Темы урока

[**Chatbot: Разработка слоя доступа к данным**](#_6zj5vaog9t7n) **1**

[Библиотека Reminder.Storage.Core: обсуждение](#_shy0dsukx5bn) 1

[Библиотека Reminder.Storage.InMemory: обсуждение](#_tq09zpf0h51i) 1

[Библиотека Reminder.Storage.Core: имплементация](#_dgvivtcn60ps) 1

[Библиотека Reminder.Storage.InMemory: имплементация](#_elxcr51y0eal) 2

[**Unit-тесты**](#_1aqfrvd3wmkr) **2**

[Самостоятельная работа](#_ibaofetivr8r) 2

[**Домашнее задание**](#_l7mphfdcuudx) **2**

# Chatbot: Разработка слоя доступа к данным

Основные сборки, отвечающей за логику работы собственно ремайндера:

**Reminder.Storage.Core**

* Библиотека с описанием интерфейсов и классов, которые будут использоваться конкретными реализациями хранилища данных
* Ей в пару будет создана сборка Reminder.Storage.Core.Tests для тестов логики классов, если она будет.

**Reminder.Storage.InMemory**

* Библиотека с реализацией хранилища данных в памяти. Реализует все интерфейсы Reminder.Storage.Core.
* Ей в пару будет создана сборка Reminder.Storage.InMemory.Tests для тестов логики хранилища.

**Reminder.Domain**

* Основная библиотека логики.

На этом уроке подробно рассматриваем только 2 первые

## Библиотека Reminder.Storage.Core: обсуждение

По слайду перечислить требования к содержанию.

## Библиотека Reminder.Storage.InMemory: обсуждение

По слайду перечислить требования к содержанию.

## Библиотека Reminder.Storage.Core: имплементация

Вместе пишем код:

* Storage.Core.ReminderItem класс
* Storage.Core.ReminderItemStatus перечисление
* Storage.Core.IReminderStorage интерфейс

## Библиотека Reminder.Storage.InMemory: имплементация

Вместе пишем код:

* Storage.InMemory.IReminderStorage интерфейс

На это отводится первая половина урока, т.е. нужно остановиться на том, на чём остановимся после первых двух часов. В следующий раз продолжим с этого места.

# Unit-тесты

Объяснять по слайдам, там всё прекрасно изложено.

Пример для демонстрации:

./Unit\_Tests/L18\_ClassWork\_UnitTests.sln

Обратить внимание, что тесты должны быть простыми, атомарными (желательно один Assert на тест).

Чем он быстрее читается, тем лучше - все значения лучше делать “in place”.

## Самостоятельная работа

Добавить к проектам Reminder.Storage.Core и Reminder.Storage.InMemory по одному проекту MSTest Test Project (.NET Core) с названиями

* Reminder.Storage.Core.Tests
* Reminder.Storage.InMemory.Tests

Покройте тестами

* Свойство TimeToAlarm класса ReminderItem библиотеки Reminder.Storage.Core,
* Метод Add(ReminderItem item) класса ReminderStorage библиотеки Reminder.Storage.InMemory.

# Домашнее задание

Завершить реализацию интерфейса IReminderStorage в библиотеке Reminder.Storage.InMemory и покрыть тестами код реализации в библиотеке Reminder.Storage.InMemory.Tests.