

РОСЖЕЛДОР

**Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Ростовский государственный университет путей сообщения»
(ФГБОУ ВО РГУПС)**

Е.В. Голубенко, А.Г. Кулькин

УПРАВЛЕНИЕ ДАННЫМИ

Учебно-методическое пособие
к расчётно-графической работе

Ростов-на-Дону
2017

УДК 681.3(07) + 06

Рецензент – доктор технических наук, профессор М.А. Бутакова

Голубенко, Е.В.

Управление данными: учебно-методическое пособие к расчётно-графической работе / Е.В. Голубенко, А.Г. Кулькин; ФГБОУ ВО РГУПС. – Ростов н/Д, 2017. – 19 с.

В учебно-методическом пособии приведены задания и методика выполнения расчётно-графической работы по дисциплине «Управление данными».

Учебно-методическое пособие предназначено для студентов направления подготовки «Информатика и вычислительная техника».

Одобрено к изданию кафедрой «Вычислительная техника и автоматизированные системы управления».

Содержание

1	Цель и содержание расчётно-графической работы.....	4
2	Краткие теоретические сведения по теории реляционных таблиц.....	5
3	Постановка задачи.....	12
	3.1 Описание контрольного примера.....	13
4	Варианты индивидуальных заданий.....	17
	Библиографический список.....	18

1 Цель и содержание расчётно-графической работы

Цель расчётно-графической работы – закрепить навыки самостоятельного проектирования реляционных баз данных, а также создания приложений в среде СУБД MicroSoft Access.

Расчётно-графическая работа состоит из пояснительной записки с описанием контрольного примера.

Пояснительная записка должна быть оформлена в соответствии с требованиями ЕСКД и ЕСПД и содержать следующие разделы:

1. Постановка задачи и анализ задания.
2. Реализация метода (алгоритма) решения задачи на компьютере.
3. Описание контрольного примера.
4. Заключение.
5. Список литературы.
6. Приложение (файл базы данных на диске).

Объём пояснительной записки 10–15 страниц машинописного текста.

В кратком докладе на защите РГР (5 мин.) студент должен изложить сущность выполненной работы: обоснование выбора принятой структуры данных: структуры реляционных таблиц и связей между таблицами, обоснование выбора ключевых полей, продемонстрировать контрольный пример.

2 Краткие теоретические сведения по теории реляционных баз данных

База данных – это компьютерный термин для обозначения совокупности информации по определенной теме или определенной прикладной области. Хранение такой информации в виде базы данных позволяет существенно облегчить ее использование и доступ к ней.

Традиционная система хранения информации состоит из людей, бумаг и картотечных шкафов. Основным носителем информации является бумага. В реальной системе делопроизводства имеются папки для входящей и исходящей информации и определенный метод ведения документации. Доступ к информации осуществляется вручную путем открывания ящика, извлечения из него нужной папки и поиска в этой папке конкретного документа. Информация тем или иным способом вносится в документы, например, с помощью пишущей машинки. Поиск осуществляется путем сортировки или копирования информации из разных документов на один лист бумаги. Для анализа или подготовки отчета применяется электронная таблица или калькулятор.

Компьютерная база данных – это ни что иное, как автоматизированная версия системы заполнения, хранения и извлечения документов. В компьютерных базах данных информация хранится в определенном формате. В таких базах данных может содержаться любая информация – от простого текста (например, имя и адрес) до сложной структуры, включая рисунки, звуки и изображения. Хранение данных в заранее известном формате позволяет извлекать данные в желаемом формате благодаря использованию разных методов вывода, таких как запросы и отчеты.

Набор программ, обеспечивающих централизованное управление массивами данных в базе, образует файловую систему, называемую системой управления базой данных **СУБД**.

Автоматизированная СУБД представляется следующим образом. Человек использует компьютер для доступа к хранящимся в таблицах данным, вводя данные в таблицы через формы ввода данных и получая их с помощью запроса. По запросам извлекаются из таблиц только необходимые данные. Затем данные форматируются с помощью программы отчета и выводятся на экран или принтер.

Реляционные базы данных представляют связанную между собой совокупность таблиц баз данных. Связь между таблицами может находить свое отражение в структуре данных, а может только подразумеваться, то есть присутствовать на неформализованном уровне. Пользователь может адресовать сложные запросы одной или нескольким таким таблицам. РСУБД выдает запрашиваемую информацию в виде форм и отчетов.

