

ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАНИЕ

# Базы данных

*Учебник*

ИЗДАТЕЛЬСТВО  
**FOLIANT**

Нур-Султан  
2019

**УДК 004**  
**ББК 32.973.26**  
**Б 17**

Авторы:  
Дэлер Эльмар  
Харди Дирк  
Троссман Хуберт

**Рецензенты:**

**Шангытбаева Г. А.** – PhD по специальности «Информационные технологии» Актюбинского регионального государственного университета им. К. Жубанова

**Нурина Г. К.** – преподаватель специальных дисциплин Актюбинского высшего политехнического колледжа, магистр техники и технологии по специальности «Информационные системы»

**Марат Г. М.** – преподаватель специальных дисциплин и производственного обучения Актюбинского высшего политехнического колледжа, магистр техники и технологии по специальности «Информационные технологии»

**Б17 Базы данных:** Учебник / Пер. с немецкого. – Нур-Султан: Фолиант, 2019. – 184 с.

ISBN 978-601-338-415-3

Материалы учебника напрямую связаны с учебной программой модуля «Проектирование баз данных». Модуль относится к циклу профессиональных модулей, обеспечивающий соответствующий уровень знаний и умений в области программирования базы данных. Рассмотрены теоретические и практические основы для планирования, реализации и программирования баз данных с использованием современных программных систем. Большое внимание при этом уделяется разъяснению взаимосвязей.

Предназначен для студентов учебных заведений технического и профессионального образования, обучающихся по специальностям 1305000 «Информационные системы», 1304000 «Вычислительная техника и программное обеспечение».

**УДК 004**  
**ББК 32.973.26**

ISBN 978-601-338-415-3

© 2018 Verlag Europa-Lehrmittel, Nourney  
3-е издание  
© Издательство «Фолиант», переводная, 2019

## Предисловие

Информатика и информационные технологии влияют практически на все социальные сферы нашей жизни. Почти все профессиональные действия и процессы в значительной степени осуществляются или поддерживаются благодаря информационным технологиям. Системы баз данных являются важным компонентом, так как от доступности, полноты и точности хранимых данных зависит дееспособность самой компании.

Настоящая книга «Базы данных – разработка, программирование, применение» предоставляет теоретические и практические основы для планирования, реализации и программирования баз данных с использованием современных программных систем. Большое значение при этом уделяется разъяснению взаимосвязей.

Являясь введением в предметную область технологий баз данных, эта книга подойдет обучающимся и студентам профессиональных училищ, колледжей, профессиональных академий, гимназий, специальных высших учебных заведений и университетов.

В отдельных главах наряду с многочисленными примерами также содержатся дифференцированные практические упражнения, которые служат для закрепления и углубления тематических областей.

Будем благодарны всем внимательным читателям, за ценные замечания, которые мы учтем в следующих изданиях нашей книги.

Нам важно Ваше мнение по этой книге!

Мы будем рады принять Ваши отзывы и предложения на электронную почту: [lektorat@europa-lehrmittel.de](mailto:lektorat@europa-lehrmittel.de).

Осень 2018 г.

