

Программа Перезапуск

Модуль SQL.

Занятие 7

Преподаватель: Марат Гарафутдинов

Что смотрели на предыдущих занятиях

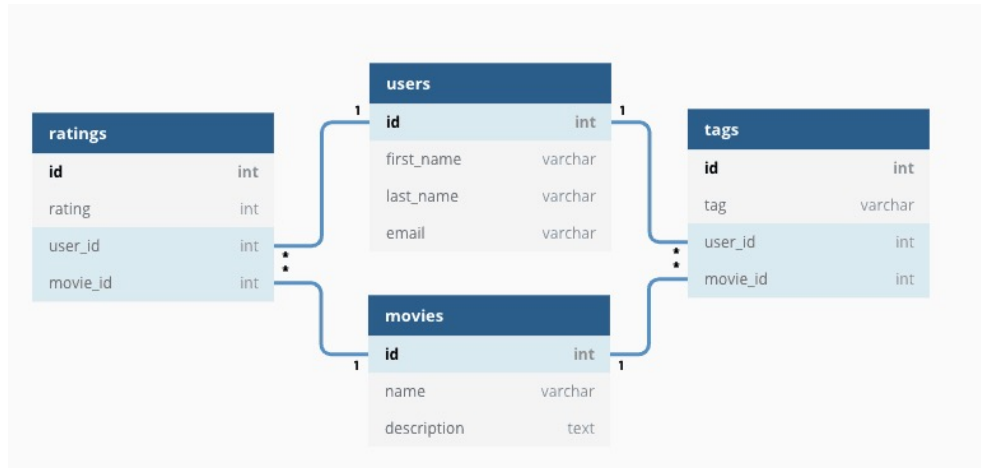
- Почему SQL нужен
- Реляционные таблицы
- Базовые операции
 - SELECT, INSERT, UPDATE, DELETE
- Операторы
 - WHERE, GROUP BY, AND, OR, NOT
- Операции JOIN
- Подзапросы
- Конструкции CASE WHEN
- Операторы ANY, ALL, IS NOT NULL
- Конструкции LIKE
- Конструкции CREATE TABLE & VIEW
- Оконные функции (ROW_NUMBER, RANK, DENSE_RANK)
- Индексы

Реляционные базы данных

Данные и связи между данными организованы с помощью таблиц.

Каждый столбец в таблице имеет имя и тип.

Каждая строка представляет отдельную запись или элемент данных в таблице, который содержит значения для каждого из столбцов.



Реляционные базы данных: колоночные

Колоночные базы данных (также нереляционные колоночные хранилища или базы данных с широкими столбцами) принадлежат к семейству NoSQL БД, но внешне похож на реляционные БД.

Как и реляционные, колоночные БД хранят данные, используя строки и столбцы, но с иной связью между элементами.

В реляционных БД все строки должны соответствовать фиксированной схеме. Схема определяет, какие столбцы будут в таблице, типы данных и другие критерии.

В колоночных базах вместо таблиц имеются структуры – «колоночные семейства».

Семейства содержат строки, каждая из которых определяет собственный формат. Строка состоит из уникального идентификатора, используемого для поиска, за которым следуют наборы имён и значений столбцов.

Следствия:

- БД удобны при работе с приложениями, требующими высокой производительности;
- данные и метаданные записи доступны по одному идентификатору;
- гарантировано размещение всех данных из строки в одном кластере, что упрощает сегментацию и масштабирование данных.

SQLite3

SQLite — это встраиваемая кроссплатформенная БД, которая поддерживает достаточно полный набор команд SQL и доступна в исходных кодах (на языке C).

Исходные коды SQLite находятся в public domain, то есть вообще никаких ограничений на использование.

Насколько SQLite популярна?

Кратко: она везде. Как минимум, на любом смартфоне.

Насколько она надежна?

Очень. При выпуске версии она проходит через ряд серьезнейших автоматических тестов (проводится ~ 2 млн тестов), покрытие кода тестами 100% (с августа 2009).

NoSQL: таблицы типа «ключ-значение»

В базах данных «ключ-значение» для хранения информации вы предоставляете ключ и объект данных, который нужно сохранить.

Например, JSON-объект, изображение или текст. Чтобы запросить данные, отправляете ключ и получаете blob-объект.