Типичным представителем реляционной СУБД является мощное приложение Microsoft Access. Access предназначен для хранения и получения данных, представления их в удобном виде и автоматизации часто выполняемых операций (например, для ведения счетов, учета материальных ценностей, планирования и т.п.). Используя Access, можно не только разрабатывать удобные формы ввода

данных, но и обрабатывать данные, а также составлять всевозможные сложные отчеты.

Производительность СУБД Access соответствует возможностям такой развитой операционной системы, как Microsoft Windows. Windows и Access, детища фирмы Microsoft, очень хорошо взаимодействуют друг с другом. Система Access работает под управлением Windows, и все преимущества Windows реализованы и в Access. Это означает, что можно вырезать, копировать, вставлять данные из любого приложения Windows в Access и обратно.

Access – *реляционная СУБД*. Прежде всего это означает, что основным элементом базы является таблица. С помощью Access можно получить доступ к любым данным любого типа и использовать одновременно несколько таблиц. Использование реляционной СУБД позволяет упростить структуру базы данных и, таким образом, облегчить выполнение работы. Можно связать таблицу Access с данными, хранящимися на большой ЭВМ или на сервере, а также использовать таблицы, созданные в других СУБД.

В Access реализовано корректное управление реляционными базами данных. Эта система поддерживает первичные и внешние ключи и обеспечивает полную поддержку целостности данных на уровне самой базы данных, что предотвращает несовместимое обновление или утрату данных. СУБД Access включает в себя программы проверки данных для предотвращения ввода некорректных значений. Каждое поле в таблице имеет определенный формат и заданный по умолчанию тип. Access поддерживает все необходимые типы полей: *Текстовый* – для хранения одной строки текста, *Числовой* – для чисел, *Денежный* – для представления денежных величин, *Дата\Время* – для времени, *Поле MEMO* – для хранения произвольной информации, *Логический* – для логических значений, *Поле объекта OLE* – для хранения объектов из других приложений, поддерживающих технологию OLE

В Access реализован специальный инструмент, называемый *Мастером*. Мастер поможет за несколько минут выполнить рутинную работу, на которую раньше уходили целые часы. Мастера задают вопросы о содержании, стиле и формате объекта, а затем создают этот объект без какого-либо вмешательства пользователя. В системе Access имеется около сотни мастеров, предназначенных для проектирования баз данных, приложений, таблиц, форм, отчетов, графиков, почтовых наклеек, элементов управления и свойств.

В Microsoft Access используется традиционная терминология баз данных – термины «база данных», «таблица», «запись», «поле» и «значение» составляют элементы базы. В базе данных содержится одна или больше таблиц (другими словами, логически сгруппированных данных). Большинство приложений имеет несколько связанных таблиц, что позволяет эффективнее представлять информацию. Приложение, использующее несколько таблиц, может манипулировать данными более эффективно, чем при использовании одной большой таблицы. Множественные таблицы упрощают ввод данных и создание отчетов, уменьшая ввод избыточных данных.

В таблице информация группируется по строкам и столбцам. Таблица состоит из строк, называемых *записями*, и столбцов, называемых *полями*. Данные в таблице упорядочены по столбцам, например, по фамилии студента (см. таб.1). Эти столбцы являются *полями*. Каждое поле имеет определенный *тип данных* (текст, число, дата и т.д.), *длину и уникальное имя*, которое идентифицирует хранящуюся в этом поле информацию.

Строки данных в таблице - это ее *записи*. Каждая строка считается отдельной величиной, к которой можно получить доступ и по которой можно отсортировать таблицу.

На пересечении строки (записи) и столбца (поля) в ячейках находится значение – собственно **данные**. Здесь приводится фрагмент базы данных (табл. 1) содержащей данные о техническом осмотре.

Таблица 1

номер	дата слачи	фамилия	марка	год	фамилия
1	23.02.02	Емельянов	Жигули	1954	Гусев
2	25.03.02	Орлова	Porsche	2000	Иванов
3	01.09.02	Тарасов	Mencelес	1995	Ширман
4	04.04.02	Бутман	Ferrari	1967	Немиров

В этой таблице колонки (поля) имеют имена: дата, фамилия владельца, марка, год, фамилия инспектора.

