|  |
| --- |
|  |
| Что такое ORM? |
|  |  |
|  | Объектно-реляционное отображение-технология программирования, которая создает слой между реляционными базами данных и объектно-ориентированными языками программирования без необходимости написания SQL-запросов. |
|  | ORM используется для упрощения процесса сохранения объектов в реляционную базу данных и их извлечения. |
|  |  |
|  | 2.Зачем используются ORM-решения при работе с базами данных? |
|  |  |
|  | Использование ORM в проекте избавляет разработчика от необходимости работы с SQL и написания большого количества кода, часто однообразного и подверженного ошибкам. |
|  | Весь генерируемый ORM код предположительно хорошо проверен и оптимизирован, поэтому не нужно в целом задумывается о его тестировании. |
|  | Это несомненно является плюсом, но в тоже время не стоит забывать и о минусах. |
|  | Основной из них — это потеря производительности. |
|  | Это происходит потому, что большинство ORM предназначены для обработки широкого спектра сценариев использования данных, гораздо большего, чем любое отдельное приложение когда-либо сможет использовать. |
|  | Кроме того, большинство современных ORM позволяют программисту при необходимости самому задавать код SQL-запросов. |
|  | Без сомнений, для небольших проектов использование ORM будет куда более оправдано, чем разработка собственных библиотек для работы с БД. |
|  |  |
|  | +меньший объем кода |
|  | +авто исп-ние паттернов – дизайн |
|  | +код хорошо протестирован |
|  | -потеря производительности |
|  |  |
|  |  |
|  | 3.Что такое Entity Framework? |
|  |  |
|  | предоставляет возможность работы с Базой данных через C# код,предоставляет возможность работы с Базой данных через C# код. |
|  | Entity Framework представляет специальную объектно-ориентированную технологию на базе фреймворка .NET для работы с данными. |
|  | Если традиционные средства ADO.NET позволяют создавать подключения, команды и прочие объекты для взаимодействия с базами данных, то Entity Framework представляет собой более высокий уровень абстракции, который позволяет абстрагироваться от самой базы данных и работать с данными независимо от типа хранилища. |
|  | Если на физическом уровне мы оперируем таблицами, индексами, первичными и внешними ключами, но на концептуальном уровне, который нам предлагает Entity Framework, мы уже работает с объектами. |
|  |  |
|  | 4.Что такое EDM в Entity Framework и из каких уровней она состоит? |
|  |  |
|  | Entity Data Model |
|  | Эта модель сопоставляет классы сущностей с реальными таблицами в БД. |
|  | Entity Data Model состоит из трех уровней: концептуального, уровень хранилища и уровень сопоставления (маппинга). |
|  | На концептуальном уровне происходит определение классов сущностей, используемых в приложении. |
|  | Уровень хранилища определяет таблицы, столбцы, отношения между таблицами и типы данных, с которыми сопоставляется используемая база данных. |
|  | Уровень сопоставления (маппинга) служит посредником между предыдущими двумя, определяя сопоставление между свойствами класса сущности и столбцами таблиц. |
|  |  |
|  | 5. Перечислите способы создания моделей. Опишите их действия. |
|  |  |
|  | \*Database first – EF создает набор классов, кот.отраж.модель конкр.БД |
|  | \*Model first – сначала разработчик созд.модель БД, по которой затем EF создает реальную БД на сервере |
|  | \*Code first – разраб.созд.класс модели д-х, кот.будут хранится в БД, а затем EF генерирует БД и ее табл. |
|  |  |
|  | 6. Назовите основные отличия EF от других технологий работы с базами |
|  | данных. |
|  |  |
|  | Если традиционные средства ADO.NET позволяют создавать подключения, команды и прочие объекты для взаимодействия с базами данных, |
|  | то Entity Framework представляет собой более высокий уровень абстракции, |
|  | который позволяет абстрагироваться от самой базы данных и работать с данными независимо от типа хранилища. |
|  |  |
|  | 7. Для чего и как используется DbContext? |
|  |  |
|  | Является базовым классом Entity Framework и предоставляет широкие возможности по работе с базой данных: создание запросов, отслеживание изменений и сохранение данных в базе. |
|  |  |
|  | 8.Как и для чего используются аннотации при настройке конфигураций |
|  | Code First? |
|  |  |
|  | Аннотации представляют настройку сопоставления моделей и таблиц с помощью атрибутов. |
|  | Большинство классов аннотаций располагаются в пространстве System.ComponentModel.DataAnnotations, которое нам надо подключить в файл c# перед использованием аннотаций. |
|  |  |
|  | 9. Как и для чего используются Fluent API при настройке конфигураций Code First? |
|  |  |
|  | Набор методов, которые определяют сопоставление между (классами и их свойствами) и (таблицами и их столбцами). |
|  |  |
|  | 10.Какие существуют способы загрузки связанных данных? |
|  |  |
|  | \*жадная загрузка (eager) |
|  | позволяет указать в запросе какие связанные д-е нужно загрузить при выполнении запроса |
|  | \*ленивая загрузка (lazy) |
|  | EF автомат.загружает д-е, при этом не загружая связанные д-е. |
|  | \*явная загрузка (explicit) |
|  | не приводит к загрузке всех связ.д-х в запросе, но при вызове навигацонного св-ва свзяанного класса, эта хагрузка не приводит к авто извлечению д-х |
|  |  |
|  | 11.В чем разница между концепциями Code First и Database First? |
|  |  |
|  | Database First и встроенный дизайнер как ни на есть лучшим образом, за исключением некоторых тонкостей (Code First их впрочем не решает), позволяют проектировать базу и ни какие скрипты писать не нужно. Более того, все необходимые классы также создаются автоматически. |
|  | Более того, Database First позволяет легко обновлять модель из базы, если такое нужно в последствии, а также легко и удобно работать с Stored Proceudres и Views. |
|  | Code-First – новая возможность, стоящая в одном ряду с Model-First и Database-First. Теперь можно сначала написать код, описывающий классы-модели, а потом фреймворк автоматически создаст БД по такому коду. |
|  | Самое приятное, что виды отношений будут распознаны – достаточно определить просто ссылки на объекты для 1:1, ICollection для отношения 1:n, и взаимные ICollection для m:n. В этом случае промежуточная таблица также создастся автоматически. Для обеспечения «ленивой» загрузки хватит ключевого слова virtual в определении свойства. |
|  |  |
|  | 12.Зачем EF использует миграции? Как создать новую миграцию? |
|  |  |
|  | Миграции позоляют вносить изменения в базу данных при изменениях моделей и контекста данных. |
|  |  |
|  | 13.Расскажите алгоритм выполнения CRUD-запросов при работе с БД |
|  | через библиотеку Entity Framework |
|  |  |
|  | CRUD(CREATE, READ, UPDATE and DELETE) – обозначает3 базовые функции, используемые при работе с БД. |
|  | Добавление-add. |
|  | Удаление-Remove. |
|  | Изменение-Entity Framework сам отслеживает все изменения, и когда вызывается метод SaveChanges(), будет сформировано SQL-выражение UPDATE для данного объекта, которое обновит объект в базе данных. |
|  | Базовым подходом к написанию запросов в Entity Framework является использование расширения языка C# - LINQ, которое предоставляет набор методов для работы с коллекциями. |
|  |  |
|  |  |