

Министерство образования и науки Российской Федерации

Калужский филиал  
федерального государственного бюджетного образовательного  
учреждения высшего образования  
**«Московский государственный технический университет  
имени Н.Э. Баумана  
(национальный исследовательский университет)»**  
(КФ МГТУ им. Н.Э. Баумана)

**С.А. Глебов**

**РЕАЛИЗАЦИЯ БАЗЫ ДАННЫХ**  
Методические указания по выполнению лабораторной работы  
по курсу «Базы данных»

Калуга – 2018

УДК 004.65  
ББК 32.972.134  
Г53

Методические указания составлены в соответствии с учебным планом КФ МГТУ им. Н.Э. Баумана по направлению подготовки 09.03.04 «Программная инженерия» кафедры «Программного обеспечения ЭВМ, информационных технологий и прикладной математики».

Методические указания рассмотрены и одобрены:

- Кафедрой «Программного обеспечения ЭВМ, информационных технологий и прикладной математики» (ФН1-КФ) протокол № 7 от «21» февраля 2018 г.


И.о. зав. кафедрой ФН1-КФ  к.т.н., доцент Ю.Е. Гагарин

- Методической комиссией факультета ФНК протокол № 2 от «28» 02 2018 г.

Председатель методической комиссии факультета ФНК  к.х.н., доцент К.Л. Анфилов

- Методической комиссией КФ МГТУ им.Н.Э. Баумана протокол № 2 от «06» 03 2018 г.

Председатель методической комиссии КФ МГТУ им.Н.Э. Баумана

 д.э.н., профессор О.Л. Перерва

Рецензент:

к.т.н., доцент кафедры ЭИУ6-КФ

 А.Б. Лачихина

Авторы

к.ф.-м.н., доцент кафедры ФН1-КФ

 С.А. Глебов

#### Аннотация

Методические указания по выполнению лабораторной работы по курсу «Базы данных» содержат руководство по разработке базы данных согласно выбранной СУБД, руководство по определению стратегии автоматической поддержки ссылочной целостности и задание на выполнение лабораторной работы.

Предназначены для студентов 3-го курса бакалавриата КФ МГТУ им. Н.Э. Баумана, обучающихся по направлению подготовки 09.03.04 «Программная инженерия».

© Калужский филиал МГТУ им. Н.Э. Баумана, 2018 г.  
© С.А. Глебов, 2018 г.

## ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ .....	4
ЦЕЛЬ И ЗАДАЧИ РАБОТЫ, ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ЕЕ ВЫПОЛНЕНИЯ.....	5
КРАТКАЯ ХАРАКТЕРИСТИКА ОБЪЕКТА ИЗУЧЕНИЯ, ИССЛЕДОВАНИЯ .....	6
ПРИМЕР ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ .....	9
ПРИМЕР СОЗДАНИЕ БАЗЫ ДАННЫХ .....	37
ЗАДАНИЕ НА ЛАБОРАТОРНУЮ РАБОТУ .....	49
ВАРИАНТЫ ЗАДАНИЙ.....	49
КОНТРОЛЬНЫЕ ВОПРОСЫ И ЗАДАНИЯ .....	52
ФОРМА ОТЧЕТА ПО ЛАБОРАТОРНОЙ РАБОТЕ .....	52
ОСНОВНАЯ ЛИТЕРАТУРА.....	53
ДОПОЛНИТЕЛЬНАЯ ЛИТЕРАТУРА .....	53

## **ВВЕДЕНИЕ**

Настоящие методические указания составлены в соответствии с программой проведения лабораторных работ по курсу «Базы данных» на кафедре «Программное обеспечение ЭВМ, информационные технологии и прикладная математика» факультета фундаментальных наук Калужского филиала МГТУ им. Н.Э. Баумана.

Методические указания, ориентированные на студентов 3-го курса направления подготовки 09.03.04 «Программная инженерия», содержат руководство по разработке базы данных согласно выбранной СУБД и руководство по определению стратегии автоматической поддержки ссылочной целостности.

## **ЦЕЛЬ И ЗАДАЧИ РАБОТЫ, ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ЕЕ ВЫПОЛНЕНИЯ**

Целью выполнения лабораторной работы является: формирование практических навыков разработки базы данных.

Основными задачами выполнения лабораторной работы являются:

1. По имеющейся даталогической модели разработать базу данных под управлением выбранной СУБД.
2. Определить стратегии автоматической поддержки ссылочной целостности.
3. Проверить функциональность.

Результатами работы являются:

- Рассмотренный принцип реализации базы данных в среде OpenOffice.
- Построенная логическая и физическая модель согласно данному варианту задания.
- Заданные связи между таблицами, свойства связей, свойства полей, ограничения на значения полей.
- Подготовленный отчет.

## КРАТКАЯ ХАРАКТЕРИСТИКА ОБЪЕКТА ИЗУЧЕНИЯ, ИССЛЕДОВАНИЯ

База данных — важнейшая составная часть информационных систем. Информационные системы предназначены для хранения и обработки больших объемов информации. Изначально такие системы существовали в письменном виде. Для этого использовались различные картотеки, папки, журналы, библиотечные каталоги и т.п. Базы данных — это реализованная с помощью компьютера информационная структура (модель), отражающая состояния объектов и их отношения.

В общем смысле термин «база данных» можно применить к любой совокупности связанной информации, объединенной вместе по определенному признаку. Примерами баз данных могут служить адресная книга, картотека, словарь, энциклопедия или кафедральный журнал. При этом в качестве базы данных рассматривается только набор данных, организованных определенным образом.

База данных предполагает наличие комплекса программных средств, обслуживающих эту базу данных и позволяющих использовать содержащуюся в ней информацию. Такие комплексы программ называются СУБД (*системы управления базой данных*). Это комплекс программных средств, предназначенных для создания структуры новой базы, наполнение ее содержимым, редактирование содержимого и визуализации информации. Под *визуализацией информации* базы понимается отбор отображаемых данных в соответствии с заданным критерием, их упорядочение, оформление и последующая выдача на устройства вывода или передачи по каналам связи.

В мире существует множество систем управления базами данных. Несмотря на то, что они могут по-разному работать с разными объектами и предоставляют пользователю различные функции и средства, большинство СУБД опираются на единый устоявшийся комплекс основных понятий.

Основные функции СУБД:

1. **Определение данных** — определить, какая именно информация будет храниться в базе данных, задать свойства данных, их тип, а также указать, как эти данные связаны между собой.

2. **Обработка данных** — данные могут обрабатываться самыми разными способами. Можно выбирать любые поля, фильтровать и сортировать данные. Можно объединять данные со связанной информацией, вычислять итоговые значения.

3. **Управление данными** — можно указать, кому разрешено знакомиться с данными, корректировать их или добавлять новую информацию. Можно определить правило коллективного доступа.

СУБД поддерживают один из возможных типов моделей данных — сетевую, иерархическую или реляционную, которые являются важнейшими признаками классификации СУБД.

Основными средствами СУБД являются:

- средства задания структуры данных;
- средства конструирования экранных форм, предназначенных для ввода данных, просмотра и обработки в диалоговом режиме;
- средства создания запросов для выборки данных при заданных условиях, а также выполнения операций по их обработке;
- средства создания отчетов из базы данных для вывода на печать результатов обработки в удобном для пользователя виде;
- языковые средства — макросы, встроенный алгоритмический язык, язык запросов и т.п., которые используются для реализации нестандартных алгоритмов обработки данных, а также процедур обработки событий в задачах пользователя;
- средства создания приложений пользователя, позволяющие объединять различные операции работы с базой данных в единый технологический процесс.

Архитектура СУБД состоит из двух основных компонентов: языка описания данных и языка манипулирования данными. Язык описания данных (ЯОД) — средства описания данных в БД и связей между ними. Средствами этого языка описывается структура БД, форматы записей, пароли, защищающие данные.

Язык манипулирования данными (ЯМД) — язык для выполнения операций над данными, позволяющий менять их строение. Для разных СУБД реализация этих уровней языков может быть различной. В одних

случаях ЯОД и ЯМД требует составления пользователем программы полностью «вручную», в других (что отражает современную тенденцию) СУБД присутствуют средства визуальной (зримой, наглядной) разработки программ. Для этого в современных СУБД имеются редакторы экранных форм, отчетов. «Кирпичиками» (инструментами) таких редакторов являются поля различных видов (поля ввода, поля вывода, вычисляемые поля), процедуры обработки различных типов (формы ввода, [таблицы](#), [отчеты](#), [запросы](#)). На основании созданных пользователем объектов программы—генераторы формируют программный код на языке конкретной машины или на промежуточном языке.

Обычно с базами данных работают две категории пользователей. Первая категория — проектировщики. Их задача состоит в разработке структуры таблиц базы данных и согласовании ее с заказчиком. Вторая категория исполнителей, работающих с базами данных, — пользователи. Они получают исходную базу данных от проектировщиков и занимаются ее наполнением и обслуживанием.

Соответственно СУБД имеет два режима работы: *проектировочный* и *пользовательский*. Первый режим предназначен для создания или изменения структуры базы и ее объектов. Во втором режиме происходит использование ранее подготовленных объектов для наполнения базы или получения данных из нее.

Выбор СУБД для практического приложения пользователем определяется многими факторами, к которым относятся:

- имеющееся техническое и базовое программное обеспечение, их конфигурация, оперативная и дисковая память;
- потребности разрабатываемых приложений пользователя;
- тип поддерживаемой модели данных, специфика предметной области и т.п.;
- требования к производительности при обработке данных;
- наличие в СУБД необходимых функциональных средств;
- наличие русифицированной версии СУБД;
- уровень квалификации пользователей и наличие в СУБД диалоговых средств разработки и работы с базой данных.



## ПРИМЕР ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ

Процесс создания базы данных можно представить в виде следующих основных этапов.

1. Определение цели создания базы данных. На первом этапе проектирования базы данных необходимо определить назначение базы данных, как она будет использоваться и какие сведения она должны содержать. Дается словесная и документальная характеристика данной предметной области. Зная это, можно определить, какие сведения будут храниться в таблицах и в полях таблиц. База данных должна отвечать требованиям конечного пользователя. Для этого необходимо определить темы, которые должна раскрывать база данных, отчеты, которые она должна выдавать.

2. Определение таблиц, которые должна содержать база данных. Определение необходимых в базе данных таблиц может оказаться самым непростым этапом процесса проектирования базы данных, поскольку результаты, которые должна выдавать база данных: отчеты, формы и т.п. — не всегда дают полное представление о структуре таблиц, по которым они создаются. Для проектирования таблиц вовсе не обязательно использовать СУБД. Разрабатываемая модель предметной области обычно представляется в виде графической схемы, начерченной на бумаге.

При разработке таблиц рекомендуется руководствоваться следующими основными принципами:

- сведения не должны дублироваться в таблице или между таблицами;
- данные, хранящиеся только в одной таблице, обновляются только в этой таблице. Это исключает возможность дублирования записей, содержащих разные сведения. Например, адрес и номер телефона каждого ученика достаточно сохранить один раз, в одной таблице;
- каждая таблица должна содержать информацию только на одну тему. Когда таблица содержит сведения только по одной теме, со сведениями по каждой теме можно работать независимо от

остальных тем. Например, адрес читателя библиотеки хранится отдельно от списка книг, взятых этим читателем, что позволяет удалить список книг, сохранив сведения о читателе.

3. Определение необходимых в таблице полей. Каждая таблица содержит сведения по конкретной теме, а каждое поле в таблице содержит конкретный факт по теме таблицы. Например, таблица сведений об ученике может содержать поля сведений об имени, отчестве, фамилии, дате рождения, адресе, номере телефона. При составлении схемы полей для каждой таблицы необходимо учитывать следующее:

- каждое поле должно быть связано с темой таблицы;
- не рекомендуется включать в таблицу данные, которые являются результатом выражения;
- данные следует разбить на наименьшие логические единицы (например, поля Имя и Фамилия, а не общее поле Имя)

4. Определение полей с уникальными значениями в каждой записи. Для связывания сведений, хранящихся в разных таблицах, например для связывания данных о читателе со всеми книгами, которые он брал, каждая таблица базы данных должна содержать поля или набор полей, однозначно определяющих каждую запись. Такое поле или набор полей называют первичным ключом.

5. Определение связей между таблицами. После разбиения сведений на таблицы и определения ключевых полей необходимо выбрать способ, которым СУБД будет объединять связанные сведения. Для этого необходимо определить связи между таблицами базы данных.

6. Усовершенствование структуры базы данных. После создания нужных таблиц, полей и связей необходимо еще раз просмотреть структуру базы данных и выявить возможные недочеты. Желательно это сделать до заполнения таблиц данными.

7. Ввод данных и создание других объектов базы данных. Если структуры таблиц отвечают поставленным требованиям, то можно вводить все данные. Затем можно создать все необходимые запросы, формы, отчеты.

## Основные принципы работы OpenOffice.org Base

Программа OpenOffice.org Base является простой системой по внесению изменений, редактированию и обработке данных. База данных OpenOffice.org Base позволяет вставлять данные из внешних ресурсов. Источником может быть Adabas, JDBC, ADO, dBase, Текстовый файл, Документ электронной таблицы или Данные из адресной книги. Кроме того, можно использовать базу данных OpenOffice.org для подключения к внешним реляционным базам данных, например к базам данных MySQL или Oracle.