В то время как поля различаются по имени, записи (строки) обычно идентифицируются по некоторой уникальной характеристике. В таблице техосмотр таким уникальным идентификатором является **номер** (идентификационный номер - первая колонка). В других таблицах это может быть табельный номер работника, шифр изделия и т.п. (Важно, что значения этого поля не могут повторяться в разных записях).

В базе данных Access хранятся данные и ассоциирующиеся с ними объекты. К объектам баз данных относятся таблицы, запросы, формы, макросы и модули. В базе данных Access содержится шесть объектов, состоящих из данных и средств, необходимых для использования Access.

Таблица – содержит собственно данные (для их отображения используется просмотр в режиме таблицы).

Запрос – позволяет искать, сортировать и извлекать определенные данные.

Форма – позволяет вводить и отображать данные в указанном формате.

Отчет – позволяет отображать и печатать форматированные данные, включая калькуляции и итоги.

Макрос – предоставляет простые команды для автоматизации выполнения задач без программирования. –

Модуль – программы, написанные на VBA. Одним из основных режимов работы с базой является режим таблицы.

Просмотр данных в режиме таблицы – это один из нескольких методов просмотра данных. В этом режиме таблица отображается в виде строк и столбцов (наподобие электронной таблицы).

В этом режиме таблицу можно листать, используя клавиши управления курсором. Кроме того, можно вносить изменения в данные. При этом нужно соблюдать осторожность, так как любые изменения в этом режиме отражаются в таблице данных.

Для извлечения информации из базы данных используется запрос. С его помощью можно выбрать и определить группу записей, удовлетворяющих определенному условию. Использование запросов перед печатью отчета позволяет выводить на печать только нужные данные. Чтобы на экране появлялись только записи, соответствующие заданному критерию, можно использовать формы. Запросы можно использовать и в процедурах, изменяющих, добавляющих или уничтожающих записи базы данных. Выбранные записи называются динамическим набором. Он будет изменяться вместе с данными в оригинальной таблице.

Формы ввода данных помогают пользователю быстро, легко и без ошибок поместить информацию в таблицу базы данных. Формы ввода и отображения данных обеспечивают более структурированный подход, чем использование режима таблицы. Тем не менее, можно по-прежнему просматривать, добавлять, изменять или удалять записи базы данных. Использование форм ввода данных - самый распространенный способ внесения данных в таблицу базы данных.

Отчеты представляют данные в том виде, в каком они должны выводиться на печать. В одной СУБД можно создавать несколько типов отчетов. Отчеты могут объединять множество таблиц, чтобы отразить сложные зависимости между различными наборами данных.

Главное окно Access - это рабочий стол системы (см. рис.2). Здесь можно открывать другие окна, каждое из которых по-своему представляет обрабатываемые данные. Ниже кратко описаны основные элементы Главного окна Access.

Строка заголовка. В строке заголовка отображается имя активной в данный момент программы. Строка заголовка главного окна Access всегда отображает имя программы Microsoft Access.

Кнопка минимизации. Щелчок по этой кнопке позволяет свернуть главное окно Access до кнопки на панели задач Windows. Access можно снова активизировать, щелкнув по кнопке на панели задач.

Кнопки восстановления размеров/максимизации.

Кнопка восстановления размеров появляется только тогда, когда главное меню Access имеет максимальный размер.

Кнопка закрытия. После щелчка на этой кнопке работа Access завершается.

Строка меню. Строка меню содержит имена нескольких подменю.

Панель инструментов. Панель инструментов - это группа пиктограмм, расположенная непосредственно под полосой меню Главное ее назначение - ускоренный вызов команд меню. Кнопки панели инструментов также могут изменяться в зависимости от выполняемых операций.

Строка состояния. В левой части строки состояния отображается информация о том, что происходит в настоящее время.

Окно базы данных. Это окно появляется при открытой базе данных - в нем сосредоточены все "рычаги управления" базой данных.

Прежде чем приступить к созданию таких объектов базы данных, как таблицы, формы и отчеты, нужно разработать их проект. Главное назначение проекта в выработке четкого пути, по которому нужно следовать при его реализации.

В окне базы данных отображаются все файлы объектов базы данных, которые можно создать с помощью Access. На самом деле база данных является единым файлом. Все создаваемые файлы объектов сохраняются в файле базы данных.

