

A decorative graphic on the left side of the slide. It consists of a blue parallelogram and a light green parallelogram, both tilted at an angle. The blue shape is in the foreground, and the green shape is partially behind it. They are set against a dark blue background with subtle diagonal lines.

Design Patterns

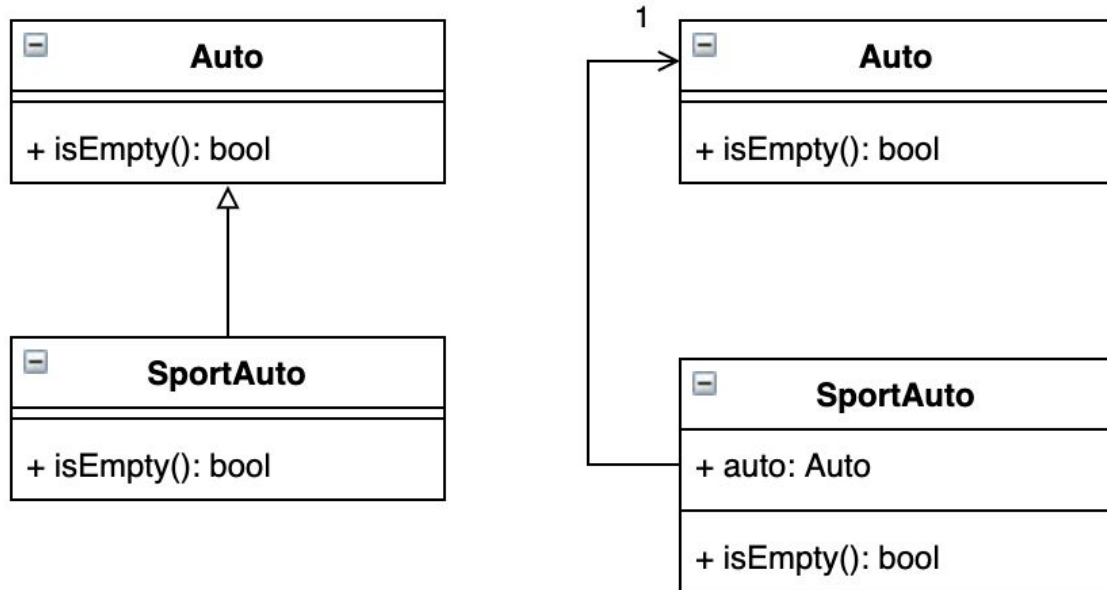
by Versus 2020

Delegation Делегирование

Фундаментальный шаблон



Суть шаблона





Проблема

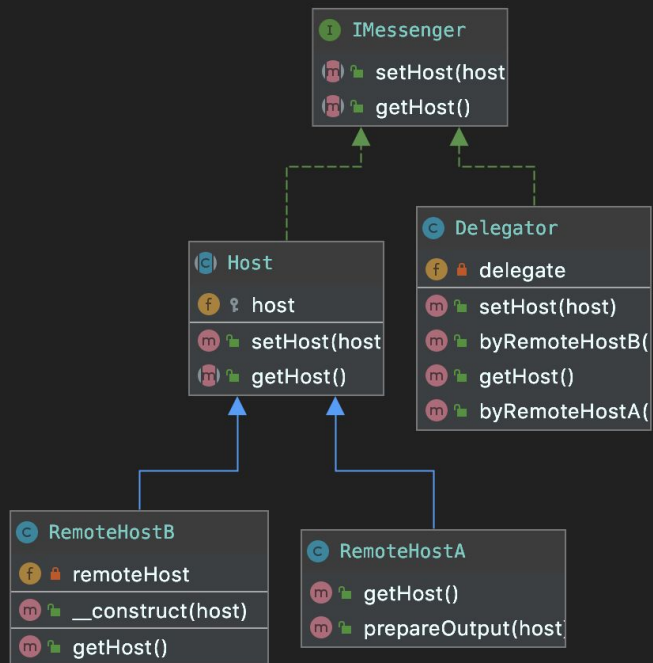
1. При расширении или изменении поведения объекта класса через наследование усложняет кодовую базу и увеличивает зависимости.
2. У вас есть подкласс, который использует только часть методов суперкласса или/и не хочет наследовать его данные.



Решение

1. Убрать наследование.
2. Композировать экземпляр бывшего класса-родителя в свойство бывшего класса-наследника.
3. Делегировать выполнение экземпляру класса-наследника.

Структура



Пример кода





Шаги реализации

1. Создайте поле в подклассе для содержания суперкласса. На первом этапе поместите в него текущий объект.
2. Измените методы подкласса так, чтобы они использовали объект суперкласса, вместо `this`.
3. Для методов, которые были унаследованы из суперкласса и которые вызываются в клиентском коде, в подклассе нужно создать простые делегирующие методы.
4. Уберите объявление наследования из подкласса.
5. Измените код инициализации поля, в котором хранится бывший суперкласс, созданием нового объекта.



Преимущества

- Возможность изменить поведение конкретного экземпляра объекта вместо создания нового класса путём наследования.
- Класс не содержит лишних методов, которые достались ему в наследство от суперкласса.
- В поле-делегат можно подставлять разные объекты, имеющие различные реализации функциональности. По сути, вы получаете реализацию паттерна проектирования Стратегия.



Недостатки

- Этот шаблон обычно затрудняет оптимизацию по скорости в пользу улучшенной чистоты абстракции.
- Приходится писать очень много простых делегирующих методов.

Связь с другими шаблонами



ИТОГ

