Отношения между моделями. Наследование

№ урока: 3 **Курс:** Microsoft Entity Framework Core

Средства обучения: SQL Server Management Studio Express with Tools, Visual Studio

Обзор, цель и назначение урока

- Отношения между моделями.
- Наследование.
- Связи («один к одному», «один ко многим», «многие ко многим»).

Изучив материал данного занятия, учащийся сможет:

- Выполнять настройку связи-отношения между моделями.
- Связывать сущности.
- Понимать работу EF Core при наследовании сущностей.

Содержание урока

- Внешние ключи и навигационные свойства.
- Каскадное удаление.
- Загрузка связанных данных.
- Один к одному.
- Один ко многим.
- Многие ко многим.
- Подход TPH.

Резюме

- **Entity Data Model** (EDM) – модель сущностей или концептуальная модель между объектной моделью и БД, согласно которой определяются правила соответствия объектов элементам базы данных.
- **Внешний ключ** (FK) — это столбец или сочетание столбцов, которое применяется для принудительного установления связи между данными в двух таблицах. Внешний ключ можно создать, определив ограничение FOREIGN KEY при создании или изменении таблицы.
- По умолчанию название внешнего ключа должно принимать одно из следующих вариантов имени:
- **Имя навигационного свойства** + **Имя ключа** из связанной сущности.
  Имя класса связанной сущности + **Имя ключа** из связанной сущности.
- Отношения между таблицами можно определить в модели только за счет использования навигационных свойств, которые могут быть указаны в обеих таблицах (двусторонняя связь) или в одной таблице (односторонняя связь).
- Атрибут **ForeignKey** используется для настройки внешнего ключа во взаимосвязи между двумя объектами в EF 6 и EF Core. Он отменяет соглашения по умолчанию. Согласно стандарту, EF создает свойство как свойство внешнего ключа, когда его имя совпадает с свойством первичного ключа связанного объекта.
- **Каскадное удаление** – представляет автоматическое удаление зависимой сущности после удаления главной.

CyberBionic systematics

CyberBionic Systematics ® 2018
19 Eugene Sverstyuk Str., 5 floor
Kyiv, Ukraine

Tel. 0 800 750 312
E-mail: edu@cbsystematics.com
Site: www.edu.cbsystematics.com | itvdn.com

- В Fluent API доступны три разных сценариях, которые управляют поведением зависимой сущности в случае удаления главной сущности:
- **Cascade**: зависимая сущность удаляется вместе с главной.
- **SetNull**: свойство-внешний ключ в зависимой сущности получает значение null.
- **Restrict**: зависимая сущность никак не изменяется при удалении главной сущности.
- **Eager loading** – позволяет загружать связанные данные с помощью метода Include(), в который передается навигационное свойство.
- **Explicit loading** – предполагает явную загрузку данных с помощью метода Load.
- **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.
- **Связь один к одному**. При связи один-к-одному одна модель хранит ссылку на один объект другой модели.
- **Связь один ко многим.** Связь один-ко-многим реализуется, если одна модель хранит ссылку на один объект другой модели, а вторая модель может ссылаться на коллекцию объектов первой модели.
- **Связь многие ко многим**. Обе модели имеют свойства-коллекции, через которые и будет осуществляться связь многие-ко-многим.
- При использовании подхода **TPH** (Table Per Hierarchy / Таблица на одну иерархию классов) для одной иерархии классов используется одна таблица. Данные базовых и производных классов сохраняются в одну таблицу, а для их отличия создается специальный столбец.

Закрепление материала

- Что такое каскадное удаление?
- Что такое внешний ключ?
- Перечислите связи.

Дополнительное задание

Используя Visual Studio, создайте проект по шаблону Console Application.
Требуется:
Используя подход Code First создайте две сущности с произвольными именами. Свяжите две сущности связью один ко многим. Заполните сущности данными. Выведите данные в консоль.

Самостоятельная деятельность учащегося

Задание 1
Используя Visual Studio, создайте проект по шаблону Console Application.
Требуется:

CyberBionic systematics
CyberBionic Systematics ® 2018
19 Eugene Sverstyuk Str., 5 floor
Kyiv, Ukraine
Tel. 0 800 750 312
E-mail: edu@cbsystematics.com
Site: www.edu.cbsystematics.com | itvdn.com

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://entityframework.codeplex.com/wikipage?title=specs

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

CyberBionic systematics

CyberBionic Systematics ® 2018
19 Eugene Sverstyuk Str., 5 floor
Kyiv, Ukraine

Tel. 0 800 750 312
E-mail: edu@cbsystematics.com
Site: www.edu.cbsystematics.com | itvdn.com