

Введение и базовые операции SQL

1 модуль

Для кого этот курс

- Узнать об одном из наиболее распространенных способов персистентного хранения данных
- Что-то слышали или сталкивались с базами данных, но знания не систематизированы
- Хотели бы сменить профиль деятельности, стать аналитиком, разработчиком, системным администратором БД
- Хотите попробовать работу с БД, чтобы понять, интересно ли вам

План курса

- **Базовые операции SQL** - *как стать на “ты” с базами данных*
- **Основы реляционных СУБД** - *как написать запрос любой сложности*
- **Проектирование БД** - *от концепции до физической структуры*
- **Использование ORM** - *связь БД с концепциями ООП*
- **Администрирование MySQL и оптимизация запросов** - *как стать системным администратором БД*
- **Нереляционные СУБД** - *когда и как сказать “Not Only SQL”*

Технологии и инструменты курса

СУБД:

- **MySQL** - доминирующая РСУБД среди свободного ПО
- **MongoDB** - наиболее популярная документоориентированная СУБД
- **Redis** - наиболее стабильная СУБД типа “ключ-значение”

Проектирование:

- **MySQL Workbench** - популярный и свободный инструмент выполнения запросов и проектирования БД
- **Ваш любимый ORM** (*наш - Django ORM*) - на любом популярном языке есть несколько вариантов

Что такое база данных

Определений много, но есть схожие характеристики:

- Хранит данные по правилам (концепция, схема)
- Можно управлять данными по правилам
- Нужна для удовлетворения информационных потребностей

Система Управления Базами Данных

DataBase Management System (DBMS)

Программное средство независимое от области базы данных

- Управление создания и использования баз данных
- Проверка ограничений и учет правил для модели БД
- Контроль доступа к данным

Основные понятия БД

Сущность - класс, хранящийся в базе данных, таблица.

Объект - экземпляр сущности.

Атрибут - свойства характеризующие сущность, название столбца в таблице.

Кортеж - строка в таблице, набор значений конкретных атрибутов.

Домен - набор допустимых значений атрибута.

Идентификатор - атрибут с уникальным значением для данной таблицы

Области применения БД

- **Банки, картотеки, любые крупные системы**, в которых нужно хранить и изменять данные, а также иметь доступ к ним с разными ролями.
(системы, где уже не обойтись xls-файликом)
- **Веб-сайты** - сервер использует БД для удобства управления информацией и взаимодействия с пользователем.
- **Приложения (мобильные и десктопные)** используют локальные базы для удобства хранения данных по некоторым правилам.
- Любой программный продукт, подразумевающий отделение бизнес-логики и уровня хранения данных.

Архитектура СУБД

Основные компоненты:

- **Ядро** - процессы, сеть, память, файловая система и т.д.
- **Диспетчер данных** - транзакции, кэш
- **Диспетчер запросов** - парсер запроса, оптимизатор, исполнитель
- **Набор инструментов для служебных операций** - резервное копирование, восстановление, мониторинг

Характеристики реляционных СУБД

Преимущества:

- Простая схема данных для пользователя.
- Логическая и физическая независимость от данных.
- Целостность и защищенность данных.
- Методологический подход к проектированию.

Недостатки:

- Относительно низкая скорость доступа к данным.
- Не универсальное решение для любой предметной области.
- Меньшая гибкость при добавлении своих типов данных и операций.

Язык SQL

Диалекты:

- Oracle
- MySQL
- PostgreSQL
- MS SQL
- SQLite
- Access

База платежной системы

Состоит из одной таблицы “Платежи”:

- Email отправителя
- Email получателя
- Сумма платежа
- Валюта платежа
- Дата платежа
- Комментарий

Выборка данных SELECT WHERE

- Варианты условий
- Выборка конкретных полей
- Условия с подзапросом

Добавление записей INSERT INTO

- Вставка одной записи
- Вставка нескольких
- Заполнение только указанных полей
- Выборка вставленных записей (`SELECT LAST_INSERT_ID` в MySQL), не сработает при одновременной вставке нескольких записей

Обновление данных UPDATE WHERE

- Условия аналогично SELECT
- Нельзя делать подзапросы в условиях, если обновляется эта же таблица

Удаление данных DELETE

- Синтаксис аналогичен SELECT

Агрегация данных

Примеры информационных потребностей:

- Суммировать значения по указанному атрибуту для записей с определенными условиями
- Статистика в разрезе значений указанного атрибута

База услуг компании

Одна таблица с полями:

- Название проекта
- Имя клиента
- Срок начала работы (не указан, если проект не начат)
- Срок сдачи (не указан, если за проект еще не завершено)
- Бюджет

Группировка данных GROUP BY

- Как работает группировка
- Ошибка not a GROUP BY expression

База онлайн-магазина

Одна таблица с полями:

- Название товара
- Название категории
- Цена
- Количество продаж товара