# Основные понятия реляционных баз данных #

Она определяет способ представления данных (структуру данных), методы защиты данных (целостность данных), и операции, которые можно выполнять с данными (манипулирование данными). Эта модель лежит в основе всех реляционных баз данных до настоящего времени.

## Основные принципы реляционных баз данных: ##

+ все данные на концептуальном уровне представляются в виде объектов, заданных в виде строк и столбцов, называемых отношением, более распространенное название – таблица;

+ в пересечение строки и столбца таблицы можно занести только одно значение;

+ все операции выполняются над целыми отношениями и результатом этих операций является отношение.

### Терминология реляционных баз данных: ###

+ отношение – это структура данных целиком, набор записей (в обычном понимании – таблица) , в примере –это Сотрудник;

+ кортеж – это каждая строка , содержащая данные (более распространенный термин – запись ), например, <001, Борин С.А, 234-01-23, программист>, все кортежи в отношении должны быть различны;

+ мощность – число кортежей в таблице (проще говоря, число записей), в данном случае 3, мощность отношения может быть любой (от 0 до бесконечности), порядок следования кортежей - неважен;

+ атрибут – это столбец в таблице (более распространенный термин – поле ), в примере – Табельный номер, Фамилия И.О., Телефон, Должность)

+ размерность – это число атрибутов в таблице, в данном случае – 4;

+ размерность отношения должна быть больше 0, порядок следования атрибутов существенен;

+ домен атрибута – это допустимые значения (неповторяющиеся), которые можно занести в поле , например для атрибута Должность домен – {инженер, программист}.

Реляционная БД состоит из таблиц.

Каждый столбец - имеет «тип и имя». Каждая строка- отдельная запись или элемент данных в таблице.

Поле в таблице (внешний ключ) может содержать ссылки содержащие в других таблицах. Что позволяет их соединять.

Высокоорганизованная структура и гибкость позволяет адаптироваться к различным типам данных

Примеры:

My SQL

PostgreSQL

MariaDB

SQLite

Перед созданием таблицы в базе данных необходимо описать ее структуру. Для этого выполняется следующая последовательность шагов:

## Дать таблице имя, пусть она будет называться book, вот некоторые правила для выбора имен таблиц: ##

### может включать английские буквы, цифры и знак подчеркивания, должно начинаться с буквы; ###

### имя должно быть уникальным в пределах базы данных. ###

## Определить структуру таблицы, из каких атрибутов(столбцов, полей) она будет состоять, в нашем случае это: ##

### title – поле для хранения названия книги; ###

### author – поле с фамилией автора книги; ###

### priсe – цена книги; ###

### amount – количество книг. ###

## Включить ключевое поле book\_id, которое является ОБЯЗАТЕЛЬНЫМ ЭЛЕМЕНТОМ каждой реляционной таблицы. Ключевое поле является уникальным для каждой записи, однозначно определяет запись и в дальнейшем будет использоваться для связей с другими таблицами. ##

# Выбор типов данных для полей #

После описания структуры таблицы необходимо выбрать типы данных для каждого поля.

## Рекомендации по выбору типов данных для полей таблицы: ##

### Выбирайте минимальный тип данных исходя из максимального значения поля. Например, если максимальный текст, который может быть записан в поле, имеет длину 25 символов, значит нужно использовать тип VARCHAR(25). ###

### Для описания ключевого поля используйте описание INT PRIMARY KEY AUTO\_INCREMENT. Это значит, что в поле будут заноситься различные целые числа, при этом они будут автоматически генерироваться (каждая следующая строка будет иметь значение ключа на 1 больше предыдущего). ###

+ book\_id - ключевой столбец, целое число, которое должно генерироваться автоматически - INT PRIMARY KEY AUTO\_INCREMENT;

+ title - строка текста, ее длина выбирается в зависимости от данных, которые предполагается хранить в поле, предположим, что название книги не превышает 50 символов - VARCHAR(50);

+ author - строка текста - VARCHAR(30);

+ price - для описание денежного значения используется числовой тип данных с двумя знаками после запятой - DECIMAL(8,2);

+ amount - целое число - INT.

# Создание таблицы (CREATE) #

Его структура :

+ ключевые слова : CREATE TABLE

+ имя создаваемой таблицы;

+ открывающая круглая скобка «(»;

+ название поля и его описание, которое включает тип поля и другие необязательные характеристики;

+ запятая;

+ название поля и его описание;

+ ...

+ закрывающая скобка «)».

## Рекомендации по записи SQL запроса ##

+ Ключевые слова: SQL не является регистрозависимым языком (CREATE и create - одно и тоже ключевое слово).

+ Ключевые слова SQL и типы данных рекомендуется записывать прописными (большими) буквами.

+ Имена таблиц и полей - строчными (маленькими) буквами.

+ SQL-запрос можно писать на нескольких строках.

+ В конце SQL-запроса ставится точка с запятой (хотя если Вы пишете один запрос, это необязательно).

INSERT INTO genre (name\_genre) VALUES ('Роман');

Заносится только значение поля name\_genre, значение ключевого поля формируется автоматически.