Рассмотрим далее, как можно использовать простые функции программы в повседневной работе. Для освоения навыков работы в программе еще раз вспомним структуру построения любой базы. Как правило, данные в базе хранятся в виде таблиц. Но если в обычном табличном редакторе элементом таблицы является ячейка, то в таблице базы основой является поле.

**Поле** — это элемент таблицы, который содержит данные определенного рода, например фамилию ученика. В режиме таблицы для представления поля используется столбец или ячейка, в этом случае имя поля является заголовком столбца таблицы.

**Запись** — полный набор данных об определенном объекте. В режиме таблицы запись изображается как строка.

База данных может состоять из нескольких таблиц, связанных между собой по различным полям. Имеется возможность выбирать и анализировать данные по различным параметрам.

В нашем пособии рассматривается СУБД **OpenOffice.org Base**, входящая в состав пакета **OpenOffice.org 2.3**, включенного в Пакет свободного программного обеспечения для общеобразовательных учреждений Российской Федерации. Внешний интерфейс **OpenOffice.org Base** наследует общие принципы пакета **OpenOffice.org**, при этом возможны незначительные отличия в зависимости от версии **OpenOffice.org**.

Главное меню содержит команды **Файл, Правка, Вид, Вставка, Сервис, Окно, Справка**. Рассмотрим некоторые команды меню.

Пункт меню **Файл**.

**Сохранить.** Сохраняет текущий файл базы данных, запрос, форму или отчет. Для файла базы данных открывается диалоговое окно сохранения файла. Для прочих объектов открывается диалоговое окно Сохранить.

**Сохранить как.** Сохраняет текущий файл базы данных под другим именем. В диалоговом окне сохранения файла выберите путь и имя файла для сохранения.

**Экспорт.** Экспортирует выбранный отчет или форму в текстовый документ. Динамический отчет экспортируется как копия содержимого базы данных на момент экспорта.

**Отправить.** Открывает подменю. **Документ как сообщение электронной почты** — открывает заданное по умолчанию почтовое приложение для отправки нового сообщения. Текущий файл базы данных добавляется как вложение. Можно ввести тему, получателей и текст письма. **Отчет по электронной почте** — открывает заданное по умолчанию почтовое приложение для отправки нового сообщения. Выбранный отчет добавляется как вложение. Можно ввести тему, получателей и текст письма. Динамический отчет экспортируется как копия содержимого базы данных на момент экспорта. **Отчет в текстовый документ** — экспортирует выбранный отчет в текстовый документ. Динамический отчет экспортируется как копия содержимого базы данных на момент экспорта.

Пункт меню **Правка.**

**Копировать** — копирует выделенный объект в буфер обмена.

**Вставить** — вставляет объект из буфера обмена. При необходимости можно вставлять формы и отчеты, включая вложенные папки, из одной базы данных в другую.

**Вставить как** — вставляет объект из буфера обмена. При необходимости можно вставлять формы и отчеты, включая вложенные папки, из одной базы данных в другую.

**Изменить** — открывает окно, где можно редактировать выбранную таблицу, запрос, форму или отчет.

**Удалить** — удаляет выбранную таблицу, запрос, форму или отчет.

**Переименовать** — переименование выбранного объекта. В зависимости от базы данных некоторые имена, символы или длина имени могут быть недопустимыми.

**Открыть** — открывает выбранный объект в том виде, в котором он сохранялся последний раз.

**Создать как представление** — преобразует выбранный запрос в представление. Исходный запрос остается в файле базы данных, а на сервере создается дополнительное представление. Для добавления представления к базе данных необходимо иметь разрешение на запись.

**Мастер форм** — запускает мастер форм для выбранной таблицы, запроса или представления.

**Мастер отчетов** — запускает мастер отчетов для выбранной таблицы, запроса или представления.

**Выделить все** — выделяет все записи, включая вложенные папки, в нижней части окна базы данных.

**База данных** — открывает подменю, содержащие команды **Свойства, Тип подключения, Дополнительные свойства**.

Пункт меню **Вид** содержит такие команды, как:

**Объекты базы данных** — открывает подменю **Формы** (выбирает контейнер для форм и отображает все формы в подробном представлении), **Отчеты** (выбирает контейнер для отчетов и отображает все отчеты в подробном представлении), **Запросы** (выбирает контейнер для запросов и отображает все запросы в подробном представлении), **Таблицы** (выбирает контейнер для таблиц и отображает все таблицы в подробном представлении).

**Сортировка** — открывает подменю, позволяющее сортировать **По возрастанию, По убыванию**.

**Предварительный просмотр** — открывается набор подменю, позволяющий отключать предварительный просмотр в окне базы данных, отображать информацию о документе формы или отчета и др.

**Обновить таблицы** — обновляет таблицы.

Пункт меню **Вставка**.

**Форма** — открывает новый текстовый документ в режиме формы.

**Мастер отчетов** — запускает мастер отчетов для выбранной таблицы, представления или запроса.

**Запрос (конструктор)** — открывает новый запрос в режиме конструктора.

**Запрос (режим SQL)** — открывает новый запрос в режиме SQL.

**Конструктор таблиц** — открывает конструктор таблиц.

**Конструктор представлений** — открывает новое представление в режиме конструктора.

**Представление (простое)** — открывает новое представление в режиме SQL.

**Папка** — открывает диалоговое окно, где можно сохранить новую папку в файл базы данных.

Пункт меню **Сервис**.

**Связи** — открывает Конструктор связей и проверяет, поддерживаются ли связи подключением базы данных.

**Управление пользователями** — открывает диалоговое окно **Управление пользователями**, если база данных поддерживает эту функцию.

**Фильтр таблиц** — открывает диалоговое окно **Фильтр таблиц**, где можно выбрать, какие из таблиц базы данных следует показать, а какие — скрыть. В списке **Фильтр** выберите таблицы, которые требуется отфильтровать. Если выбрать самую верхнюю таблицу в иерархии, будут выбраны все таблицы этой иерархии. Если выбрать в иерархии таблицу более низкого уровня, вышестоящие таблицы этой иерархии не выбираются.

**SQL** — открывает диалоговое окно "SQL", где можно вводить инструкции SQL.

### **Создание базы данных в OpenOffice.org Base**

Чтобы создать новый файл базы данных, можно пойти двумя путями. Если уже открыт какой-либо документ OpenOffice.org, то достаточно выбрать команду меню **Файл — Создать — База данных**. Второй путь предполагает запуск программы **OpenOffice.org Base** через меню графической оболочки **KDE, Gnome**. Как правило, для этого необходимо открыть главное меню, затем **Офис — «OpenOffice.org Base»**. Во время работы часто возникают вопросы по выполнению тех или иных задач. **OpenOffice.org Base** имеет хорошую

справочную систему, которая поможет ответить практически на любой вопрос. Для вызова справки выберете пункт **Вызов справки**.

Создание новой базы данных происходит с помощью мастера и состоит из нескольких этапов. На первом этапе в диалоговом окне предлагается выбрать, создавать новую базу данных или открыть уже существующую (рис. 1).

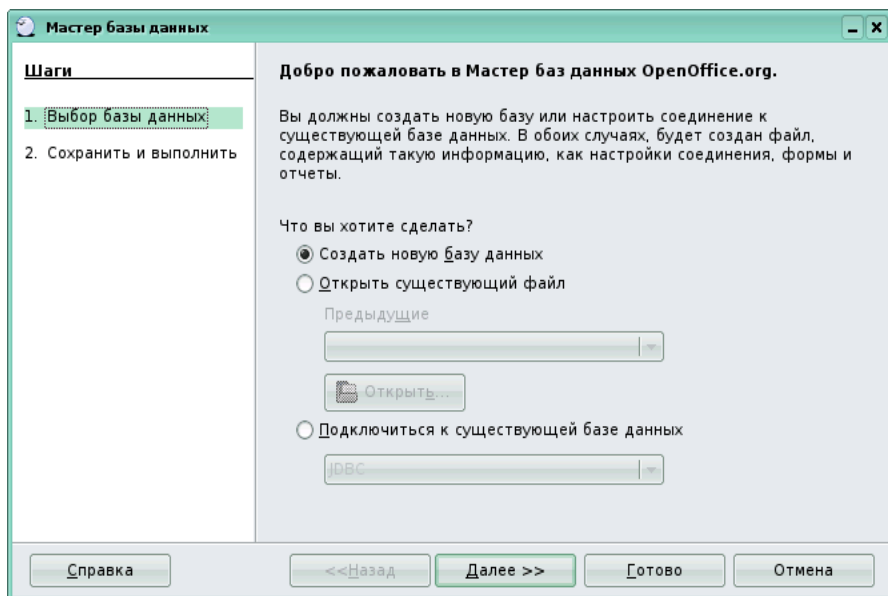


Рис. 1

Чтобы создать новую, необходимо выбрать **Создать новую базу данных** и нажать на кнопку **Далее**.

На втором этапе мастер предлагает два варианта: открыть базу данных для редактирования или сразу приступить к созданию таблиц (рис. 2). После выбора варианта необходимо нажать на кнопку **Готово**.

Чтобы открыть файл базы данных в процессе работы, выберите команду меню **Файл — Открыть**. Из списка **Тип файла** выберите для просмотра только **Документы базы данных**. Выберите документ базы данных и щелкните на кнопку **Открыть**.

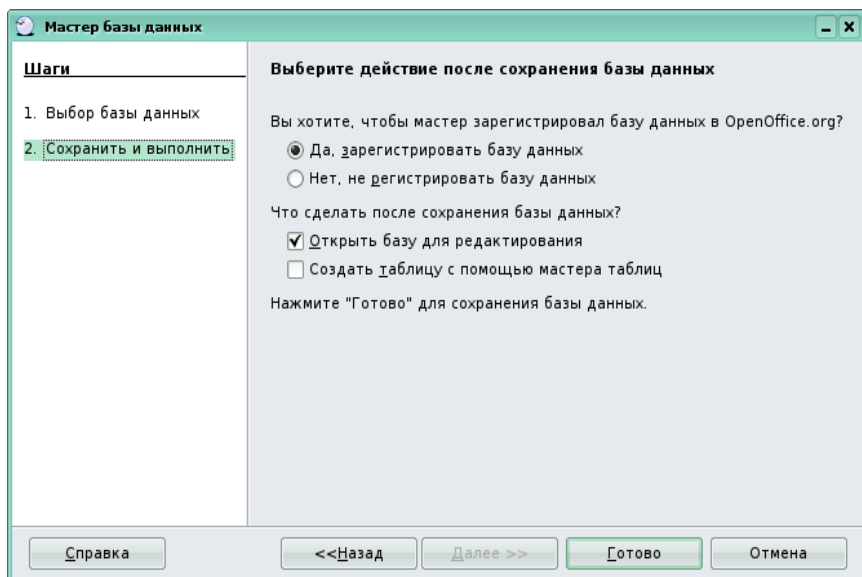


Рис. 2

## Открытие базы данных

Создав базу данных (с нуля, на основе уже существующей базы данных), вы будете периодически открывать ее для внесения изменений или проведения различных операций. Для того чтобы открыть базу данных, выполните следующие действия.

1. Щелкните на кнопке **Открыть** на панели инструментов, и на экране появится предназначенное для этого диалоговое окно (рис. 3).

2. Укажите необходимую папку в списке **Папка** и щелкните на имени файла базы данных, имеющего расширение \* .odb.

3. Щелкните по кнопке **ОК** (рис. 4).

После этого вы можете приступить к формированию таблиц базы данных.



