****

Минобрнауки России

федеральное государственное бюджетное образовательное учреждение

высшего образования

«Санкт-Петербургский государственный технологический институт

(технический университет)»

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Направление | | | **27.03.03** | Системный анализ и управление | |
| Направленность | | | | Системный анализ и управление в химической технологии | |
| Факультет | | | | Информационных технологий и управления | |
| Кафедра | | | | Системного анализа и информационных технологий | |
| ***Учебная дисциплина*** | | | | ***Базы данных*** | |
| Курс | 2 |  | | Группа | 496 |

**КУРСОВАЯ РАБОТА**

**Тема:**

**Разработка реляционной базы данных автоматизации учета заказов**

Студент \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Васильев В.В.

Руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Мамаева Г.А.

Оценка за курсовую работу \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись руководителя)

Санкт-Петербург

2020

****

Минобрнауки России

федеральное государственное бюджетное образовательное учреждение

высшего образования

«Санкт-Петербургский государственный технологический институт

(технический университет)»

**ЗАДАНИЕ НА КУРСОВУЮ РАБОТУ**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Направление | | | **27.03.03** | Системный анализ и управление | | |
| Направленность | | | | Системный анализ и управление в химической технологии | | |
| Факультет | | | | Информационных технологий и управления | | |
| Кафедра | | | | Системного анализа и информационных технологий | | |
| ***Учебная дисциплина*** | | | | **Базы данных** | | |
| Курс | 2 |  | | Группа | 496 | |
| Студент | | | | Васильев Владислав Валерьевич | |

**Тема:**Разработка базы данных автоматизации рабочих процессов детского развлекательного комплекса

**Исходные данные к работе (источники):**

1. Базы данных : учебное пособие / В. И. Халимон, Г.А. Мамаева, А.Ю. Рогов, В.Н. Чепикова; СПбГТИ(ТУ). Каф. систем. анализа и информ. технологий. – Электрон. текстовые дан. – СПб. : СПбГТИ(ТУ), 2017. – 217 с. (ЭБ).
2. Мамаева, Г.А. Система управления базами данных Microsoft Access : Учебное пособие / Г. А. Мамаева, В. Н. Чепикова ; СПбГТИ(ТУ). Каф. систем. анализа и информ. технологий. – Электрон. текстовые дан. – СПб. СПбГТИ(ТУ), 2018. – 52 с. (ЭБ).
3. Волк, В.К. Базы данных. Проектирование, программирование, управление и администрирование : Учебник / В. К. Волк. – Электрон. текстовые дан. – СПб: Лань, 2019. – 244 с. (ЭБС Лань).
4. Советов, Б.Я. Базы данных. Теория и практика : Учебник для вузов по направ. "Информатика и вычислительная техника" и "Информационные системы" / Б.Я. Советов, В. В. Цехановский, В. Д. Чертовской. – 2 –е изд., стер. – М. : Высш. шк., 2007. – 463 с.

***Перечень вопросов, подлежащих разработке***

**Введение**

1 **Аналитический обзор**

2 **Цели и задачи работы**

3 **Основная часть**

3.1 Проектирование базы данных

3.1.1. Этапы проектирования

3.1.2. Анализ предметной области

3.1.3. Построение инфологической (концептуальной) модели предметной   
области

3.1.4. Выбор СУБД

3.1.5. Даталогическое проектирование

3.1.5.1. Приведение схемы к 3NF

3.1.5.2. Разработка схемы данных

3.1.6. Физическое проектирование

3.2 Разработка базы данных средствами СУБД MS Access

3.2.1 Описание таблиц

3.2.2 Схема данных

3.2.3 Разработка запросов

3.2.4 Разработка пользовательских форм

3.2.5 Разработка отчетов

3.2.6 Разработка макросов

3.3 Руководство пользователю базой данных

3.3.1 Назначение и возможности базы данных

3.3.2 Правила и порядок работы с базой данных

4 **Выводы по работе**

Список литературных источников

***Требования к аппаратному и программному обеспечению***

Аппаратное обеспечение: PC на базе микропроцессора Intel Core 2 Duo (3 ГГц), ОЗУ 4 Гб, НЖМД 500 Гб, монитор ЖК (17″), клавиатура, USB –порты, мышь.

Программное обеспечение: операционная система Microsoft Windows 7 Professional, СУБД MS Access 2010, текстовый процессор MS Word 2010.

***Требования к оформлению***

Курсовая работа должна быть оформлена в соответствии с требованиями СТО СПбГТИ 044 –2012 КС УКДВ. Виды учебных занятий. Курсовой проект. Курсовая работа. Общие требования

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Дата выдачи задания** | |  | **20.10.2020 г.** | | |
| **Срок представления к защите** | |  | **22.12.2020 г.** | | |
| **Зав. кафедрой** | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 22.12.2020 | | |  | д.т.н., профессор | |
| Мусаев А.А. | |
|  | (подпись, дата) | | |  |  | |
|  |  | | |  |  | |
|  |  | | |  |  | |
| **Руководитель,**  **Должность** | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_22.12.2020 | | |  | к.э.н., доцент  Мамаева Г.А. | |
|  | (подпись, дата) | | |  |  | |
| **Задание принял**  **к выполнению** | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 22.12.2020 | | |  | Васильев В.В. | |
|  | (подпись, дата) | | |  |  | |

**Содержание**

[Введение 3](#_Toc59496727)

[1 Аналитический обзор 4](#_Toc59496728)

[2 Цели и задачи работы 6](#_Toc59496729)

[3 Основная часть 7](#_Toc59496730)

[3.1 Проектирование базы данных 7](#_Toc59496731)

[3.1.1 Этапы проектирования 7](#_Toc59496732)

[3.1.2 Анализ предметной области 8](#_Toc59496733)

[3.1.3 Построение инфологической (концептуальной) модели предметной области 9](#_Toc59496734)

[3.1.4 Выбор СУБД 12](#_Toc59496735)

[3.1.5 Даталогическое проектирование 14](#_Toc59496736)

[3.1.6 Физическое проектирование 22](#_Toc59496737)

[3.2 Разработка базы данных средствами СУБД MS Access 24](#_Toc59496738)

