**Министерство образования и молодежной политики СК**

**ГБПОУ «Лермонтовский региональный многопрофильный колледж»**

**Работа допущена к защите**

Зав.кафедрой «Программирование

в компьютерных системах»

\_\_\_\_\_\_\_\_\_\_\_\_\_\_Г.Д.Томулевич

«\_\_\_\_»\_\_\_\_\_\_\_\_\_\_2015 г.

**КУРСОВАЯ РАБОТА**

###### Пояснительная записка

на тему:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Прокат автомобилей*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

**ПМ.02 Разработка и администрирование баз данных**

**специальность 09.02.03 «Программирование в компьютерных системах»**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Выполнил студент гр. | | ***ПР4*** |  |  | | **В.С.Меренчук** |
|  | | группа |  | подпись | | Ф.И.О. студента |
| Проверил |  | | | | **А.А.Светличный** | |
|  | оценка и подпись | | | | Ф.И.О. преподавателя | |

Лермонтов, 2015

СОДЕРЖАНИЕ

cтр.

[ВЕДЕНИЕ 3](#_Toc437960970)

[1. ПЛАНИРОВАНИЕ СОЗДАНИЯ БАЗЫ ДАННЫХ «ПРОКАТ АВТОМОБИЛЕЙ» 4](#_Toc437960971)

[2. Составление требований заказчика 6](#_Toc437960972)

[3. Проектирование БАЗЫ ДАННЫХ «ПРОКАТ АВТОМОБИЛЕЙ» 7](#_Toc437960973)

[3.1. Визуальное моделирование 7](#_Toc437960974)

[4. Разработка БАЗЫ ДАННЫХ «ПРОКАТ АВТОМОБИЛЕЙ» 8](#_Toc437960975)

[4.1. Создание таблиц 8](#_Toc437960976)

[4.2. Создание форм 10](#_Toc437960977)

[4.3. Создание запросов 14](#_Toc437960978)

[4.4. Создание отчетов 20](#_Toc437960979)

[5. Тестирование БАЗЫ ДАННЫХ «ПРОКАТ АВТОМОБИЛЕЙ» 23](#_Toc437960980)

[6. Сопровождение БАЗЫ ДАННЫХ «ПРОКАТ АВТОМОБИЛЕЙ» 24](#_Toc437960981)

[Заключение 26](#_Toc437960982)

[Список литературы 27](#_Toc437960983)

# ВЕДЕНИЕ

В настоящее время большинство организаций используют различные базы данных для автоматизации процессов автоматизации обработки информации, удобства её эксплуатации, повышения надёжности хранения данных, сокращения числа возможных ошибок в работе, предотвращения избыточности данных, а также в целях повышения производительности.

Состав баз данных и их размер определяются направлением деятельности конкретного предприятия и его мощностью. В связи с тем, что современные информационные системы оперируют большими объемами и сложными структурами данных, средства управления базами данных можно выделить в отдельную группу информационных систем, отвечающих за управление сложными структурированными данными.

Сегодня можно с уверенностью утверждать, что решение широкого круга задач в любой сфере деятельности человека сегодня практически невозможно без использования оперативно управляемых баз данных.

Основные идеи современной информационной технологии базируются на концепции, согласно которой данные должны быть организованы в базы данных с целью адекватного отображения изменяющегося реального мира и удовлетворения информационных потребностей пользователей.

Эти базы данных создаются и функционируют под управлением специальных программных комплексов – систем управления базами данных (СУБД).

Увеличение объема и структурной сложности хранимых данных, расширение круга пользователей информационных систем привели к широкому распространению наиболее удобных и простых для понимания реляционных (табличных) СУБД.

# 1. ПЛАНИРОВАНИЕ СОЗДАНИЯ БАЗЫ ДАННЫХ «ПРОКАТ АВТОМОБИЛЕЙ»

В представлении информации в базах данных можно выделить два уровня абстракции: информационную модель, которая обычно бывает, понятна пользователю этой базы, и физическую модель данных, которая связана с подробностями физического хранения данных в дисковых файлах.

Информационная модель отображает программное обеспечение в терминах фактов, явлений, событий и предметов (а также связей между ними). Описание абстрактной информационной модели для представления в компьютере выполняется средствами модели данных, которую поддерживает СУБД.

Средствами модели данных строится логическая схема информационной модели для внутреннего представления. При этом модель данных, которую поддерживает СУБД, определяется: допустимой структурой данных (разнообразием и количеством типов описываемых с помощью модели объектов), множеством допустимых операций над данными и ограничениями, необходимыми для обеспечения контроля над целостностью данных.

Модели данных, поддерживаемые СУБД, делят на сетевые, иерархические и реляционные. В этой связи различают сетевые, иерархические и реляционные СУБД. В настоящее время ведутся активные работы по созданию объектно-ориентированных СУБД, и хотя имеются функционирующие примеры объектно-ориентированных СУБД, эти работы существенно тормозятся отсутствием теории соответствующей модели данных.