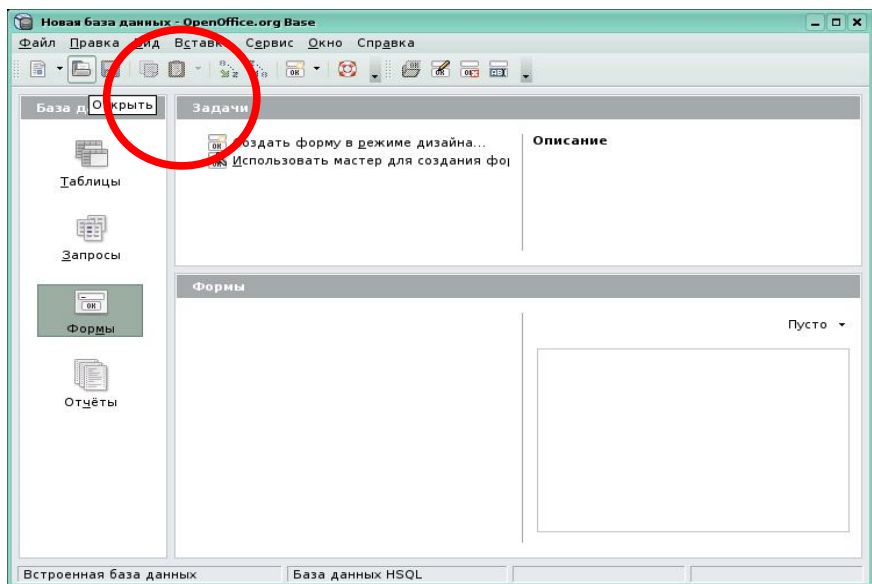


Рис. 3

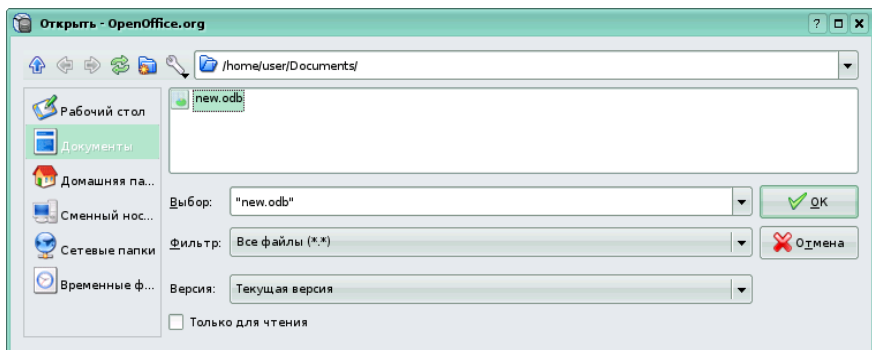


Рис. 4

### Создание новой таблицы

После того как создан новый файл базы данных, редактор откроет основное рабочее окно базы (рис. 5). Здесь можно работать с таблицами, запросами, формами и пр. Создавать, редактировать, управлять ими. Причем практически каждую форму можно создавать в двух режимах — с помощью мастера (для начинающих) или с помощью режима дизайна (для опытных пользователей).

Чтобы воспользоваться помощью мастера таблиц, следует щелкнуть на кнопке **Таблицы** в окне базы данных. В программе OpenOffice.org Base предусмотрено три различных способа создания таблицы базы данных: **Создать таблицу в режиме дизайна** (вся работа по определению структуры таблицы и ее созданию ложится на разработчика), **Использовать мастер для создания таблицы** (позволяет частично автоматизировать процесс создания таблицы), **Создать представление**.

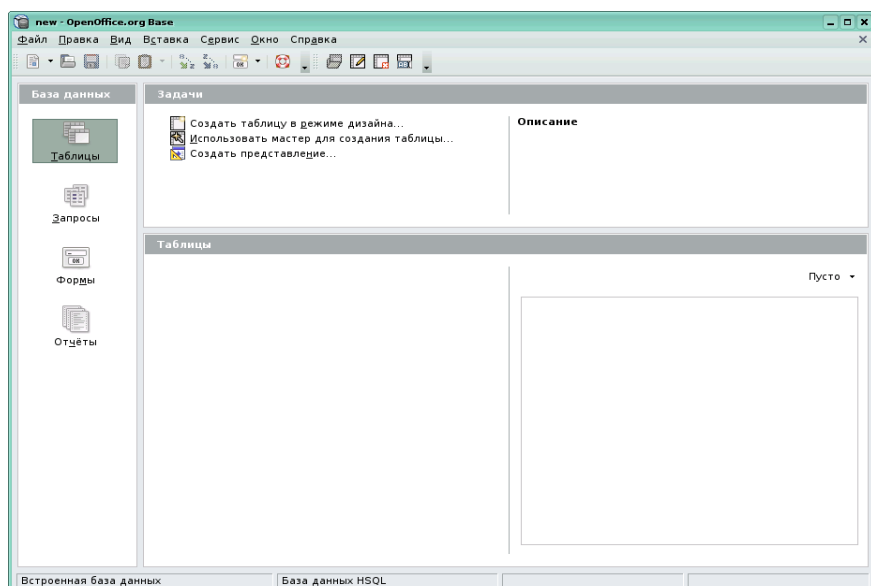


Рис. 5

Рассмотрим создание таблицы с помощью мастера. Для создания таблицы откройте файл базы данных, в которой требуется создать новую таблицу. Щелкните значок **Таблицы**, а справа щелчком левой кнопки мыши выберите **Использовать мастер для создания таблиц...** (рис. 6).

Мастер создания таблиц предлагает множество уже готовых шаблонов. Остается выбрать подходящий и использовать необходимые поля. Создание таблицы начинается с определения ее категории с помощью переключателей **Деловой**, **Персональный**. Затем в списке

**Примеры таблиц** следует выбрать название наиболее подходящего образца. После щелчка по названию выбранного образца формируется соответствующее содержимое списка **Переменные поля**.

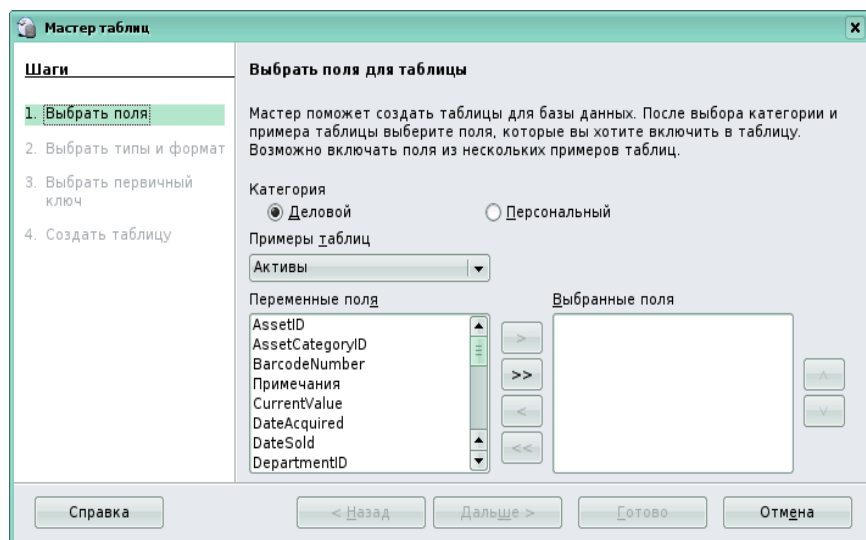



Рис. 6

Следует просмотреть этот список и отобрать те поля, которые предусмотрены в проекте таблицы. Чтобы поле попало в структуру формируемой таблицы, его нужно выделить, а затем щелкнуть по кнопке .

Поле, случайно попавшее в список, можно исключить из него. При желании любое поле, включенное в список выбранных полей, можно переименовать. Это можно сделать в окне следующего шага (рис. 7).

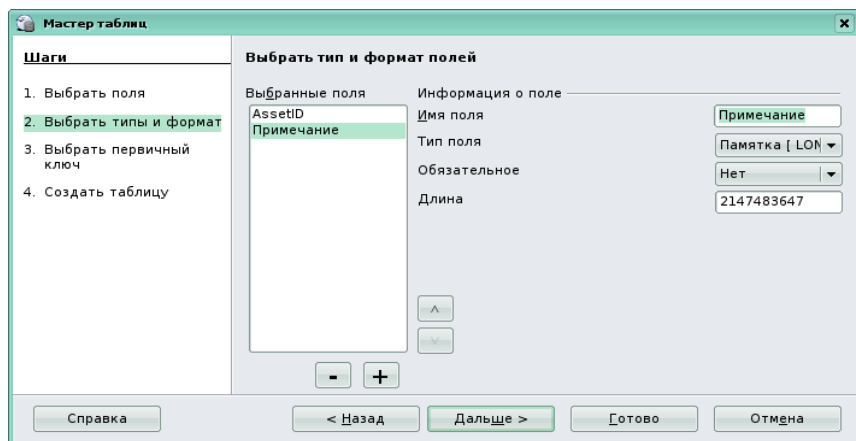


Рис. 7

На третьем шаге происходит задание первичного ключа. В последнем окне мастер предлагает три варианта дальнейших действий: **Немедленно вставить данные, Модифицировать дизайн таблицы, Создать форму на основе данной таблицы.** Следует сделать выбор и щелкнуть на кнопке **Готово**. В случае выбора первого варианта происходит переключение в режим таблицы, в котором можно выполнить первичное заполнение базы. В случае выбора второго варианта происходит переключение в режим дизайна, в котором можно произвольным образом изменить структуру созданной таблицы. При выборе третьего варианта мастер таблиц автоматически создаст форму, обеспечивающую более удобный стиль заполнения базы.

Далее рассмотрим процесс создания новой таблицы в режиме дизайна. Чтобы создать новую таблицу в режиме дизайна, выберите **Создание таблицы в режиме дизайна**.

Теперь можно создать поля для таблицы в режиме дизайна. Создадим таблицу «Ученики», содержащую информацию об учениках. Наша таблица будет содержать такие поля: порядковый номер, имя, отчество, фамилия, дата рождения, возраст, класс, параллель, улица, дом, квартира, классный руководитель.

Введем новые поля в строки сверху вниз. Щелкните ячейку **Имя поля** и введите имя для каждого поля данных (рис. 8).

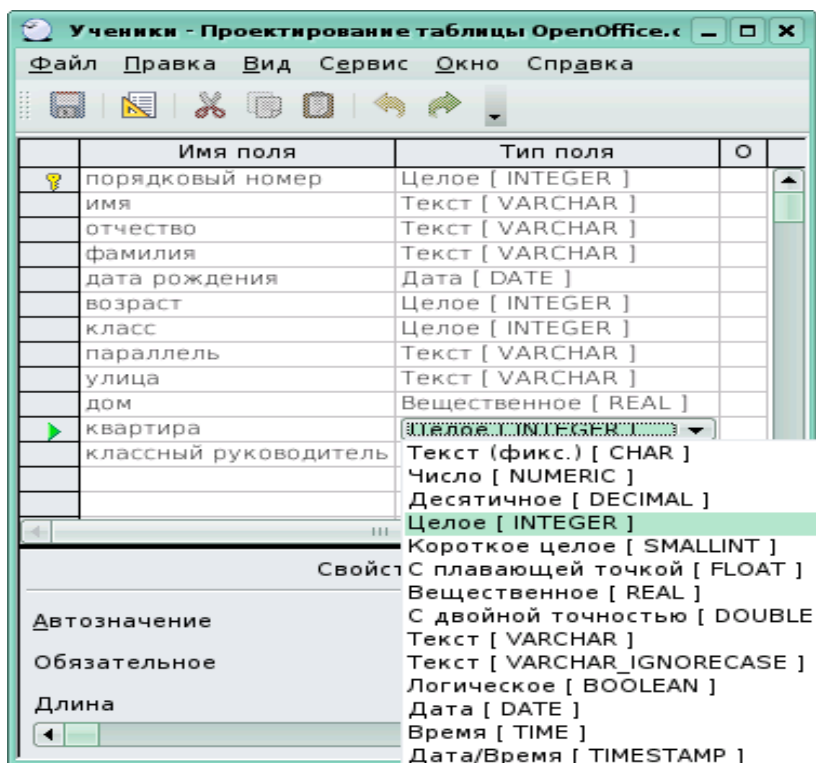


Рис. 8

В следующей ячейке справа определите **Тип поля**. Щелкнув в этой ячейке, выберите тип поля из списка (см. рис. 8).

В нашем примере поля будут иметь следующий тип:

порядковый номер — целое,

- имя — текст,
- отчество — текст,
- фамилия — текст,
- дата рождения — дата,
- возраст — целое,
- класс — целое,
- параллель — текст,

- улица — текст,
- дом — вещественное,
- квартира — целое,
- классный руководитель – целое (номер).

В каждое поле можно вводить данные, соответствующие заданному типу. Например, нельзя ввести текст в числовое поле. Поля записок в формате dBase III являются ссылками на внутренне управляемые текстовые файлы, которые могут содержать до 64 Кбайт текста.

Для каждого поля можно ввести **Описание**. Текст описания будет возникать в виде всплывающей подсказки при наведении курсора на заголовок в представлении таблицы.

Ниже вводятся **Свойства поля** для каждого выбранного поля данных (рис. 9).

В зависимости от типа базы данных некоторые возможности ввода могут быть недоступны.

В поле **Значение по умолчанию** введите содержимое по умолчанию для каждой новой записи. Это содержимое потом можно будет изменить.

В поле **Обязательное** укажите, может ли поле оставаться пустым.

В поле **Длина** можно открыть список с вариантами выбора. После того как все поля будут заполнены, откройте таблицу для заполнения, для этого в окне базы данных щелкните два раза левой кнопкой мыши на названии таблицы.

