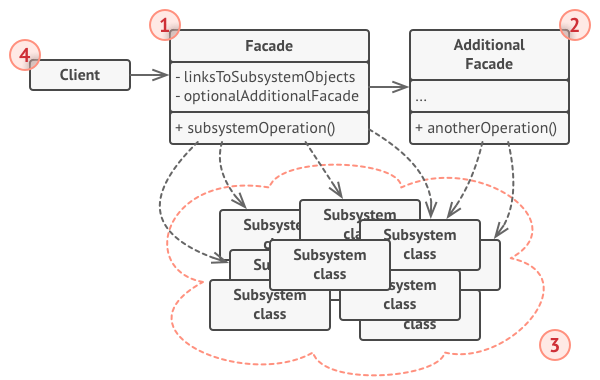
**ФАСАД** - предоставляет простой интерфейс к сложной системе классов, библиотеке или фреймворку. Также обеспечивает логическую изоляцию клиента от подсистемы, состоящей из многих компонентов - т.е. принцип общения только с друзьями. Основан на композиции и делегировании.

ПРИМЕНЕНИЕ:

- когда нужно представить простой или урезанный интерфейс к сложной подсистеме

- когда хотите разложить подсистему на отдельные слои

РЕАЛИЗАЦИЯ:

**1. ФАСАД**: предоставляет быстрый доступ к определённой функциональности подсистемы. Знает, каким классам нужно переадресовать запрос, и какие данные для этого нужны. Может использоваться как клиентом, так и другими фасадами. Например, "кнопка для подготовки всего к написанию трека" и "кнопка для завершения всего после написания трека".

**2. ДОПОЛНИТЕЛЬНЫЙ ФАСАД**: можно ввести дополнительный фасад, чтобы не «захламлять» единственный фасад разнородной функциональностью.

**3. СЛОЖНАЯ ПОДСИСТЕМА**: состоит из множества разнообразных классов. Чтобы заставить их что-то делать, нужно знать подробности устройства подсистемы, порядок инициализации объектов и так далее. Классы подсистемы не знают о существовании фасада и работают друг с другом напрямую. Например, "свет", "компьютер", "Reaper", "Миди устройство".

**4. КЛИЕНТ**: использует фасад вместо прямой работы с объектами сложной подсистемы. Например, я.

ПЛЮСЫ:

- изолирует клиентов от компонентов сложной подсистемы

МИНУСЫ:

- рискует стать божественным объектом, привязанным ко всем классам программы