Авторы и издатели

## СОДЕРЖАНИЕ

Предисловие .....	3
1 Основные понятия о базе данных .....	9
1.1 Использование баз данных .....	9
1.1.1 Примеры использования баз данных .....	9
1.1.2 Проблемы хранения данных в базах данных .....	10
1.1.3 Задачи СУБД .....	11
1.2 Системные архитектуры .....	13
1.2.1 Базы данных для настольных ПК (однопользовательский доступ) .....	13
1.2.2 Базы данных на ПК для нескольких пользователей (многопользовательский доступ) .....	13
1.2.3 Базы данных клиент-сервер .....	13
1.3 Модели баз данных .....	14
1.3.1 Реляционные базы данных .....	14
1.3.2 Объектно-ориентированные базы данных .....	14
1.3.3 Иерархические и сетевые базы данных .....	14
1.4 Архитектура системы управления базами данных (СУБД) .....	15
1.4.1 Трехуровневая архитектура (трехслойная архитектура) .....	15
1.5 Этапы проектирования базы данных .....	17
1.6 Упражнения к Главе 1 .....	17
2 Реляционные системы баз данных .....	18
2.1 Реляционные системы баз данных .....	18
2.1.1 Таблицы и связи .....	18
2.1.2 Ключи и связи .....	19
2.2 Модель типа «сущность – связь» .....	21
2.3 Примеры с решением для модели типа «сущность–связь» .....	24
2.3.1 Обработка заказа .....	24
2.3.2 Поставщики и товары .....	24
2.4 Упражнения к Главе 2 .....	25
3 Разработка и нормализация базы данных .....	29
3.1 Разработка базы данных .....	29
3.1.1 Процедуры разработки программного обеспечения .....	30
3.2 Нормализация .....	30
3.2.1 Нормальные формы .....	30

3.2.2	Пример для нормализации: дистанционная торговля .....	33
3.2.3	Другие нормальные формы .....	36
3.2.4	Условия целостности .....	36
3.3	Упражнения к Главе 3 .....	37
4	Программное обеспечение для моделирования баз данных .....	40
4.1	DB-Designer .....	40
4.1.1	Загрузка и установка .....	40
4.1.2	Создание таблиц .....	43
4.1.3	Реляционное связывание таблиц .....	45
4.1.4	Запись строк базы данных .....	46
4.1.5	Создание ER-диаграммы .....	46
4.1.6	Forward Engineering (Прямое проектирование) .....	50
4.2	Microsoft VISIO .....	53
4.2.1	Запуск диаграммы модели базы данных .....	53
4.2.2	Создание таблиц .....	54
4.2.3	Добавление столбцов .....	55
4.2.4	Добавление связей .....	56
4.2.5	Реверсивный инжиниринг .....	57
4.2.6	Создание индексов .....	60
4.2.7	Создание представлений (Views) .....	62
4.2.8	Создание условий проверки полей .....	64
5	Разработка базы данных в среде Access .....	65
5.1	Создание таблиц .....	65
5.2	Определение связей и ссылочной целостности .....	67
5.3	Формы .....	69
5.3.1	Создание формы .....	69
5.3.2	Подчиненные формы .....	70
5.3.3	Управление базами данных с помощью кнопок .....	72
5.4	Макросы .....	73
5.5	Создание отчета .....	74
5.6	Создание запросов к базе данных .....	76
5.7	Упражнения к Главе 5 .....	78
6	Язык базы данных SQL .....	79
6.1	Стандарты SQL .....	79
6.2	Создание, изменение и удаление таблиц .....	80
6.3	Запросы выбора с помощью SELECT .....	81
6.3.1	Ограничение запросов на выбор с условиями .....	81

6.3.2 DISTINCT .....	82
6.3.3 Представление содержимого поля в условии WHERE.....	82
6.3.4 Оператор BETWEEN .....	83
6.3.5 Оператор IN .....	83
6.3.6 Работа с нулевыми значениями .....	84
6.3.7 Сортировка данных.....	84
6.3.8 Ограничение результатов запроса .....	
6.3.8 Ограничение результатов запроса .....	85
6.3.9 Функции в SELECT-запросах .....	86
6.3.10 Группировка данных.....	89
6.3.11 Запросы по нескольким таблицам (JOIN).....	90
6.3.12 Подзапросы.....	95
6.4 Редактирование данных с помощью SQL.....	96
6.4.1 Вставка строк базы данных.....	96
6.4.2 Удаление строк базы данных .....	97
6.4.3 Обновление данных .....	97
6.5 Согласованность базы данных.....	98
6.6 Транзакции.....	99
6.7 Упражнения к Главе 6 .....	100
7 LibreOffice Base .....	103
7.1 Создание базы данных.....	103
7.2 Создание связей между таблицами .....	113
7.3 Запись строк базы данных.....	115
7.4 Подключение к другим базам данных .....	116
7.6 Формы .....	123
8 Базы данных в сети Интернет .....	128
8.1 Среда разработки ХАМРР.....	128
8.2 Работа компонентов .....	128
8.2.1 Веб-сервер .....	128
8.2.2 Установка среды разработки ХАМРР .....	129
8.2.3 Запуск компонентов .....	129
8.3 Язык сценариев PHP .....	130
8.3.1 Введение .....	130
8.3.2 Написание сценария PHP .....	130
8.3.3 Переменные в PHP .....	131
8.3.4 Массивы (Arrays) .....	131
8.3.5 Работа с массивами .....	135
8.3.2 Редактирование символьных строк.....	135
8.3.7 Операции с файлами с помощью PHP .....	136