Наиболее популярными СУБД на рынке программного обеспечения и удобными для доступа из VB- и VBA-кода являются Microsoft Access, Microsoft SQL Server и Oracle.

Существует мнение о том, что Access – наиболее простая в освоении и использовании СУБД. Отчасти, это результат маркетинговых усилий компании Microsoft. Те, кто начинал работу с такой СУБД, как dBase III (IV или V), скорее всего, с этим не согласятся. Но, тем не менее, для новичка эта система имеет довольно дружественные средства, которые дают возможность без глубоких знаний теории баз данных создавать несложные объекты для хранения и простой обработки данных.

С другой стороны, специалист по Access может использовать эту систему для работы в сети и доступа к более эффективным системам баз данных, подобным SQL Server. Одним из удобных свойств Access баз данных является возможность хранения всей информации о базе данных в одном файле с расширением mdb (mdb-файле). Сразу следует отметить, что подразумевается под термином «информация о базе данных». В mdb-файле находятся все сведения о некоторой базе данных (с таблицами, связями, запросами, диалоговыми фермами и отчетами), с которыми может работать СУБД, в том числе – Access. Это, в частности, означает, что попытка каким-либо образом использовать mdb-файл в системе, которая не имеет СУБД (например, самой Access), «понимающей» mdb-формат, обречена на неудачу.

Access можно использовать не только для работы с mdb-файлами, эта система позволяет создавать базы данных – таблицы и связи между ними и интерфейсные элементы для работы с этими (и другими, в том числе не только mdb-файлами) базами — диалоговые формы, отчеты, запросы и т.д.

Формы, разработанные в Access, можно применять для, создания клиентского интерфейса, а VB или VBA для формирования кода прикладной задачи. Многие разработчики применяют Access для реализации клиент/серверных приложений.

# 2. Составление требований заказчика

Технические задания являются ключевым разделом аукционной документации. В них должны быть предельно подробно определены стоящие перед Исполнителем задачи и объем работ. Технические задания должны быть составлены настолько детально, чтобы позволить Исполнителю правильно разработать методологию выполнения задания и соответственно оценить его стоимость. Заказчику принципиально важно учитывать, что на этапе реализации Гос. контракта у него нет права в одностороннем порядке менять технические требования, вводить новые виды работ, требовать исполнения каких-то дополнительных заданий. Соответственно, принципиально важно, чтобы уже в аукционной документации был приведен полный и конечный перечень работ, которые должен будет выполнить Исполнитель.

На этапе составления требований заказчика были выявлены следующие требования:

* Создать БД Прокат автомобилей;
* Обеспечить удобство пользования БД;
* Создать таблицы содержащие информацию об:
  + Автомобилях;
  + Клиентах;
  + Договорах;
* Обеспечить признак того что автомобиль находится в прокате

# 3. Проектирование БАЗЫ ДАННЫХ «ПРОКАТ АВТОМОБИЛЕЙ»

## 3.1. Визуальное моделирование

Под моделью ПО (Программного обеспечения) в общем случае понимается формализованное описание системы ПО на определенном уровне абстракции. Каждая модель определяет конкретный аспект системы, использует набор диаграмм и документов заданного формата, а также отражает точку зрения и является объектом деятельности различных людей с конкретными интересами, ролями или задачами.

Графические (визуальные) модели представляют собой средства для визуализации, описания, проектирования и документирования архитектуры системы. Разработка модели системы ПО промышленного характера в такой же мере необходима, как и наличие проекта при строительстве большого здания. Это утверждение справедливо как в случае разработки новой системы. Хорошие модели являются основой взаимодействия участников проекта и гарантируют корректность архитектуры. Поскольку сложность систем повышается, важно располагать хорошими методами моделирования. Хотя имеется много других факторов, от которых зависит успех проекта, но наличие строгого стандарта языка моделирования является весьма существенным.

Состав моделей, используемых в каждом конкретном проекте, и степень их детальности в общем случае зависят от следующих факторов:

* сложности проектируемой системы;
* необходимой полноты ее описания;
* знаний и навыков участников проекта;
* времени, отведенного на проектирование.

# 4. Разработка БАЗЫ ДАННЫХ «ПРОКАТ АВТОМОБИЛЕЙ»

## 4.1. Создание таблиц

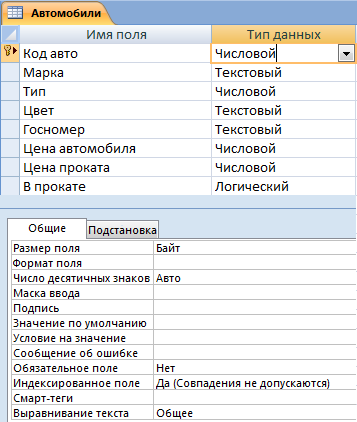
Создаваемая информационная система хранит информацию об автомобилях и прокатах, хранит сведения о клиентах и договорах, о прокате автомобилей. Основными элементами информационной системы являются база данных и приложение для работы с ней. Приложение, как правило, включает формы для ввода данных, запросы, отчеты и другие элементы.

