



---

# **Bài 22**

# **Structural Design Pattern**

Module: ADVANCED PROGRAMMING WITH JAVA

- Trình bày được ý nghĩa của các Structural Design Pattern
- Triển khai được Adapter Design Pattern
- Triển khai được Façade Design Pattern
- Triển khai được Proxy Design Pattern



---

# Structural Design Pattern

# Structural Design Pattern

---



- Structural Design Pattern là các mẫu thiết kế tập trung vào các mối liên quan giữa các thực thể
- Structural Design Pattern giúp tổ chức các lớp và đối tượng một cách hợp lý để tạo nên các cấu trúc lớn hơn (component, module...)
- Các Structural Design Pattern giúp cho việc thiết kế các mối quan hệ giữa các thực thể trở nên dễ dàng hơn

# Các Structural Design Pattern thông dụng

---



- Adapter
- Decorator
- Facade
- Proxy
- Bridge
- Composite
- Flyweight
- Private Class Data



---

# Adapter Pattern

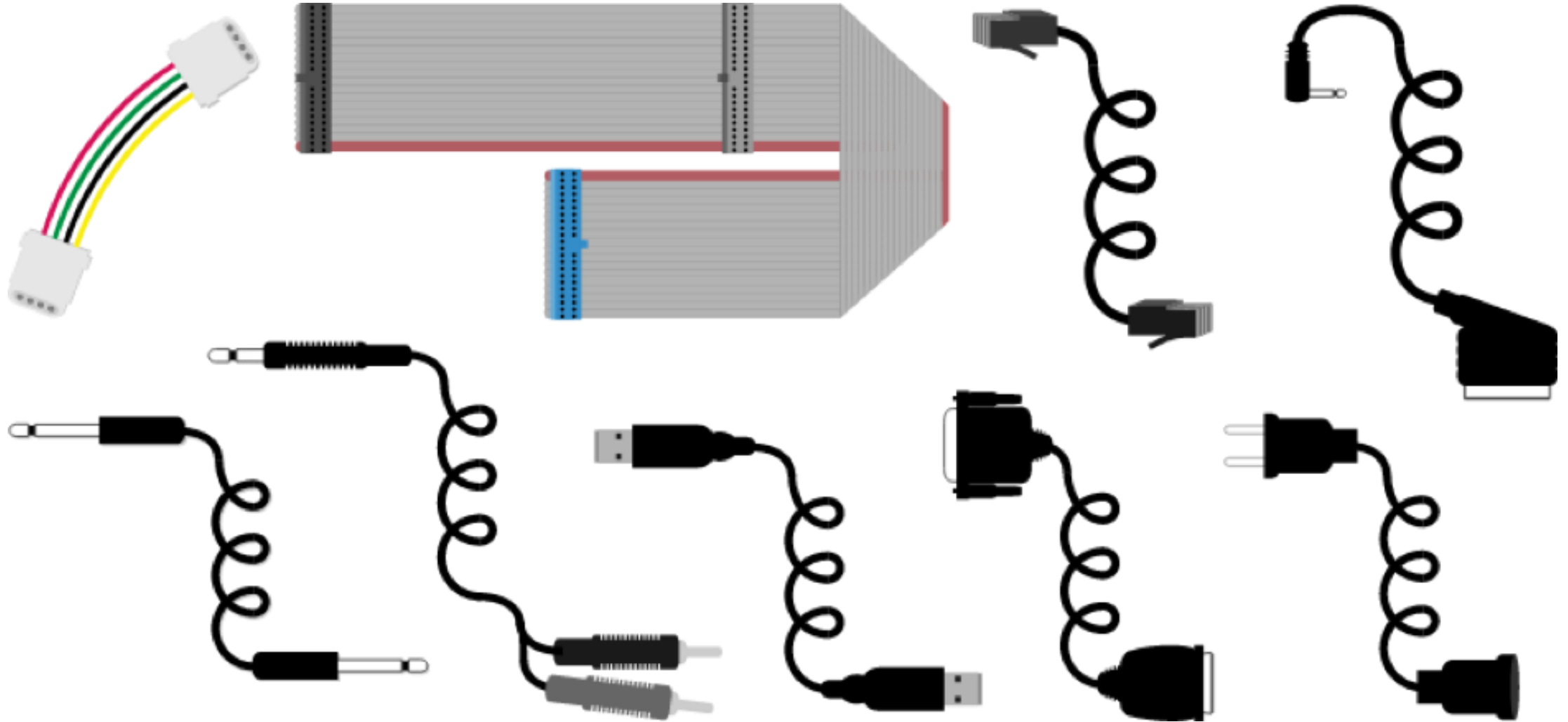
# Adapter Pattern

---



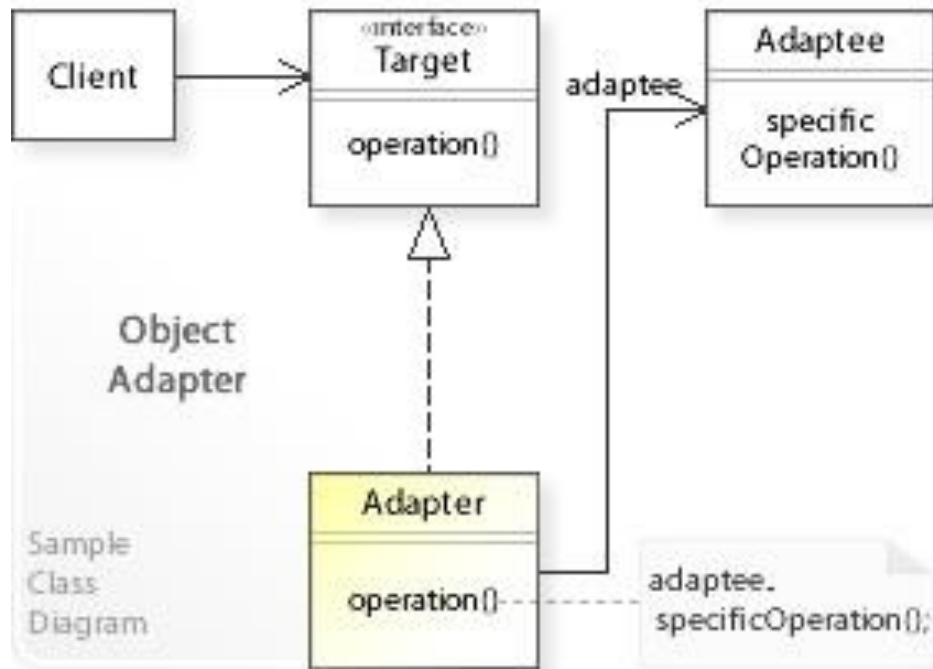
- Chuyển đổi giao diện của một hay nhiều lớp có sẵn thành một giao diện khác mà client mong muốn
- Cho phép các lớp có các giao diện khác nhau có thể dễ dàng giao tiếp tốt với nhau thông qua giao diện chuyển tiếp trung gian mà không cần thay đổi code của lớp có sẵn cũng như lớp đang viết.
- Cung cấp một giao diện “bọc ngoài” tương thích cho một hệ thống có sẵn
- Đối ứng phù hợp với các component cũ cho hệ thống mới