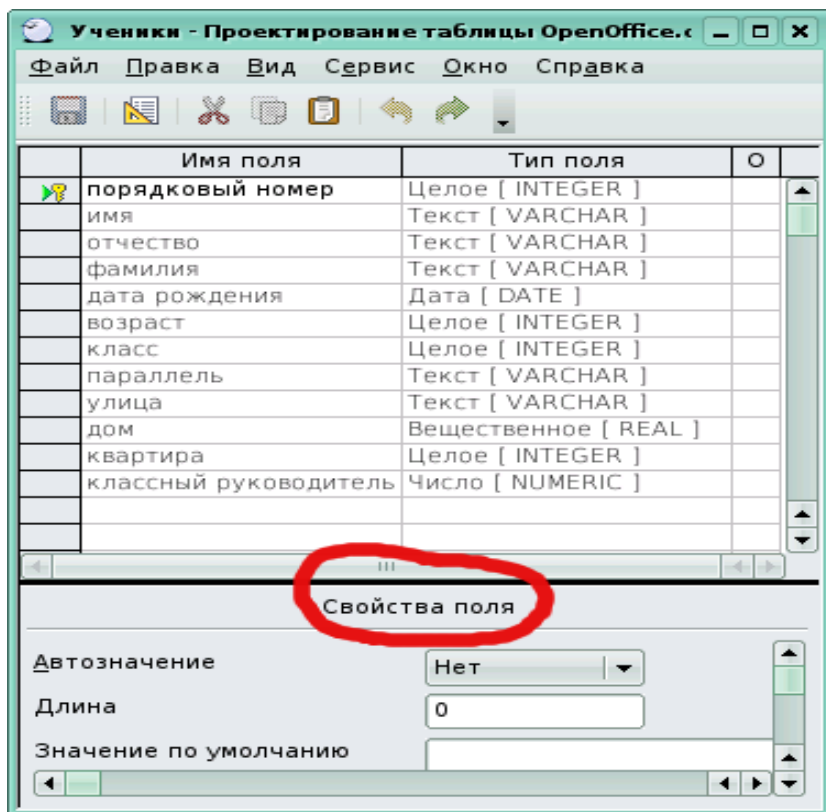


Рис. 9

Заполненная таблица выглядит следующим образом (рис. 10):

порядков...	имя	отчество	фамилия	дата рождения	возраст	класс	парал...	улица	дом	кварт...	классный руковод...
1	Иван	Аркадьевич	Белоногов	02.12.96	12	5	а	Ленина	17	114	1
2	Светлана	Андреевна	Сидорова	15.04.96	12	5	б	Парковая	104	26	2
3	Ирина	Ивановна	Аллаева	21.10.96	12	5	а	Садовая	12	68	1

Рис. 10

Записи в таблице можно отсортировать. Для этого:

1. Выделите тот столбец, по которому будет проводиться сортировка.

2. Выберите на панели инструментов кнопку **"По возрастанию"** или **"По убыванию"**.

В нашей таблице мы отсортировали по алфавиту поле «Фамилия», результат сортировки показан на рис. 11.

порядков...	имя	отчество	фамилия	дата рождения	возраст	класс	парал...	улица	дом	кварт...	классный руковод...
3	Ирина	Ивановна	Аллаева	21.10.96	12	5	а	Садовая	12	68	1
1	Иван	Аркадьевич	Белоногов	02.12.96	12	5	а	Ленина	17	114	1
2	Светлана	Андреевна	Сидорова	15.04.96	12	5	6	Парковая	104	26	2

Рис. 11

## Связи

При существовании связей между именем поля в одной таблице и именем поля в другой таблице эти связи можно использовать в запросе.

Если, к примеру, имеется электронная таблица «Ученики» и электронная таблица, в которой хранится информация о классных руководителях — «Учителя», то существует связь между полями данных. Теперь, чтобы создать запрос о классном руководителе ученика, необходимо получить данные из двух электронных таблиц. Для этого требуется указать в OpenOffice.org Base, какая связь существует между данными этих электронных таблиц.

Чтобы приступить к созданию связей, выберите пункт меню **Сервис** — **Связи** (рис. 12).



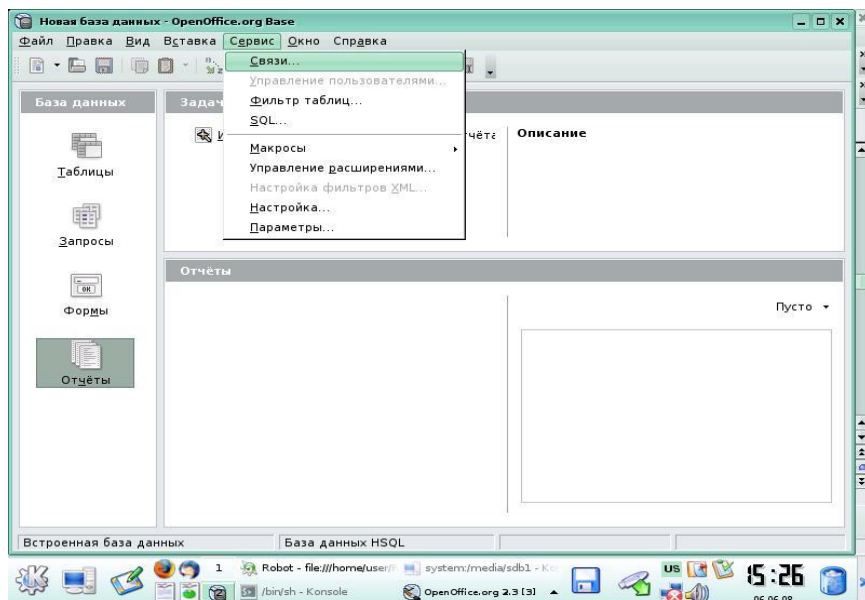


Рис. 12

Следует помнить, что создавать или изменять связи между открытыми таблицами нельзя. Поэтому вначале необходимо закрыть все ранее открытые таблицы базы данных и открыть окно базы данных.

Если в базе еще не определены связи между таблицами, автоматически откроется диалоговое окно **Добавить таблицы** (рис. 13).

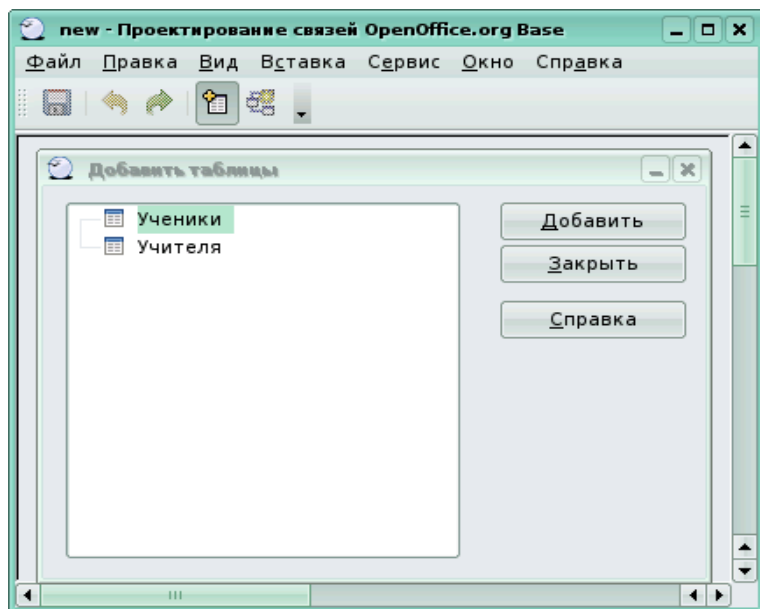


Рис. 13

В открытом окне **Добавить таблицы** следует выполнить двойные щелчки по названиям таблиц, которые участвуют в связи, а затем закрыть его. Кроме того, можно перетащить нужные таблицы, удерживая левую кнопку мыши. В нашем примере для наглядности добавим все таблицы базы данных.

Далее есть два варианта создания связи. Выбираем пункт меню **Вставка — Создать связь**. Появляется модальный диалог (рис. 14).

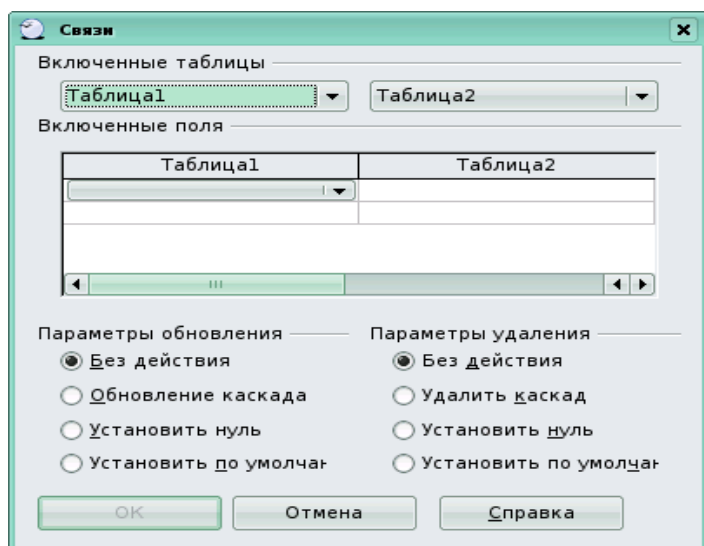


Рис. 14

В нем мы осуществляем все необходимые настройки и в результате получаем отображение всех таблиц и связей между ними (рис. 15).

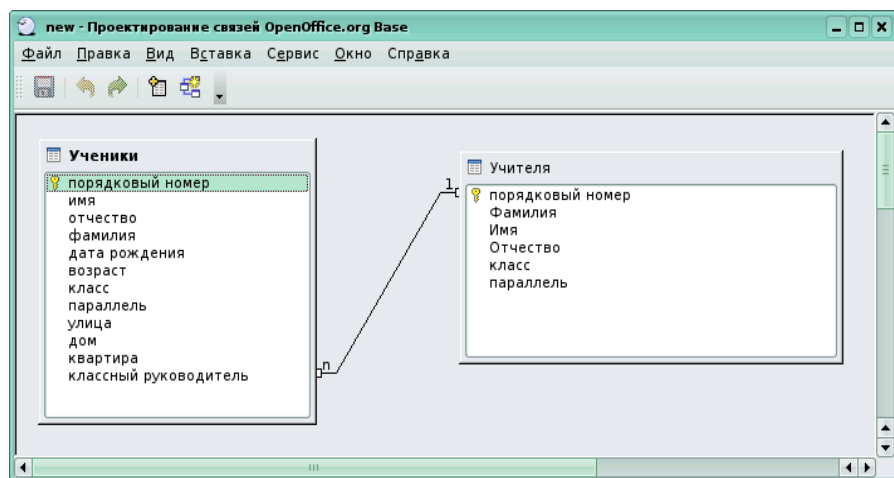


Рис. 15

В диалоговом окне (см. рис. 14) выбираются таблицы, определяются включенные поля, задаются параметры обновления (эти параметры вступают в силу при изменении поля первичного ключа) и удаления (эти параметры вступают в силу при удалении поля первичного ключа). Можно задать следующие параметры обновления:

- **Без действия** — указывается, что любое изменение первичного ключа не должно затрагивать остальные внешние ключевые поля.
- **Обновление каскадом** — происходит обновление всех внешних ключевых полей при изменении соответствующего первичного ключа (обновление каскадом).
- **Установить ноль** — если соответствующий первичный ключ изменен, этот параметр используется для задания значения "ПУСТО" всем внешним ключевым полям. Это значение указывает, что поле пустое.
- **Установить по умолчанию** — если соответствующий первичный ключ изменен, этот параметр используется для задания значения по умолчанию всем внешним ключевым полям. При создании соответствующей таблицы значение по умолчанию внешнего ключевого поля определяется при задании свойств данного поля.

Аналогичный смысл имеют параметры удаления.

Создать связь можно и следующим образом: щелкните имя поля в таблице (например, имя поля "класс"), удерживая кнопку мыши, и перетащите имя поля на имя поля другой таблицы. Если отпустить кнопку мыши, появляется линия, соединяющая эти два поля в двух окнах.

Чтобы удалить связь между двумя таблицами, щелкните соединительную линию и нажмите клавишу **DELETE**, или выделите соединительный вектор с помощью клавиши **TAB**, вызовите контекстное меню, нажав клавиши **SHIFT+F10**, и выберите команду **Удалить**.

Связь между таблицами можно редактировать, для этого можно воспользоваться контекстным меню связи.

## Создание запросов

Если вам часто приходится обращаться только к подмножеству своих данных, которые могут быть хорошо определены условием фильтра, рекомендуется составить запрос. По сути, это просто новое представление отфильтрованных данных. Открыв запрос, вы увидите текущие данные в виде таблицы, которую вы определили.

Создавать запросы можно в режиме **Мастер запросов** или в режиме **Дизайн запросов**. Для создания запроса в режиме **Мастер запросов** необходимо:

Открыть файл базы данных, в котором нужно создать новый запрос.

В левой области окна базы данных щелкнуть значок **Запросы**.

Установить флажок **Использовать мастер** для создания запросов.

Чтобы создать запрос в режиме **Дизайн запросов**, щелкните значок **Запросы** в документе базы данных, а затем щелкните **Создать запрос в режиме дизайна**.

Определение запроса выполняется в нижней области окна. Чтобы определить запрос, укажите имена полей базы данных, которые требуется включить, а также условия отображения полей. Чтобы переупорядочить столбцы в нижней области конструктора, перетащите заголовок столбца в новое расположение или выберите столбец и нажмите **CTRL** и клавишу со стрелкой. Создадим запрос, который будет выводить фамилии учеников старше 10 лет (рис. 16).

