

СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ
SIBERIAN FEDERAL UNIVERSITY

Институт космических и информационных технологий
Объектно-ориентированное программирование

Контроллер

к.т.н. Якунин Юрий Юрьевич

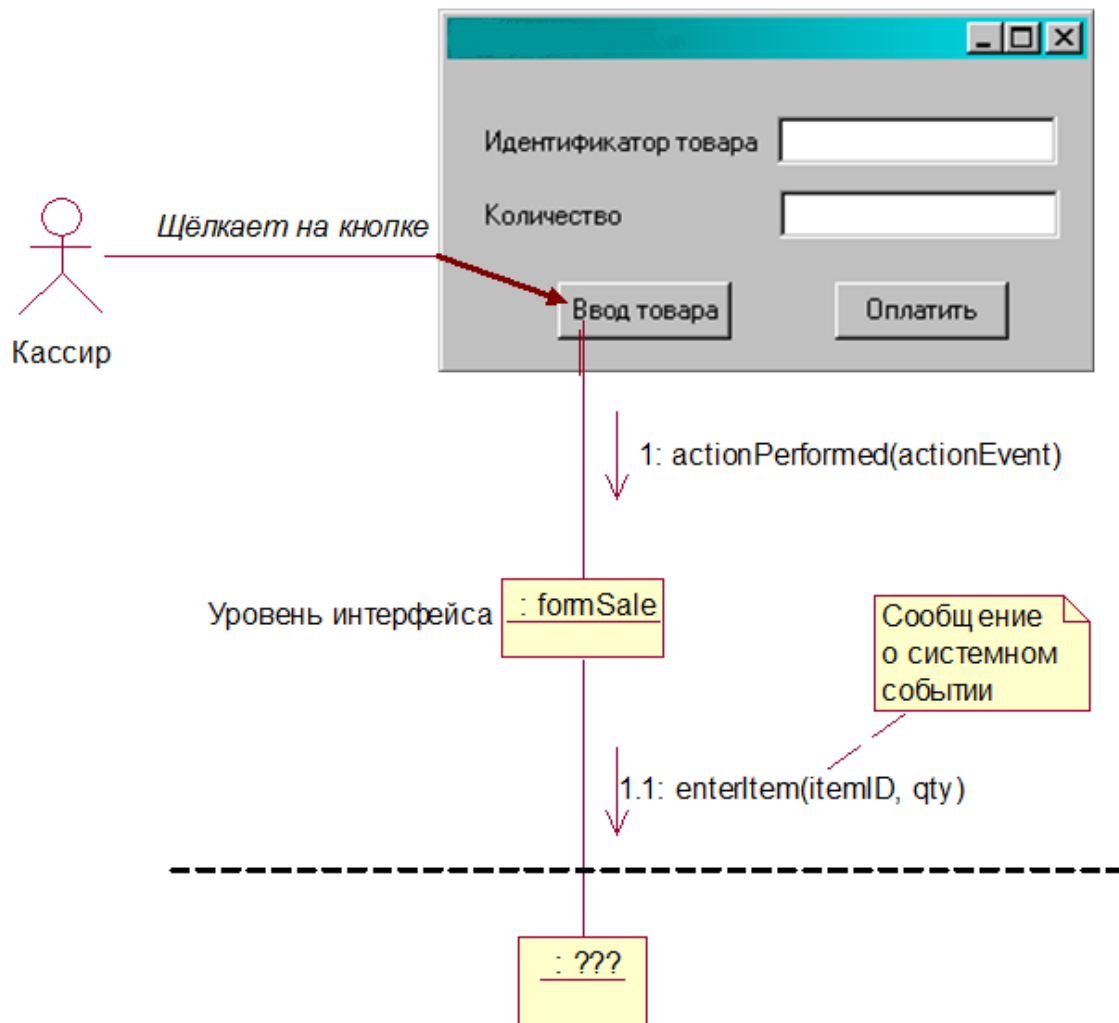
Шаблон Controller

Контроллер

- **Проблема.** Кто должен отвечать за обработку входных системных событий?
- **Решение.** Делегирование обязанностей по обработке системных сообщений классу, удовлетворяющему одному из следующих условий:
 - Класс представляет всю систему в целом, устройство или подсистему (внешний контроллер)
 - Класс представляет сценарий некоторого варианта использования, в рамках которого выполняется обработка всех системных событий, и обычно называется Handler, Coordinator или Session (контроллер варианта использования или контроллер сеанса)

Шаблон Controller

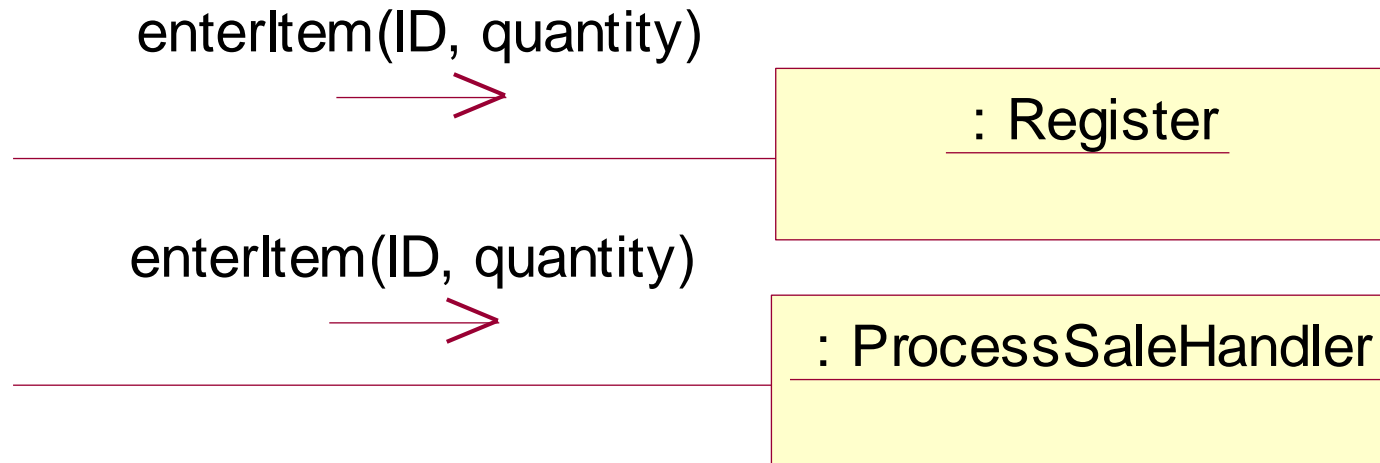
Пример



Какой класс
должен
выступать в роли
контроллера для
системных
событий
типа enterItem
или
endSale?

Шаблон Controller

Варианты решения



Выбор наиболее подходящего контроллера определяется зацеплением и связыванием

Шаблон Controller

Преимущества

- **Улучшение условий для повторного использования компонентов.** Применение этого шаблона обеспечивает обработку процессов предметной области на уровне реализации объектов, а не на уровне интерфейса. Делегирование выполнения системных операций специальному контроллеру облегчает повторное использование логики обработки подобных процессов в последующих приложениях
- **Контроль состояния варианта использования.** Иногда необходимо удостовериться, что системные операции выполняются в некоторой определённой последовательности. Например, необходимо гарантировать, чтобы операция `makePayment` выполнялась только после операции `endSale`, для чего необходимо накапливать информацию о последовательности событий. Для этой цели удобно использовать контроллер, особенно контроллер варианта использования