Основной частью создаваемой информационной системы является база данных. База данных получила имя «Прокат авто»

На примере таблицы Автомобили рассмотрим структуру таблиц.

Таблица Автомобили содержит информацию обо всех автомобилях, которые находятся в автопарке фирмы. На рис. 1 изображена структура таблицы. Перечислим ее поля:

1. *Код авто* – это ключевое поле, у которого тип данных счетчик, а размер поля – байт;
2. *Марка* – это текстовое поле, размер поля 30 символов;
3. *Тип* – это поле внешнего ключа, т.е. из выпадающего списка можно выбрать значение которые хранятся в другой таблице;
4. *Цвет* – это текстовое поле, размер поля 30 символов;
5. *Госномер* – это текстовое поле, размер поля 6 символов;
6. *Цена автомобиля* – это числовое поле, формат поля денежный;
7. *Цена проката* – это числовое поле, формат поля денежный;
8. *В прокате* – это поле логического типа, формат поля да/нет;



1. Структура таблица *Автомобили*

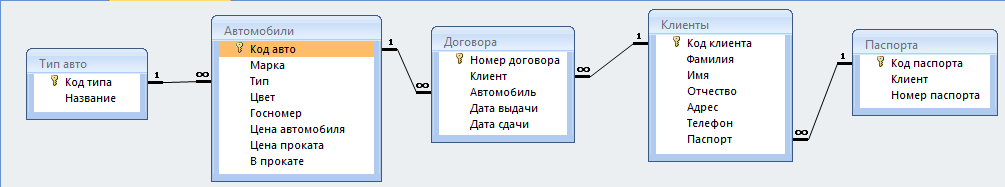
Схема данных является графическим образом БД. Она используется различными объектами Access для определения связей между несколькими таблицами. Схема данных обеспечивает автоматический согласованный доступ к полям таблиц. Она же обеспечивает целостность взаимосвязанных данных при корректировке таблиц.

После того, как созданы таблицы БД, можно установить связи между ними, обратившись к схеме данных. Запуск схемы данных выполняется через значок "Схема данных" http://lab314.brsu.by/roleg/BD_TiG/theory/images/access03_01.jpg на панели инструментов Access.

Связь между таблицами устанавливает отношения между совпадающими значениями в ключевых полях, обычно между полями, имеющими одинаковые имена в обеих таблицах. В большинстве случаев с ключевым полем одной таблицы, являющимся уникальным идентификатором каждой записи, связывается внешний ключ другой таблицы.

Обязательным условием при установлении связи является совпадение связываемых полей по типу и формату.

На рис. 2 изображена схема данных БД *Прокат автомобилей*

**

1. Схема данных БД Прокат автомобилей

## 4.2. Создание форм

Формы используют для реализации ввода, редактирования и просмотра данных в таблицах базы данных. Простые формы используют для работы с одной таблицей. Составные формы используют для работы с двумя таблицами.

В данной БД были созданы следующие формы:

1. *Главная форма*;
2. Форма *Автомобили*;
3. Форма *Договора*;
4. Форма *Клиенты*

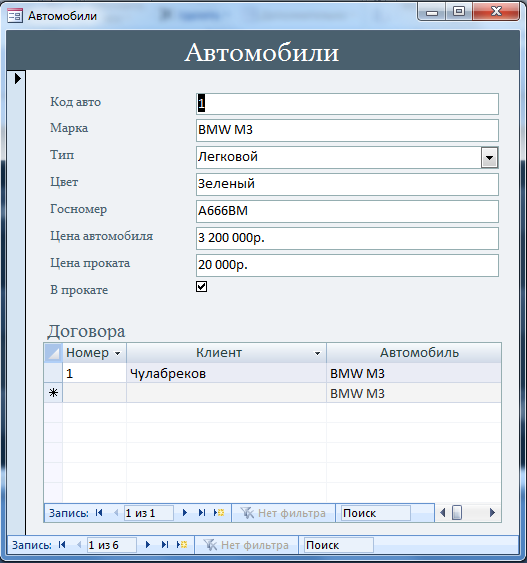
На главной форме (рис. 3) находятся кнопки перехода к другим формам.