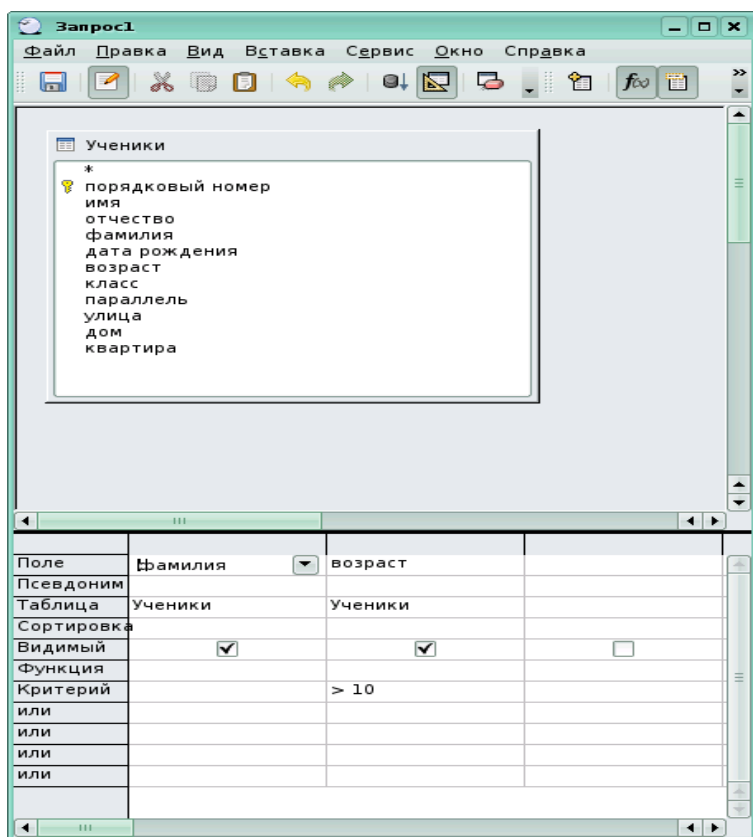


Рис. 16

Для формирования запроса выбираем поля **Фамилия** и **Возраст**. Указываем, что это поля видимые. Для поля **Возраст** вводим критерий выбора «>10». После этого закрываем окно запроса. При закрытии окна редактор предложит сохранить запрос, сохраняем его, указав имя запроса.

Чтобы проверить запрос, дважды щелкните его имя в документе базы данных.

Результат запроса отобразится в таблице, аналогичной представлению источника данных, также результаты запроса можно посмотреть не выходя из режима дизайна, для этого нужно выбрать

пункт меню **Вид** — **Вид** или просто нажать клавишу **F4**. В нашем примере получились следующие результаты запроса (рис. 17):



The screenshot shows a window titled 'new: Запрос1' with a menu bar (Файл, Правка, Вид, Сервис, Окно, Справка) and a toolbar. Below the toolbar is a table with two columns: 'фамилия' and 'возраст'. The table contains three rows of data. At the bottom of the window, there is a status bar showing 'Запись 1 из 3'.

фамилия	возраст
Белоногов	12
Сидорова	12
Аллаева	12

Запись 1 из 3

Рис. 17

## Создание форм

Рассмотрим создание новой формы с помощью мастера форм. Для создания новой формы в OpenOffice.org Base можно использовать

### Мастер форм:

Откройте файл базы данных, в котором нужно создать новую форму.

В левой области окна базы данных щелкните значок **Формы**.

Выберите **Использовать мастер для создания формы**.

Мастер форм предложит вам следующие шаги:

Шаг 1. Выбор поля (рис. 18).

Выбираем таблицу или запрос, по которым мы будем создавать форму. В нашем примере для таблицы «Ученики» мы выбрали поля Фамилия и Класс.

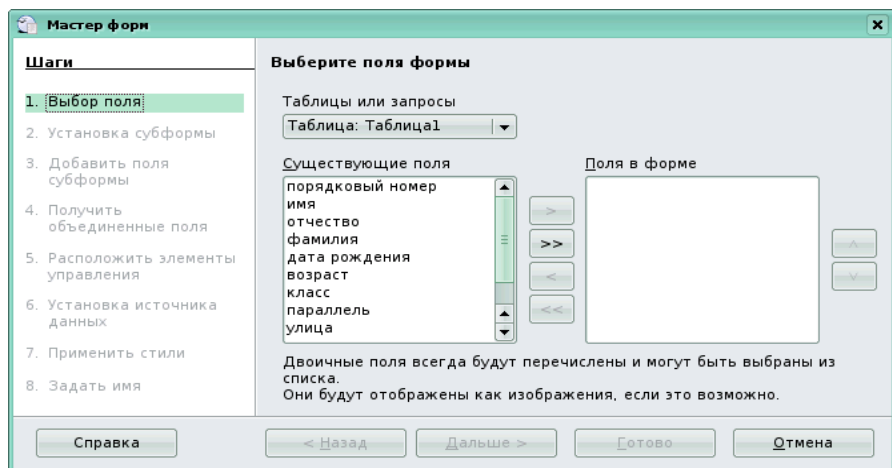


Рис. 18

## Шаг 2. Установка субформы.

Эта форма является вложенной в другую форму. Используется для отображения данных из таблиц или запросов типа «один ко многим». Если вы выбрали **Добавить субформу**, вам нужно выполнить два дополнительных шага, в которых вы выбираете необходимые вам пункты, аналогично шагу 1. В нашем случае субформа не нужна, и мы переходим сразу к шагу 5.

## Шаг 5. Расположить элементы управления.

На этом шаге выберите, каким образом будут расположены элементы на форме (рис. 19).



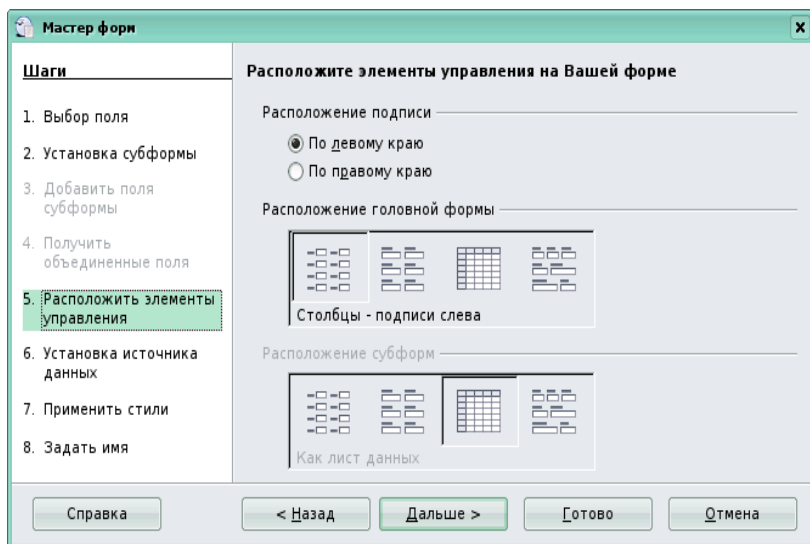


Рис. 19

Шаг 6. Установка источника данных.

На данном этапе производится выбор режима источника данных.

В зависимости от ваших потребностей форма может использоваться для отображения всех данных или только для ввода новых данных, можно запретить изменение существующих данных, добавление и удаление данных.

Мы выбираем вариант «только для ввода новых данных».

Шаг 7. Применить стили.

Выберите цвет формы и вид ее элементов.

Шаг 8. Задать имя.

Задайте имя формы и выберите действие после заполнения формы. В результате получаем следующую форму (рис. 20).

Возможно также создание новой формы вручную. Для этого необходимо:

Открыть файл базы данных, в котором нужно создать новую форму.

В левой области окна базы данных щелкнуть значок **Формы**.

Щелкнуть **Создание формы в режиме конструктора**.

Откроется новый текстовый документ. Чтобы вставить в форму элементы управления, используйте **Элементы управления форм**.

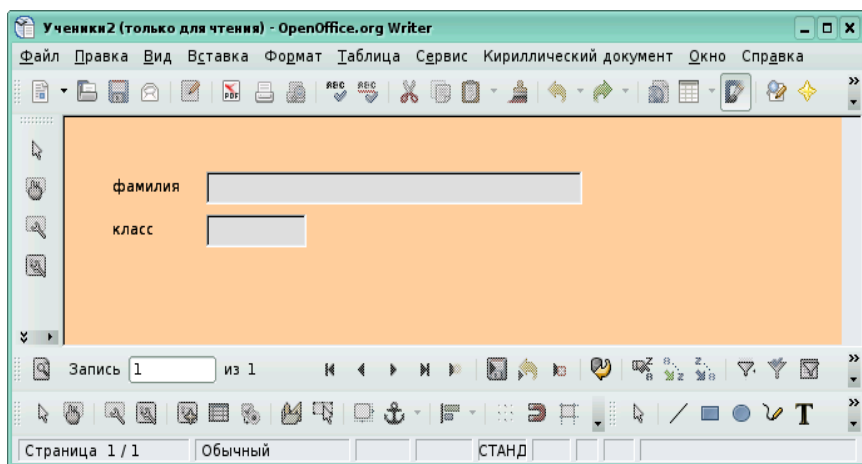


Рис. 20

### Создание отчетов

Отчет является эффективным средством наглядного представления данных в печатном документе. Отчет — это текстовый документ, отображающий текущие данные или данные на время создания отчета. Разработчик имеет возможность эффективно управлять размером, внешним видом и структурой всех компонентов отчета.

Большинство отчетов связаны с одной или несколькими таблицами и/или запросами базы данных, поля которых являются источником формирования для включаемых в отчет записей. Отчет может включать любые комбинации полей из каждой таблицы или запроса, на основе которых он создается.

Для создания нового отчета в OpenOffice.org Base можно использовать **Мастер отчетов**:

Откройте файл базы данных, в которой необходимо создать новый отчет.

В левой области окна базы данных щелкните значок **Отчеты**.

Выберите задачу **Использовать мастер для создания отчета** (рис. 21).

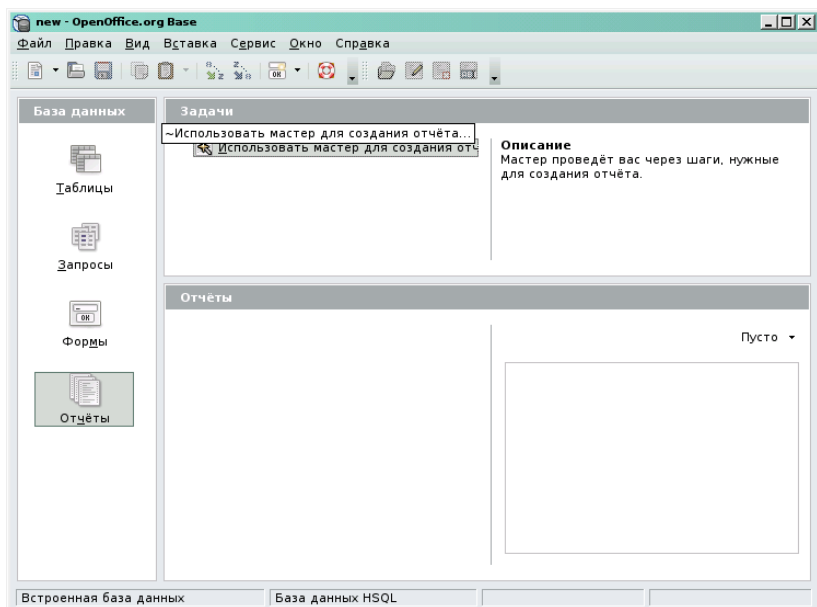


Рис. 21

В результате откроется окно первого шага мастера построения отчетов (рис. 22).

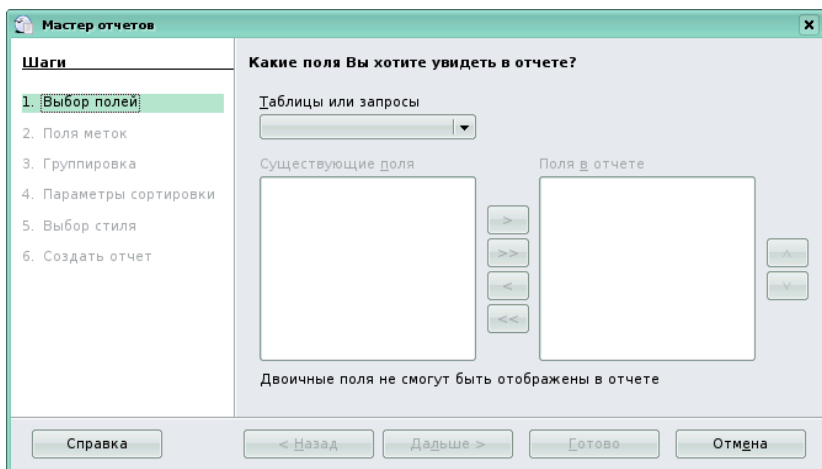


Рис. 22

Описанным ранее способом с помощью списка **Таблицы или запросы** в этом окне следует сформировать список **Поля в отчете** и с помощью кнопки **Далее** перейти к следующему шагу.