[3.2.1 Описание таблиц 24](#_Toc59496739)

[3.2.2 Схема данных 25](#_Toc59496740)

[3.2.3 Разработка запросов 23](#_Toc59496741)

[3.2.4 Разработка пользовательских форм 25](#_Toc59496742)

[3.2.5 Разработка отчетов 28](#_Toc59496743)

[3.2.3.4 Разработка макросов 31](#_Toc59496744)

[3.3 Руководство пользователю базой данных 32](#_Toc59496745)

[3.3.1 Назначение и возможности базы данных 32](#_Toc59496746)

[3.3.2 Правила и порядок работы с базой данных 33](#_Toc59496747)

[4 Выводы по работе 34](#_Toc59496748)

[Список литературных источников 35](#_Toc59496749)

# Введение

Идея открытия автосервиса очень перспективна в наши дни. Автомобиль – это имущество, требующее постоянных сложений. Ежегодно в нашей стране увеличивается количество автомобильных средств, требующих своевременного ремонта и технического обслуживания. Предоставление услуг по ремонту автомобилей осуществляют автосервисы. Никто не застрахован от поломок машины, поэтому и услуги автосервиса актуальны для большинства россиян.  
 Сервисная деятельность подразделяется на большое количество направлений (в частности станции технического обслуживания). Структура сервисной деятельности должна в общих чертах повторять структуру потребностей человека, соответствовать ей. Востребованность автомастерских обусловлена сложностью конструкции транспортных средств, и как следствие необходимостью осуществления ремонта квалифицированными специалистами.

Целью моей курсовой работы является анализ деятельности предприятия по оказанию автомобильных услуг, внедрение информационных технологий в процесс работы автосервисов. Результатом выполнения работы является создание готовой информационной системы для учета доходности предприятия.

# 1 Аналитический обзор

**История БД**

Период развития баз данных – 70-е годы 20 века. Концепция баз данных широко распространяется благодаря повышению характеристик аппаратного обеспечения компьютеров. Идет успешное внедрение систем, поддерживающих иерархическую и сетевую структуры данных.

Период становления – начало 60-х – начало 70-х гг. В этот период появляется сам термин «база данных» и создается несколько первоначальных систем. Основой появления баз данных явилось предложение конца 50-х годов использовать файлы для хранения исходных данных. Основное требование к таким файловым системам – быть совместно используемым хранилищем данных. В последующем стало очевидным, что совместно используемые данные, должны обладать специфическими свойствами, в частности: независимость данных, отсутствие дублирования и противоречивости, контроль прав доступа к данным, эффективная техника доступа к данным, а также многие другие.  
 Период зрелости – 80-е годы. Реляционная модель получила полное теоретическое обоснование. Промышленные реляционные системы получают широкое распространение во всех сферах человеческой деятельности. Реляционные системы практически вытеснили с мирового рынка ранние СУБД иерархического и сетевого типа.

Появление персональных компьютеров сделал принципиальным вопрос удобства использования программ, что также относилось и к СУБД. На протяжении всего этого периода интенсивно развивается внешний интерфейс взаимодействия пользователей с базами данных.

Изначально базы данных разрабатывались для хранения и обработки символьной информации и традиционно использовались в таких сферах, как обработка экономической информации, статистика, банковское дело, системы резервирования, информационные системы различного направления.

Появление спроса к базам данных в нетрадиционных сферах их применения, системы автоматизации проектирования, издательское дело и другие, потребовали хранения в базах данных и обработки изображений, звуков, полнотекстовой информации.

Этот период также характеризуется теоретическими и экспериментальными исследованиями в области баз знаний. Разрабатываются многочисленные экспертные системы, использующие базы знаний. В подавляющем большинстве случаев базы знаний разрабатываются на основе реляционных СУБД.

**Виды БД и СУБД**

Система управления базами данных (СУБД) – это программное обеспечение, которое взаимодействует с конечными пользователями, приложениями и самой базой данных для сбора и анализа данных. Программное обеспечение СУБД дополнительно включает в себя основные средства, предоставляемые для администрирования базы данных.

СУБД обычно классифицируются в соответствии с моделями баз данных, которые они поддерживают. Реляционные СУБД стали доминировать в 1980-х годах. Такие модели управления можно охарактеризовать простотой, удобным табличным представлением и возможностью использования формального аппарата алгебры отношений и реляционного исчисления для обработки данных. Microsoft Access, SQL Server, Oracle, MySQL – все это реляционные СУБД. Реляционные СУБД используют язык структурированных запросов SQL для описания, изменения и извлечения данных.

В 2000-х годах стали популярны нереляционные базы данных, называемые NoSQL и NewSQL – горизонтально масштабируемые, распределенные и разрабатываемые в открытых кодах, в которых делается попытка решить проблемы масштабируемости и доступности за счёт атомарности и согласованности данных.

**СУБД MS Access**

Microsoft Access – система управления базами данных, которую фирма Microsoft неизменно включает в состав профессиональной редакции Microsoft Office. СУБД Access занимает одно из ведущих мест среди систем разработки баз данных для малого и среднего бизнеса.

Access характеризуется простотой, гибкостью, русификацией, наличием разнообразных мастеров, конструкторов, надежностью работы.

Access рекомендуется использовать для разработки простых приложений и персональных баз данных с ограниченным объемом (несколько сотен тысяч записей) информации для небольших предприятий.

# 2 Цели и задачи работы

Целью данной курсовой работы является проектирование базы данных автоматизированной информационной системы обеспечения деятельности автосервиса.

Задачи курсовой работы:

1. Анализ предметной области
2. Выделение сущностей и отношений между ними
3. Разработка инфологической и даталогической модели
4. Создание объектов базы данных в среде разработки
5. Разработка интерфейса
6. Оформление пояснительной записки и графического материала

# 3 Основная часть

## 3.1 Проектирование базы данных

### 3.1.1 Этапы проектирования

Проектирование баз данных – это процесс создания схемы базы данных и определения необходимых ограничений целостности. Основными задачами проектирования баз данных являются:

* Обеспечение хранения в БД всей необходимой информации.
* Обеспечение возможности получения данных по всем необходимым запросам.
* Сокращение избыточности и дублирования данных.
* Обеспечение целостности данных (правильности их содержания): исключение противоречий в содержании данных, исключение их потери и т.д.

