**Вопросы 9 ООП**

1. Что такое ORM?

ORM - это Object-Relational Mapping, технология программирования, которая позволяет связывать объекты в коде с записями в базе данных. Она позволяет работать с базами данных через объекты, нам не нужно писать SQL-запросы вручную.

ORM упрощает процесс работы с базами данных. Предоставляет простой способ выполнения операций CRUD (create, read, update, delete)

2. Зачем используются ORM-решения при работе с базами данных? Опишите их достоинства и недостатки.

ORM-решения

– *упрощают процесс работы* с базами данных и сокращают количество кода, необходимого для выполнения задач, связанных с БД.

– Они позволяют *ускорить разработку* и снизить количество ошибок и опечаток, которые могут возникнуть при ручном написании SQL-запросов.

– *Кроссплатформенность*: ORM позволяет работать одинаково с различными реляционными СУБД,

Однако, использование ORM-решений может привести к **снижению производительности** при выполнении сложных запросов.

3. Что такое Entity Framework?

Entity Framework - это ORM-решение от Microsoft, предназначенное для работы с базами данных на платформе .NET. Оно позволяет работать с различными СУБД, в том числе с MS SQL Server, PostgreSQL, MySQL и другими.

4. Что такое EDM в Entity Framework и из каких уровней она состоит?

EDM - Entity Data Model - это модель данных в Entity Framework, которая состоит из трех уровней:

* концептуальный (описывает сущности и их отношения)
* хранилище (описывает хранилище базы данных) и
* соединительный (описывает маппинг между концептуальным и хранилище слоями).

5. Перечислите способы создания моделей. Опишите их действия.

Существуют следующие способы создания моделей: Code First, Database First и Model First.

**Code First** позволяет создавать модели на основе классов в коде,

**Database First** – генерируем классы на основе существующей базы данных

**Model First** - путем создания модели сущностей визуально в дизайнере.

6. Назовите основные отличия EF от других технологий работы с базами данных.

Основным отличием EF от других технологий работы с базами данных является то, что *EF позволяет работать с БД, используя объекты в коде, а не SQL-запросы.*

EF также *позволяет выполнить миграции* схемы базы данных в коде и контролировать изменения базы данных в процессе разработки приложения.

EF может использоваться в различных архитектурных решениях, таких как Code First, Model First и Database First.

7. Для чего и как используется DbContext?

DbContext - это класс в Entity Framework, который используется для взаимодействия с базой данных Он предоставляет доступ к DbSet, которые представляют собой коллекции объектов из БД, и к другим функциям для работы с БД.

8. Как и для чего используются аннотации при настройке конфигураций Code First?

Аннотации используются для настройки конфигураций моделей в Code First, таких как указание первичного ключа, индексов, ограничений и другой схемы для маппинга объектов в БД. Они также могут использоваться для указания правил валидации

9. Как и для чего используются Fluent API при настройке конфигураций Code First?

Fluent API - это более гибкий способ настройки конфигураций моделей в Code First, позволяющий более точную настройку маппинга объектов в БД. Он используется для указания сложных связей между сущностями, индексации, ограничений и других деталей маппинга.

10. Какие существуют способы загрузки связанных данных?

Существуют три способа загрузки связанных данных: жадная загрузка, ленивая загрузка и явная загрузка.

**Жадная загрузка** - это загрузка всех связанных данных вместе с главными объектами при первом доступе к ним.

**Ленивая загрузка** - это загрузка связанных данных только при первом доступе к связанным объектам.

**Явная загрузка** - это загрузка связанных данных вручную при необходимости.

11. В чем разница между концепциями Code First и Database First?

Концепция Code First - это подход к разработке приложений, при котором модель базы данных создается на основе классов в коде, а не на основе базы данных. Database First, наоборот, позволяет создавать модели на основе существующей базы данных.

12. Зачем EF использует миграции? Как создать новую миграцию?

Миграции - это способ контроля изменений в базе данных. Они позволяют автоматически обновлять схему базы данных при любых изменениях в моделях, без необходимости ручного изменения БД.

Создание новой миграции происходит при помощи команды **Add-Migration** в консоли диспетчера пакетов в Visual Studio.

13. Расскажите алгоритм выполнения CRUD-запросов при работе с БД через библиотеку Entity Framework.

1. Создание экземпляра DbContext для работы с БД.

2. Выборка данных с использованием методов DbSet и LINQ-запросов.

*3. Создание объектов сущностей (Entity) для добавления новых записей в БД с использованием метода Add объекта DbSet.*

*4. Обновление сущностей в БД с использованием метода Update объекта DbSet.*

*5. Удаление сущностей из БД с использованием метода Remove объекта DbSet.*

6. Вызов метода SaveChanges для сохранения изменений в БД.

Алгоритм выполнения CRUD-запросов в Entity Framework следующий: 1.сначала создается экземпляр DbContext и DbSets для доступа к объектам в БД.

2. Затем, при выполнении операции чтения, Entity Framework создает SQL-запрос и отправляет его в БД.

3Результаты запроса затем собираются в объекты моделей и возвращаются в код. Для операций вставки, обновления и удаления объектов в БД, Entity Framework сгенерирует соответствующие SQL-запросы и выполнит их.