# Adapter trong đời sống

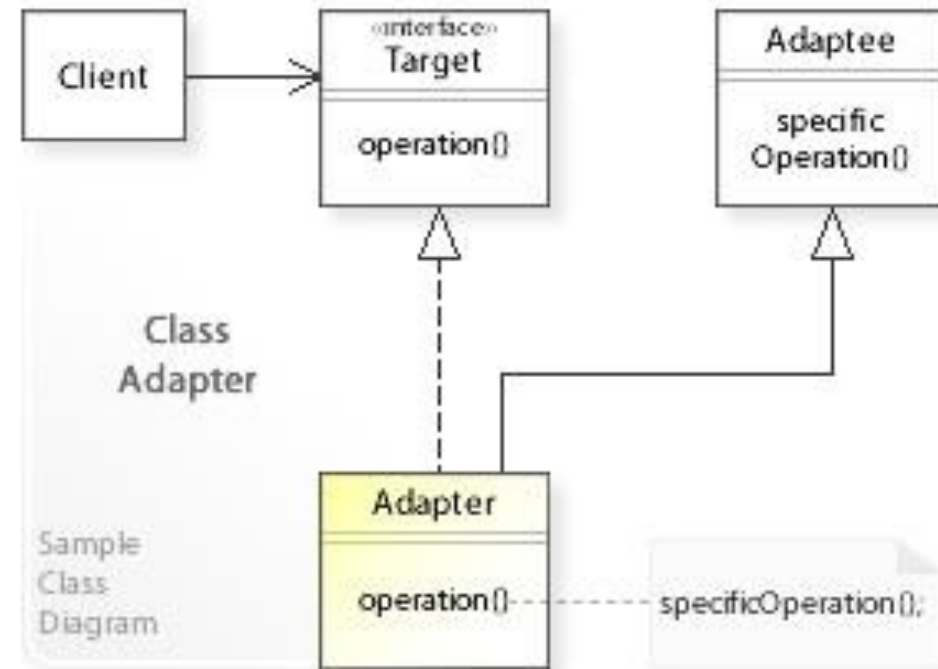




# Triển khai Adapter



Aggregation/Composition



