

Содержание

Об авторе	13
Благодарности	14
Введение	15
Для кого предназначена эта книга	16
СУБД, рассмотренные в книге	16
Файлы примеров	17
Условные обозначения	17
Ждем ваших отзывов!	19
Урок 1. Основы SQL	21
Терминология баз данных	21
Что такое SQL	28
Попробуйте сами	29
Резюме	31
Урок 2. Извлечение данных из таблиц	33
Инструкция SELECT	33
Извлечение отдельных столбцов	34
Извлечение нескольких столбцов	37
Извлечение всех столбцов	38
Извлечение уникальных строк	39
Ограничение результатов запроса	41
Использование комментариев	45
Резюме	47
Упражнения	48
Урок 3. Сортировка полученных данных	49
Сортировка записей	49
Сортировка по нескольким столбцам	51
Сортировка по положению столбца	53

Указание направления сортировки	54
Резюме	57
Упражнения	58
Урок 4. Фильтрация данных	59
Предложение WHERE	59
Операторы в предложении WHERE	61
Резюме	67
Упражнения	68
Урок 5. Расширенная фильтрация данных	69
Комбинирование предложений WHERE	69
Оператор IN	75
Оператор NOT	77
Резюме	79
Упражнения	80
Урок 6. Фильтрация с использованием метасимволов	81
Оператор LIKE	81
Советы по использованию метасимволов	89
Резюме	90
Упражнения	91
Урок 7. Создание вычисляемых полей	93
Что такое вычисляемые поля	93
Конкатенация полей	95
Выполнение арифметических вычислений	101
Резюме	104
Упражнения	104
Урок 8. Функции обработки данных	105
Что такое функция	105
Применение функций	107
Резюме	116
Упражнения	117

Урок 9. Итоговые вычисления	119
Итоговые функции	119
Итоговые вычисления для уникальных значений	128
Комбинирование итоговых функций	130
Резюме	131
Упражнения	131
Урок 10. Группирование данных	133
Принципы группирования данных	133
Создание групп	134
Фильтрация по группам	137
Группирование и сортировка	141
Порядок предложений в инструкции SELECT	143
Резюме	144
Упражнения	145
Урок 11. Подзапросы	147
Что такое подзапросы	147
Фильтрация с помощью подзапросов	147
Использование подзапросов в качестве вычисляемых полей	152
Резюме	156
Упражнения	157
Урок 12. Соединение таблиц	159
Что такое соединение	159
Создание соединения	163
Резюме	172
Упражнения	173
Урок 13. Создание расширенных соединений	175
Использование псевдонимов таблиц	175
Другие виды соединений	177

Использование соединений совместно с итоговыми функциями	184
Правила создания соединений	186
Резюме	187
Упражнения	188
Урок 14. Комбинированные запросы	189
Что такое комбинированные запросы	189
Создание комбинированных запросов	190
Резюме	198
Упражнения	199
Урок 15. Добавление данных	201
Способы добавления данных	201
Копирование данных из одной таблицы в другую	210
Резюме	212
Упражнения	212
Урок 16. Обновление и удаление данных	213
Обновление данных	213
Удаление данных	216
Советы по обновлению и удалению данных	218
Резюме	220
Упражнения	220
Урок 17. Создание таблиц и работа с ними	221
Создание таблиц	221
Обновление таблиц	228
Удаление таблиц	231
Переименование таблиц	232
Резюме	233
Упражнения	233

Урок 18. Представления	235
Что такое представления	235
Создание представлений	240
Резюме	248
Упражнения	249
Урок 19. Хранимые процедуры	251
Что такое хранимые процедуры	251
Зачем нужны хранимые процедуры	252
Выполнение хранимых процедур	255
Создание хранимых процедур	256
Резюме	262
Урок 20. Обработка транзакций	263
Что такое транзакции	263
Управление транзакциями	266
Резюме	272
Урок 21. Курсоры	273
Что такое курсоры	273
Работа с курсорами	275
Резюме	280
Урок 22. Расширенные возможности SQL	281
Что такое ограничения	281
Что такое индексы	290
Что такое триггеры	294
Безопасность баз данных	296
Резюме	297
Приложение А. Сценарии демонстрационных таблиц	299
Демонстрационные таблицы	299
Получение демонстрационных таблиц	304

Приложение Б. Синтаксис инструкций SQL	309
ALTER TABLE	309
COMMIT	310
CREATE INDEX	310
CREATE PROCEDURE	310
CREATE TABLE	311
CREATE VIEW	311
DELETE	312
DROP	312
INSERT	312
INSERT SELECT	312
ROLLBACK	313
SELECT	313
UPDATE	314
Приложение В. Типы данных в SQL	315
Строковые типы данных	316
Числовые типы данных	318
Типы данных даты и времени	320
Бинарные типы данных	321
Приложение Г. Резервированные слова SQL	323
Приложение Д. Ответы на упражнения	329
Предметный указатель	345

УРОК 1

Основы SQL

На этом уроке вы узнаете, что такое SQL и что такое запросы к базам данных.