На втором шаге **Поля меток** указывается, что будут обозначать поля. В окне третьего шага можно осуществлять группировку записей в отчет. Записи отчета можно группировать на основе значений одного или нескольких полей. Выберите поля, по которым будет сгруппирован итоговый отчет. В отчете можно группировать до четырех полей. При группировке нескольких полей OpenOffice.org Base создает вложенные группы согласно их уровню группировки. В образце отчета отображается получаемая при этом структура отчета, в котором каждый следующий уровень сдвинут вправо по отношению к предыдущему уровню и обозначены поля, отображаемые на каждом уровне.

В окне четвертого шага **Параметры сортировки** мастер предоставляет возможность выбрать поля, по которым требуется сортировать отчет. Поля можно сортировать по четырем уровням, на каждом — по возрастанию или по убыванию. Сгруппированные поля можно сортировать только внутри соответствующих групп.

На следующем шаге **Разметка** выберите разметку из разных шаблонов и стилей и ориентацию страницы (книжную или альбомную).

Отчет можно создать как статический или как динамический отчет. При открытии динамического отчета он отображается с текущим содержимым. При открытии статического отчета всегда отображаются те данные, которые содержались в этом документе в момент его создания.

## ПРИМЕР СОЗДАНИЕ БАЗЫ ДАННЫХ

В заключение рассмотрим пример создания базы данных «Библиотека». На этом примере покажем все стадии создания базы, формирования отчетов и запросов. Рекомендуем читателю повторить рассматриваемый пример или создать свой.

В базе данных «Библиотека» будем хранить следующую информацию: название книги, автор, шифр книги, название издательства, адрес издательства, главный редактор, количество экземпляров книги.

Очевидно, что если поместить всю эту информацию в одну таблицу, мы столкнемся с проблемой избыточных данных. Для того чтобы избежать этого, информация разбивается на несколько таблиц, что позволяет оптимально использовать память, этот процесс называется нормализацией. В нашем случае создадим две таблицы: 1) таблица «Книги», содержащая поля название книги, шифр, автор, название издательства, количество экземпляров, и так как нам необходимо ключевое поле, добавим поле номер книги; 2) таблица «Издательства», содержащая поля название издательства, адрес издательства, главный редактор, в данном случае ключевым полем выберем название издательства. Между этими таблицами будет связь «один ко многим», так как несколько книг могут издаваться в одном издательстве (таблица «Книги»), названия издательств не совпадают (таблица «Издательства»).

Теперь можно приступить к непосредственному созданию базы данных. Открываем приложение **OpenOffice.org Base**. Напомним, что процесс создания новой базы описан в п. 2.3 настоящего пособия. Назовем нашу базу «Библиотека».

После того как база создана, приступим к созданию таблиц. Создадим наши таблицы в режиме дизайна (см. п. 2.5). Первой создадим таблицу «Книги».

В колонку **Имя поля** записываем названия полей в удобной для нас очередности, в колонке **Тип поля** выбираем тип данных, которые будет содержать это поле (рис. 23).

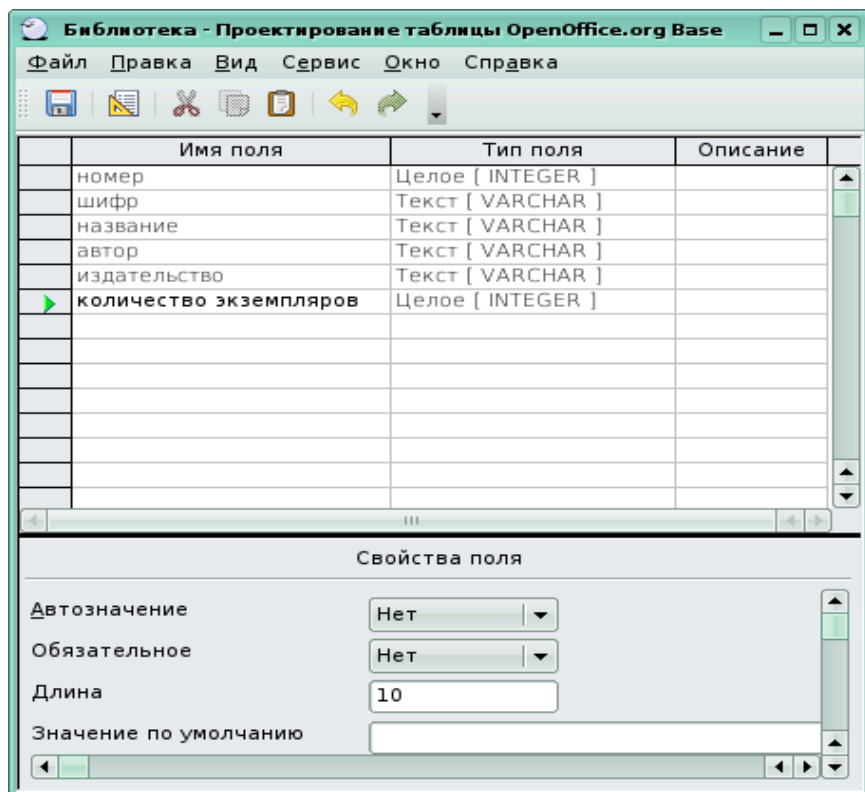


Рис. 23

Номер книги делаем целым числом, шифр — текстовым, так как он может содержать и буквы и цифры; название книги, автор и название издательства тоже текстовые, а количество экземпляров — целое число.

После заполнения всех полей создаем первичный ключ. Напомним, что в данной таблице первичным ключом у нас будет номер. Щелкаем правой кнопкой мыши на прямоугольник слева от поля номер, в выпавшем контекстном меню выбираем пункт **Первичный ключ** (рис. 24).

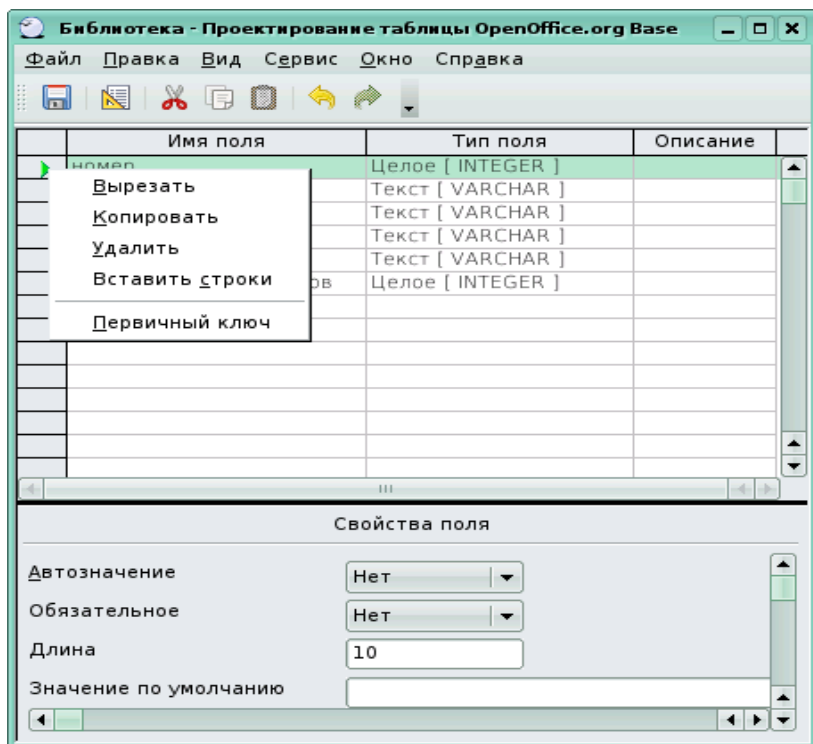


Рис. 24

Сохраняем изменения и закрываем окно редактирования. Аналогичным образом создаем таблицу «Издательства», которая имеет поля — название издательства, адрес издательства, главный редактор, с ключевым полем — название издательства.

Если мы посмотрим в главное окно базы данных, то увидим, что в правом нижнем разделе **Таблицы** появились наши таблицы «Книги», «Издательства» (рис. 25). Теперь заполним обе таблицы. Начнем с таблицы «Книги». Щелкнем на нее два раза левой кнопкой мыши, таблица откроется для заполнения, внесем в нее данные (рис. 26). Аналогичным образом заполним таблицу «Издательства» (рис. 27).

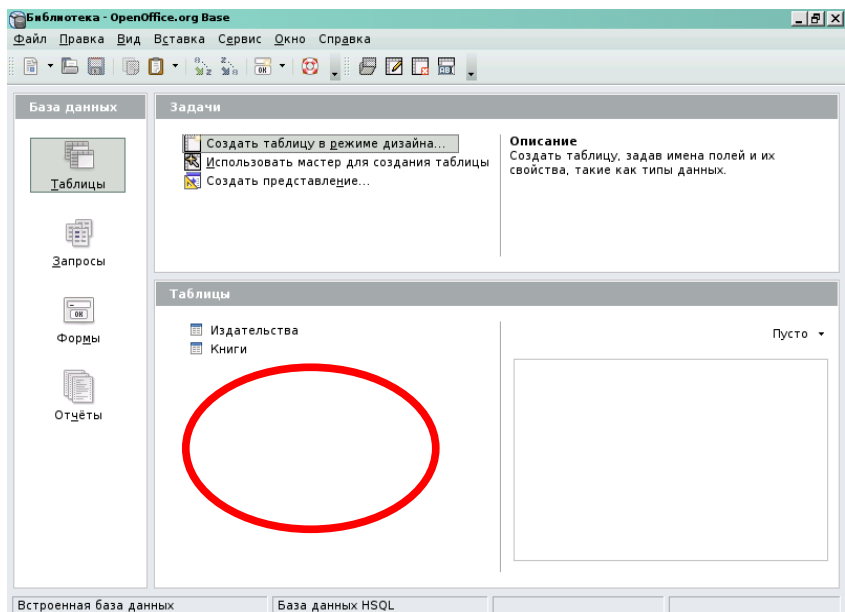


Рис. 25

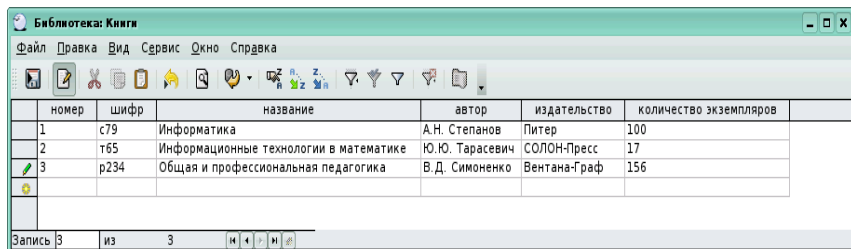


Рис. 26

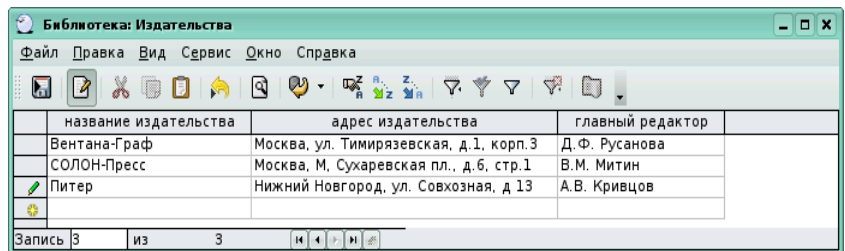


Рис. 27



После того как все таблицы созданы, приступаем к созданию связей. Этот этап подробно рассмотрен в п. 2.6 **Связи**, предлагаем читателю проработать его самостоятельно, результат представлен на рис. 28.

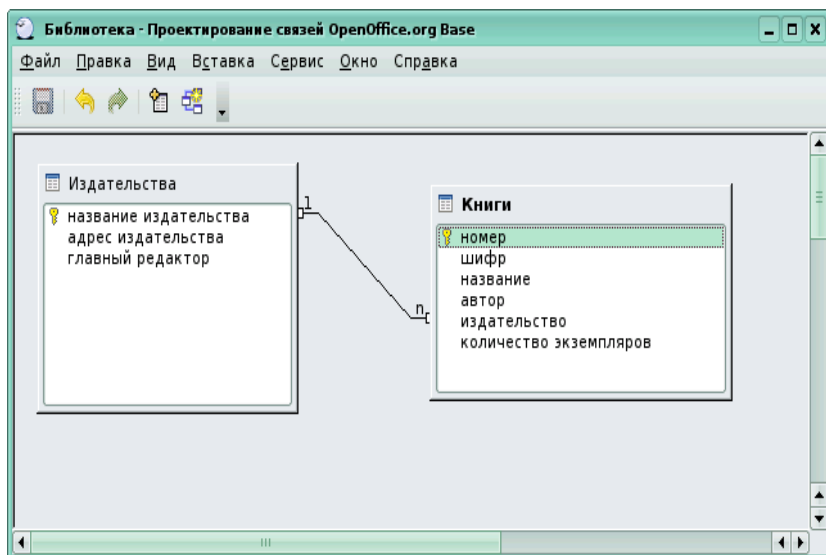


Рис. 28

Теперь приступим к созданию запросов. Для начала создадим запрос сортировки данных по таблице «Книги». Для этого заходим в режим дизайна запросов и добавляем туда таблицу «Книги», после чего выбираем интересующие нас поля и ставим сортировку «по возрастанию» для поля «название» (рис. 29).