Inheritance

# Demo

---





---

# Facade Design Pattern

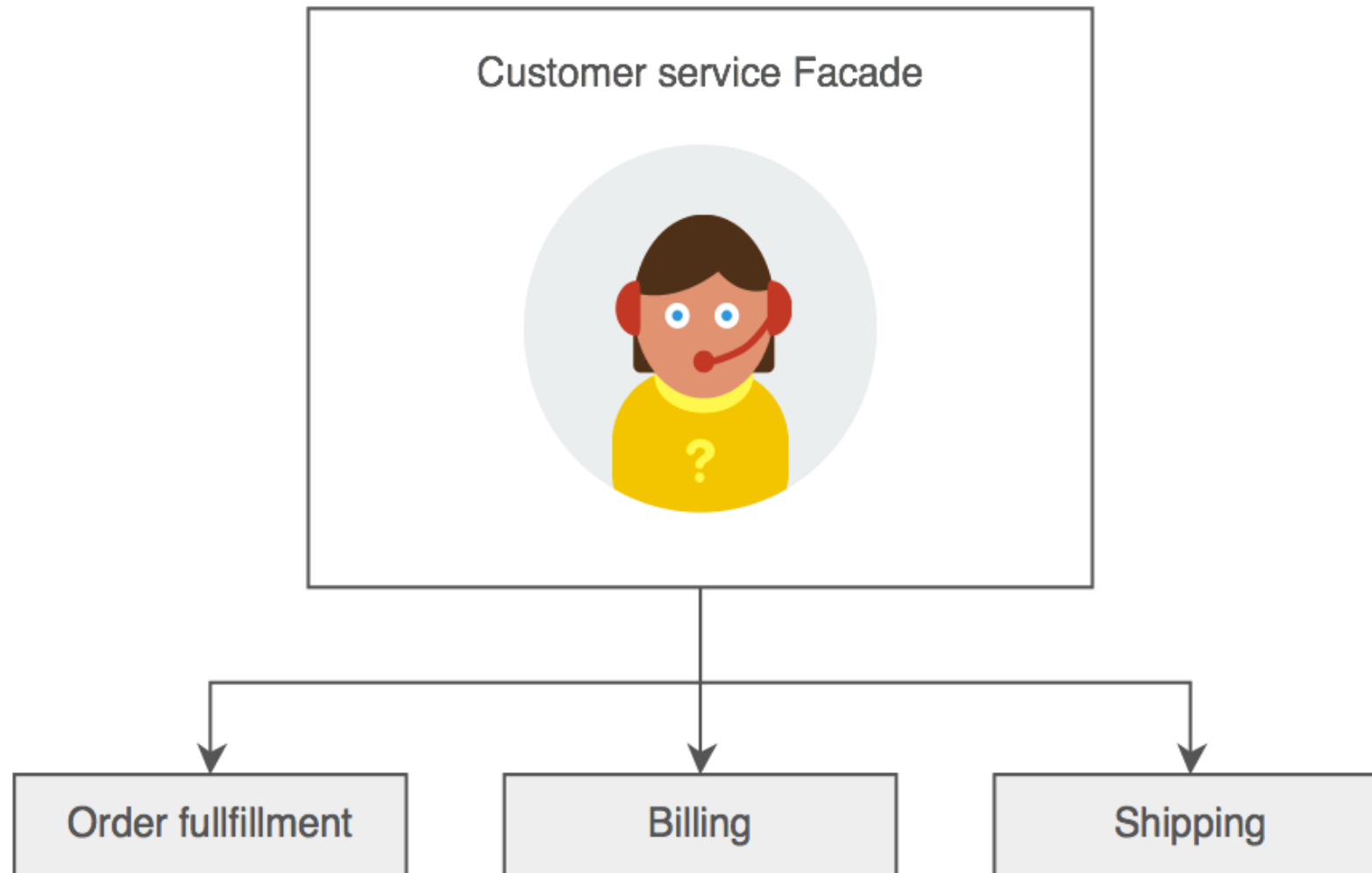