1. Главная форма БД Прокат автомобилей

На рис. 4 изображена форма Автомобили. На ней находятся поля таблицы Автомобили. При добавлении нового автомобиля в БД, нужно:

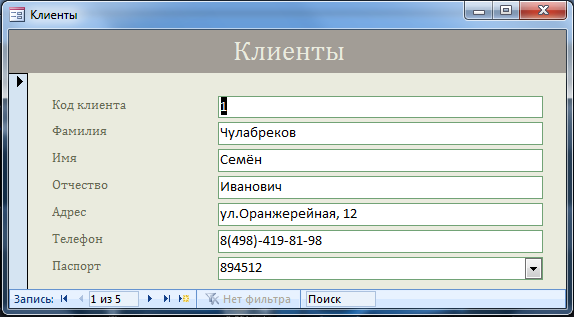
1. Указать в поле *Код авто* уникальное значение для нового автомобиля;
2. В поле *Марка* указать марку добавляемого автомобиля;
3. В поле *Тип* из выпадающего списка выбрать тип автомобиля;
4. В поле *Цвет* указать цвет автомобиля;
5. В поле *Госномер* указать номера автомобиля;
6. В поле *Цена автомобиля* указать актуальную цену автомобиля;
7. В поле *Цена проката* указать цену в которую обойдется прокат данного автомобиля;
8. В поле *В прокате* указать находится автомобиль в прокате или нет



1. Форма Автомобили в БД Прокат автомобилей

На рис. 5 изображена форма Клиенты. На ней находится информация об клиентах которые берут автомобили в прокат. Для добавления нового клиента в БД, нужно:

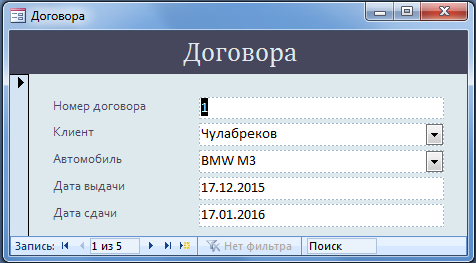
1. Указать в поле *Код клиента* уникальное значение для нового клиента;
2. В поле *Фамилия* указать фамилию клиента;
3. В поле *Имя* указать имя клиента;
4. В поле *Отчество* указать отчество клиента;
5. В поле *Адрес* указать адрес клиента;
6. В поле *Телефон* указать контактный телефон клиента;
7. В поле *Паспорт* из выпадающего списка выбрать паспорт клиента;



1. Форма Клиенты в БД Прокат автомобилей

На рис. 6 изображена форма Договора. В ней находится информация об клиентах которые взяли автомобили в прокат и сроки возврата автомобиля в автопарк. Для добавления нового договора в БД Прокат автомобилей, нужно:

1. Указать в поле *Код договора* уникальное значение для нового договора;
2. В поле *Клиент* из выпадающего списка выбрать клиента;
3. В поле *Автомобиль* из выпадающего списка выбрать автомобиль;
4. В поле *Дата выдачи* указать дату начала проката автомобиля;
5. В поле *Дата сдачи* указать дату окончания проката автомобиля;



1. Форма Договора в БД Прокат автомобилей

## 4.3. Создание запросов

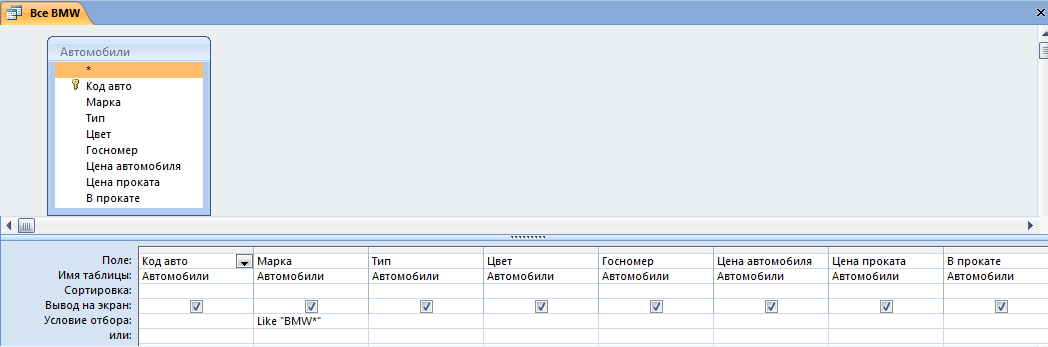
Запрос – это средство выбора необходимой информации из базы данных. Вопрос, сформированный по отношению к базе данных, и есть запрос. Применяются два типа запросов: по образцу QBE и структурированный язык запросов SQL.

QBE – запрос по образцу, средство для отыскания необходимой информации в базе данных. Он формируется не на специальном языке, а путем заполнения бланка запроса в окне Конструктора запросов.