Процесс проектирования БД представляет собой последовательность переходов от неформального словесного описания информационной структуры предметной области к формализованному описанию объектов предметной области в терминах некоторой модели.

В общем случае можно выделить следующие этапы проектирования (рис. 1):



Рисунок 1 — Этапы проектирования базы данных

### 3.1.2 Анализ предметной области

Предметная область информационной системы – это материальная система или система, характеризующая элементы материального мира, информация о которой хранится и обрабатывается. Предметная область рассматривается как некоторая совокупность реальных объектов и связей между ними.

Основными задачами для сервисных предприятий является обеспечение удовлетворенности клиентов как номенклатурой и самим процессов обслуживания, так и качеством выполняемых работ.

Рассмотрим типичные бизнес-процессы автосервиса на неавтоматизированном производстве – предприятие-автосервис.

Процедура оказания услуг клиенту:

1) Клиент приезжает в автосервис.

2) Мастер осматривает автомобиль и выявляет перечень необходимых услуг.

3) В случаи необходимости производится закупка недостающих деталей.

4) Производится ремонт автомобиля, с записью оказываемых услуг в документ.

5) На основе оказываемых услуг составляется счет на оплату ремонта.

6) Вся информация о заказе заносится в документы отчетности.

Проанализировав ситуацию в автосервисе и выявив все минусы, постараемся создать такую систему, которая бы автоматизировала следующие операции:

1. Регистрация заказа;
2. Поиск и редактирование заказа ;
3. Просмотри информации о клиенте и его автомобиле;
4. Формирование документов отчетности.

В результате вся работа с бумажными документами будет автоматизирована.

Автоматизация позволит сократить ручные операции, ускорить обработку информации, повысить точность расчетов.

Для реализации автоматизации нам необходимо разработать базу данных формирования и учета заказов на оказание услуг клиентами автосервиса. Создание такой базы необходимо для анализа спроса и объемов оказываемых услуг, учета доходов от оказания услуг за определенные периоды времени, изучения предпочтений клиентов.

Все это даст возможность нивелировать проблемы падения спроса и повысить эффективность работы автосервиса за счет получения своевременной информации для принятия правильных управленческих решений по организации его работы.

После детального анализа оказалось возможным выделить следующие сущности будущей базы данных:

ЗАКАЗЫ – сущность, содержащая информацию о клиентах, датах обращения на предприятия, видах оказываемых услуг и информацию о мастерах – исполнителях работ.

ДИРЕКТОРА – сущность, содержащая контактную информацию о директоре каждого предприятия (ООО «Везёт», ООО «Монтаж», ООО «Колесо»).

СПРАВОЧНИК ВИДОВ РАБОТ – сущность, содержащая список предоставляемых услуг автосервисами и информацию о них.

СПРАВОЧНИК – сущность, содержащая контактную информацию о предприятиях и исполнителях работ.

### 3.1.3 Построение инфологической (концептуальной) модели предметной области

Построение информационной модели предметной области предполагает определение атрибутов и первичных ключей, выделение сущностей и определение связей между сущностями.

Нотации – графические модели, которые используются, для отображения сущностей с их атрибутами и связей между сущностями.

В данной курсовой работе мы будем использовать нотацию Чена, которая предоставляет богатый набор средств моделирования данных, включая ERD диаграммы, а также диаграммы атрибутов и диаграммы категоризации.

Общепринятым видом графического изображения реляционной модели данных является ER-диаграмма, на которой сущности изображаются прямоугольниками, соединенные между собой связями. Такое графическое представление облегчает восприятие структуры базы данных по сравнению с текстовым описанием.

Основные преимущества ER-моделей:

* наглядность;
* модели позволяют проектировать базы данных с большим количеством объектов и атрибутов.
* ER-модели реализованы во многих системах автоматизированного проектирования баз данных.

Основные элементы ER-моделей:

* объекты (сущности);
* атрибуты объектов;
* связи между объектами;

Сущность – это множество экземпляров реальных или абстрактных объектов: человек, место, вещь, событие, состояние, концепция, идея, предмет и т.п., обладающих общими атрибутами или характеристиками, и о которых необходимо хранить информацию.

Связь – это функциональная зависимость между сущностями. Каждая сущность обладает атрибутами. Перечисленные выше сущности содержат различные атрибуты.

Атрибут – это свойство объекта, характеризующее его экземпляр.

Перечислим выделенные сущности и их атрибуты:

**ЗАКАЗЫ** (Дата, Номер заказа, Код предприятия, Марка автомобиля, Код работ, Код исполнителя)

**СПРАВОЧНИК** (Код предприятия, Наименование предприятия, Адрес предприятия, Телефон, Код исполнителя, ФИО)

**ДИРЕКТОРА ПРЕДПРИЯТИЙ** (Код предприятия, Фамилия, Имя, Отчество, Телефон, Электронная почта)

**СПРАВОЧНИК ВИДОВ РАБОТ** (Код работы, Марка автомобиля, Наименование работы, Стоимость работы)

Графически в модели Чена связь изображается в виде линии, связывающей две сущности.

Связь типа один к одному означает, что один экземпляр первой сущности (левой) связан с одним экземпляром второй сущности (правой).

Связь типа один ко многим означает, что один экземпляр первой сущности (левой) связан с несколькими экземплярами второй сущности (правой).

Связь типа многие ко многим означает, что каждый экземпляр первой сущности может быть связан с несколькими экземплярами второй сущности, и каждый экземпляр второй сущности может быть связан с несколькими экземплярами первой сущности.

В нашем случае все сущности соединяются классическими связями типа один ко многим.