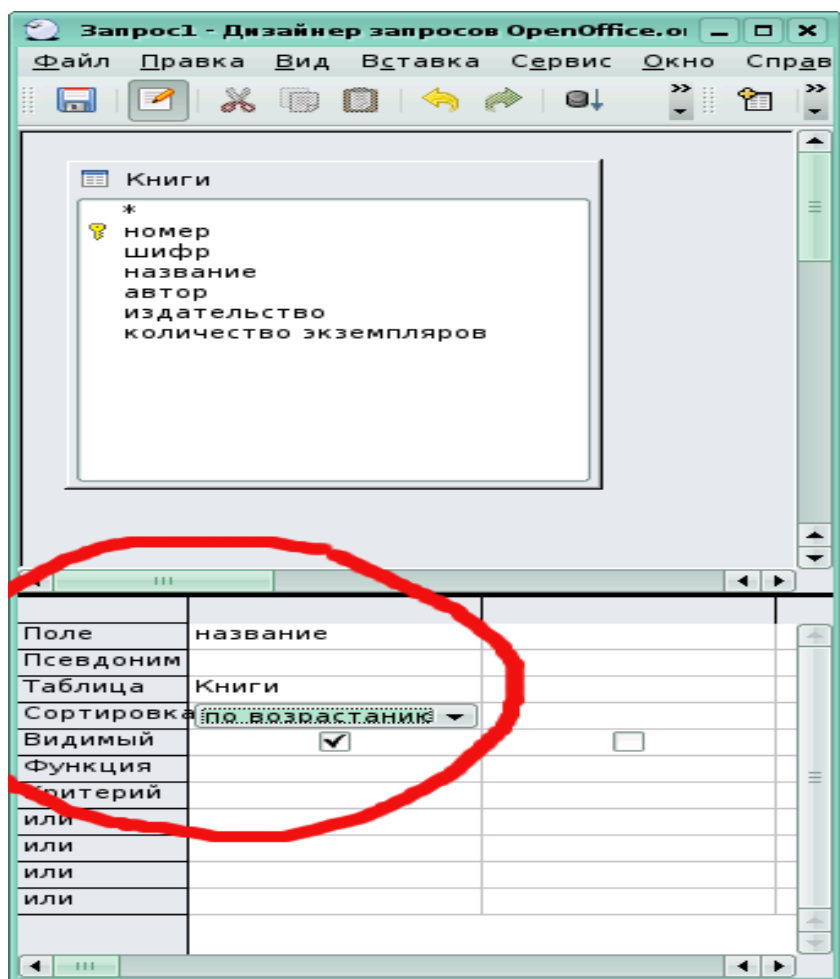


Рис. 29

Результаты данного запроса показаны на рис. 30.

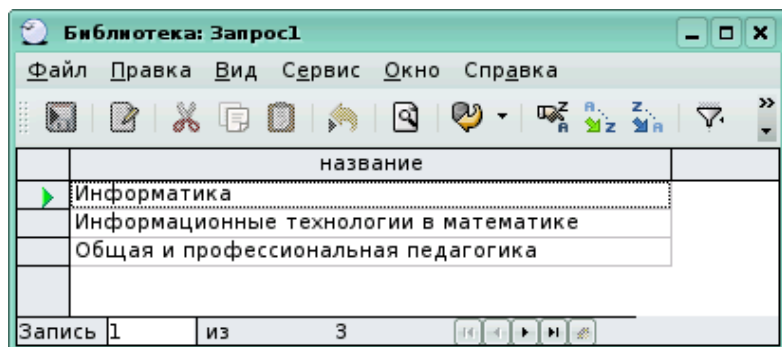


Рис. 30

Для того чтобы знать, сколько различных наименований книг хранится в библиотеке, создадим следующий запрос. В режиме дизайна добавляем таблицу «Книги», для подсчета книг можно воспользоваться полем «номер книги». Выбираем это поле, пишем псевдоним «количество наименований книг» и выбираем функцию «счетчик (Count)» (рис. 31)

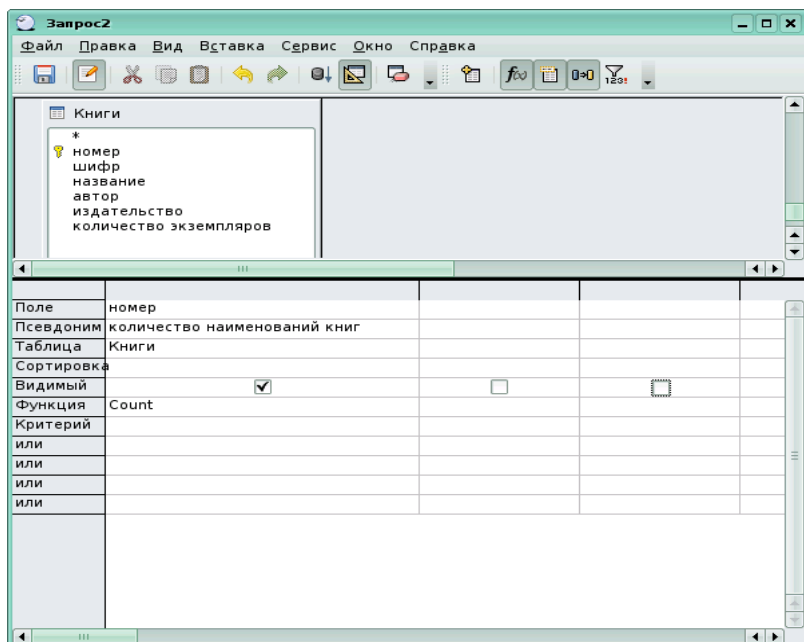


Рис. 31

При этом «Псевдоним» несет смысловую нагрузку. Результат запроса представлен на рис. 32.

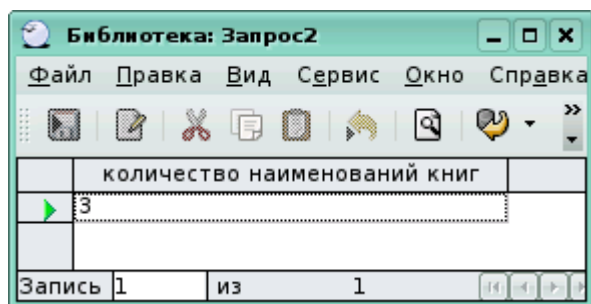


Рис. 32

Далее можно создать форму, формы могут создаваться на основе таблиц или запросов. Мы создадим форму, которая будет отображать все данные о книге с возможностью их добавления.

Открываем **Мастер форм** и приступаем к выполнению шагов.

Шаг 1. Помещаем в форму все поля таблицы «Книги» (рис. 33).

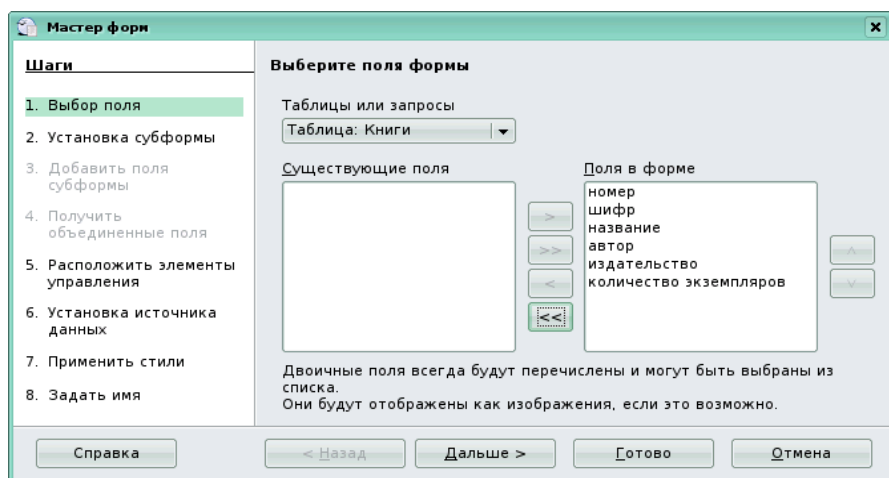


Рис. 33

Шаг 2. Выбираем добавление субформы.

Шаг 3. Выбираем поля для субформы (это будут поля «адрес» и «главный редактор» таблицы «Издательства») (рис. 34).

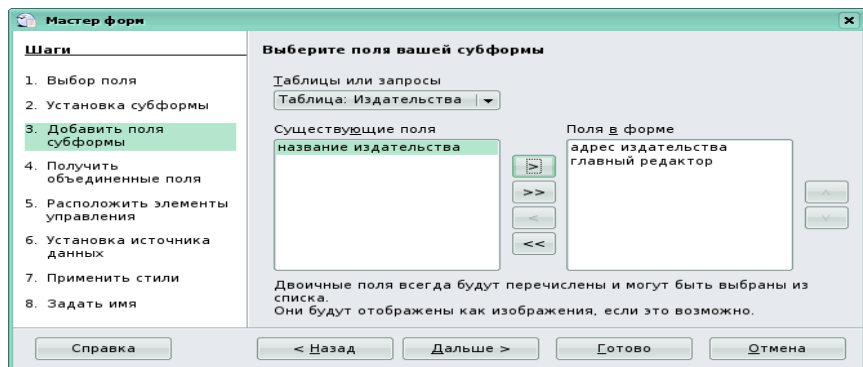


Рис. 34

Шаг 4. На этом шаге не меняем ничего.

Шаг 5. Выбираем расположение элементов на форме. В данном случае столбцы подписи расположим слева и на основной форме и на субформе.

Шаг 6. Здесь задаем условие, что в форме будут отображены все данные, но их нельзя изменять и удалять, можно только добавлять новые данные (рис. 35).

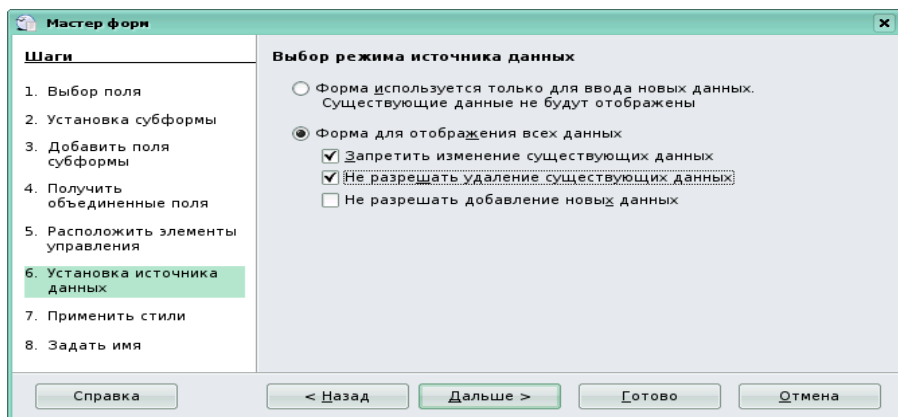


Рис. 35

Шаг 7. Выбираем оформление формы. Мы выбрали светло-синий фон и трехмерный вид поля.

Шаг 8. На заключительном шаге задаем имя формы как «информация о книгах» и отмечаем, что после заполнения формы хотим с ней работать (рис. 36).

The screenshot shows a window titled "Мастер форм" (Form Master) with a close button (X) in the top right corner. The window is divided into two main sections. On the left, under the heading "Шаги" (Steps), is a list of eight steps: 1. Выбор поля (Field selection), 2. Установка субформы (Subform setup), 3. Добавить поля субформы (Add subform fields), 4. Получить объединенные поля (Get combined fields), 5. Расположить элементы управления (Arrange controls), 6. Установка источника данных (Set data source), 7. Применить стили (Apply styles), and 8. Задать имя (Assign name), which is currently highlighted with a green background. On the right, under the heading "Укажите имя формы" (Specify form name), there is a text input field containing "информация о книгах" (information about books). Below this, under the heading "Действия после заполнения формы" (Actions after filling the form), there are two radio button options: "Работа с формой" (Work with form), which is selected, and "Модифицировать форму" (Modify form). At the bottom of the window, there are five buttons: "Справка" (Help), "< Назад" (Back), "Дальше >" (Next), "Готово" (Finish), and "Отмена" (Cancel).

Рис. 36

После завершения последнего шага жмем кнопку «Готово», на этом создание формы закончено (рис. 37).