Создать базу данных можно с помощью команды **Создать** из меню **Файл** или кнопки **Создать базу данных** на панели инструментов. Созданной базе данных необходимо присвоить имя. По умолчанию имя принято db1.mdb. Можно ввести вместо него другое имя. При этом создается пока пустая база данных. После создания или открытия базы данных на экране появляются дополнительные пункты меню.

Кнопка **Создать** окна базы данных позволяет создать новый объект, а кнопка **Открыть** – открыть существующий объект. Для внесения изменений в существующий объект можно использовать кнопку **Конструктор**.

Создание структуры таблицы - многошаговый процесс. Для быстрого конструирования таблицы необходимо выполнить следующие шаги:

1. Создать структуру новой (пустой) таблицы.
2. Ввести имена полей, тип данных и их описание.
3. Определить свойства для каждого поля.
4. Установить первичный ключ.
5. Для нужных полей создать индексы.
6. Сохранить структуру.

В окне конструктора таблиц содержатся такие области:

1. Область ввода полей.
2. Область ввода свойств.

Область ввода полей предназначена для ввода имен полей и типов данных, а также для описания отдельных полей, а **область свойств полей** - для ввода других опций поля, называемых свойствами. Они включают размер поля, формат, маску ввода, подпись, значение по умолчанию, условие на значение, сообщение об ошибке, обязательное поле, пустые строки и индексированное поле.

Поля создаются путем ввода имени поля и типа данных поля в каждую строку области ввода полей окна конструктора таблиц. Описание поля - это опция, которая указывает на назначение поля. При вводе данных в режиме таблицы она появляется в строке состояния. После ввода имен полей и их типов в области свойств следует указать, как поля будут использоваться.

При создании таблицы следует придерживаться тщательно продуманного плана. Однако даже при наличии плана можно вносить изменения. Например, чтобы вставить нужное поле, следует поместить курсор в существующее поле и выбрать команду **Поле** из меню **Вставка**. В таблице появится новая строка, а все существующие поля будут смещены вниз. Удалить поле можно тремя способами:

1. Выбрать поле, щелкнув на маркере строки (идентификатор строки с левой стороны) и нажав кнопку **Delete**.

2. Выбрав поле, а затем выполнив команду **Удалить строку** из меню **Правка**.

3. Выбрав поле и щелкнув по кнопке **Удалить строку** на панели инструментов.

Очень просто изменить расположение полей. Для этого необходимо щелкнуть на маркере поля (идентификатор строки с левой стороны) и перетащить его в новое место.

Каждая таблица должна иметь **первичный ключ** – одно или несколько полей, которые делают запись уникальной. Если не был задан уникальный ключ, Access создаст его автоматически.

Завершенную структуру нужно сохранить, выбрав команду **Сохранить** из меню **Файл**. При первом сохранении Access попросит указать имя таблицы. Это имя должно состоять не более чем из 64 символов. При создании нескольких таблиц в базе данных может понадобиться использовать эти таблицы в других базах данных. В окне базы данных можно выполнять множество операций с таблицами, включая следующие:

1. Переименование таблиц
2. Удаление таблиц.
3. Копирование таблиц в базе данных.
4. Копирование таблицы из другой базы.

Для ввода значений в поле таблицы следует поместить курсор в это поле и ввести значение. При редактировании записи маркер записи принимает вид карандаша. При вводе новой записи (на маркере записи изображен карандаш) автоматически появляется следующая строка.

К любой записи можно перейти, разместив в ней курсор или щелкнув на ней мышью. Для перемещения по записям можно использовать вертикальную полосу прокрутки. Кнопки со стрелками на полосе прокрутки позволяют перемещать указатель записи на одну запись за один щелчок. Команда **Перейти** из меню **Правка** предоставляет несколько вариантов выбора, обеспечивающих быстрое перемещение по таблице (первая запись, последняя запись, следующая запись, предыдущая запись, новая запись).

Найти значение в поле можно тремя способами:

1. Выбрать команду **Найти** из меню **Правка**.
2. Щелкнуть на кнопке **Найти** на панели инструментов.
3. Воспользоваться комбинацией клавиш **Ctrl+F**.