Диаграмма сущность-связь предметной области предприятия автосервиса представлена на рисунке 2:

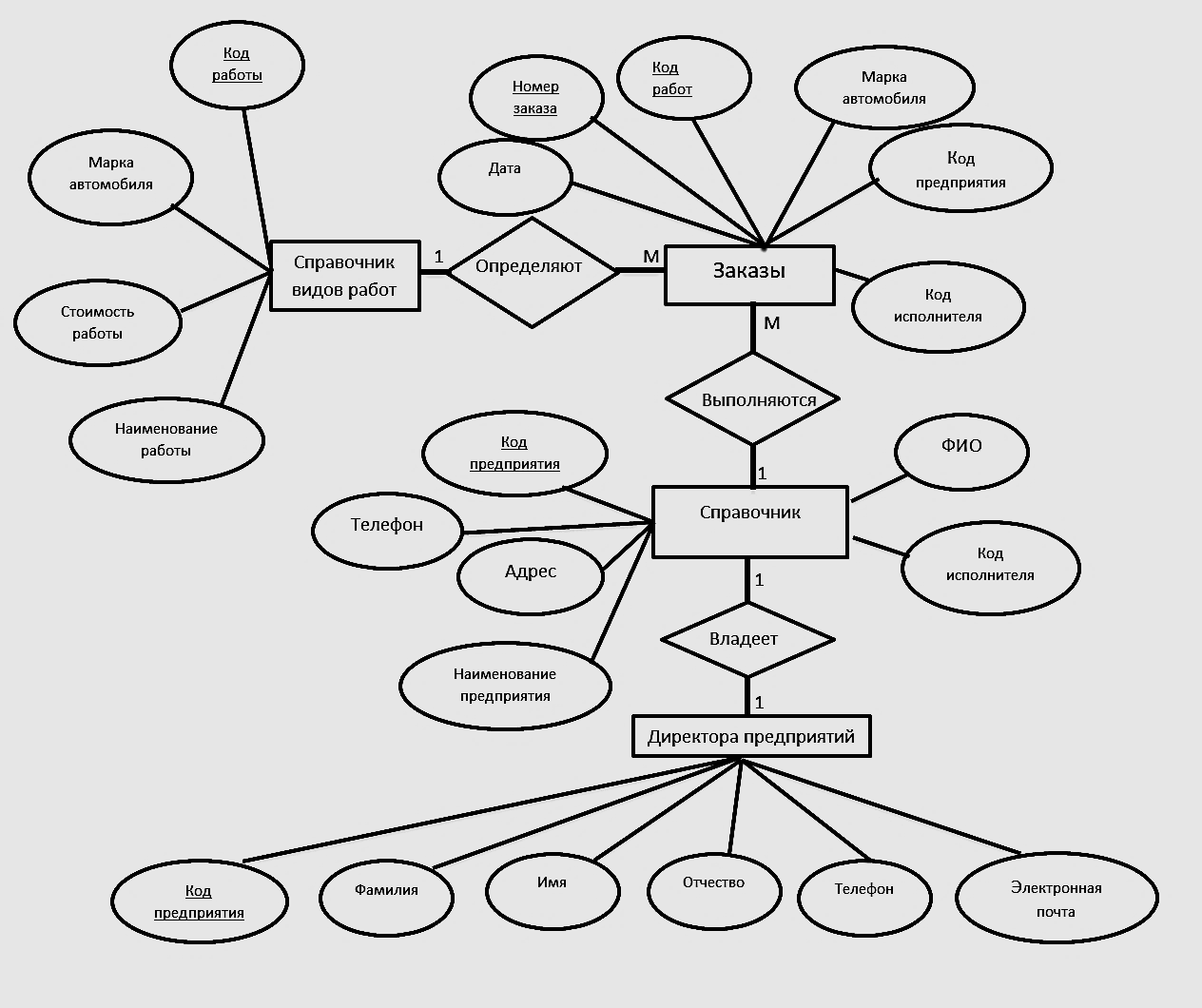


Рисунок 2 — ER-модель, переоформленная в системе обозначений Чена

### 3.1.4 Выбор СУБД

Перед тем, как приступить к определению логической структуры проектируемой БД, необходимо принять решение, с использованием какой стандартной СУБД будет реализовываться проект в дальнейшем.

Анализ информационных задач показывает, что для реализации требуемых функций проектируемой базы данных подходят почти все СУБД для ПЭВМ: SQL Server, Microsoft Access, Oracle, MySQL и т.д. Все они поддерживают реляционную модель данных и предоставляют разнообразные возможности для работы с данными.

Для учебного примера и с учетом относительно небольшой базы для учета заказов автосервиса рассмотрим возможность описания логической схемы БД и, в дальнейшем, реализации этого проекта в среде СУБД MS Access.

Обоснуем достоинства этой СУБД для нашего примера:

1. Доступность (наличие лицензионного офиса с предустановленной СУБД MS Access, тогда как для серверных CУБД типа MS SQL Server потребуются дополнительные расходы).

2. Access функционирует под управлением операционной системы Windows и обладает стандартизованным интерфейсом приложений Windows.

3. Взаимодействует почти со всеми продуктами MS Office.

4. Расположение всех объектов, которыми оперирует Access, в одном файле, позволяет без труда переносить программу на другие ПК.

5. Как реляционная СУБД, Access обеспечивает доступ ко всем типам данных и позволяет одновременно использовать несколько таблиц базы данных.

6. В MS Access есть богатые возможности по интерфейсу к базе данных, формы, различные элементы управления, отчёты, модули и макросы.

7. Access позволяет создавать и выполнять запросы на выборку, добавление данных, удаление и обновление. Запрос можно легко создавать в конструкторе как на QBE (Query By Example) - языке запросов по образцу, так и на SQL (Structured Query Language) - структурированном языке запросов.

8. СУБД достаточно простая, компактная, хорошо документирована.

9. Очень высокая скорость разработки, изменения и дополнения.

10. Нет необходимости в специальном администрировании.

11. Реализован доступ к БД практически из любого места, где можно воспользоваться веб-браузером.

12. Реализована надёжная система защиты от несанкционированного доступа к файлам.

14. Программой Access поддерживается механизм OLE (связывание и встраивание объектов) и механизм DDE (динамический обмен данными).

### 3.1.5 Даталогическое проектирование

#### 3.1.5.1 Приведение схемы данных к 3 НФ

Логическое (даталогическое) проектирование – отображение инфологической модели на модель данных, используемую в конкретной СУБД, например на реляционную модель данных.

