

Базы данных



СУБД MySQL

Автор: Гасанов Эльмир Вагидович
computerschool@yandex.ru

Лекция №1



Введение в базы данных.

База данных

База данных – это совокупность данных, организованных по определенным правилам.

База данных – это совокупность данных, систематизированных таким образом, чтобы эти данные могли быть найдены и обработаны с помощью ЭВМ.

Логическая структура данных, хранимых в базе, называется моделью представления данных или моделью данных.

Основными моделями данных являются иерархическая, сетевая, реляционная и объектно-ориентированная.

Реляционные базы данных

В настоящее время наибольшее распространение получили реляционные базы данных.

В реляционной базе данных информация хранится в виде таблиц, состоящих из строк и столбцов.

Таблицы называются отношениями (relations), строки – записями или кортежами, столбцы – полями или атрибутами.

Пример таблицы реляционной БД

Таблица Students:

Таблица 1.1

name	facultet	gruppa
Иванов Иван	Бизнес-информатика	101
Иванов Петр	Экономика	223
Петров Петр	Дизайн	302

Первичный ключ

Таблица реляционной базы данных является неупорядоченным множеством, состоящим из 0 или более записей. Каждая запись должна быть уникальной. Для каждой таблицы необходимо определить комбинацию атрибутов (полей), однозначно определяющих каждую запись. Такой набор из одного или более атрибутов называется первичным ключом (**PRIMARY KEY**).

Первичный ключ, состоящий из одного атрибута называется простым первичным ключом, а состоящий из множества атрибутов – составным первичным ключом.

Определение или создание первичного ключа

Для определения первичного ключа необходимо просмотреть все поля создаваемой таблицы и найти поле или группу полей, информация которых была бы уникальной для каждой записи.

Первичный ключ должен состоять из минимального набора атрибутов, однозначно идентифицирующих запись.

Когда единственно возможный ключ состоит из слишком многих атрибутов, то наилучшее решение этой проблемы заключается в том, чтобы в таблицу добавить поле, которое будет содержать идентификационный номер.

Таблица Students с первичным ключом:

Таблица 1.2

id_stud	name	facultet	gruppa
1	Иванов Иван	Бизнес-информатика	101
2	Иванов Петр	Экономика	223
3	Петров Петр	Дизайн	302

Построение структуры реляционной базы данных

Проектирование базы данных является искусством и приобретается с опытом.

Перед созданием структуры данных необходимо выявить (определить) основные сущности предметной области.

Сущность – это нечто, о чем нужно хранить информацию в базе данных.

Затем определяются все атрибуты (свойства) сущности, которые будут храниться в базе данных.

Структуру базы данных в начале проектирования представляют в виде таблицы, содержащей все атрибуты сущности.

Нормализация баз данных

При проектировании реляционной базы данных основной задачей является построение эффективной структуры данных.

Реляционная база данных считается эффективной, если она обладает следующими характеристиками:

- отсутствие избыточности информации;
- обеспечение целостности данных.

Для создания эффективной структуры данных используется процесс, называемый нормализацией.

Нормальные формы

В теории нормализации баз данных разработаны достаточно формализованные подходы по разбиению данных, обладающих сложной структурой, среди нескольких таблиц.

Теоретические правила, которым отвечает структура таблицы, называются нормальными формами. Теория нормализации оперирует с шестью нормальными формами таблиц. Эти формы предназначены для уменьшения избыточной информации.

Лежащая в основе нормализации математическая теория довольно сложна, но для практического применения ее можно сформулировать в виде довольно простых правил.

При практическом проектировании баз данных форма Бойса-Кодда, четвертая и пятая формы, как правило не используются, поэтому мы ограничимся рассмотрением первых трех нормальных форм.

Нормальные формы - Бойса-Кодда, четвертая, пятая были разработаны для специальных случаев, которые редко встречаются на практике.