8.3.8 Права доступа к файлам .....	138
8.3.9 Работа с формами .....	139
8.4 Система управления базами данных MySQL .....	140
8.4.1 Работа с клиентами MySQL работает .....	141
Клиентская часть mysql .....	141
8.4.2 Предоставление и отмена прав доступа .....	143
8.4.3 Редактирование базы данных MySQL с помощью PHP .....	145
8.5 Обмен данными через интерфейсы ODBC .....	147
9 Доступ к базе данных через Java .....	150
9.1 Доступ к базе данных через Java .....	150
9.1.2 Загрузка драйверов JDBC и соединение .....	150
9.1.3 Доступ к базе данных SQLite .....	151
9.1.4 Отмена команд, отличных от SELECT .....	154
9.1.5 Получение метаданных .....	155
9.2 Обратиться к другим базам данных .....	157
9.2.1 Добавление драйвера .....	157
9.2.2 Другие драйверы базы данных .....	158
9.3 Упражнения к Главе 9 .....	158
10 Доступ к базе данных с помощью .NET и C# .....	160
10.1 Доступ к базе данных с помощью .NET и C # .....	160
10.1.1 Подключение базы данных под .NET Framework .....	160
10.1.2 Использование провайдеров и установление соединения .....	161
10.1.3 Пример доступа к базе данных на ACCESS .....	161
10.1.4 Отмена команд, отличных от SELECT .....	164
10.1.5 DataAdapter и набор данных .....	166
10.2 Использование мастера базы данных Visual C# .....	169
10.2.1 Подключение базы данных .....	169
10.2.2 Автоматическое подключение элементов управления Windows Forms .....	172
10.2.3 Автоматическое подключение элементов управления WPF .....	174
10.3 Задачи к Главе 10 .....	179



# 1 Основные понятия о базе данных

## 1.1 Использование баз данных

База данных – это набор сохраненных данных, которые находятся в логической взаимосвязи и управляются **системой управления базами данных (СУБД)**. Эти данные используются, например, прикладными программами и пользователями предприятия.

### Примечание

Базы данных являются логически связанными наборами данных.

### 1.1.1 Примеры использования баз данных

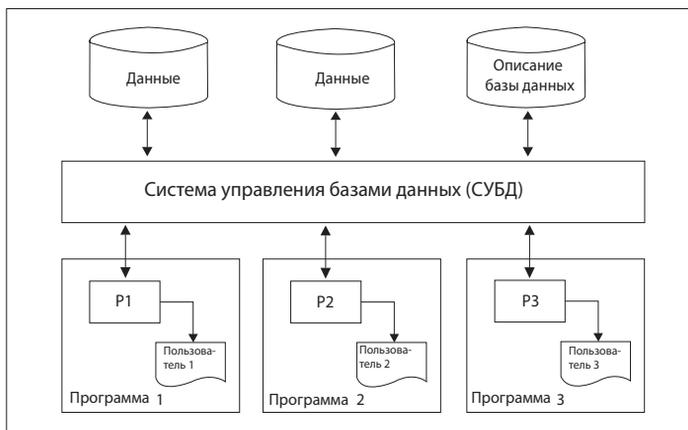
Базы данных часто играют главную роль в использовании предприятием компьютерных технологий. Везде, где рабочие процессы обрабатываются с помощью компьютера, требуется хранение больших объемов данных.