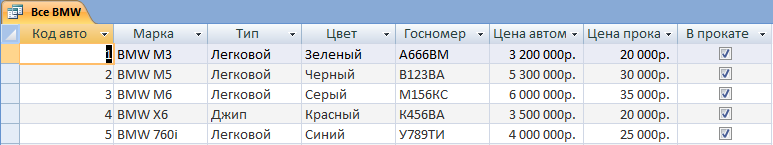
SQL **–** это запросы, которые составляются (программистами) из последовательности SQL инструкций. Эти инструкции задают, что надо сделать с входным набором данных для генерации выходного набора. Все запросы Access строит на основе SQL запросов, чтобы посмотреть их, необходимо в активном окне проектирования запроса выполнить команду Вид/SQL.

Существует несколько типов запросов: на выборку, на обновление, на добавление, на удаление, перекрестный запрос, создание таблиц. Наиболее распространенным является запрос на выборку. Запросы на выборку используются для отбора нужной пользователю информации, содержащейся в таблицах. Они создаются только для связанных таблиц.

На рис. 7 изображен запрос в режиме конструктора на выборку автомобилей, у которых марка начинается с *BMW.* На рис. 8 приведен пример выполнения данного запроса.

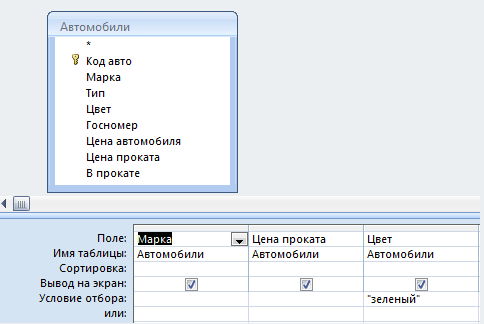


1. Запрос в режиме конструктора на выборку автомобилей BMW

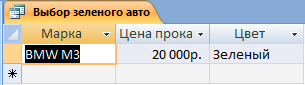
**

1. Пример выполнения запроса на выборку автомобилей BMW

На рис. 9 изображен запрос в режиме конструктора на выборку автомобилей, у которых цвет зеленый. На рис. 10 приведен пример выполнения данного запроса.

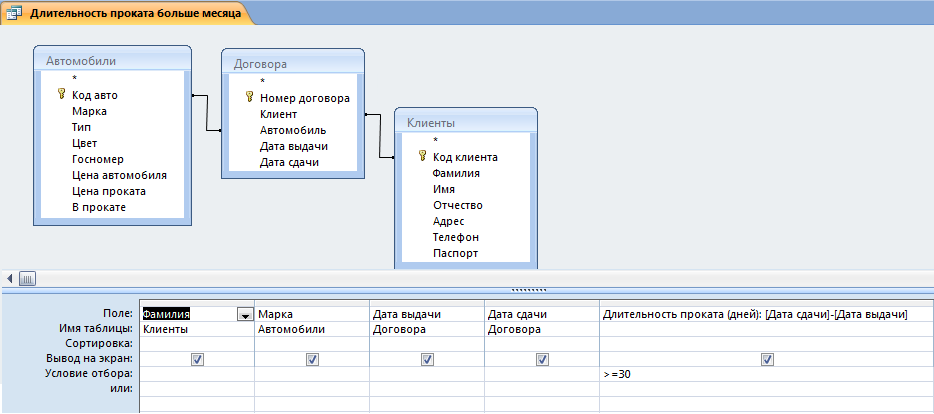


1. Запрос в режиме конструктора на выборку автомобилей зеленого цвета

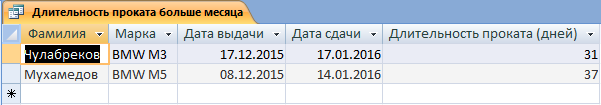


1. Пример выполнения запроса на выборку автомобилей зеленого цвета

На рис. 11 изображен запрос в режиме конструктора на выборку автомобилей и клиентов, у которых длительность проката автомобиля больше месяца. На рис. 12 приведен пример выполнения данного запроса.

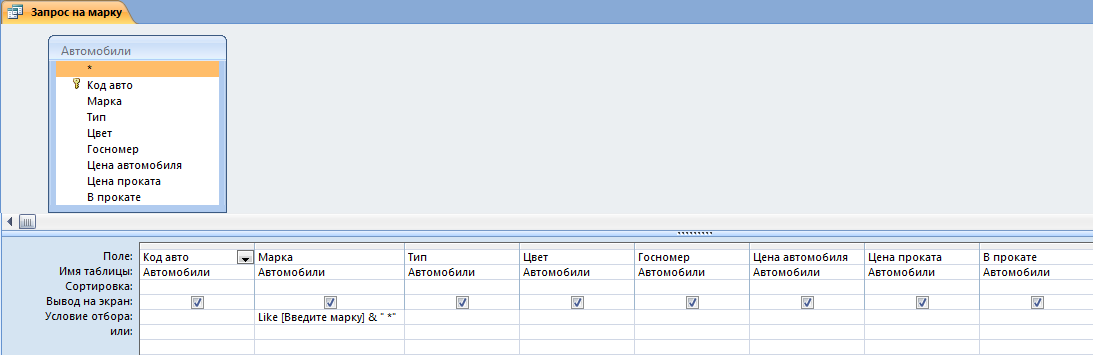


1. Запрос в режиме конструктора на выборку автомобилей и клиентов, у которых длительность проката автомобиля больше месяца