Терминология баз данных

Раз вы читаете книгу по SQL, значит, вам так или иначе необходимо работать с базами данных. SQL предназначен для создания запросов к базам данных, поэтому, прежде чем перейти к рассмотрению самого языка, очень важно ознакомиться с основными понятиями баз данных.

Хотите вы того или нет, но вы постоянно пользуетесь базами данных. Каждый раз, когда вы выбираете имя в адресной книге электронной почты, вы обращаетесь к базе данных. Когда вы выполняете поиск в Интернете, то посылаете запросы к базе данных. Когда вы регистрируетесь на офисном компьютере, то вводите свое имя и пароль, которые затем сравниваются со значениями, хранящимися в базе данных. И даже когда вы вставляете свою пластиковую карту в банкомат, проверка PIN-кода и остатка на счете идет через базу данных.

Но, несмотря на то что мы постоянно имеем дело с базами данных, для многих остается загадкой, что же это такое. Отчасти непонимание связано с тем, что разные люди вкладывают разный смысл в одни и те же термины. Поэтому мы начнем с определения наиболее важных терминов, относящихся к базам данных.



Основные концепции

Ниже дан очень краткий обзор основных понятий баз данных. Он предназначен для того, чтобы освежить ваши знания либо дать самые общие представления, если вы новичок. Понимание основ баз данных необходимо для изучения SQL, поэтому рекомендую найти хорошую книгу по базам данных и пополнить свой багаж знаний.

Базы данных

Термин *база данных* используется в самых разных областях, но применительно к SQL мы будем считать базу данных набором записей, хранящихся неким упорядоченным способом. Проще всего рассматривать базу данных как шкаф для хранения документов. Шкаф — это просто физический объект для хранения данных, независимо от того, что это за данные и как они упорядочены.



База данных

Контейнер (обычно файл или группа файлов), предназначенный для хранения упорядоченных данных.



Неправильное употребление термина приводит к путанице

Люди часто используют термин *база данных* для обозначения программного обеспечения, управляющего базой данных. Это ведет к путанице. В действительности такое программное обеспечение называется *СУБД* (система управления базами данных). База данных — это хранилище, создаваемое и управляемое посредством СУБД. Что именно представляет собой такое хранилище, зависит от конкретной СУБД.

Таблицы

Когда вы храните документы в шкафу, вы стараетесь не сваливать их в кучу. Вместо этого вы раскладываете их по соответствующим папкам.

В базах данных такая папка называется таблицей. *Таблица* — это структурированный файл, в котором могут храниться данные определенного типа. В таблице может находиться список клиентов, каталог продукции и любая другая информация.



Таблица

Структурированный набор данных определенного типа.

Ключевой момент заключается в том, что данные, хранимые в таблице, должны быть одного типа или взяты из одного списка. Никогда не храните список клиентов и список заказов в одной таблице базы данных. Это затрудняет поиск и получение информации. Лучше создать две таблицы для каждого из списков.

Каждая таблица базы данных обладает уникальным именем, идентифицирующим ее, и никакая другая таблица в базе данных не может называться таким именем.



Имена таблиц

Уникальность имени таблицы достигается комбинацией нескольких компонентов, включая имя базы данных и самой таблицы. В качестве части уникального имени в некоторых базах данных применяется также имя владельца. Это означает, что нельзя использовать два одинаковых имени таблицы в одной базе, но в разных базах данных имена таблиц могут повторяться.

Таблицы имеют определенные характеристики и свойства, определяющие, каким образом в них хранятся данные. Сюда входит информация о том, какие данные могут храниться в таблицах, как они распределены по таблицам, как называются отдельные информационные блоки и многое другое. Подобный набор информации, описывающей таблицу, называется *схемой*. Схемы служат для описания как отдельных таблиц в базе данных, так и базы данных в целом (а также для описания связей между таблицами, если таковые имеются).



Схема

Информация о базе данных, а также о структуре и свойствах ее таблиц.

Столбцы и типы данных

Таблицы состоят из *столбцов*, в которых хранятся отдельные фрагменты информации.



Столбец

Одиночное поле таблицы. Все таблицы состоят из одного или нескольких столбцов.

Чтобы лучше понять это, представьте себе таблицу базы данных в виде сетки ячеек, как в Excel. В каждом столбце сетки находится определенная часть информации. Например, в таблице клиентов в одном столбце находится номер клиента, в другом — его имя. Адрес, город, штат, почтовый индекс — все это хранится в отдельных столбцах.



