

Предисловие

Языки программирования постоянно появляются и исчезают, и сегодня используется очень мало языков, которые имеют корни, уходящие в прошлое более чем на десятилетие. Вот некоторые из них: COBOL, который все еще довольно активно используется в мейнфреймах; Java, который родился в середине 1990-х и стал одним из самых распространенных языков программирования; C, который по-прежнему довольно популярен при разработке операционных систем и серверов, а также для встроенных систем. В области баз данных у нас имеется SQL, который появился в 1970-х годах.

SQL был разработан как язык для создания, выборки и обработки данных из реляционных баз данных, которые существуют уже более 40 лет. Однако за последнее десятилетие или около того приобрели большую популярность другие платформы данных, такие как Hadoop, Spark и NoSQL, занявшие свои ниши на рынке реляционных баз данных. Однако, как описано в нескольких последних главах этой книги, язык SQL постоянно развивается, чтобы упростить выборку данных на различных платформах независимо от того, где хранятся данные — в таблицах, документах или простых “плоских” файлах.

Зачем изучать SQL

Независимо от того, будете ли вы использовать реляционную базу данных, если вы работаете в области науки о данных или бизнес-аналитиком либо сталкиваетесь с некоторым иным аспектом анализа данных, вам, вероятно, потребуется знание SQL, а также других языков/платформ, таких как Python и R. Данные окружают нас в огромных количествах (и их объемы продолжают возрастать быстрыми темпами), и люди, способные извлекать из них значимую информацию, пользуются большим спросом.

Почему стоит использовать данную книгу

Есть много книг, в которых к вам относятся как к чайнику, идиоту или какому-то иному простаку, но эти книги, как правило, поверхностны. На другом конце спектра — справочные руководства, в которых подробно описывается каждая перестановка каждой инструкции языка, что может быть полезно, только если у вас уже есть хорошее представление о том, что вы хотите сделать, и нужно только вспомнить синтаксис. В этой книге мы попытаемся найти золотую середину, начав с некоторых азов языка SQL, пройдя через его основы и продолжив более “продвинутыми” функциями, которые позволят вам по-настоящему проявить себя. Кроме того, в последней главе показано, как можно запрашивать информацию из нереляционных баз данных, — вопрос, достаточно редко рассматриваемый в книгах, являющихся вводным курсом.

Структура книги

Книга состоит из 18 глав и двух приложений.

Глава 1, “Небольшая предыстория”

История компьютеризованных баз данных, в том числе реляционной модели и языка SQL.

Глава 2, “Создание и наполнение базы данных”

Создание базы данных MySQL, а также таблиц, используемых в качестве примеров в этой книге, и заполнение таблиц данными.

Глава 3, “Запросы”

Инструкция `select` и наиболее распространенные конструкции (`select`, `from`, `where`).

Глава 4, “Фильтрация”

Типы условий, которые можно использовать в конструкции `where` инструкции `select`, `update` или `delete`.

Глава 5, “Запросы к нескольким таблицам”

Использование запросов нескольких таблиц посредством их соединения.

Глава 6, “Работа с множествами”

Наборы данных и их взаимодействие в рамках запросов.

Глава 7, “Генерация, обработка и преобразование данных”

Встроенные функции, используемые для управления данными или их преобразования.

Глава 8, “Группировка и агрегация”

Способы агрегирования данных.

Глава 9, “Подзапросы”

Подзапросы и их использование.

Глава 10, “Соединения”

Дальнейшее исследование типов соединения таблиц.

Глава 11, “Условная логика”

Применение условной логики в инструкциях `select`, `insert`, `update` и `delete`.

Глава 12, “Транзакции”

Транзакции и их использование.

Глава 13, “Индексы и ограничения”

Индексы и ограничения.

Глава 14, “Представления”

Создание интерфейса для защиты пользователей от сложных данных.

Глава 15, “Метаданные”

Словарь данных, его преимущества и работа с ним.

Глава 16, “Аналитические функции”

Функциональность, используемая для создания рейтингов, промежуточных итогов и других значений, активно используемых в отчетности и анализе.

Глава 17, “Работа с большими базами данных”

Технологии, упрощающие управление и обход очень больших баз данных.

Глава 18, “SQL и большие данные”

Вариации языка SQL, позволяющие извлекать данные из нереляционных платформ данных.

Приложение А, “Схема базы данных Sakila”

Схема базы данных, используемая для всех примеров в книге.

Приложение Б, “Ответы к упражнениям”

Ответы к упражнениям.

Соглашения, принятые в этой книге

В этой книге используются следующие типографские соглашения.

Курсив

Используется для обозначения новых терминов.

Моноширинный шрифт

Используется для оформления листингов программ, а также внутри абзацев для ссылки на такие элементы программ, как имена переменных или функций, базы данных, типы данных, переменные среды, инструкции и ключевые слова, а также URL-адреса, адреса электронной почты, имена и расширения файлов.