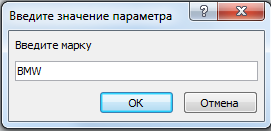


1. Пример выполнения запроса на выборку автомобилей и клиентов, у которых длительность проката автомобиля больше месяца

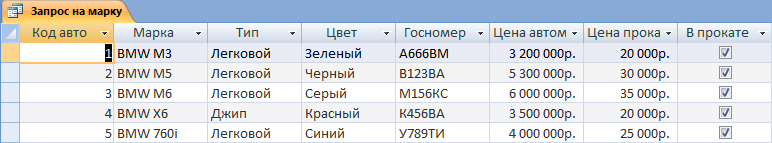
На рис. 13 изображен запрос в режиме конструктора. При запуске запроса (рис. 14) появляется окно с просьбой ввести марку автомобиля. На рис. 15 приведен пример выполнения данного запроса.



1. Запрос в режиме конструктора, на выборку автомобилей по марке введенной пользователем

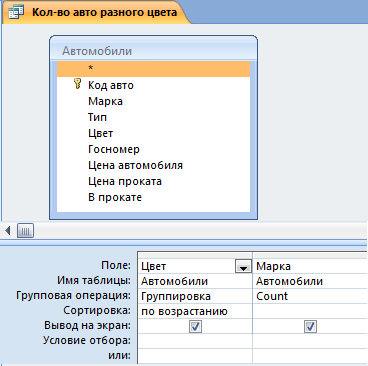


1. Всплывающее окно для ввода марки автомобиля

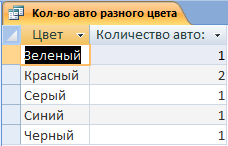


1. Пример выполнения запроса на выборку автомобилей по марке введенной пользователем

На рис. 16 изображен запрос в режиме конструктора. В нем подсчитывается сколько машин разного цвета. На рис. 17 изображен пример выполнения данного запроса.

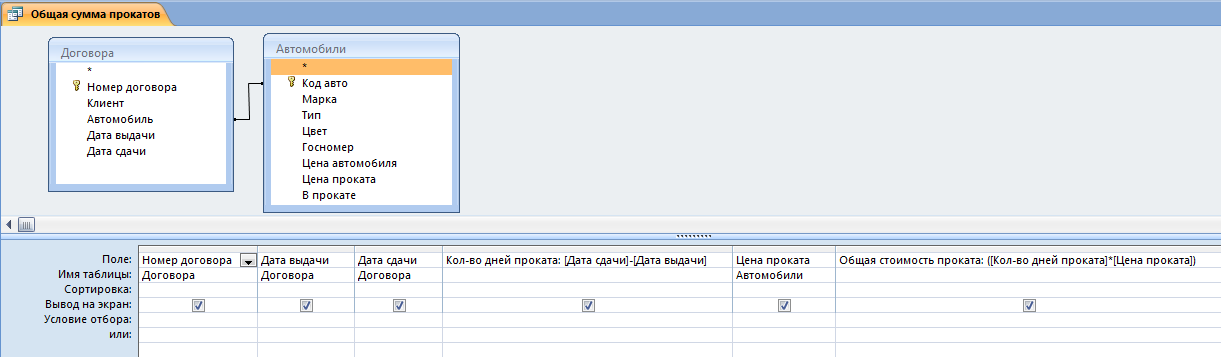


1. Запрос в режиме конструктора, на подсчет автомобилей разного цвета

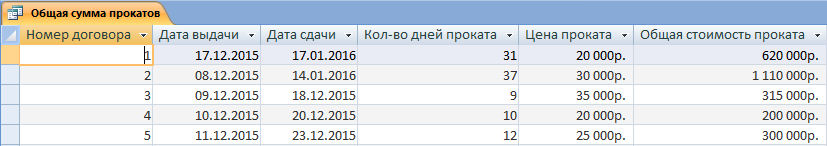


1. Пример выполнения запроса на подсчет автомобилей разного цвета

На рис. 18 изображен запрос в режиме конструктора. В нем подсчитывается общая сумма затраченная на прокат автомобилей. На рис. 19 изображен пример выполнения данного запроса.



1. Запрос в режиме конструктора, на подсчет общей суммы затраченной на прокат автомобиля



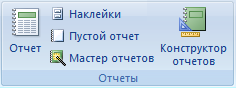
1. Пример выполнения запроса на подсчет общей суммы затраченной на прокат автомобиля

## 4.4. Создание отчетов

Отчет – это объект базы данных, который используется для вывода на экран, в печать или файл структурированной информации. Отчеты позволяют извлечь из таблиц или запросов базы данных необходимую информацию и представить ее в виде удобном для восприятия. Отчет содержит заголовок, область данных, верхний и нижний колонтитулы, примечание и разбит на страницы.