Следствия:

- хранилища обеспечивают быстрый и малозатратный доступ;
- часто хранят данные конфигураций и информацию о состоянии данных, представленных словарями или хэшем;
- нет жёсткой схемы отношения между данными, поэтому в таких БД часто хранят одновременно различные типы данных;
- разработчик отвечает за определение схемы именования ключей и за то, чтобы значение имело соответствующий тип/формат.

Key	Value
K1	AAA,BBB,CCC
K2	AAA,BBB
K3	AAA,DDD
K4	AAA,2,01/01/2015
K5	3,ZZZ,5623

NoSQL: Документная база данных

Документные базы данных (также документоориентированные БД или хранилища документов), совместно используют базовую семантику доступа и поиска хранилищ ключей и значений.

Такие БД также используют ключ для уникальной идентификации данных.

Разница между хранилищами «ключ-значение» и документными БД заключается в том, что вместо хранения blob-объектов, документоориентированные базы хранят данные в структурированных форматах – JSON, BSON или XML.

Следствия:

- база данных не предписывает определённый формат или схему;
- каждый документ может иметь свою внутреннюю структуру;
- документные БД являются хорошим выбором для быстрой разработки;
- в любой момент можно менять свойства данных, не изменяя структуру или сами данные.

	R1C1	R1C2	R1C3	R1C4
	R2C1	R2C2	R2C3	R2C4
	R3C1	R3C2	R3C3	R3C4
	R4C1	R4C2	R4C3	R4C4

Relational data model
Highly-structured table organization with rigidly-defined data formats and record structure



Document data model
Collection of complex documents with arbitrary, nested data formats and varying "record" format.

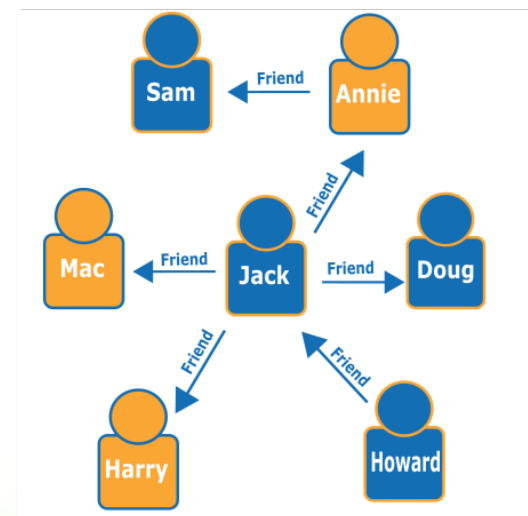
NoSQL: графовая база данных

Вместо сопоставления связей с таблицами и внешними ключами, графовые базы данных устанавливают связи, используя узлы, рёбра и свойства.

Графовые базы представляют данные в виде отдельных узлов, которые могут иметь любое количество связанных с ними свойств.

Следствия:

- фокусируются на связях между элементами;
- явно отображает связи между типами данных;
- не требуют пошагового обхода для перемещения между элементами;
- нет ограничений в типах представляемых связей.



https://sandbox.neo4j.com/?_gl=1*1of3ftu*_ga*MTU2NTgyMTI1Mi4xNjlwOTI1Mzly*_ga_DL38Q8KGQC*MTYyMTI1NjU0OC40LjEuMTYyMTI1NjYxMy4w&_ga=2.79382167.2089558875.1621248209-1565821252.1620925322

Linux: Работаем с Google Cloud Shell

<https://console.cloud.google.com/>

pwd текущая директория

ls список файлов и директорий в текущей (ls -al выведет красивее)

cp <a> копирует <a> в

cp -R <a> рекурсивно копирует <a> в

mv <a> перемещает <a> в

mkdir <имя папки> создать папку

cd <path to dir> перейти в папку по адресу <path to dir>

touch filename создаст пустой файл с именем filename

man <имя команды> руководство по команде

также рекомендуется гуглить первое время

Linux: Работаем с Google Cloud Shell и PostgreSQL

<https://console.cloud.google.com/>

Команда: `psql -h 52.157.159.24 -U student0 sql_ex_for_student0`

Далее вводим пароль

Демонстрация и решение упражнений