Распределение данных по столбцам

Очень важно правильно распределить данные по нескольким столбцам. Например, название города и штата, а также почтовый индекс всегда должны находиться в отдельных столбцах. Это позволяет сортировать и фильтровать данные по столбцам (например, для поиска всех клиентов из определенного штата или города). Если названия города и штата будут храниться в одном столбце, то это сильно затруднит сортировку и фильтрацию данных по штатам.

Когда вы распределяете данные по столбцам, уровень дробления определяется вами и требованиями конкретной базы данных. Например, адреса обычно хранятся в виде названия улицы и номера дома. Это удобно, если только однажды вы не решите отсортировать таблицу по названиям улиц. В таком случае предпочтительно отделять номера домов от названий улиц.

С каждым столбцом базы данных связан определенный *тип данных*, который определяет, какие данные могут храниться в этом столбце. Например, если в столбце должны содержаться числа (допустим, количество товаров в заказе), то тип данных будет числовым. Если в столбце необходимо хранить даты, заметки, денежные суммы и т.п., то для всех этих данных предусмотрены соответствующие типы.



Тип данных

Тип разрешенных для хранения данных. Каждому столбцу таблицы присваивается тип, который разрешает хранить в нем только определенную информацию.

Типы данных ограничивают характер информации, которую можно хранить в столбце (например, предотвращают ввод алфавитных символов в числовое поле). Типы данных также помогают корректно сортировать инфор-

мацию и играют важную роль в оптимизации использования места на диске. Таким образом, выбору типов данных для столбцов создаваемой таблицы необходимо уделить особое внимание.



Совместимость типов данных

Типы данных и их названия служат одним из основных источников несовместимости в SQL. Базовые типы данных обычно поддерживаются всеми СУБД примерно одинаково, в отличие от некоторых сложных типов. Более того, иногда вы будете сталкиваться с тем, что один и тот же тип данных в разных СУБД называется по-разному. К сожалению, с этим ничего нельзя поделать, просто следует помнить о таких вещах при создании схем таблиц.

Строки

Данные в таблице хранятся в строках, и каждая запись содержится в своей строке. Возвращаясь к сравнению с сеткой ячеек Excel, можно сказать, что ее вертикальные ряды представляют собой столбцы таблицы, а горизонтальные ряды — это строки.

Например, в таблице клиентов информация о каждом клиенте хранится в отдельной строке. Число строк в таблице равно числу записей о клиентах.



Строка

Отдельная запись в таблице.



Записи или строки?

Часто пользователи баз данных говорят о *записях*, имея в виду *строки*. Эти два термина взаимозаменяемы.

Первичные ключи

В каждой строке таблицы должно быть несколько столбцов, которые уникальным образом идентифицируют ее. В таблице клиентов это могут быть номера клиентов, тогда как в таблице заказов таким столбцом может служить идентификатор заказа. В таблице со списком сотрудников можно использовать столбец с номерами сотрудников, а в таблице со списком книг уникальным идентификатором будет служить ISBN.



Первичный ключ

Столбец (или набор столбцов), значения которого уникальным образом идентифицируют каждую строку таблицы.

Столбец (или набор столбцов), уникально идентифицирующий каждую строку таблицы, называется *первичным ключом*. Первичный ключ нужен для обращения к конкретной строке. Без него выполнять обновление или удаление строк таблицы было бы очень затруднительно, так как не было бы никакой гарантии, что изменяются нужные строки.



Всегда определяйте первичные ключи

Несмотря на то что первичные ключи не являются обязательными, большинство разработчиков баз данных создают их для каждой таблицы, чтобы в будущем иметь возможность выполнять любые манипуляции с данными.

Любой столбец таблицы может быть выбран в качестве первичного ключа, если соблюдаются следующие условия.

- Две разные строки не могут иметь одно и то же значение первичного ключа.

- Каждая строка должна иметь определенное значение первичного ключа (столбцы первичного ключа не могут содержать значения NULL).
- Значения в столбце первичного ключа не могут быть изменены.
- Значения первичного ключа нельзя использовать дважды. (Если строка удалена из таблицы, то ее первичный ключ нельзя в дальнейшем назначать другим строкам.)

В качестве первичного ключа обычно выбирается только один столбец таблицы. Но данное требование не обязательно, и первичным ключом могут служить несколько столбцов. При этом приведенные выше правила должны соблюдаться для всех столбцов первичного ключа, а все комбинации их значений должны быть уникальными (в отдельных столбцах значения могут повторяться).

Существует еще один важный тип ключа — *внешний ключ*, но его мы рассмотрим на уроке 22.

Что такое SQL

SQL (Structured Query Language) — это язык структурированных запросов, который был специально разработан для взаимодействия с базами данных.