Первая нормальная форма (1NF)

Таблица имеет форму 1NF если:

1. содержит информацию об одной сущности;
2. имеет первичный ключ;
3. все атрибуты содержат одно значение.

Исходная таблица Sudents:

Таблица 1.3

<u>id stud</u>	<u>name</u>	<u>facultet</u>	<u>gruppa</u>	<u>email</u>
1	Иванов Иван	Бизнес- информатика	101	mail1@ya.ru mail1@mail.ru
2	Иванов Петр	Экономика	223	mail2@ya.ru mail2@mail.ru
3	Петров Петр	Дизайн	302	mail3@ya.ru mail3@mail.ru

Пример нормализации

Таблица Students не находится в первой нормальной форме, т.к. не выполняется условие 3 (поле email содержит более одного значения).

Для выполнения условия 3 необходимо создать вторую таблицу Students_email, разместив в ней электронные адреса.

При этом требуется сопоставить электронный адрес его владельцу, представленному в первой таблице.

Для этого во вторую таблицу помещают первичный ключ из первой таблицы.

Таблица 1.4

<u>id_stud</u>	name	<u>facultet</u>	<u>gruppa</u>
1	Иванов Иван	Бизнес- информатика	101
2	Иванов Петр	Экономика	223
3	Петров Петр	Дизайн	302

Students

Таблица 1.5

<u>id_stud</u>	email
1	mail1@ya.ru
1	mail1@mail.ru
2	mail2@ya.ru
2	mail2@mail.ru
3	mail3@ya.ru
3	mail3@mail.ru

Students_email

Внешний ключ

Так как номера студентов уникальны, то всегда можно будет узнать, какому студенту соответствует данный электронный адрес.

Столбец `id_stud` в таблице `Students_email` называется внешним ключом (**FOREIGN KEY**). Он ссылается на первичный ключ таблицы `Students`.

Ключи, на которые ссылаются внешние ключи, называются родительскими ключами.

Вторая нормальная форма (2NF)

Таблица находится во второй нормальной форме (2NF) если:

- она удовлетворяет условиям первой нормальной формы;
- любое неключевое поле зависит от полного первичного ключа.

Из данного определения следует, что понятие второй нормальной формы применимо только к таблицам, имеющим составной первичный ключ.

Третья нормальная форма (3NF)

Таблица находится во третьей нормальной форме (3NF) если:

- она удовлетворяет условиям второй нормальной формы;
- ни одно из неключевых полей не зависит от другого неключевого поля.

Система управления базами данных (СУБД)

Для взаимодействия пользователя с базами данных используются системы управления базами данных (СУБД).

Система управления базами данных – это программное обеспечение, обеспечивающее управление созданием и использованием баз данных.

СУБД различают по используемой модели данных. СУБД, основанные на реляционной модели данных, называются реляционными СУБД.

СУБД MySQL

Одной из самых популярных систем управления базами данных, используемых в настоящее время в Web-среде является СУБД MySQL.

По сравнению с другими данная система предоставляет ряд преимуществ:

- она свободно доступна для загрузки из Internet и установки на практически любой машине;
- ее легко установить на многих операционных системах (включая Windows и Unix);
- она проста в использовании и включает в себя несколько удобных инструментов администрирования.

СУБД MySQL

Тип Реляционная СУБД

Разработчик MySQL AB (до 2008),
Sun Microsystems (2008-2010),
Oracle (с 2010)

Написана на C, C++

Операционная система Кроссплатформенное ПО

Последняя версия 5.6.15 (3 декабря 2013)

Лицензия GNU General Public License или
коммерческая лицензия

Сайт mysql.com

Язык SQL

Стандартным языком, используемым для взаимодействия с базами данных является язык структурированных запросов SQL (Structured Query Language).

Язык SQL используется для извлечения, добавления и манипулирования данными, размещаемыми в базах данных.

Язык SQL

Появился в: 1974

**Авторы: Дональд Чемберлин
Рэймонд Бойс**

Релиз: SQL:2008 (2008)