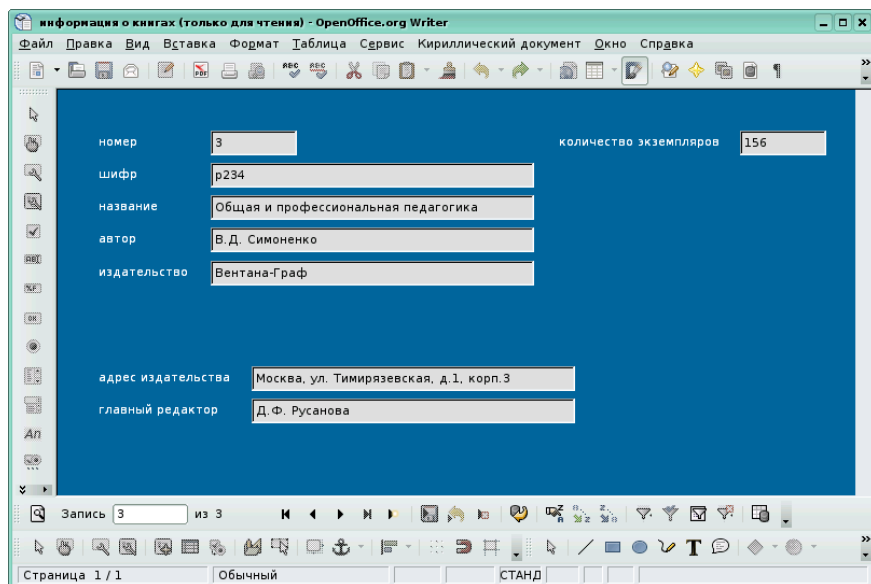


Рис. 37

Форма открывается в виде текстового документа в окне редактора OpenOffice.org Writer. Если читателя заинтересовала работа в этом редакторе, он может обратиться к пособию А.В. Литвиновой «Создание и редактирование текстов в среде OpenOffice.org», разработанному по пакету СПО.

Для создания отчета выбираем **Мастер отчетов**. На рис. 38 приведен пример отчета. Отчеты, построенные на основе таблиц, будут статическими, то есть будут отображать только ту информацию, которая содержалась в таблице во время их создания. Напомним, что есть возможность создать динамический отчет. При этом происходит сохранение отчета в виде шаблона. При открытии динамического отчета он отображается с текущим содержанием.

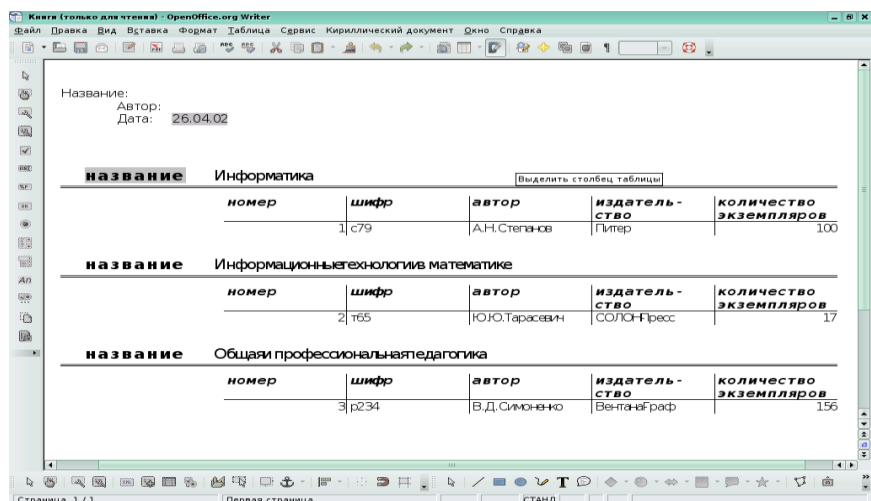


Рис. 38

На этом создание элементарной базы данных «Библиотека» закончено. Естественно, эта база является лишь «пробой пера», но она также является основой, опираясь на которую, можно построить более расширенную и более функциональную базу данных.

Таким образом, мы рассмотрели основные этапы создания базы данных и последующей работы с ней. Для более детального знакомства с процессом проектирования и создания баз данных предлагаем читателю изучить дополнительную литературу из списка.



## **ЗАДАНИЕ НА ЛАБОРАТОРНУЮ РАБОТУ**

Реализовать базу данных в среде Open Office Base в соответствии с вариантом аналогично приведенному примеру.

### **ВАРИАНТЫ ЗАДАНИЙ**

1. Географические объекты. СТРАНЫ (название, площадь, население, столица), ГОРОДА (название, страна, население, географические координаты: широта, долгота), РЕКИ и ОЗЕРА (тип, название, длина, макс. глубина, в каких странах находится).

2. Протоколы нарушения правил дорожного движения. ПРОТОКОЛ (номер, дата, место совершения, нарушитель, инспектор, автомобиль, нарушенные статьи КоАП, объяснение нарушителя, смягчающие и отягчающие обстоятельства), ИНСПЕКТОРА (ФИО, звание, дата рождения), КоАП (статья, часть, пункт, наименование, меры взыскания)

3. Субъекты и города Российской Федерации. СУБЪЕКТ (код региона, название, тип /область, край, автономный округ/, столица, губернатор, население, площадь, с какими субъектами граничит) НАСЕЛЕННЫЙ ПУНКТ (наименование, телефонный код, тип населенного пункта, население, регион, мэр)

4. Автобусные маршруты. МАРШРУТ (номер маршрута, номер пункт и время отправления, пункт и время прибытия, время в пути, промежуточные пункты с указанием времени прибытия и отправления) НАСЕЛЕННЫЕ ПУНКТЫ (наименование, наличие авто- и ж/д вокзала, аэропорта, гостиниц)

5. Регистрация автотранспортных средств. ТРАНСПОРТНОЕ СРЕДСТВО /ТС/ (номер паспорта, производитель, модель, кузов, двигатель, цвет, мощность и др. возможные характеристики ТС) СВИДЕТЕЛЬСТВА О РЕГИСТРАЦИИ /СоР/ (номер свидетельства, ТС, государственный регистрационный номер, владелец /ФИО, адрес/, должностное лицо, выдавшее СоР) УЧЕТ СоР (номер свидетельства, дата выдачи или изъятия)

6. Банковские счета. СЧЕТ (номер, владелец счета, дата открытия, сумма, сотрудник, открывший счет) ОПЕРАЦИИ (дата, счет, сумма, сотрудник, проводивший операцию) СОТРУДНИКИ (ФИО, адрес, телефон, дата приема на работу) КЛИЕНТЫ (ФИО, адрес, паспортные данные и др.)

7. Телефонная компания. АБОНЕНТ (номер телефона, тарифный план, сумма на счете, ФИО владельца, дата приобретения) ЗВОНКИ (исходящий номер, входящий номер, время, длительность) ОПЕРАЦИИ (пополнение счета, списание средств /звонки, SMS/, сумма, дата)

8. Учебные планы. УЧЕБНЫЙ ПЛАН (специальность, семестр, даты начала/окончания семестра, начала/окончания сессии, начала/окончания каникул). ДИСЦИПЛИНЫ (наименование, количество часов лекций, упражнений, лабораторных, форма контроля /экзамен, зачет, дифференцированный зачет/, читающая кафедра, учебный план к которому относится дисциплина). КАФЕДРЫ (шифр, наименование, зав. кафедрой). ГРУППЫ (специальность, номер, выпускающая кафедра)

9. Прокат. ОБЪЕКТЫ (наименование, категория, страховочная стоимость, дата ввода в эксплуатацию, стоимость аренды в сутки/в месяц, характеристики, процент износа) КАТЕГОРИИ (наименование, срок эксплуатации) ДОГОВОРА (номер, дата, данные клиента, объекты, взятые в прокат и на какой срок)

10. Учет основных средств предприятия. ОБЪЕКТЫ (наименование, категория, балансовая стоимость, дата ввода в эксплуатацию, характеристики, материально-ответственное лицо /МОЛ) КАТЕГОРИИ (наименование, процент ежемесячно начисляемого износа) МОЛ (ФИО, должность, дата приема на работу) ИЗНОС (объект, дата начисления, сумма)

11. Библиотечный абонемент. КНИГИ (ISBN, автор, наименование, объем, переплет, стоимость) ЧИТАТЕЛИ (ФИО, паспортные данные, телефон, дата регистрации) ЖУРНАЛ (книга, читатель, дата выдачи, срок)

12. Оплата кредита. КЛИЕНТ (ФИО, паспортные данные, данные работы, доход) КРЕДИТ (сумма, когда, кому, под какой процент, на какой срок выдан, назначение) ОПЛАТА (дата, сумма, кредит, плательщик)

13. Ж/д билеты БИЛЕТ (поезд, дата, станция и время отправления, станция и время назначения, вагон, место) ПОЕЗД (номер, пункт и время отправления, пункт и время прибытия, кол-во вагонов люкс/купе/плацкарт) СТАНЦИИ (наименование, населенный пункт, область)

14. Путевки. КЛИЕНТЫ (ФИО, паспортные данные, адрес, телефон) ОТЕЛЬ (страна, название, адрес, телефон, управляющий, сведения о номерах, бассейнах и т.п.) ПУТЕВКИ (клиент, отель, даты заезда/отъезда, номер)

15. Заявки ЖЭКа. ДОМА (улица, номер, этажность, дата ввода в эксплуатацию) КВАРТИРЫ (дом, номер, подъезд, этаж, ответственный квартиросъемщик) ЖИЛЬЦЫ (ФИО, квартира, дата прописки, д/р) ЗАЯВКА (Когда и кем из жильцов подана заявка, содержание, исполнитель заявки, срок исполнения)

16. Складской учет. ТОВАРЫ (артикул, наименование, производитель) ПРИХОД (товар, количество) РАСХОД (товар, количество)

17. Учет заказов на изготовление книжной продукции. Книга (название, автор, издательство, формат, объем, тираж, переплет) изготавливается тремя рабочими (наборщик, печатник, переплетчик) из материалов (основная бумага, бумага переплета).

18. Таксопарк. Учет заказов на такси. (время приема заказа, номер телефона клиента, адрес, водитель, автомобиль).

19. Театральные кассы. Какие спектакли, где идут, кто режиссер, актерский состав, сведения о билетах в разные категории мест (партер, амфитеатр, балкон).

20. ОСАГО. Фиксируется факт ДТП (время, место, участники, обстоятельства, объяснения и т.д.), данные страховых полисов участников ДТП (страховая компания, номер полиса).

21. Кадровое агентство (биржа труда). Человек (м.б. безработный, м.б. нет) подает резюме, в котором указывается — помимо очевидных атрибутов — стаж, последнее место работы и должность, последняя зарплата, специальность и т.д. Работодатель составляет вакансии: место и должность, з/п, особенности работы и т.д. В специальной таблице нужно вести подбор вакансий и резюме по специальности, по зарплате и т.д.

## **КОНТРОЛЬНЫЕ ВОПРОСЫ И ЗАДАНИЯ**

1. Дайте определение термину «база данных»
2. Раскройте понятие система управления базами данных.
3. Перечислите модели баз данных.
4. Дайте определение реляционной базе данных.
5. Дайте определение термину «таблица».
6. Дайте определение термину «первичный ключ»
7. Опишите роль форм в БД.
8. Опишите роль запросов в БД.
9. Опишите роль отчетов в БД.
10. Опишите процесс создания базы данных.
11. Перечислите режимы, использующиеся в OpenOffice.org Base для создания таблиц.
12. Перечислите нормальные формы представления данных.
13. Опишите назначение значения Null поля таблицы.

## **ФОРМА ОТЧЕТА ПО ЛАБОРАТОРНОЙ РАБОТЕ**

На выполнение лабораторной работы отводится 2 занятия (4 академических часа: 3 часа на выполнение и сдачу лабораторной работы и 1 час на подготовку отчета).

Номер варианта студенту выдается преподавателем.

Отчет на защиту предоставляется в печатном виде.

Структура отчета (на отдельном листе(-ах)): титульный лист, формулировка задания (вариант), ход выполнения работы, результаты выполнения работы, выводы.

## ОСНОВНАЯ ЛИТЕРАТУРА

1. Карпова, Т.С. Базы данных: модели, разработка, реализация : учебное пособие / Т.С. Карпова. - 2-е изд., исправ. - Москва : Национальный Открытый Университет «ИНТУИТ», 2016. - 241 с. : ил. ; То же [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=429003>
2. Давыдова, Е.М. Базы данных [Электронный ресурс] : учеб. пособие / Е.М. Давыдова, Н.А. Новгородова. — Электрон. дан. — Москва : ТУСУР, 2007. — 166 с. — Режим доступа: <https://e.lanbook.com/book/11636>. — Загл. с экрана.
3. Харрингтон, Д. Проектирование объектно ориентированных баз данных [Электронный ресурс] — Электрон. дан. — Москва : ДМК Пресс, 2007. — 272 с. — Режим доступа: <https://e.lanbook.com/book/1231>. — Загл. с экрана.

## ДОПОЛНИТЕЛЬНАЯ ЛИТЕРАТУРА

4. . Голицина О.Л., Максимов Н.В., Попов И.И. Базы данных: учеб. пособие. – М.: Форум:Инфра-М, 2007.
5. 5. Гагарин Ю.Е. Применение языка SQL в MS Access: учебно-методическое пособие. – М.: МГТУ им. Н.Э. Баумана, 2012.

### Электронные ресурсы:

1. Научная электронная библиотека <http://eLIBRARY.RU>
2. Электронно-библиотечная система <http://e.lanbook.com>
3. Электронно-библиотечная система «Университетская библиотека онлайн» <http://biblioclub.ru>
4. Электронно-библиотечная система IPRBook <http://www.iprbookshop.ru>