Для реляционных СУБД даталогическая модель – набор таблиц, обычно с указанием ключевых полей, связей между таблицами.

Если инфологическая модель построена в виде ER-диаграмм (или других формализованных средств), то даталогическое проектирование представляет собой построение таблиц по определённым формализованным правилам, а также нормализацию этих таблиц. Этот этап может быть в значительной степени автоматизирован.

Определение состава базы данных:

При переходе от инфологической модели к даталогической следует иметь в виду, что инфологическая модель включает в себя всю информацию о предметной области, необходимую и достаточную для проектирования БД.

Основная цель логического проектирование базы данных – устранение избыточности и дублирования информации. Для удовлетворения этих потребностей Э.Коддом предложен аппарат нормализации отношений.

Нормализация отношений – это приведение отношений к виду, позволяющему устранить дублирование, обеспечить непротиворечивость данных, хранимых в БД, и уменьшить трудозатраты на ведение БД.

Согласно теории нормализации, выделяют шесть нормальных форм:

1. Первая нормальная форма (1NF)
2. Вторая нормальная форма (2NF)
3. Третья нормальная форма (3NF)
4. Нормальная форма Бойса-Кодда (BCNF)
5. Четвертая нормальная форма (4NF)
6. Пятая нормальная форма, форма проекции-соединения (5NF)

Следование каждой новой форме повышает нагрузку на систему, поэтому в большинстве случаев ограничиваются первыми тремя нормальными формами.

Первая нормальная форма (1NF). Отношение между таблицами называется приведённым к первой нормальной форме, если все его атрибуты являются атомарными.

Вторая нормальная форма (2NF). Отношение приведено ко второй нормальной форме, если оно находится в первой нормальной форме, и каждый неключевой атрибут функционально полно зависит от составного ключа.

Третья нормальная форма (3NF). Отношение находится в третьей нормальной форме, если оно находится во второй нормальной форме, и каждый неключевой атрибут не зависит от ключа транзитивно.

Транзитивная зависимость присутствует в отношении, если существует два неключевых поля, первое из которых зависит от ключа, а второе от первого.

База данных считается нормализованной, если ее таблицы представлены как минимум в третьей нормальной форме.

Согласно заданию, необходимо спроектировать базу данных для автоматизации учёта заказов автосервиса. Используя метод синтеза, целью которого является получение целого на основе описания составляющей его частей, нужно определить и описать сущности и атрибуты предметной области.

Построим первоначальную таблицу Заказы с атрибутами данной предметной области (таблица 1):

Таблица 1 – Заказы

|  |
| --- |
| Заказы |
| Номер заказа |
| Код работы  Повторяющиеся данные о работах. |
| Наименование работы |
| Стоимость работы |
| Марка автомобиля |
| Код предприятия  Повторяющиеся данные о предприятиях. |
| Наименование предприятия |
| Адрес предприятия |
| Телефон предприятия  Повторяющиеся данные об исполнителях работ. |
| Код исполнителя |
| ФИО исполнителя |
| Фамилия директора |
| Имя директора  Повторяющиеся данные о директорах предприятий. |
| Отчество директора |
| Телефон директора |
| Эл. почта директора |
| Дата |

Так как данные о работах повторяются, выделим их в отдельную сущность «Справочник видов работ» и свяжем ее по полю «Код работы» (таблица 2):

Таблица 2 – Декомпозиция таблицы «Заказы» на «Заказы» и «Справочник видов работ»

|  |
| --- |
| Заказы |
| Номер заказа  **М** |
| Код работ  **1** |
| Марка автомобиля |
| Код предприятия |
| Наименование предприятия |
| Адрес предприятия |
| Телефон предприятия |
| Код исполнителя |
| ФИО исполнителя |
| Фамилия директора |
| Имя директора |
| Отчество директора |
| Телефон директора |
| Эл. почта директора |
| Дата |

|  |
| --- |
| Справочник видов работ |
| Код работ |
| Наименование работы |
| Стоимость работы |
| Марка автомобиля |

Из таблицы «Заказы» нужно вычленить еще одну повторяющуюся сущность - «Справочник предприятий» и свяжем ее по полю «Код предприятия» (таблица 3):

Таблица 3 – Таблицы «Заказы», «Справочник видов работ», «Справочник предприятий» после декомпозиции.

|  |
| --- |
| Заказы |
| Номер заказа  **М** |
| Код работ |
| Марка автомобиля  **М** |
| Код предприятия  **1** |
| Код исполнителя |
| ФИО исполнителя |
| Фамилия директора |
| Имя директора |
| Отчество директора  **1** |
| Телефон директора |
| Эл. почта директора |
| Дата |

|  |
| --- |
| Справочник предприятий |
| Код предприятия |
| Наименование  предприятия |
| Адрес предприятия |
| Телефон предприятия |

|  |
| --- |
| Справочник видов работ |
| Код работ |
| Наименование работы |
| Стоимость работы |
| Марка автомобиля |

Вычленим из таблицы «Заказы» повторяющуюся сущность «Справочник исполнителей работ» и свяжем ее по полю «Код исполнителя» (таблица 4):

Таблица 4 – Таблицы «Заказы», «Справочник Видов работ», «Справочник предприятий», «Справочник исполнителей работ» после декомпозиции.

|  |
| --- |
| Заказы |
| Номер заказа  **М** |
| Код работ  **1** |
| Марка автомобиля  **М** |
| Код предприятия  **М** |
| Код исполнителя |
| Фамилия директора |
| Имя директора |
| Отчество директора  **1** |
| Телефон директора |
| Эл. почта директора |
| Дата |

|  |
| --- |
| Справочник предприятий |
| Код предприятия |
| Наименование  предприятия |
| Адрес предприятия |
| Телефон предприятия |

|  |
| --- |
| Справочник видов работ |
| Код работ |
| Наименование работы |
| Стоимость работы |
| Марка автомобиля |

|  |
| --- |
| Справочник исполнителей работ  **1** |
| Код исполнителя |
| ФИО исполнителя |

Вынесем информацию о директорах в отдельную таблицу и свяжем по полю «Код предприятия» с таблицей «Справочник предприятий» (таблица 5):

