

Базы данных и СУБД. Введение в SQL. Основные понятия. Язык запросов SQL

База данных – это совокупность связанных данных, организованных по определенным правилам, предусматривающим общие принципы описания, хранения и манипулирования, независимая от прикладных программ. **База данных** является информационной моделью предметной области. Обращение к **базам данных** осуществляется с помощью системы управления **базами данных (СУБД)**. **СУБД** обеспечивает поддержку создания **баз данных**, централизованного управления и организации доступа к ним различных пользователей.

Итак, мы пришли к выводу, что хранить данные независимо от программ, так, что они связаны между собой и организованы по определенным правилам, целесообразно. Но вопрос, как хранить данные, по каким правилам они должны быть организованы, остался открытым. Способов существует множество (кстати, называются они моделями представления или хранения данных). Наиболее популярные – объектная и **реляционная** модели данных.

Автором **реляционной модели** считается Э. Кодд, который первым предложил использовать для обработки данных аппарат теории множеств (объединение, пересечение, разность, декартово произведение) и показал, что любое представление данных сводится к совокупности двумерных таблиц особого вида, известного в математике как отношение.

Таким образом, **реляционная база данных** представляет собой набор таблиц (точно таких же, как приведенная выше), связанных между собой. Строка в таблице соответствует сущности реального мира (в приведенном выше примере это информация о человеке).

Примеры **реляционных СУБД**: *MySql*, *PostgreSQL*.

В основу **объектной модели** положена концепция объектно-ориентированного программирования, в которой данные представляются в виде набора объектов и классов, связанных между собой родственными отношениями, а работа с объектами осуществляется с помощью скрытых (инкапсулированных) в них методов.

Примеры **объектных СУБД**: *Cache*, *GemStone* (от *Servio Corporation*), *ONTOS* (*ONTOS*).

В последнее время производители **СУБД** стремятся соединить два этих подхода и проповедуют объектно-реляционную модель представления данных. Примеры таких **СУБД** – *IBM DB2 for Common Servers*, *Oracle8*.

Поскольку мы собираемся работать с *MySql*, то будем обсуждать аспекты работы только с **реляционными базами данных**. Нам осталось рассмотреть еще два важных понятия из этой области: **ключи** и **индексирование**, после чего мы сможем приступить к изучению языка запросов **SQL**.

ИНДЕКСИРОВАНИЕ

Одна из основных задач, возникающих при работе с базами данных, – это задача поиска. При этом, поскольку информации в базе данных, как правило, содержится много, перед программистами встает задача не просто поиска, а эффективного поиска, т.е. поиска за сравнительно небольшое время и с достаточной точностью. Для этого (для оптимизации производительности запросов) производят индексирование некоторых полей таблицы. Использовать индексы полезно для быстрого поиска строк с указанным значением одного столбца. Без индекса чтение таблицы осуществляется по всей таблице, начиная с первой записи, пока не будут найдены соответствующие строки. Чем больше таблица, тем больше накладные расходы. Если же таблица содержит индекс по рассматриваемым столбцам, то база данных может быстро определить позицию для поиска в середине файла данных без просмотра всех данных. Это происходит потому, что база данных помещает проиндексированные поля поближе в памяти, так, чтобы можно было побыстрее найти их значения. Для таблицы, содержащей 1000 строк, это будет как минимум в 100 раз быстрее по сравнению с последовательным перебором всех записей. Однако в случае, когда необходим доступ почти ко всем 1000 строкам, быстрее будет последовательное чтение, так как при этом не требуется операций поиска по диску. Так что иногда индексы бывают только помехой. Например, если копируется большой объем данных в таблицу, то лучше не иметь никаких индексов. Однако в некоторых случаях требуется

задействовать сразу несколько индексов (например, для обработки запросов к часто используемым таблицам).

Если говорить о MySQL, то там существует три вида индексов: PRIMARY, UNIQUE, и INDEX, а слово ключ (KEY) используется как синоним слова индекс (INDEX). Все индексы хранятся в памяти в виде B-деревьев.

PRIMARY – уникальный индекс (ключ) с ограничением, что все индексируемые им поля не могут иметь пустого значения (т.е. они NOT NULL). Таблица может иметь только один первичный индекс, но он может состоять из нескольких полей.

UNIQUE – ключ (индекс), задающий поля, которые могут иметь только уникальные значения.

INDEX – обычный индекс (как мы описали выше). В MySQL, кроме того, можно индексировать строковые поля по заданному числу символов от начала строки.

СУБД MySQL

MySQL – это реляционная система управления базами данных. То есть данные в ее базах хранятся в виде логически связанных между собой таблиц, доступ к которым осуществляется с помощью языка запросов SQL. MySQL – свободно распространяемая система, т.е. платить за ее применение не нужно. Кроме того, это достаточно быстрая, надежная и, главное, простая в использовании СУБД, вполне подходящая для не слишком глобальных проектов.

Работать с MySQL можно не только в текстовом режиме, но и в графическом. Существует очень популярный визуальный интерфейс (кстати, написанный на PHP) для работы с этой СУБД. Называется он PhpMyAdmin. Этот интерфейс позволяет значительно упростить работу с базами данных в MySQL.

Язык SQL

Итак, мы в общих чертах познакомились с основными понятиями теории баз данных, установили и настроили для работы MySQL. Теперь самое время научиться манипулировать данными, хранящимися в базах данных. Для этого нам понадобится SQL – структурированный язык запросов. Этот язык дает возможность создавать, редактировать и удалять информацию, хранящуюся в базах данных, создавать новые базы данных и многое другое. SQL является стандартом ANSI (Американский национальный институт стандартов) и ISO (Международная организация по стандартизации).

Если говорить о MySQL, то она соответствует начальному уровню SQL92, содержит несколько расширений этого стандарта и стремится к полной поддержке стандарта ANSI SQL99, но без ущерба для скорости и качества кода.

Далее, говоря об основах языка SQL, будем придерживаться его реализации в СУБД MySQL.

Основные операторы языка SQL

Функции любой СУБД включают:

1. создание, удаление, изменение базы данных (БД);
2. добавление, изменение, удаление, назначение прав пользователя;
3. внесение, удаление и изменение данных в БД (таблиц и записей);
4. выборку данных из БД.

К первым двум функциям имеют доступ только администраторы СУБД или привилегированные пользователи. Рассмотрим, как решаются последние две задачи (на самом деле это семь задач).

Прежде чем что-либо делать с данными, нужно создать таблицы, в которых эти данные будут храниться, научиться изменять структуру этих таблиц и удалять их, если потребуется. Для этого в языке SQL существуют операторы CREATE TABLE, ALTER TABLE и DROP TABLE.