После использования любого из этих методов появится диалоговое окно. Значения поля **Просмотр** задают направление поиска. Для ограничения поиска по заданному полю следует поместить курсор в то поле, которое необходимо использовать для поиска, и установить флажок **Только в текущем поле**. В текстовом поле **Образец** вводится искомое значение. Значение можно ввести так, как оно выглядит в поле, а можно использовать приведенные ниже специальные символы: * - соответствует любому количеству любых цифр или любых символов,

? - соответствует любому текстовому символу;

- соответствует любой цифре или любому символу.

В раскрывающемся списке Совпадения содержится три варианта выбора:

1. С любой частью поля.
2. Поля целиком.
3. С начала поля.

Значением, установленным по умолчанию, является значение **Поля целиком**. Флажок **С учетом регистра** определяет, будут ли различаться прописные и строчные буквы. По умолчанию они не различаются.

Обычно данные выводятся в сетке. С помощью команды **Ячейки** из меню **Формат** можно задать, будет ли отображаться эта сетка и как она будет выглядеть. В диалоговом окне **Вид сетки** содержатся опции отображения линий сетки. Используя флажки **Линии сетки**, можно исключить **линии По горизонтали** и **По вертикали**.

Табличные данные можно просматривать в различных режимах, но режим формы обеспечивает максимальную гибкость. Форма позволяет отображать одновременно все поля одной или нескольких записей. Можно создать пять типов форм:

1. В один столбец.
2. Ленточная.
3. Главная\подчиненная.
4. Табличная.
5. Диаграмма.

Формы предоставляют большую гибкость при вводе данных по сравнению со строками и столбцами. Можно не только помещать вычисляемые поля в форму, но и добавлять расширенные правила проверки корректности ввода и элементы управления.

Главное назначение базы данных (хранение и предоставление информации) может потребоваться сразу после ввода данных, либо спустя годы. Слово **запрос** (query) первоначально означало спрашивать или узнавать. Запрос можно представить как вопрос, имеющий отношение к информации, которая содержится в таблицах базы данных. **Запрос** – это требование предоставить информацию, накопленную в таблицах. Информацию можно получить с помощью инструментов запроса. Access включает множество типов запросов, которые можно разбить на пять категорий:

1. Запрос на выборку. Самый распространенный тип запроса. Извлекает данные из одной или нескольких таблиц и результаты отображает в объекте в режиме таблицы, в котором допускается изменение записей (при некоторых ограничениях). Используется для группировки записей, а также для вычисления сумм, средних значений, подсчета количества записей и расчета итоговых значений других типов

2. Запрос с параметрами. Это запрос, при запуске которого открывается диалоговое окно с приглашением ввести определенные сведения. Эти запросы удобно использовать как базовый источник данных для форм и отчетов

3. Перекрестный запрос. Отображает результат статистических расчетов, выполненных по данным из одного поля. Эти результаты группируются по двум наборам данных в формате перекрестной таблицы

4. **SQL.** Это запрос, создаваемый с помощью инструкции SQL.

5. **Запрос на изменения.** Это запрос, который позволяет, выполнив одну операцию, внести изменения во многие записи.

Отчеты используются для представления данных в удобном виде. Отчет можно вывести на экран или распечатать на принтере. В нем можно группировать и сортировать данные в любом порядке, получать итоговые значения, средние значения и другие статистические величины, а также помещать в него графические объекты.

При работе с базой данных часто приходится выполнять одни и те же задачи. Этот процесс можно автоматизировать с помощью макросов. *Макрос* - такой же объект Access, как таблица, запрос, форма и отчет. Он создается для автоматического выполнения определенных действий или совокупности действий. С помощью макроса можно имитировать выбор команды меню или перемещение мыши. Любая команда, которую необходимо выполнить, называется **макрокомандой**. Макрос создается в окне макросов; запускается он после ввода его характеристик во вкладке События окна свойств формы или отчета.

В базах данных используются модули, которые можно разделить на две основные категории – модули общей области и модули форм и отчетов. *Модули общей области* независимы от отдельных форм и отчетов. Такие модули можно использовать для хранения программ, разработанных в виде процедур, которые будут применяться во всем приложении. Любым событиям при работе с формами или отчетами могут быть назначены процедуры обработки событий. Процедуры обработки событий достаточно эффективны и предоставляют дополнительные удобства при работе с базой данных.

