Реляционная модель данных

**Реляционная модель данных** - логическая модель данных, прикладная теория построения баз данных, которая является приложением теории множеств к задачам обработки данных. Принципы реляционной модели были сформулированы в 1969—1970 годах Э. Ф. Коддом (E. F. Codd).

Термин "реляционный" означает, что теория основана на математическом понятии отношение (relation). Физическим представлением отношения в теории множеств является таблица.

Поэтому на практике можно говорить, что реляционная модель представляет собой совокупность данных, состоящую из набора двумерных таблиц.

Основные компоненты реляционной модели данных:

* Структуры данных;
* Операции, которые можно выполнять над данными;
* Ограничения, связанные с обеспечением целостности данных;

**Понятие таблицы базы данных**

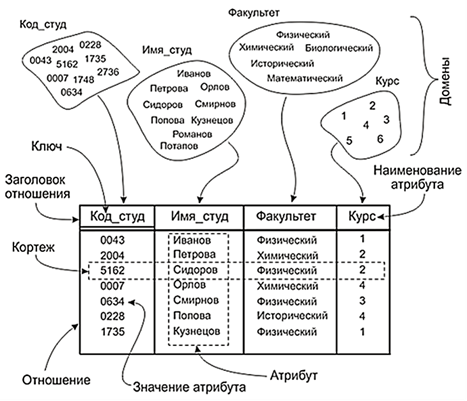
Таблица – физическое представление отношения. Таблица является структурной единицей хранения информации в базе данных.

Структура таблицы в реляционной базе характеризуется следующим:

* Она состоит из совокупности столбцов;
* каждый столбец имеет уникальное, то есть не повторяющееся в других столбцах, имя;
* Последовательность столбцов в таблице не существенна;
* все строки таблицы организованы по одинаковой структуре, то есть имеют одно и то же количество реквизитов и имеют одинаковую длину;
* В таблице нет одинаковых строк;
* Количество строк в таблице практически не ограничено;
* Последовательность строк в таблице не существенна;
* При выполнении манипуляций с таблицей все строки и столбцы могут просматриваться в произвольном порядке безотносительно к их содержанию и смыслу;

Можно дать такие нестрогие определения основных терминов реляционной модели:

* Строки таблицы называются **кортежами (tuple)**, а столбцы **атрибутами (attribute)**;
* Количество кортежей в отношении называется **кардинальным числом** отношения, а количество атрибутов называется степенью отношения;
* Каждый атрибут в отношении имеет наименование, которое указывается в заголовочной части отношения;
* **Ключ отношения** – это атрибут или набор атрибутов такой, что не существует строк, для которых значение или комбинация значений ключевых атрибутов являются одинаковыми. Таким образом, ключ является уникальным идентификатором кортежей отношения;
* **Домен отношения** – это множество всех возможных значений конкретного атрибута;



Управление базами данных

Для управления базами данных нужны такие компоненты:

* Специализированный язык программирования для работы с данными;
* Системы, которые физически реализую модель данных и язык программирования;

**Реляционные системы управления базами данных**

Наиболее популярные реляционные СУБД (Relational Database Management System):

* MS SQL Server;
* IBM DB2
* Oracle Database;
* MySQL;
* PostgreSQL;
* Informix;
* Sybase;

**Блокирующий и версионный механизмы**

**Транзакция** (англ. transaction) – группа последовательных операций с базой данных, которая представляет собой логическую единицу работы с данными. Транзакция может быть выполнена либо целиком и успешно, соблюдая целостность данных и независимо от параллельно идущих других транзакций, либо не выполнена вообще и тогда она не должна произвести никакого эффекта.

Параллельное выполнение транзакций способно привести базу в несогласованное состояние даже в тех случаях, когда каждая транзакция, выполняющаяся отдельно от других, производит полностью корректные изменения.

Поэтому очередность операций различных транзакций должна тем или иным образом регулироваться.

**Блокировочный механизм**

В основе лежит протокол двухфазной блокировки. Перед чтением или изменением объект (запись) блокируется. То есть другим транзакциям запрещается изменять или даже читать этот объект до тех пор, пока первая транзакция не закончит с ним работать.