Таблица 5 – Таблицы «Заказы», «Справочник Видов работ», «Справочник предприятий», «Справочник исполнителей работ», «Директора предприятий» после декомпозиции.

|  |
| --- |
| Заказы |
| Номер заказа  **М** |
| Код работ |
| Марка автомобиля  **М**  **1**   |  | | --- | | Справочник видов работ | | Код работ | | Наименование работы | | Стоимость работы | | Марка автомобиля |   **1** |
| Код исполнителя  **М** |
| Код предприятия |
| Дата |

|  |
| --- |
| Справочник видов работ |
| Код работ |
| Наименование работы |
| Стоимость работы |
| Марка автомобиля |

|  |
| --- |
| Справочник исполнителей работ |
| Код исполнителя |
| ФИО исполнителя |

**1**

|  |
| --- |
| Справочник видов работ |
| Код работ |
| Наименование работы |
| Стоимость работы |
| Марка автомобиля |

**1**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Справочник предприятий  **1**   |  | | --- | | Справочник видов работ | | Код работ | | Наименование работы | | Стоимость работы | | Марка автомобиля |   **1** |
| Код предприятия |
| Наименование  предприятия |
| Адрес предприятия |
| Телефон предприятия |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Директора предприятий  **1**   |  | | --- | | Справочник видов работ | | Код работ | | Наименование работы | | Стоимость работы | | Марка автомобиля |   **1** |
| Фамилия директора |
| Имя директора |
| Отчество директора |
| Телефон директора |
| Эл. почта директора  **1**   |  | | --- | | Справочник видов работ | | Код работ | | Наименование работы | | Стоимость работы | | Марка автомобиля |   **1** |
| Код предприятия |
|  |

В ходе нормализации был проведен процесс реорганизации данных, путем ликвидации повторяющихся групп и иных противоречий с целью приведения таблиц к виду, позволяющему осуществлять корректное редактирование данных (3НФ). Таблицы на рисунке 5 соответствуют третьей нормальной форме, так как каждый не ключевой атрибут функционально полно зависит от составного ключа и не зависит от этого ключа транзитивно.

#### 3.1.5.2 Разработка схемы данных

На стадии даталогического проектирования выполнено описание логической структуры проектируемой базы данных, которое на языке СУБД называется даталогической моделью базы данных (рис. 3):

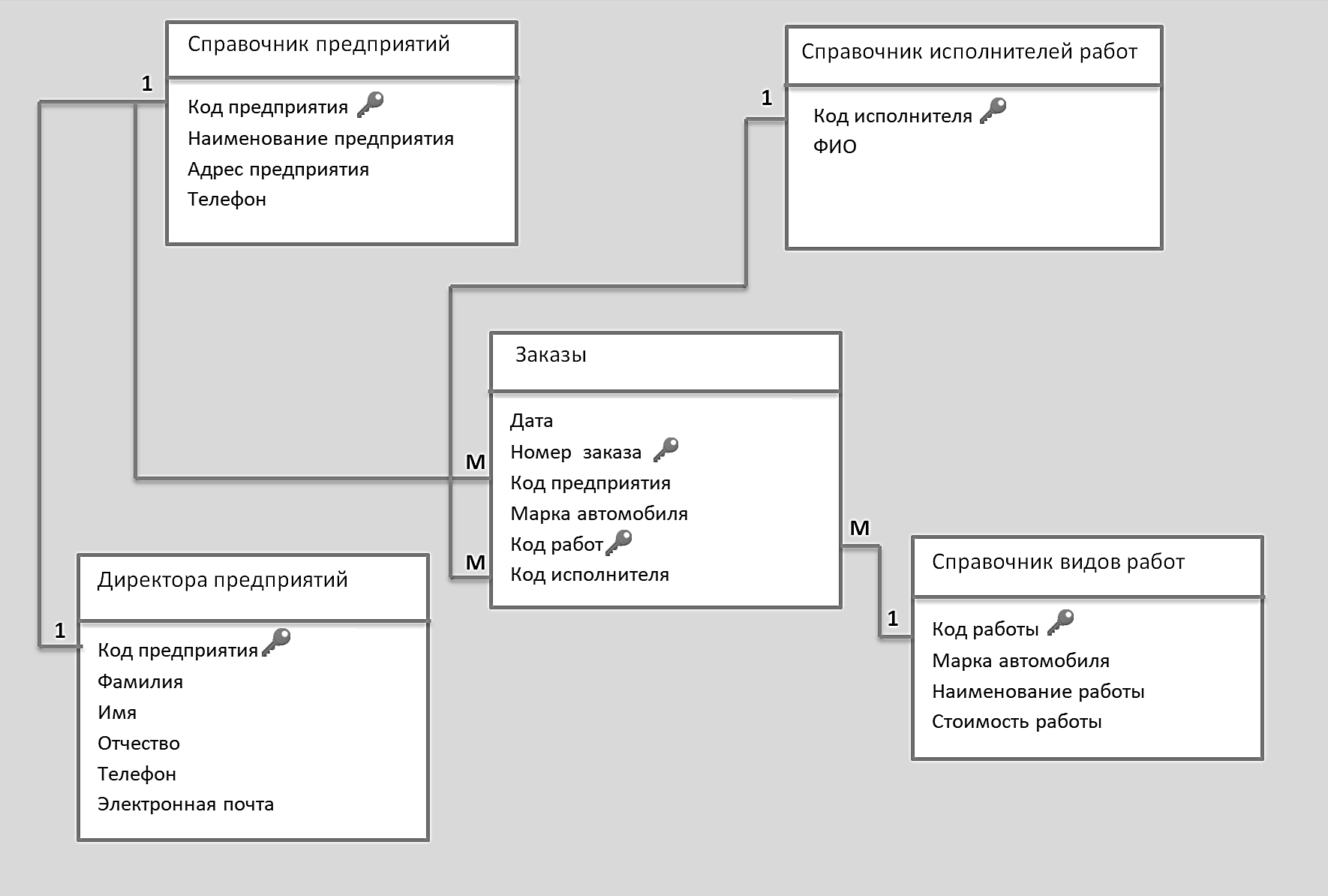


Рисунок 3 — даталогическая модель базы данных

### 3.1.6 Физическое проектирование

Физическое проектирование базы данных – процесс подготовки описания реализации базы данных на вторичных запоминающих устройствах; на этом этапе рассматриваются основные отношения, организация файлов и индексов, предназначенных для обеспечения эффективного доступа к данным, а также все связанные с этим ограничения целостности и средства защиты.