В Microsoft Access 2007 для создания отчетов можно использовать различные средства (рис. 20):

* Мастер отчетов
* Конструктор отчетов
* Инструмент Отчет
* Пустой отчет



1. Панель создания отчетов

Отчеты целесообразно выполнять с помощью *Мастера* или других указанных инструментов, а дорабатывать их, т.е. вносить необходимые изменения можно в режиме макета или конструктора. В Microsoft Access 2007 предусмотрено два режима внесения изменений и дополнений в отчет: режим макета и режим конструктора.

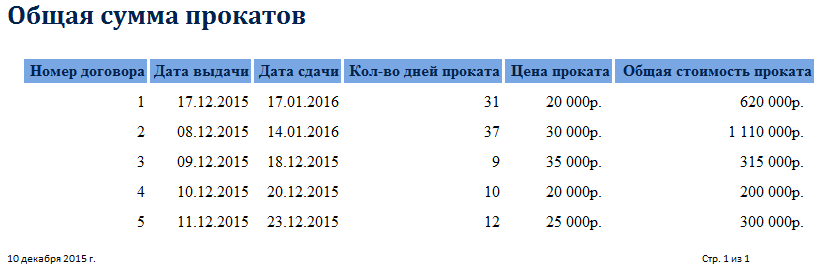
Режим макета – это более наглядный режим редактирования и форматирования (изменения) отчетов, чем режим конструктора. В тех случаях, когда в режиме макета невозможно выполнить изменения в отчете, то целесообразно применять режим конструктора.

На рис 21 приведен фрагмент выполнения отчета *Автомобили.*



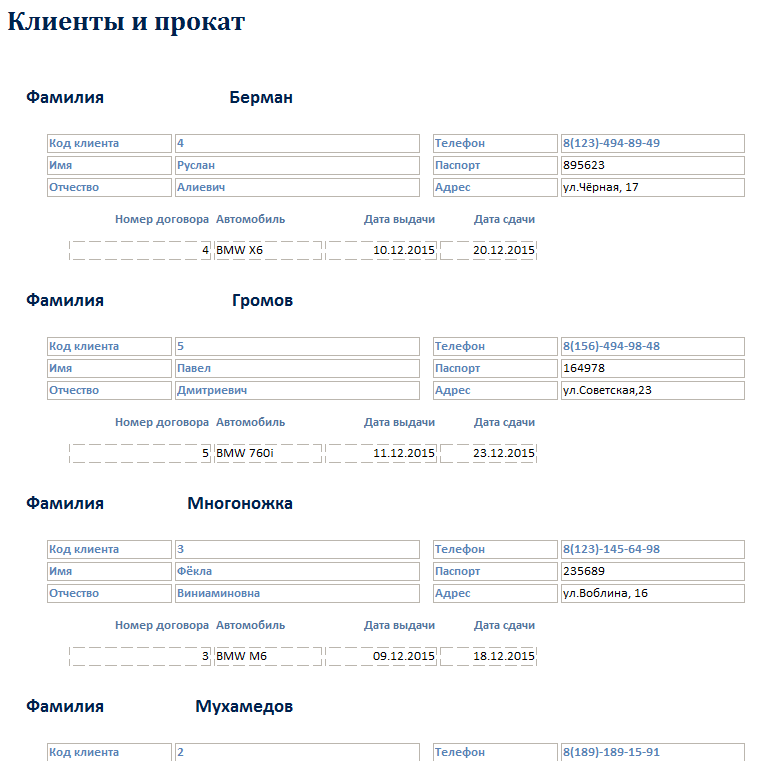
1. Фрагмент отчета автомобили

На рис 22 приведен фрагмент выполнения отчета *Общая сумма прокатов.*



1. Фрагмент отчета общая сумма прокатов

На рис 23 приведен фрагмент выполнения отчета *Клиенты.*



1. Фрагмент отчета клиенты

# 5. Тестирование БАЗЫ ДАННЫХ «ПРОКАТ АВТОМОБИЛЕЙ»

Тестирование программного обеспечения – проверка соответствия между реальным и ожидаемым поведением программы, осуществляемая на конечном наборе тестов, выбранном определенным образом.

В более широком смысле, тестирование – это одна из техник контроля качества, включающая в себя активности по планированию работ, проектированию тестов, выполнению тестирования и анализу полученных результатов.

Верификация – это процесс оценки системы или её компонентов с целью определения удовлетворяют ли результаты текущего этапа разработки условиям, сформированным в начале этого этапа тестирования. Т.е. выполняются ли наши цели, сроки, задачи по разработке проекта, определенные в начале текущей фазы.