# Facade Design Pattern

---

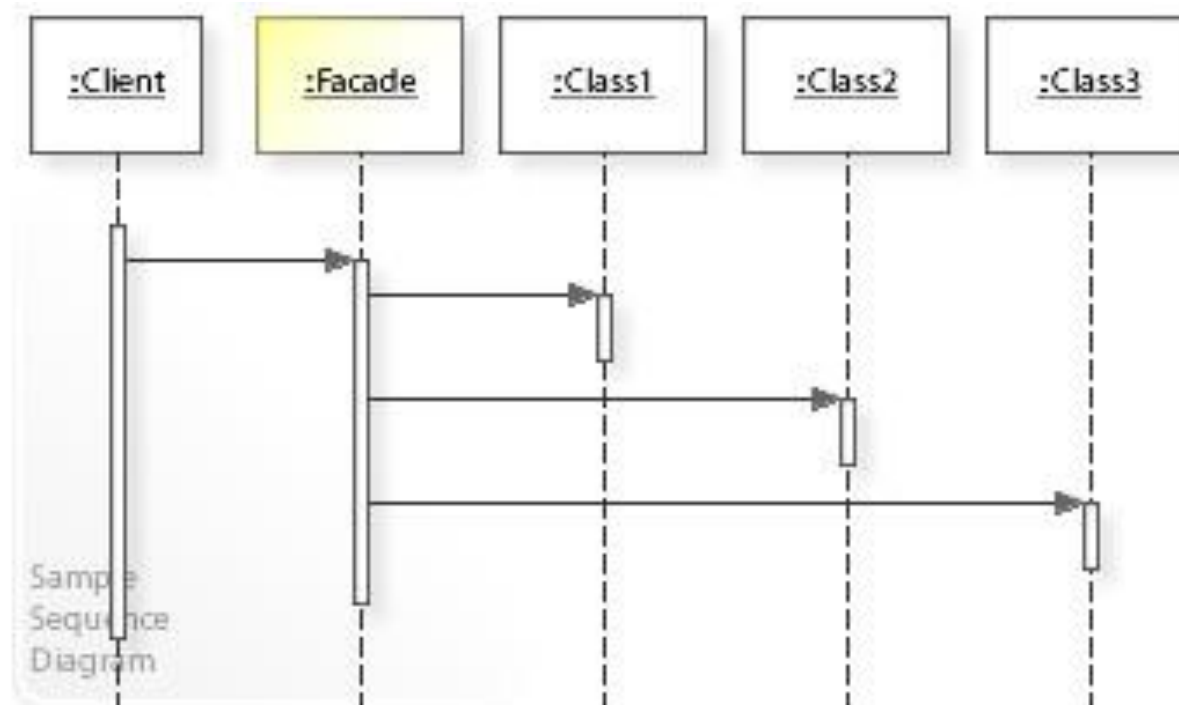
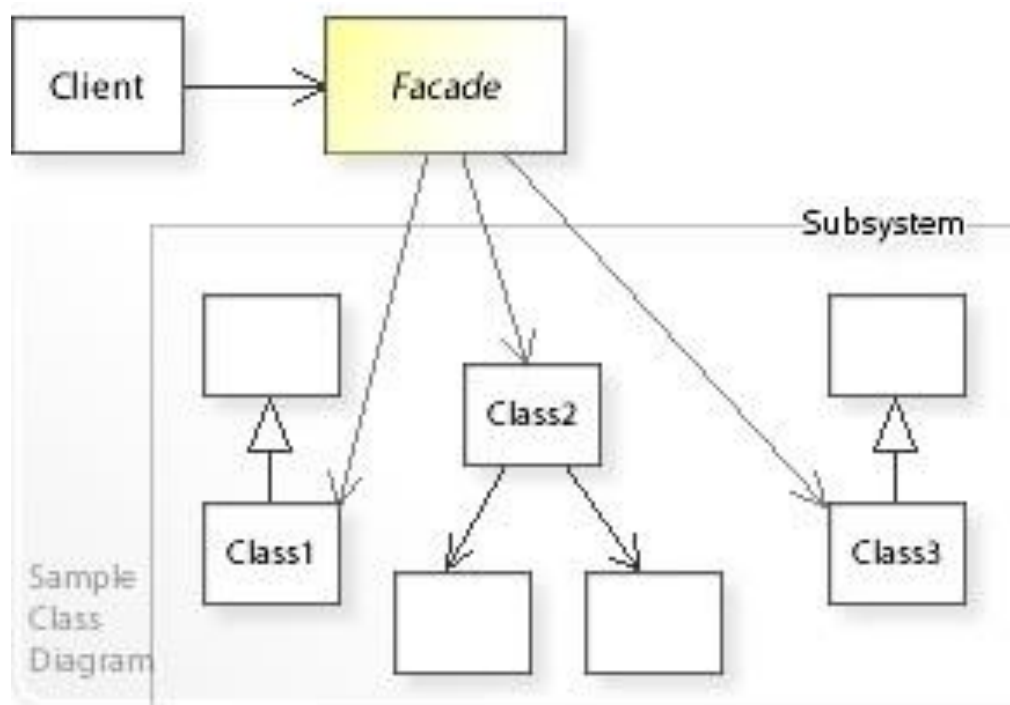