Физическое проектирование является третьим и последним этапом создания проекта базы данных, при выполнении которого проектировщик принимает решения о способах реализации разрабатываемой базы данных. Во время предыдущего этапа проектирования была определена логическая структура базы данных (которая описывает отношения и ограничения в рассматриваемой прикладной области). Хотя эта структура не зависит от конкретной целевой СУБД, она создается с учетом выбранной модели хранения данных, например реляционной, сетевой или иерархической. Однако, приступая к физическому проектированию базы данных, прежде всего необходимо выбрать конкретную целевую СУБД. Поэтому физическое проектирование неразрывно связано с конкретной СУБД. Между логическим и физическим проектированием существует постоянная обратная связь, так как решения, принимаемые на этапе физического проектирования с целью повышения производительности системы, способны повлиять на структуру логической модели данных.

Как правило, основной целью физического проектирования базы данных является описание способа физической реализации логического проекта базы данных.

В случае реляционной модели данных под этим подразумевается следующее:

* создание набора реляционных таблиц и ограничений для них на основе информации, представленной в глобальной логической модели данных;
* определение конкретных структур хранения данных и методов доступа к ним, обеспечивающих оптимальную производительность СУБД;
* разработка средств защиты создаваемой системы.

Этапы концептуального и логического проектирования больших систем следует отделять от этапов физического проектирования. На это есть несколько причин.

* + Они связаны с совершенно разными аспектами системы, поскольку отвечают на вопрос, что делать, а не как делать.
  + Они выполняются в разное время, поскольку понять, что надо сделать, следует прежде, чем решить, как это сделать.
  + Они требуют совершенно разных навыков и опыта, поэтому требуют привлечения специалистов различного профиля.

Физическое проектирование состоит из следующих этапов:

* + Перенос глобальной логической модели данных в среду целевой СУБД.
  + Реализация бизнес-правил предприятия в среде целевой СУБД.
  + Проектирование физического представления базы данных.
  + Анализ транзакций.
  + Выбор файловой структуры.
  + Определение вторичных индексов.
  + Анализ необходимости ввода контролируемой избыточности данных.
  + Определение требований к дисковой памяти.
  + Разработка механизмов защиты.
  + Определение прав доступа.
  + Организация мониторинга и настройка функционирования системы.

Физическое проектирование базы данных представляет собой процесс подготовки описания реализации этой базы во вторичной памяти. Создается описание таблиц базы данных и выбранных для них структур хранения, а также методов доступа, которые будут использоваться для эффективного доступа к данным. Высококачественный проект реализации таблиц базы данных может быть создан только разработчиками, хорошо знающими все функциональные возможности выбранной целевой СУБД.

## 3.2 Разработка базы данных средствами СУБД MS Access

### 3.2.1 Описание таблиц

Описание таблиц базы данных автосервис представлено в таблицах (6 – 10):

Таблица 6 — сущность Заказы

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование | Тип данных | Размер | Ключевые поля |
| Дата | Дата/время | Краткий формат даты |  |
| Номер заказа | Числовой | Длинное целое | \* |
| Код предприятия | Числовой | Длинное целое |  |
| Марка автомобиля | Текстовый | 30 |  |
| Код работ | Числовой | Длинное целое | \* |
| Код исполнителя | Числовой | Длинное целое |  |

Таблица 7 — сущность Справочник видов работ

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование | Тип данных | Размер | Ключевые поля |
| Код работы | Числовой | Длинное целое | \* |
| Марка автомобиля | Текстовый | 30 |  |
| Наименование работы | Текстовый | 30 |  |
| Стоимость работы | Денежный | 8 байт |  |

Таблица 8 — сущность Справочник предприятий

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование | Тип данных | Размер | Тип поля |
| Код предприятия | Числовой | Длинное целое | \* |
| Наименование предприятия | Текстовый | 30 |  |
| Адрес предприятия | Текстовый |  |  |
| Телефон | Числовой | Длинное целое |  |

Таблица 9 — сущность Справочник исполнителей

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование | Тип данных | Размер | Ключевые поля |
| Код исполнителя | Числовой | Длинное целое | \* |
| ФИО | Текстовый | 30 |  |

Таблица 10 — сущность Директора предприятий

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование | Тип данных | Размер | Ключевые поля |
| Код предприятия | Числовой | Длинное целое | \* |
| Фамилия | Текстовый | Длинное целое |  |
| Имя | Текстовый | 30 |  |
| Отчество | Текстовый | 30 |  |
| Телефон | Числовой | Длинное целое |  |
| Электронная почта | Текстовый | 30 |  |

### 3.2.2 Схема данных

После формирования структуры таблиц создается схема данных, в которой устанавливаются связи между таблицами. Схема данных проектируемой информационной системы представлена на рисунке 4.