Валидация – это определение соответствия разрабатываемого ПО ожиданиям и потребностям пользователя, требованиям к системе.

Тестирование на разных уровнях производится на протяжении всего жизненного цикла разработки и сопровождения программного обеспечения.

Уровень тестирования определяет то, над чем производятся тесты: над отдельным модулем, группой модулей или системой, в целом. Проведение тестирования на всех уровнях системы - это залог успешной реализации и сдачи проекта.

Различают несколько уровней тестирования:

1. [Компонентное или Модульное тестирование;](http://www.protesting.ru/testing/levels/component.html)
2. [Интеграционное тестирование;](http://www.protesting.ru/testing/levels/integration.html)
3. [Системное тестирование;](http://www.protesting.ru/testing/levels/system.html)
4. [Приемочное тестирование;](http://www.protesting.ru/testing/levels/acceptance.html)

# 6. Сопровождение БАЗЫ ДАННЫХ «ПРОКАТ АВТОМОБИЛЕЙ»

Программные средства являются одним из наиболее гибких видов промышленных изделий и эпизодически подвергаются изменениям в течение всего времени их использования.

Иногда достаточно при корректировке программного обеспечения внести только одну ошибку для того, чтобы резко снизилась его надежность или его корректность при некоторых исходных данных.

Для сохранения и повышения качества программного обеспечения необходимо регламентировать процесс модификации и поддерживать его соответствующим тестированием и контролем качества. В результате программное изделие со временем обычно улучшается как по функциональным возможностям, так и по качеству решения отдельных задач.

Работы, обеспечивающие контроль и повышение качества, а также развитие функциональных возможностей программ, составляют процесс сопровождения.

В процессе сопровождения в программное обеспечение вносятся следующие изменения, значительно различающиеся причинами и характеристиками;

* исправление ошибок – это корректировка программ, выдающих неправильные результаты в условиях, ограниченных техническим заданием и документацией. Исправление ошибок требуют около 20% общих затрат на сопровождение.
* регламентированная документами – это адаптация программного обеспечения к условиям конкретного использования, с учетом характеристик внешней среды или конфигурации аппаратуры, на которой предстоит функционировать программам. Адаптация занимает около 20% общих затрат на сопровождение.
* *Модернизация* – это расширение функциональных возможностей или улучшение характеристик решения отдельных задач в соответствии с новым или дополнительным техническим заданием на программное изделие. Модернизация занимает до 60% общих затрат на сопровождение.

На диаграмме ниже наглядно продемонстрировано какая часть сопровождения занимает какую долю.

# Заключение

На примере базы данных *Прокат автомобилей* мы познакомились с инструментом разработки баз данных Microsoft Access. С его помощью можно быстро создавать деловые приложения для различных сфер деятельности человека. В то же время СУБД Access имеет архитектурные ограничения (например, максимальный размер базы данных не более одного гигабайта), которые не позволяют использовать этот инструмент для управления большими промышленными распределенными базами данных. Для таких целей применяются клиент-серверные СУБД Oracle, IBM DB2, Microsoft SQL Server, Sybase и ряд других.

Microsoft Access – это функционально полная реляционная СУБД. В ней предусмотрены все необходимые средства для определения и обработки данных, а также для управления ими при работе с большими объектами информации. Области применения Access обозначены достаточно ясно. Во-первых, пользователи этой системы являются непрограммирующие персоналы-люди, близкие к вычислительной технике, но не имеющие достаточно времени на ее изучение, поскольку она лежит вне области их профессиональных интересов и служит лишь подспорьем в работе. Таких пользователей привлекает легкость изучения программы, возможность решить большинство проблем без программирования, а также средства быстрого создания приложений. Достоинством Access является то, что она имеет очень простой графический интерфейс, который позволяет не только создавать собственно базу данных, но и разрабатывать простые и сложные приложения. В отличие от других настольных СУБД Access хранит всю информацию в одном файле, хотя и распределяется и по разным таблицам.

# Список литературы

1. Базы данных: учебное пособие / Н.Ю. Братченко. Ставрополь: Сев-КавГТУ, 2011. 195 с.
2. Партыка Т.Л., Попов И.И. Информационная безопасность. Учебное пособие для студентов учреждений среднего профессионального образования. М. ИНФРА-М, 2014. 368с.
3. Информатика: Базовый курс / Под редакцией С.В. Симоновича, Издательский дом «Питер», 2015. 640с.
4. Завгородний В.И. Комплексная защита информации в компьютерных системах: Учебное пособие. М. Егоров Н.А. 2015. 264с.
5. Мельников В.В. Защита информации в компьютерных системах. М.Финансы и статистика. Электронинформ. 2015. 368с.
6. Информатика: Базовый курс. С.В. Симонович и др. СПб. Питер. 2015. 458с.