**Версионный механизм**

Принцип действия версионности основан на том, что транзакция, изменяя данные, порождает новую копию (версию) данных, с которой и работает. Другим транзакциям эта версия не видна, до тех пор, пока первая не зафиксируется. При этом даже после фиксации первой транзакции, устаревшая версия какое-то время сохраняется для корректной работы транзакций, стартовавших до завершения работы первой, но еще не успевших зафиксироваться.

**Язык SQL**

**SQL (structured query language)** – язык программирования, применяемый для создания, модификации и управления данными в произвольной реляционной базе данных, управляемой соответствующей системой управления базами данных.

SQL стандартизирован ANSI (American National Standards Institute), поэтому часто употребляется аббревиатура ANSI SQL. В настоящее время действует стандарт, принятый в 2003 году (SQL:2003) с небольшими модификациями, внесёнными позже (SQL:2008).

Несмотря на стандартизацию ANSI существует множество реализаций SQL. Но несмотря на это, стандарт гарантирует, что основные команды (такие как SELECT, UPDATE, DELETE, INSERT, WHERE) будут выполнятся одинаково.

SQL может выполнять такие действия с базой данных:

* SQL может выполнять запросы к базе данных;
* SQL может извлекать данные из базы;
* SQL может вставлять записи в базу данных;
* SQL может обновлять записи в базе данных;
* SQL может удалять записи из базы данных;
* SQL может создавать новую базу данных;
* SQL может создавать новые таблицы в базе данных;
* SQL может создавать хранимые процедуры в базе данных;
* SQL может создавать представления (views) в базе данных;
* SQL может устанавливать разрешения для таблиц процедур и представлений;
* SQL может создавать простые и сложные ключи и устанавливать связи по ним;

SQL состоит из четырех отдельных частей:

* **Язык определения данных (DDL)** используется для определения структур данных, хранящихся в базе данных. Операторы DDL позволяют создавать, изменять и удалять отдельные объекты в БД. Допустимые типы объектов зависят от используемой СУБД и обычно включают базы данных, пользователей, таблицы и ряд более мелких вспомогательных объектов, например, роли и индексы;
* **Язык манипуляции данными (DML)** используется для извлечения и изменения данных в БД. Операторы DML позволяют извлекать, вставлять, изменять, удалять и объединять данные в таблицах. Все операторы DML носят декларативный характер;
* **Язык определения доступа к данным (DCL)** используется для контроля доступа к данным в БД. Операторы DCL применяются к привилегиям и позволяют выдавать и отбирать права на применение определенных операторов DDL и DML к определенным объектам БД;
* **Язык управления транзакциями (TCL)** используется для контроля обработки транзакций в БД. Обычно операторы TCL включают commit для подтверждения изменений, сделанных в ходе транзакции, rollback для их отмены и savepoint для разбиения транзакции на несколько меньших частей;

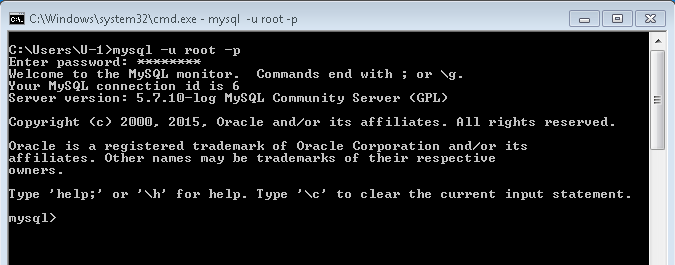
Описанные части SQL это не отдельные языки, а подмножества команд, разделенные по функциональному признаку.

Основными командами можно считать базовые команды DDL и DML:

|  |  |
| --- | --- |
| **DML** | **DDL** |
| SELECT (выбрать)  INSERT (вставить)  UPDATE (обновить)  DELETE (удалить) | CREATE DATABASE (создать базу данных)  CREATE TABLE (создать таблицу)  CREATE VIEW (создать виртуальную таблицу)  CREATE INDEX (создать индекс)  CREATE TRIGGER (создать триггер)  CREATE PROCEDURE (создать сохраненную процедуру)  ALTER DATABASE (модифицировать базу данных)  ALTER TABLE (модифицировать таблицу)  ALTER VIEW (модифицировать виртуальную таблицу)  ALTER INDEX (модифицировать индекс)  ALTER TRIGGER (модифицировать триггер)  ALTER PROCEDURE (модифицировать сохраненную процедуру)  DROP DATABASE (удалить базу данных)  DROP TABLE (удалить таблицу)  DROP VIEW (удалить виртуальную таблицу)  DROP INDEX (удалить индекс)  DROP TRIGGER (удалить триггер)  DROP PROCEDURE (удалить сохраненную процедуру) |