В отличие от других языков (таких, как Java, C или Python), SQL состоит из относительно небольшого количества слов английского языка. Так было изначально задумано. SQL разрабатывался для решения одной конкретной задачи — предоставлять простой и эффективный способ чтения и записи информации из баз данных.

Каковы же преимущества SQL?

- SQL не относится к числу проприетарных языков, используемых поставщиками конкретных СУБД. Почти все ведущие СУБД поддерживают SQL, по-

этому знание данного языка позволит вам взаимодействовать практически с любой базой данных.

- SQL легко изучить. Его немногочисленные инструкции состоят из простых английских слов.
- Несмотря на кажущуюся простоту, SQL — это очень мощный язык. Умело пользуясь его инструкциями, можно выполнять очень сложные операции с базами данных.

Вот почему стоит изучить SQL.



Расширения SQL

Многие поставщики СУБД расширили возможности SQL, добавив в язык дополнительные операторы или инструкции. Эти расширения необходимы для обеспечения дополнительной функциональности или для упрощения определенных операций. И хотя часто они бывают очень полезными, подобные расширения специфичны для конкретной СУБД и редко поддерживаются более чем одним поставщиком. Стандарт SQL контролируется комитетом ANSI и соответственно называется ANSI SQL. Все крупные СУБД, даже те, у которых есть собственные расширения, поддерживают ANSI SQL. Отдельным реализациям присвоены собственные имена (PL-SQL в Oracle, Transact-SQL в Microsoft SQL Server и др.).

Чаще всего в книге рассматривается именно ANSI SQL. В тех редких случаях, когда используется разновидность SQL, относящаяся к определенной СУБД, об этом упоминается отдельно.

Попробуйте сами

Подобно изучению любого другого языка, лучше всего попробовать применить SQL на практике. Для этого вам понадобится база данных и СУБД, в которой можно выполнять SQL-запросы.



Какую СУБД выбрать?

Для работы с примерами книги требуется СУБД. Но какую из них выбрать?

К счастью, приводимый в книге код SQL может выполняться практически в любой СУБД. Соответственно, вы вправе выбирать то, что для вас удобнее и проще.

Тут есть два варианта. Прежде всего, можете установить СУБД на свой компьютер, чтобы иметь полный доступ к базам данных. Впрочем, для многих именно установка и настройка СУБД — самая сложная часть в изучении SQL. Альтернативный вариант — получить доступ к серверной (или облачной) СУБД, чтобы ничего не требовалось устанавливать и конфигурировать.

Для выбора доступно множество СУБД. Вот несколько рекомендаций.

- MySQL (как и ее ответвление MariaDB) — самое лучшее решение, потому что это легко устанавливаемая бесплатная СУБД с поддержкой для всех операционных систем. Имеется утилита командной строки, позволяющая вводить SQL-код, но есть и графическая утилита MySQL Workbench, которую нужно загружать и устанавливать отдельно.
- Для пользователей Windows доступно приложение Microsoft SQL Server Express. Это бесплатная версия популярной СУБД SQL Server, включающая клиентский модуль SQL Server Management Studio.

Теперь что касается выбора серверной (или облачной) СУБД.

- Если вы изучаете SQL для применения на работе, то, скорее всего, у вас на фирме уже используется какая-то СУБД. Попросите администратора создать для вас учетную запись и предоставить вам утилиту, с помощью которой можно вводить и тестировать SQL-запросы.
- Облачные СУБД — это экземпляры, выполняющиеся на виртуальных серверах, благодаря чему вам не нужно ничего устанавливать у себя на компьютере. Такие СУБД предлагаются всеми ведущими постав-

щиками облачных служб (включая Google, Amazon и Microsoft). К сожалению, конфигурирование этих СУБД (включая настройку удаленного доступа) — нетривиальная задача, зачастую требующая большего объема работы, чем при установке локальной СУБД. Исключение составляют разве что Oracle Live SQL и IBM Db2 on Cloud, предлагающие бесплатную версию с веб-интерфейсом.

Ссылки на все упомянутые СУБД доступны на сайте книги по адресу <http://forta.com/books/0135182794>. По мере обновления СУБД эта страница тоже будет обновляться.

Во всех упражнениях книги используются реальные инструкции SQL и полноценные таблицы базы данных. В приложении А описываются все демонстрационные таблицы и рассказывается о том, как их получить (или создать самостоятельно), чтобы можно было выполнять примеры каждого урока.

Кроме того, начиная с урока 2 в конце каждой главы приводятся упражнения. Это даст вам возможность закрепить полученные знания и применить их для решения задач, не рассмотренных в самой главе. Ответы на упражнения приводятся в приложении Д.

Резюме

На первом уроке вы узнали, что такое SQL и для чего он нужен. В связи с тем что SQL применяется для взаимодействия с базами данных, мы также рассмотрели основную терминологию баз данных.