



C# Basic



Проверить, идет ли запись

Меня хорошо видно && слышно?



Observable, Immutable и Concurrent коллекции



Преподаватель



Ягур Алексей

Руководитель курса



Правила вебинара



Активно
участвуем



Off-topic обсуждаем
в Telegram
#csharp-basic-2023-04



Задаем вопрос
в чат или голосом



Вопросы вижу в чате,
могу ответить не сразу

Условные обозначения



Индивидуально



Время, необходимое
на активность



Пишем в чат



Говорим голосом



Документ



Ответьте себе или
задайте вопрос

Цели вебинара

После занятия вы сможете:

1. Строить коллекции с подпиской на события
2. Использовать Concurrent-коллекции для синхронизации доступа к коллекции из разных потоков
3. Работать с Immutable-коллекциями



План полета



Observable collections

Concurrent collections

Immutable collections

Кодинг

Рефлексия

Тестирование

Observable, Immutable и Concurrent коллекции

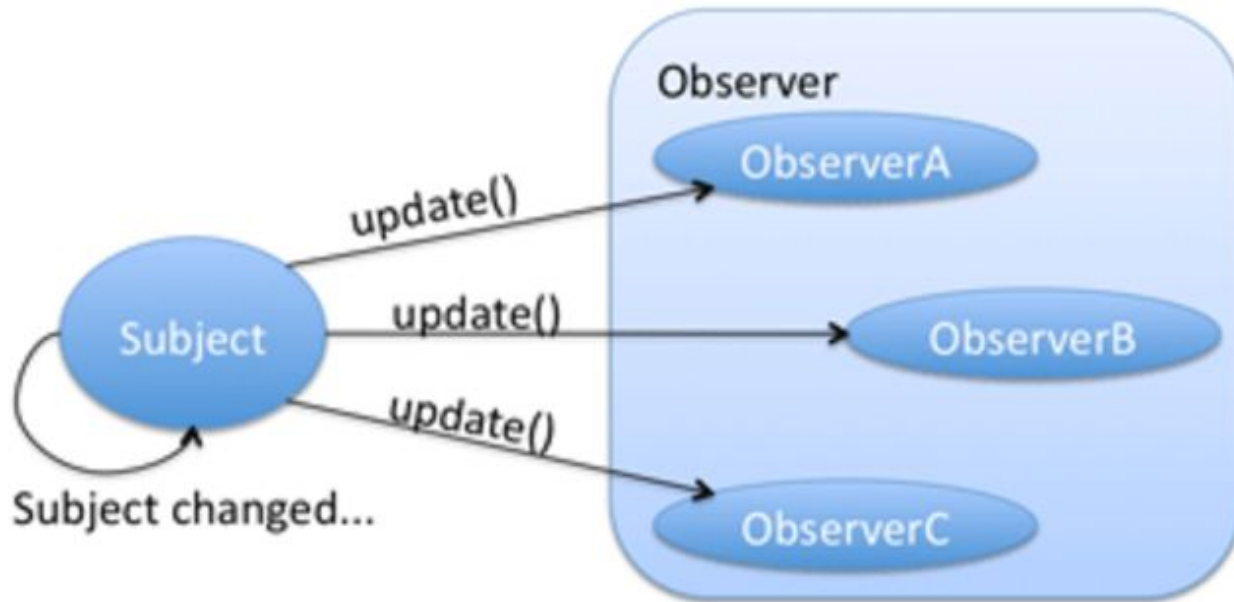
Observable collections



Что такое

- List + Observable паттерн
- Позволяет организовать подписку на события изменения коллекции

Что такое Observable паттерн?



Что такое ObservableCollection

- List + Observable паттерн
- Позволяет организовать подписку на события изменения коллекции
- Содержит `NotifyCollectionChangedEventHandler`? `CollectionChanged` для реагирования на изменение коллекций

Concurrent collections

Проблема

- Простая коллекция – какой поток создал коллекцию, тот ею и управляет
- Можно ускорить работу с коллекцией – несколько потоков
- Нужно организовать синхронизацию доступа к коллекции
- Простая коллекция – не подходит, непредсказуемое поведение

Решение

Concurrent collections

Варианты

- `ConcurrentDictionary`
- `ConcurrentBag`
- `ConcurrentStack`
- `ConcurrentQueue`
- `BlockingCollection`

Свойства

- В общем случае – медленнее не-concurrent
- API для защиты от «гонки потоков» (race condition)
- Можно безопасно перебирать элементы через foreach

Immutable collections

Синопис

- «Классические» коллекции (например, List) могут меняться в одном потоке
- В общем случае – удобно и безопасно
- Но иногда коллекция несет и семантический смысл (например, состояние базы данных)
- Тогда использование изменяемых коллекций может повредить читаемости кода
- Для этого существуют Immutable (неизменяемый) интерфейс

Варианты

- `ImmutableStack`
- `ImmutableList`
- `ImmutableArray`
- `ImmutableDictionary`

Практика

Выводы и рефлексия

Цели вебинара

Проверка достижения целей

1. Познакомились с Observable Collections и как ими пользоваться
2. Рассмотрели, как одновременная работа с коллекциями может быть нестабильной и как решать такие проблемы
3. Посмотрели концепцию неизменяемых коллекций

Рефлексия



По какому вопросу захотелось глубже изучить информацию?



Понимаете ли вы, как применять на практике то, что узнали на вебинаре. Если да, то как?

Следующий вебинар



6 сентября 2023

Модуль 5: Работа с коллекциям

LINQ запросы



Ссылка на вебинар
будет в ЛК за 15 минут



Материалы
к занятию в ЛК —
можно изучать



Обязательный материал
обозначен красной
лентой



Тестирование

**Заполните, пожалуйста,
опрос**

Спасибо за внимание!
Приходите на следующие
вебинары