3 Постановка задачи

Создать базу данных Технический осмотр. В таблицах базы данных должна содержаться информация: номер записи, дата сдачи, фамилия владельца автомобиля, марка машины, год изготовления, фамилия инспектора. С помощью запросов получить информацию: кто из владельцев прошёл техосмотр в указанный период, какие марки автомобилей были осмотрены в заданный период времени, в какие дни проходили технический осмотр с определённым годом выпуска, в какие дни был произведён осмотр машин некоторым инспектором. Создать форму Технический осмотр, содержащую информацию: номер записи, дата сдачи, фамилия владельца автомобиля, марка машины, год изготовления, фамилия инспектора. Создать макросы, запускающие запросы. Создать кнопочную форму, запускающую макросы и форму.

Для реализации этой базы данных необходимо создать 4 таблицы и заполнить их. Создать схему данных. Создать четыре запроса на выборку, одну форму, четыре макроса и кнопочную форму для удобства пользования базой данных.

3.1 Описание контрольного примера

Сначала необходимо создать 4 таблицы, в которых будет храниться информация. Для этого необходимо перейти на вкладку Таблицы и нажать кнопку Создать и в режиме конструктора ввести имя поля и тип данных. Номер записи – счётчик, дата сдачи – дата/время, фамилия владельца – текстовый, марка машины - числовой, год изготовления – числовой, фамилия инспектора – числовой. Затем создать еще 3 таблицы в которых будет информация о годе выпуска автомобиля, инспекторе и автомобилях. Так как Год изготовления в таблице Год ни с чем не сравнивается и над ним не производится никаких арифметических операций, то его можно сделать текстовым типом. После того как поля и типы данных в таблицах определены все таблицы заполнены можно приступить к созданию формы. Но сначала необходимо установить связи между таблицами (схема данных). Для этого необходимо нажать кнопку Схема данных на панели управления. Затем добавить таблицы и установить связи между ними.

Для того, чтобы создать форму необходимо перейти на вкладку Формы. Нажать кнопку Создать и с помощью Мастера форм создать форму: выбрать все поля для формы, нажав кнопку >>, затем внешний вид формы в один столбец, выбрав стиль формы Сумерки и, наконец, дав форме имя нажать кнопку Готово. Форма готова.

Теперь перейдём к созданию запросов. Перейти на вкладку Запросы. Создадим сначала Запрос по году выпуска. Для этого нажать кнопку Создать нажать раздел Конструктор и добавить таблицы Технический осмотр и Год, затем добавить поля дата сдачи из таблицы Технический осмотр и год изготовления из таблицы Год. В строке Условие отбора введём текст [Введите интересующий вас год] и, закрывая запрос сохраним его под именем Запрос по году выпуска. Точно так же создадим три других запроса: Запрос по дате и автомобилю – добавим поля дата сдачи из таблицы Технический осмотр и автомобили из таблицы Автомобили, а в строке Условие отбора введём текст Between [Введите начальную дату:] And [Введите конечную дату:] и сохраним запрос под именем Запрос по дате и автомобилю; Запрос по дате и фамилии – добавим поля фамилия владельца и дата сдачи из таблицы Технический осмотр, а в строке Условие отбора введём текст Between [Введите начальную дату:] And [Введите конечную дату:] и сохраним запрос под именем Запрос по дате_и фамилии; Запрос по инспектору – добавим поля фамилия инспектора из таблицы Инспекторы и дата сдачи из таблицы Технический осмотр, а в строке Условие отбора введём текст [Введите интересующую вас фамилию] и сохраним запрос под именем Запрос по инспектору. Все необходимые запросы созданы.

Перейдём на вкладку Макросы и нажмём кнопку Создать. В поле Макрокоманда выберем Открыть запрос. В Аргументах макрокоманды в поле Имя запроса выберем Запрос по инспектору, режим – таблица, режим данных – только чтение. Точно так же создадим три других запроса, только выбирая другие имена запросов: Запрос по году выпуска, Запрос по дате_и автомобилю,

Запрос по дате_и фамилии. И дадим имена макросам Вызов запроса по инспектору, Вызов запроса по году выпуска, Вызов запроса по дате_и автомобилю, Вызов запроса по дате_и фамилии.