Моноширинный курсив

Используется для оформления текста, который следует заменить значениями, вводимыми пользователем, или значениями, определенными из контекста.

Полужирный моноширинный шрифт

Используется для оформления команд или другого текста, который пользователь должен набирать буквально.



Обозначает совет, предложение или общее примечание. Например, здесь такие примечания используются, чтобы указать новые полезные функции в Oracle9i.



Обозначает предупреждение или предостережение, например предупреждение о том, что некоторая инструкция SQL может иметь непредвиденные последствия, если не использовать ее с осторожностью.

Использование примеров из этой книги

Чтобы экспериментировать с данными, использованными в примерах в этой книге, у вас есть два варианта.

- Загрузите и установите сервер MySQL версии 8.0 (или более поздней) и загрузите пример базы данных Sakila по адресу <https://dev.mysql.com/doc/index-other.html>.
- Перейдите по адресу <https://www.katacoda.com/mysql-db-sandbox/scenarios/mysql-sandbox>, чтобы получить доступ к “песочнице” MySQL, в которой база данных Sakila загружена в экземпляр MySQL. Вам потребуется создать (бесплатную) учетную запись Katacoda, а затем щелкнуть на кнопке **Start Scenario**.

Если вы выберете второй вариант, то, как только вы запустите сценарий, будет установлен и запущен сервер MySQL, а затем загружены схема и данные Sakila. Когда все будет готово, появится стандартное приглашение `mysql>`, и вы сможете начать посылать запросы в базу данных. Это, безусловно, самый простой вариант, и я думаю, что большинство читателей выберет именно его; если он вам нравится, смело переходите к следующему разделу.

Если вы предпочитаете иметь собственную копию данных и хотите, чтобы внесенные вами изменения были постоянными, или если вы просто заинтересованы в установке сервера MySQL на собственной машине, то, возможно, вы предпочтете первый вариант. Вы можете также использовать сервер MySQL, размещенный в такой среде, как Amazon Web Services или Google Cloud. В любом случае вам нужно будет выполнить установку и настройку самостоятельно, так как этот материал выходит за рамки данной книги. Как только ваша база данных станет доступной, вам нужно будет выполнить еще несколько шагов, чтобы загрузить базу данных Sakila.

Сначала запустите клиент командной строки `mysql` и укажите пароль, а затем выполните следующие действия.

1. Перейдите по адресу <https://dev.mysql.com/doc/index-other.html> и загрузите файлы “sakila database” в разделе Example Databases.
2. Поместите файлы в локальный каталог, например в `C:\temp\sakila-db` (используется в следующих двух шагах; воспользуйтесь путем к вашему реальному каталогу).
3. Введите `source c:\temp\sakila-db\sakila-schema.sql`; и нажмите `<Enter>`.

4. Введите source `c:\temp\sakila-db\sakila-data.sql`; и нажмите <Enter>.

Теперь у вас должна быть рабочая база данных, заполненная всеми данными, необходимыми для примеров в этой книге.

Благодарности

Я хотел бы поблагодарить моего редактора Джеффа Блейла (Jeff Bleiel) за то, что он помог сделать это издание реальностью, а также Томаса Нилда (Thomas Nield), Энн Уайт-Уоткинс (Ann White-Watkins) и Чарльза Живра (Charles Givre), которые были столь любезны, что просмотрели мою книгу и высказали свои замечания. Спасибо также Деб Бейкер (Deb Baker), Джесс Хаберман (Jess Haberman) и всему персоналу O'Reilly Media, которые участвовали в создании этой книги. Наконец, я благодарю свою жену Нэнси (Nancy) и моих дочерей Мишель (Michelle) и Николь (Nicole) за поддержку и терпение.

Ждем ваших отзывов!

Вы, читатель этой книги, и есть главный ее критик. Мы ценим ваше мнение и хотим знать, что было сделано нами правильно, что можно было сделать лучше и что еще вы хотели бы увидеть изданным нами. Нам интересны любые ваши замечания в наш адрес.

Мы ждем ваших комментариев и надеемся на них. Вы можете прислать нам электронное письмо либо просто посетить наш веб-сайт и оставить свои замечания там. Одним словом, любым удобным для вас способом дайте нам знать, нравится ли вам эта книга, а также выскажите свое мнение о том, как сделать наши книги более интересными для вас.

Отправляя письмо или сообщение, не забудьте указать название книги и ее авторов, а также свой обратный адрес. Мы внимательно ознакомимся с вашим мнением и обязательно учтем его при отборе и подготовке к изданию новых книг.

Наши электронные адреса:

E-mail: info.dialektika@gmail.com

WWW: <http://www.dialektika.com>