

Создание моделей в Entity Framework Core

№ урока: 2 **Курс:** Microsoft Entity Framework Core 2.0.

Средства обучения: Компьютер с установленной Visual Studio

Обзор, цель и назначение урока

- Создание моделей данных
- Настройка моделей данных
- Настройка конфигурации
- Миграции

Изучив материал данного занятия, учащийся сможет:

- Создавать собственные модели данных
- Настраивать сопоставления модели на БД
- Создавать собственные конфигурации

Содержание урока

- Модели, Fluent API и аннотации данных
- Определение моделей
- Свойства моделей
- Сопоставление таблиц и столбцов
- Настройка ключей и индексов
- Генерация значений свойств и столбцов
- Ограничения свойств
- Типы данных
- Конфигурация моделей

Резюме

- **Fluent API** представляет набор методов, которые определяют сопоставление между классами и их свойствами и таблицами и их столбцами. Как правило, функционал Fluent API задействуется при переопределении метода `OnModelCreating`
- **Аннотации** представляют настройку классов моделей с помощью атрибутов. Большинство подобных атрибутов располагаются в пространстве `System.ComponentModel.DataAnnotations`, которое нам надо подключить в файл `c#` перед использованием аннотаций.
- **Entity Data Model (EDM)** – модель сущностей или концептуальная модель между объектной моделью и БД, согласно которой определяются правила соответствия объектов элементам базы данных.
- **Модель Entity Data Model (модель EDM)** – это спецификация для определения данных, используемых приложениями, построенными на основе платформы Entity Framework. Приложения используют определенные моделью EDM сущности и связи в домене приложения в схеме макета. Схема макета используется для создания программируемых классов, используемых кодом приложения. Структуры хранилища, материализующие данные для приложений в этой модели, представлены в другой схеме, называемой схемой хранилища. Спецификация сопоставления соединяет схему макета и схему хранилища.

- Модель EDM использует три основных понятия для описания структуры данных: **тип сущности, тип ассоциации и свойство**. Это самые важные основные понятия в описании структуры данных в любой реализации модели EDM.
- **Тип сущности** – это фундаментальный блок построения для описания структуры данных при помощи модели EDM. В концептуальной модели типы сущностей конструируются из свойств и описывают структуру основных концептуальных элементов верхнего уровня, таких как клиенты и заказы в приложении предприятия.
- **Тип ассоциации** (также, называемый ассоциацией) – это фундаментальный блок построения для описания связей в модели EDM. В концептуальной модели ассоциация представляет собой связь между двумя типами сущностей (такими как клиент и заказ).
- **Типы сущностей** содержат свойства, которые определяют их структуру и характеристики. Например, тип сущности «Клиент» может иметь свойства, такие как идентификатор клиента, имя и адрес.
- **Миграции** позволяют вносить изменения в базу данных при изменениях моделей и контекста данных. В миграции определяются два метода: Up() и Down().

Закрепление материала

- Что позволяет сделать миграция
- С помощью каких инструментов можно настроить модель

Дополнительное задание

Используя Visual Studio, создайте проект по шаблону Console Application.

Требуется:

Используя подход Code First создайте две сущности с произвольными именами. Используя Fluent API и аннотации данных настроить их сопоставление на БД

Самостоятельная деятельность учащегося

Задание 1

Выучите основные понятия и техники создания EDM (Entity Data Model).

Задание 2

Используя Visual Studio, создайте проект по шаблону Console Application.

Требуется:

Используя подход Code First создайте три сущности с произвольными именами. Свяжите три сущности связью многие ко многим. Заполните сущности данными. Выведите данные в консоль.

Рекомендуемые ресурсы

https://ru.wikipedia.org/wiki/ADO.NET_Entity_Framework

[https://msdn.microsoft.com/ru-ru/library/ee382825\(v=vs.110\).aspx](https://msdn.microsoft.com/ru-ru/library/ee382825(v=vs.110).aspx)

<https://entityframework.codeplex.com/wikipage?title=specs>

<http://www.entityframeworktutorial.net/entityframework6/introduction.aspx>