

ВВЕДЕНИЕ В SQLALCHEMY

ВВЕДЕНИЕ

Задачи урока

- 01 Понять, что такое ORM и для чего это используется в программировании
- 02 Узнать, что такое SQLAlchemy

SQLAlchemy

это популярная библиотека Python, которая предоставляет собой удобный способ работы с базами данных с использованием языка SQL. Она основана на концепции **"ORM"** — **объектно-реляционное отображение**, что позволяет разработчикам работать с данными в виде объектов Python, а не напрямую с SQL-запросами

SQLAlchemy



Поддерживает различные системы управления базами данных, включая SQLite, PostgreSQL, MySQL и многие другие



Предлагает продвинутые возможности, позволяющие создавать гибкие и масштабируемые приложения, включая отложенную загрузку данных, поддержку транзакций и многое другое

В SQLAlchemy работа с базой данных происходит посредством создания моделей данных, которые представляют таблицы в базе данных. Модели данных могут содержать атрибуты, соответствующие полям таблицы, а также методы для выполнения операций **CRUD** (создание, чтение, обновление, удаление) с данными

SQLAlchemy, также, предоставляет возможность создавать и выполнять SQL-запросы непосредственно, если требуется более сложная логика или оптимизация запросов

Преимущества SQLAlchemy



Гибкость

Предоставляет различные уровни абстракции: от простых запросов и операций до продвинутых функций, таких как отображение наследования, сложные связи и динамическая конструирования запросов



Большое и активное сообщество

SQLAlchemy активно развивается и имеет **большое и активное сообщество**, что обеспечивает хорошую поддержку и доступ к множеству дополнительных модулей и расширений

Вывод

SQLAlchemy — мощный инструмент для работы с базами данных в Python, который позволяет разработчикам эффективно взаимодействовать с различными СУБД и упрощает процесс разработки приложений, связанных с базами данных

ORM Object-Relational Mapping или объектно-реляционное отображение

это подход, который позволяет разработчикам работать с базами данных, представляя данные в виде объектов в языках программирования, таких как Python, вместо того, чтобы напрямую работать с SQL-запросами и реляционной моделью данных

Важность ORM заключается в следующих аспектах

✦ 01 Упрощение разработки

ORM облегчает взаимодействие с базами данных, позволяя разработчикам работать с данными в более естественном и объектно-ориентированном стиле. Они могут использовать знакомые понятия, такие как классы, объекты, методы, атрибуты, для работы с данными, вместо того, чтобы писать сложные SQL-запросы

✦ 02 Повышение производительности

ORM может автоматически генерировать и оптимизировать SQL-запросы за вас. Он может выполнять дополнительные оптимизации запросов, такие как кэширование данных или умную предзагрузку связанных объектов, что может существенно повысить производительность вашего приложения

✦ 03 Переносимость

ORM-фреймворки, такие как SQLAlchemy, облегчают переносимость приложений между различными СУБД. Они предоставляют абстракцию от конкретной реализации базы данных и позволяют писать код, который будет работать с разными СУБД без изменений

✦ 04 Безопасность

ORM может предотвратить некоторые уязвимости безопасности, связанные с неправильным использованием SQL-запросов. Он может автоматически обрабатывать экранирование символов и предотвращать атаки SQL-инъекциями

✦ 05 Тестируемость

ORM позволяет создавать модели данных и работать с ними в изолированных тестовых средах, что облегчает тестирование и создание модульных тестов для вашего приложения

✦ 06 Сопровождаемость

ORM упрощает поддерживаемость кода, так как предоставляет удобный способ работы с базой данных. Использование ORM позволяет сосредоточиться на бизнес-логике приложения, вместо того, чтобы заботиться о деталях взаимодействия с базой данных

Вывод

ORM является мощным инструментом, который упрощает разработку, повышает производительность и обеспечивает гибкость и переносимость в работе с базами данных. Он является важным компонентом в разработке многих современных приложений, особенно тех, которые работают с большим объемом данных или требуют гибкого доступа к различным СУБД

ИТОГИ

- ✓ **ORM** - упрощает разработку приложений и абстрагирует нас от итоговой базы
- ✓ **SqlAlchemy** - Самая популярная библиотека которая предоставляет удобный способ работы с БД