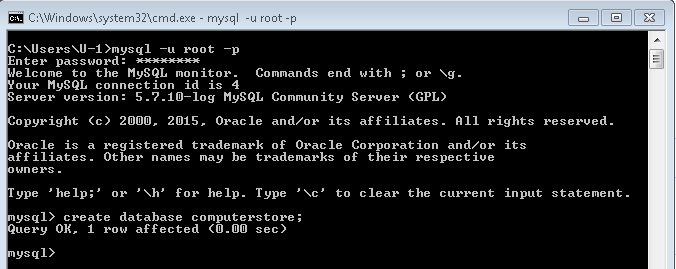
Практика

Используя консоль операционной системы установить соединение с сервером MySql (локальная машина):

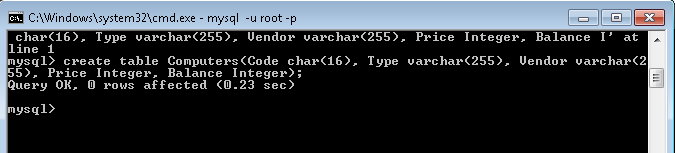


Выход: exit или quit.

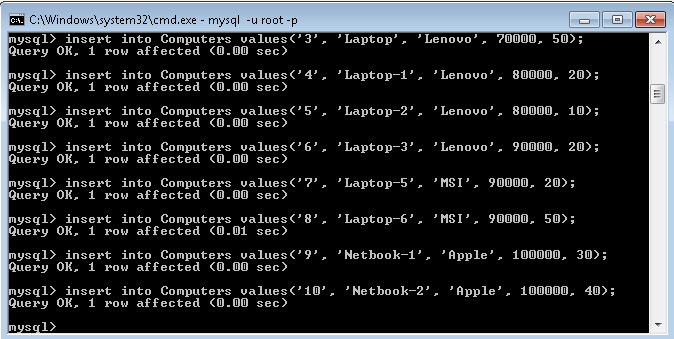
Создать базу данных (например computerstore)



Создать таблицу для хранения компьютеров (например, Computers). В таблице Computers нужно хранить тип компьютера, название производителя, цену, количество на складе.

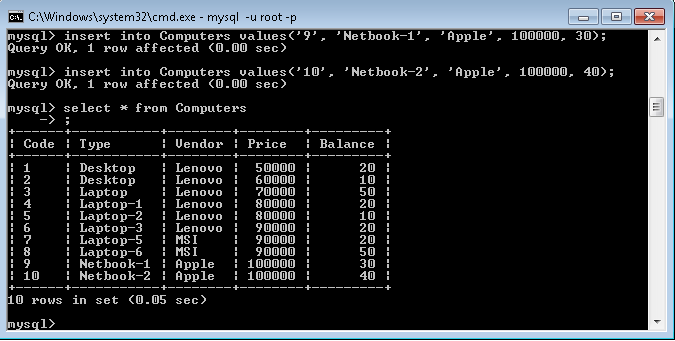


Сохрани в таблице Computers информацию об десяти компьютерах.



Текущая версия dbForge 6.3.358 требует установки совместимости с MySQL 5.7.\*. Нужно выполнить в консоли MySQL команду:

set @@global.show\_compatibility\_56=ON;

Вывести всю информацию из таблицы Computers в консоль своей операционной системы

1. <http://rdbms.narod.ru/faq/>
2. <http://www.bseu.by/it/tohod/lekcii2_3.htm>
3. <https://rsdn.ru/article/db/yukonvers.xml>
4. <https://ru.wikipedia.org/wiki/SQL>
5. <http://www.w3schools.com/sql/sql_intro.asp>
6. <http://progopedia.ru/language/sql/>
7. <https://habrahabr.ru/post/255361/>