### Примеры

- Банки и страховые компании работают с системами управления базами данных. В базе данных структурирована вся информация о счетах, проводках и клиентах. Конфиденциальность и безопасность данных имеют наивысший приоритет.
- В компаниях любого размера и отраслях с системой планирования ресурсов (ERP, Enterprise Resource Planning), данные о клиентах, сотрудниках или товарах хранятся и обрабатываются с помощью системы управления базами данных.
- Автоматизированное управление складом также требует использования баз данных. База данных склада содержит упорядоченную информацию о многочисленных поставщиках, товарах и их запасах.
- Информационные системы в Интернете (например, Wikipedia) управляют своими статьями, используя базы данных.
- Компании хранят в хранилищах данных (Data Warehouses) различную информацию для анализа данных и помощи в принятии решений в различных сферах бизнеса. Таким же образом, например, институт изучения рынков хранит собственные данные и данные третьих лиц для их дальнейшей обработки.



Прикладные программы, например, программы для управления складом или персоналом, могут параллельно обращаться к общим данным посредством СУБД



### 1.1.2 Проблемы хранения данных в базах данных

При хранении данных с помощью баз данных могут возникнуть многочисленные проблемы:

- **Избыточность**  
Данные сохраняются несколько раз (дублируются), что делает дорогостоящим внесение изменений в массив данных, а вероятность ошибок возрастает. Одни и те же данные должны быть изменены несколько раз в разных местах. Например, если изменения многократно сохраненных данных производится только в одном месте, то массив данных будет ошибочным.
- **Несоответствия**  
Если данные редактируются и изменяются одновременно несколькими пользователями или программами, это может привести к несогласованному состоянию данных. Доступ к данным не синхронизирован, в например, если текущий счет обрабатывается одновременно двумя пользователями, то оба видят одинаковое сальдо: 2000 евро. Теперь, если пользователь А снимет со счета 1000 евро и сохранит эту операцию, а пользователь В одновременно внесет 500 евро и сохранит эту операцию, то оба значения в базе данных: как 1000, так и 2500 евро будут противоречивыми и неверными.
- **Защита данных**  
Доступ к чтению и записи информации возможен на всем массиве данных. Защита данных – в зависимости от используемой операционной системы – может быть реализована с помощью настройки прав доступа или шифрования.
- **Отсутствие независимости данных**  
Управление данными обычно возможно только с помощью соответствующего прикладного программного обеспечения. Если необходимо изменить структуру данных, необходимо внести изменения как в прикладном программном обеспечении, так и в программе реструктуризации файлов. Также, для оценки одних и тех же файлов другого приложения необходимо создать собственное управление данными для этого нового приложения.

Для того, чтобы пользователь мог легко и четко управлять данными, ему нужна система управления базами данных – СУБД. Таким образом, **система базы данных (СБД)** состоит из комбинации базы данных (БД) и системы управления.

Наиболее распространенными СУБД являются: Microsoft Access, LibreOffice Base, MySQL, Paradox, Oracle и MS SQL Server.



### Примечание:

СУБД хранит и организует данные без избыточности, с необходимым уровнем безопасности и защитой данных. СУБД не зависит от прикладных программ (приложений), которые получают доступ к данным.