- Cung cấp một giao diện nhất quán cho các giao diện khác ở trong các hệ thống con
- Định nghĩa một giao diện ở mức cao (higher-level) để giúp cho các hệ thống con dễ sử dụng hơn
- Bao gói các hệ thống phức tạp với một giao diện đơn giản hơn

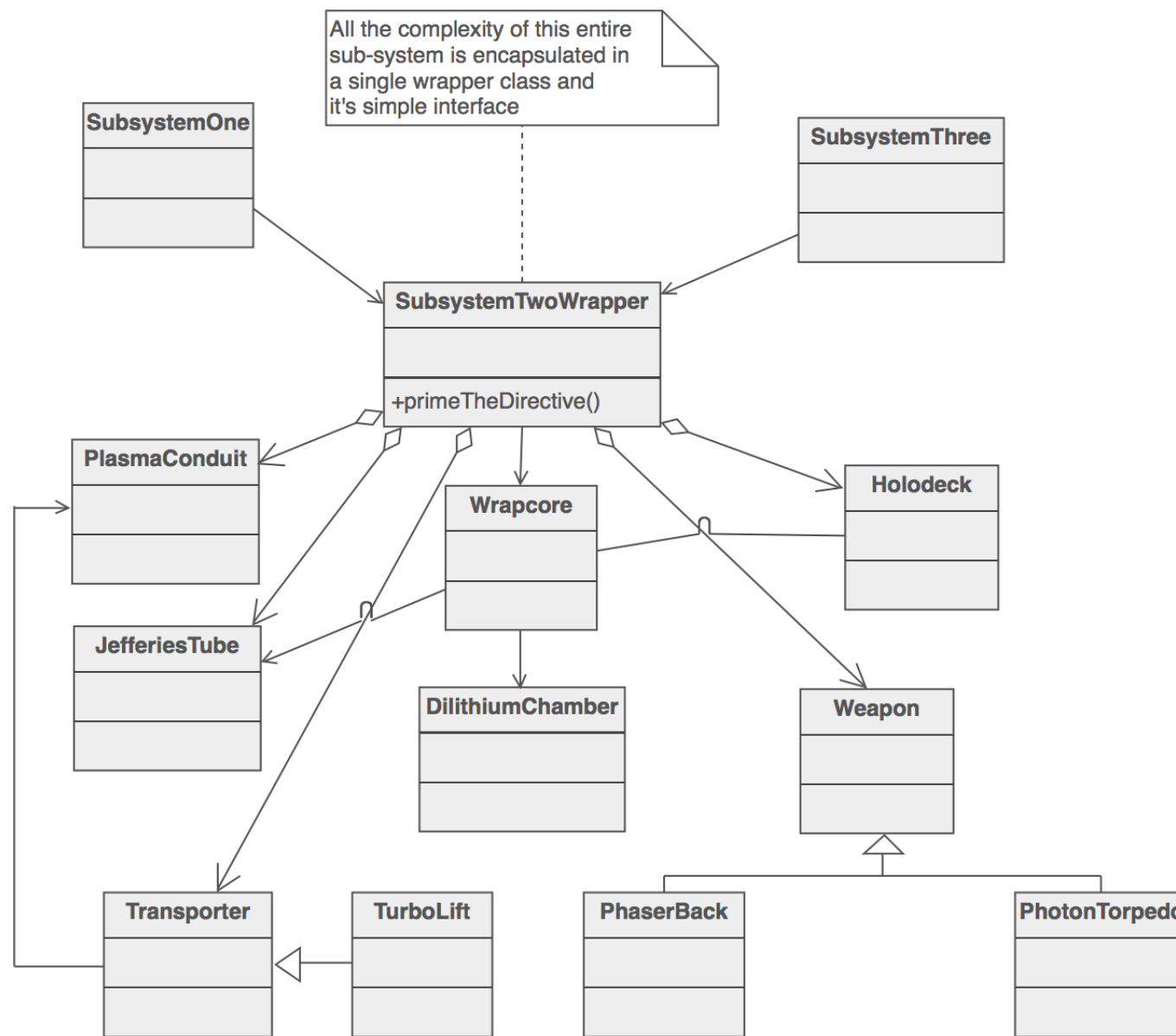
# Facade trong thực tế



# Triển khai Facade Design Pattern



# Triển khai Facade Design Pattern: Minh hoạ



# Demo Facade Design Pattern

---







---

# Proxy Design Pattern

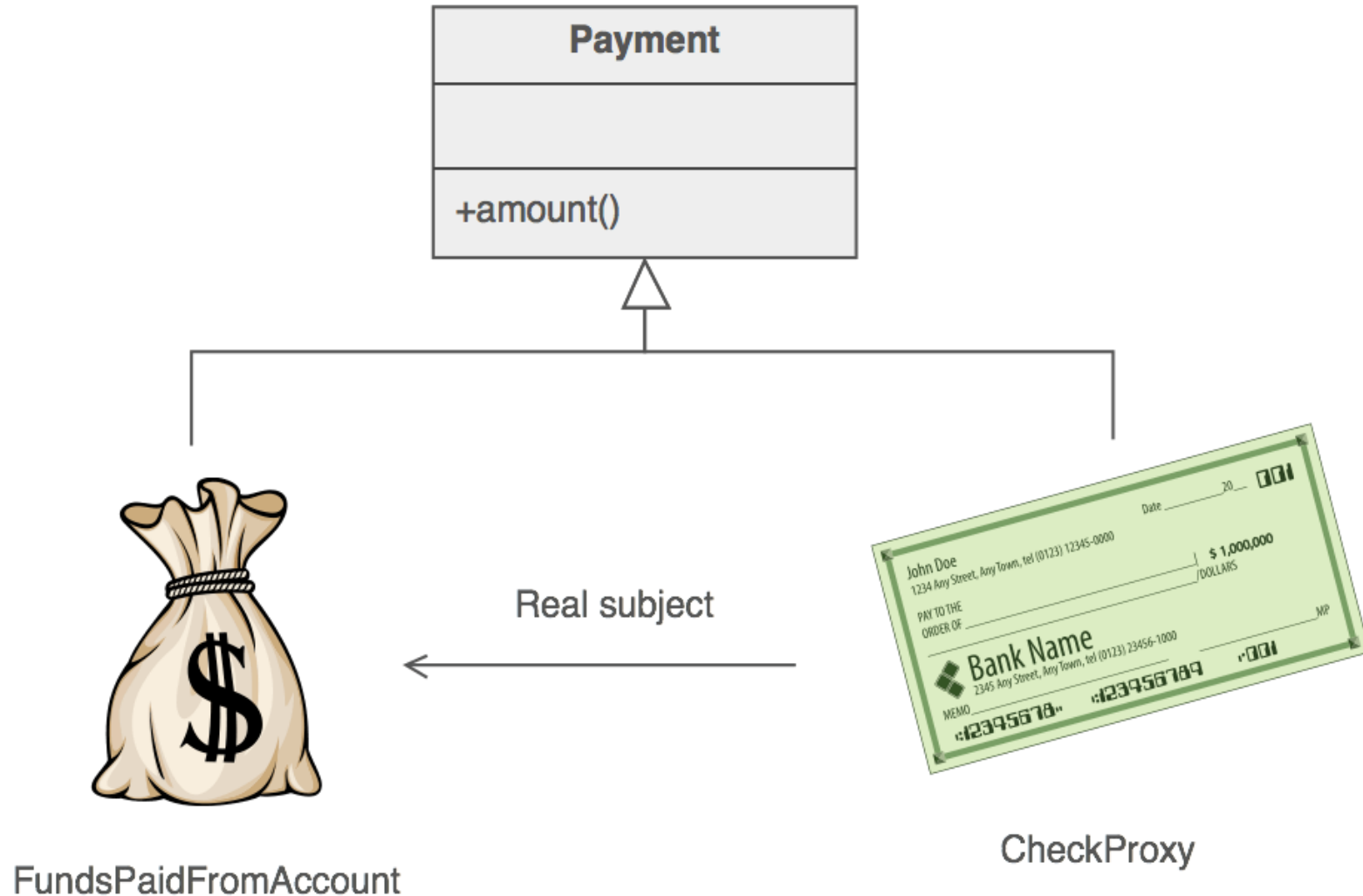
# Proxy Design Pattern

---

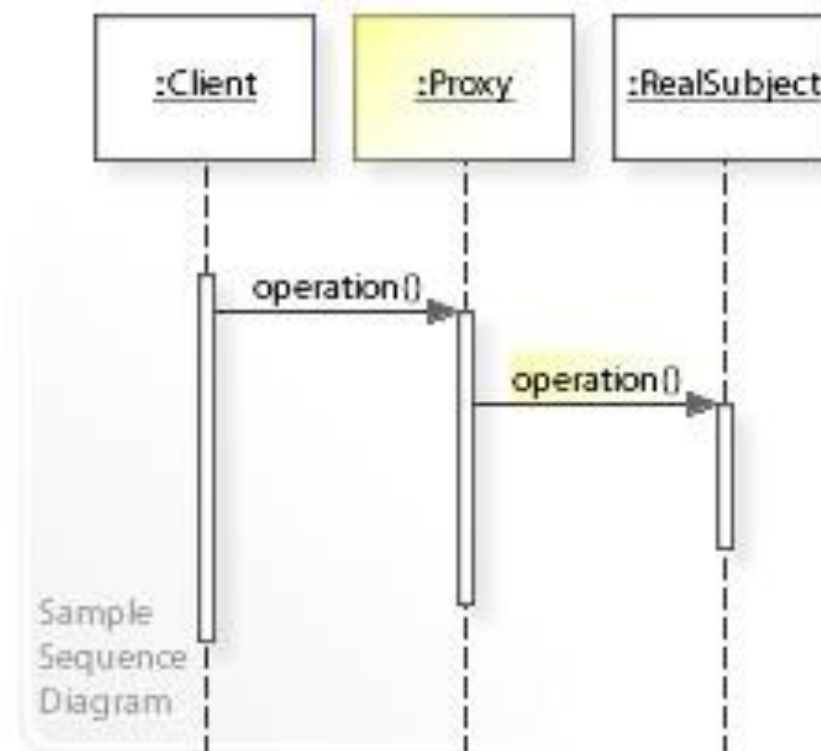