Остаётся только создать Кнопочную форму для удобства пользования базой данных. Выберем меню Сервис, пункт Надстройки, команду Диспетчер кнопочных форм. Далее нажмём кнопку Изменить, затем кнопку Создать. Далее в строке Текст вводим название кнопки, в строке команда выбираем Открытие формы в режиме редактирования, а в строке Форма – форму. Точно так же создаются другие четыре кнопки: в строке Текст вводим название кнопки, в строке команда выбираем Запуск макроса, в строке макрос выбираем необходимый макрос. Эти действия производим пока не будут созданы кнопки, вызывающие все макросы. И создаём ещё одну кнопку: кнопку Выход. Для этого необходимо в строке Команда выбрать Выход из приложения, присвоив ей (кнопке) имя Выход и нажмём кнопку ОК.

Заполненные таблицы

Таблица Технический осмотр

номер	дата сдачи	фамилия	марка	фамилия	кол гола
1	23.02.16	Емельянов	1	1	1
2	25.03.16	Орлова	2	4	2
3	01.09.16	Тарасов	3	3	3
4	04.04.16	Бутман	4	5	2
5	30.03.16	Сомов	5	2	1
6	15.01.16	Ломов	2	3	2
7	10.06.16	Лопатин	3	4	3
8	05.08.16	Тяпкин	2	5	4
9	09.12.16	Войнова	1	2	1
10	30.11.16	Карпов	1	3	3
11	17.09.16	Кирьянов	5	1	4

Таблица Автомобили

кол автом	автомобили
1	Fiat
2	Ferrari
3	BMW
4	Mercedes
5	Lexus

Таблица Инспекторы

кол инспек	фамилия
1	Гусев
2	Ляпкин
3	Горин
4	Иванов
5	Запилин

Таблица Год

кол	год
1	2005
2	2015
3	2008
4	1999

Примеры работы запросов

Запрос по году выпуска

дата слачи	год
01.09.16	2008
10.06.16	2008
30.11.16	2008

Запрос по инспектору

дата слачи	фамилия
25.03.16	Иванов
10.06.16	Иванов

Запрос по дате_и фамилии

Начальная дата 1.01.16

Конечная дата 30.07.16

фамилия	дата слачи
Емельянов	23.02.16
Орлова	25.03.16
Бутман	04.04.16
Сомов	30.03.16
Ломов	15.01.16
Лопатин	10.06.16

Запрос по дате_и автомобилю

Начальная дата 1.01.16

Конечная дата 30.07.17

автомобили	дата
Fiat	23.02.16
Ferrari	25.03.16
Mercedes	04.04.16
Lexus	30.03.16
Ferrari	15.01.16

Схема данных



4 Варианты индивидуальных заданий

1. Автотранспортное предприятие.
2. Автосервис.
3. Аэропорт.
4. Школа
5. Среднее специальное учебное заведение (техникум, колледж).
6. Санаторий.
7. Детский сад.
8. Рекламное агентство.
9. Дистанция сигнализации и связи.
10. Сортировочная горка.
11. Грузовая железнодорожная станция.
12. Больница.
13. Гостиница.
14. Парк аттракционов.
15. Сервис мобильных телефонов.
16. Филармония
17. Локомотивное депо.
18. Банк
19. Фондовая биржа
20. Вагонное депо
21. Склад
22. Поликлиника
23. Турфирма
24. Риэлтерская фирма
25. Аптека

Библиографический список

1 Кузин, А.В. Базы данных : учеб. пособие для вузов/ А. В. Кузин, С. В. Левонисова. -5-е изд., испр.. -М.: Академия, 2012. -315 с.:а-ил.

2 Казак, А.А. Разработка реляционных баз данных : учеб.-метод. пособие/ А.А. Казак; РГУПС. -Ростов н/Д, 2011. -40 с.:а-ил.

3 Карпова, Т.С. Базы данных: модели, разработка, реализация.- СПб.: Питер, 2013.

4 Хансен, Г. Базы данных: разработка и управление. / Г. Хансен, Дж. Хансен. – М.: Бином, 1999.

Учебное издание

Голубенко Евгений Владимирович
Кулькин Александр Георгиевич

УПРАВЛЕНИЕ ДАННЫМИ

Печатается в авторской редакции

Технический редактор А.В. Артамонов

Подписано в печать 14.08.17. Формат 60×84/16.

Бумага газетная. Ризография. Усл. печ. л. 1,16.

Тираж экз. Изд. № 901. Заказ .

Редакционно-издательский центр ФГБОУ ВО РГУПС.

Адрес университета: 344038, г. Ростов н/Д, пл. Ростовского Стрелкового Полка
Народного Ополчения, д. 2.