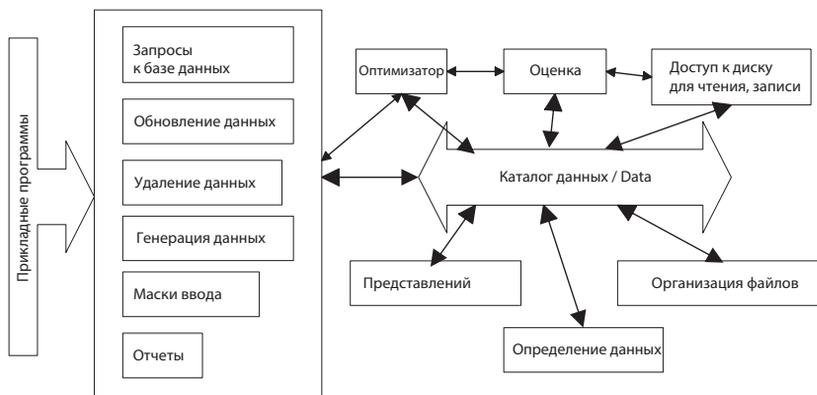
Прикладные программы не имеют прямого доступа к данным, а направляют запросы системе управления базами данных. База данных представляет собой набор логически связанных данных в определенной предметной области, например, данные клиента и данные заказа. СУБД реализует обмен данными между базой данных и их пользователями, например, прикладными программами. Она обеспечивает доступ к данным, предоставляя централизованное управление и контроль. СУБД управляет пользователями, их доступом к базе данных и правами доступа пользователей. Кроме того, при помощи СУБД обеспечивается защита от ошибок аппаратной части и программного обеспечения, поэтому при системном сбое данные не будут потеряны или смогут быть восстановлены. Внесение изменений в структуру базы данных не требует изменения прикладных программ.

### 1.1.3 Задачи СУБД

В 1982 году британский математик Эдгар Франк Кодд разработал 9 пунктов требований к СУБД:

1. Интеграция данных = единое управление всеми необходимыми данными.
2. Операции с данными = массив данных позволяет выполнять поиск, изменение и сохранение данных.
3. Каталог данных = (Data Dictionary) содержит описание базы данных.
4. Пользовательские представления = каждое приложение требует разных представлений (Views) массива данных.
5. Мониторинг согласованности = мониторинг целостности данных гарантирует правильность данных в БД.
6. Контроль доступа = доступ к данным может контролироваться и, при необходимости, пользователю может быть отказано в доступе.
7. Транзакции = изменения в БД могут быть сгруппированы как единицы.
8. Синхронизация = при конкурирующих транзакциях общие данные должны быть синхронизированы.
9. Резервное копирование данных = позволяет восстановить массив данных после конфликта, например, сбоя системы.

Разница между системой баз данных и накоплением отдельных файлов заключается в том, что в системе баз данных данные централизованно управляются системой управления базами данных – СУБД. Прикладные программы получают доступ к общим данным параллельно через СУБД.



### Архитектура СУБД

**Каталог данных** (Data Dictionary) описывает, каким образом на внутреннем уровне реализуется хранение данных. Это – центральный каталог всей информации, важной для управления данными. В частности, выделяют следующие компоненты:

- Определение организаций файл, определение путей доступа к файлам,
- Определение концептуальных данных, определение пользовательских представлений, оптимизация дескрипторов базы данных,
- Оценка запросов и изменений и контроль доступа к диску.

Отдельные **транзакции** являются автономными запросами к массиву данных. Например, при переводе денежных средств 100 € снимаются со счета А и вносятся на счет В. Информация о транзакциях хранится **в журнале транзакций**. Журнал транзакций содержит информацию о начале и конце транзакции, а также об измененном массиве данных до и после транзакции. На основе данных журнала, транзакции могут быть отслежены или сторнированы (отменены).

Программное обеспечение для управления транзакциями представляет одновременный доступ к данным. Параллельные транзакции синхронизируются для обеспечения целостности базы данных.

### Пример

На авиарейс Штутгарт-Берлин было забронировано 1 место. Клиент А в г. Ульм входит в программное обеспечение для бронирования, одновременно с клиентом В в г. Штутгарт и видит то же самое свободное место. Теперь, когда клиент А бронирует место, нажатие кнопки бронирования будет эквивалентно доступу записи к строке базы данных. В этот самый момент строка базы данных доступна клиенту А исключительно для транзакции. Если бронирование и, таким образом, транзакция будет успешно завершено, клиент В не может забронировать то же самое место. Клиент В видит, что место занято.

Налицо следующие преимущества:

- Все программы работают с одной базой данных, т. е. данные для всех одинаково актуальны.
- Одноразовое сохранение данных для всех приложений.
- Независимый одновременный доступ к общим данным под центральным управлением.