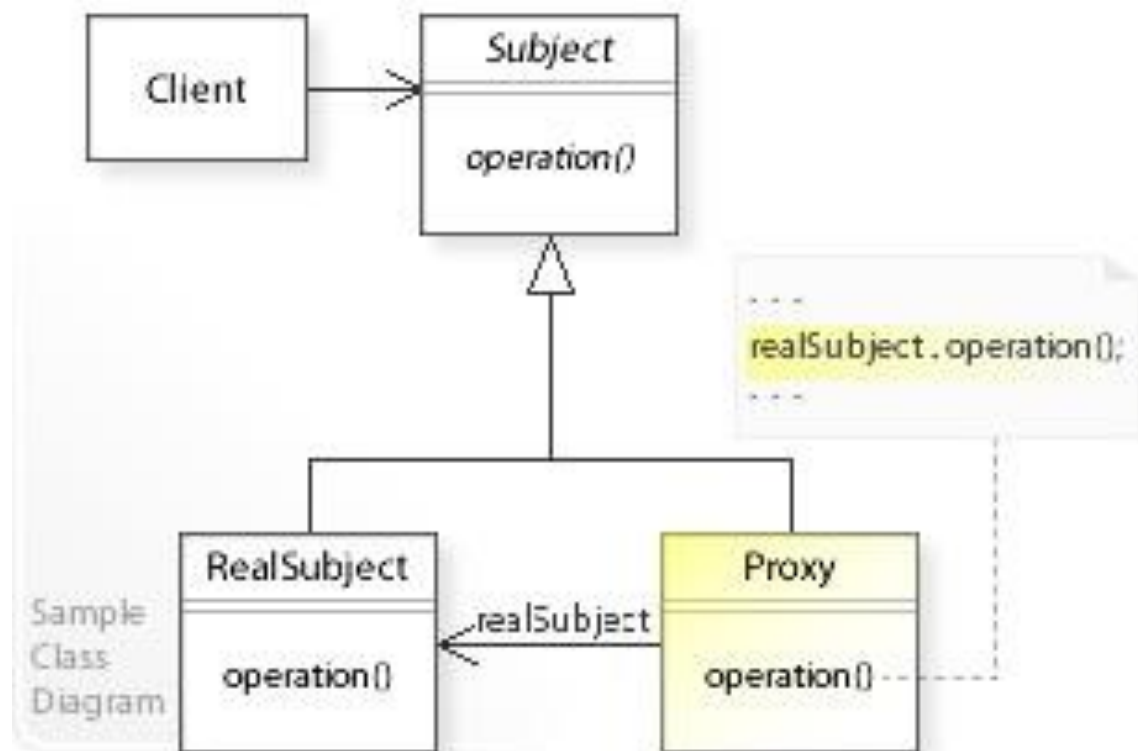


- Đưa ra một đại diện, thay thế cho một đối tượng để kiểm soát quyền truy cập vào nó.
- Tạo thêm một lớp trung gian để hỗ trợ các truy cập phân tán, được kiểm soát hoặc thống minh hơn
- Tạo một lớp bao gói bên ngoài đối tượng thật để bảo vệ nó khỏi các sự phức tạp khác không cần thiết

# Proxy Design Pattern trong đời sống



# Triển khai Proxy Design Pattern



# Demo Proxy Design Pattern

---



# [Thực hành] Triển khai Adapter

---



# [Thực hành] Triển khai Facade

---



# [Thực hành] Triển khai Proxy

---





# Tổng kết

---



- Structural Design Pattern giúp cho việc thiết lập mối quan hệ giữa các thực thể trở nên dễ dàng hơn
- Adapter Pattern chuyển đổi giao diện của một hay nhiều lớp có sẵn thành một giao diện khác mà client mong muốn
- Façade Pattern cung cấp một giao diện nhất quán cho các giao diện khác ở trong các hệ thống con
- Proxy Pattern đưa ra một đại diện, thay thế cho một đối tượng để kiểm soát quyền truy cập vào nó.

---

# Hướng dẫn

- Hướng dẫn làm bài thực hành và bài tập
- Chuẩn bị bài tiếp: Behavior Design Pattern