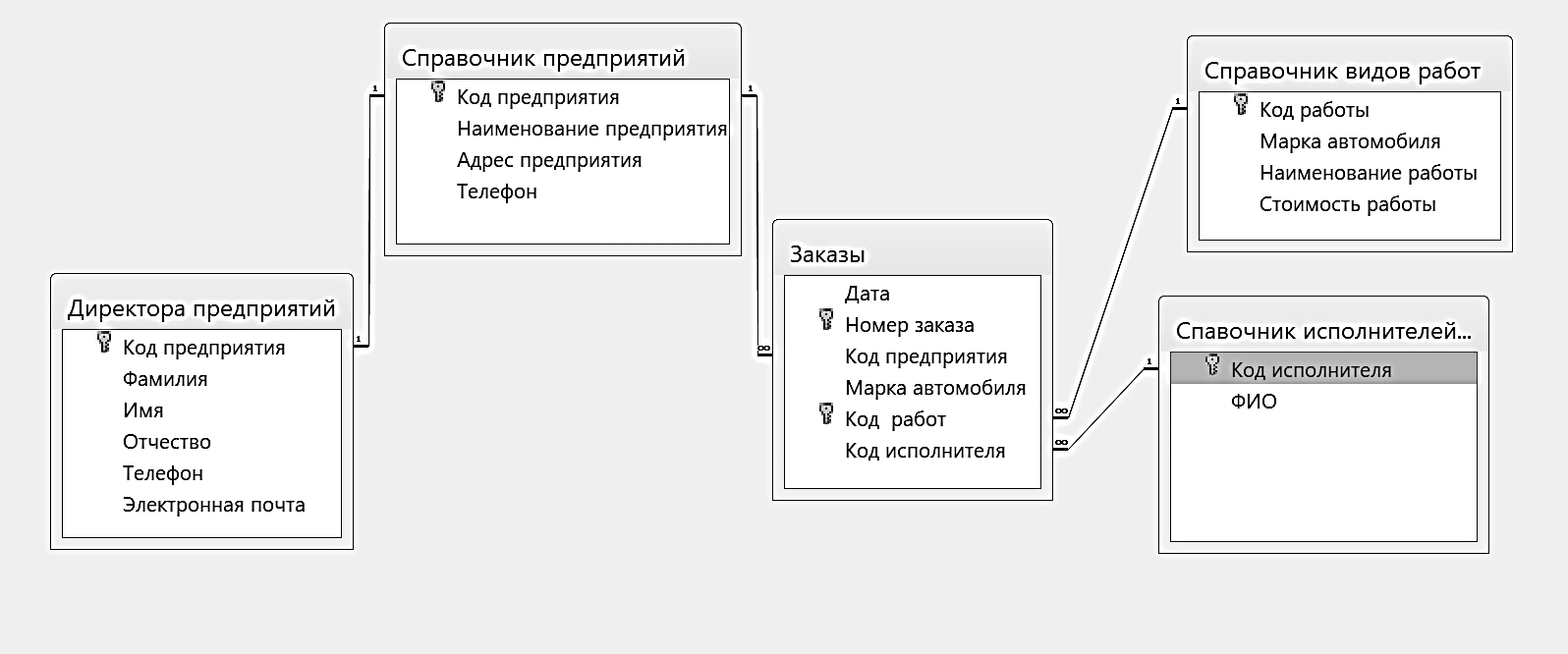


Рисунок 4 — схема данных базы данных автосервиса

### 3.2.3 Разработка запросов

Запросы – это объект базы данных, который служит для извлечения данных из таблиц и предоставления их пользователю в удобном виде. Особенность запросов состоит в том, что они черпают данные из базовых таблиц и создают на их основе временную таблицу.

Все запросы делятся на две группы: запросы -выборки, запросы -действия.

Запросы-выборки осуществляют выборку данных из таблиц в соответствии с заданными условиями.

Запросы-действия позволяют модифицировать данные в таблицах: удалять, обновлять, добавлять записи.

В данной БД представлены следующие запросы:

1. Запрос *Счет на оплату заказа* – заказ, выбираемый пользователем и подробная информация о нем (рис. 5, 6).

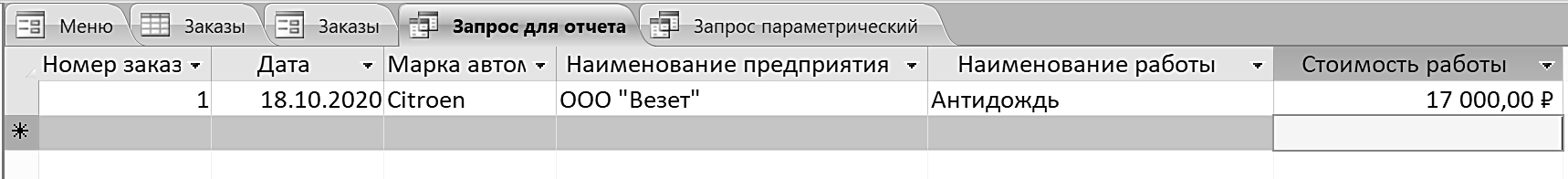


Рисунок 5 — Запрос: *счет на оплату заказа* в режиме таблицы

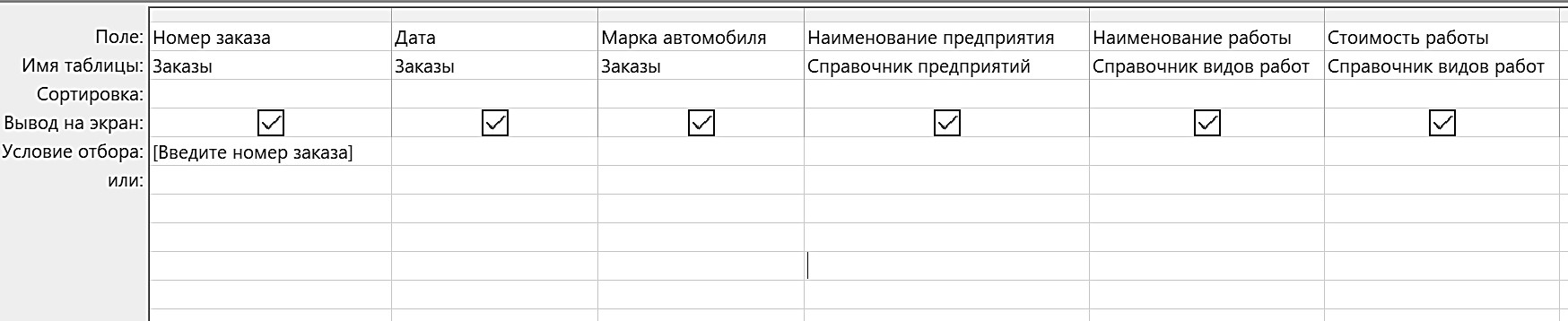


Рисунок 6 — Запрос: *счет на оплату заказа* в режиме конструктора

1. *Запрос для составления квартального отчета* – выбор заказов за определенный период времени и подробная информация о них (рис. 7, 8, 9).

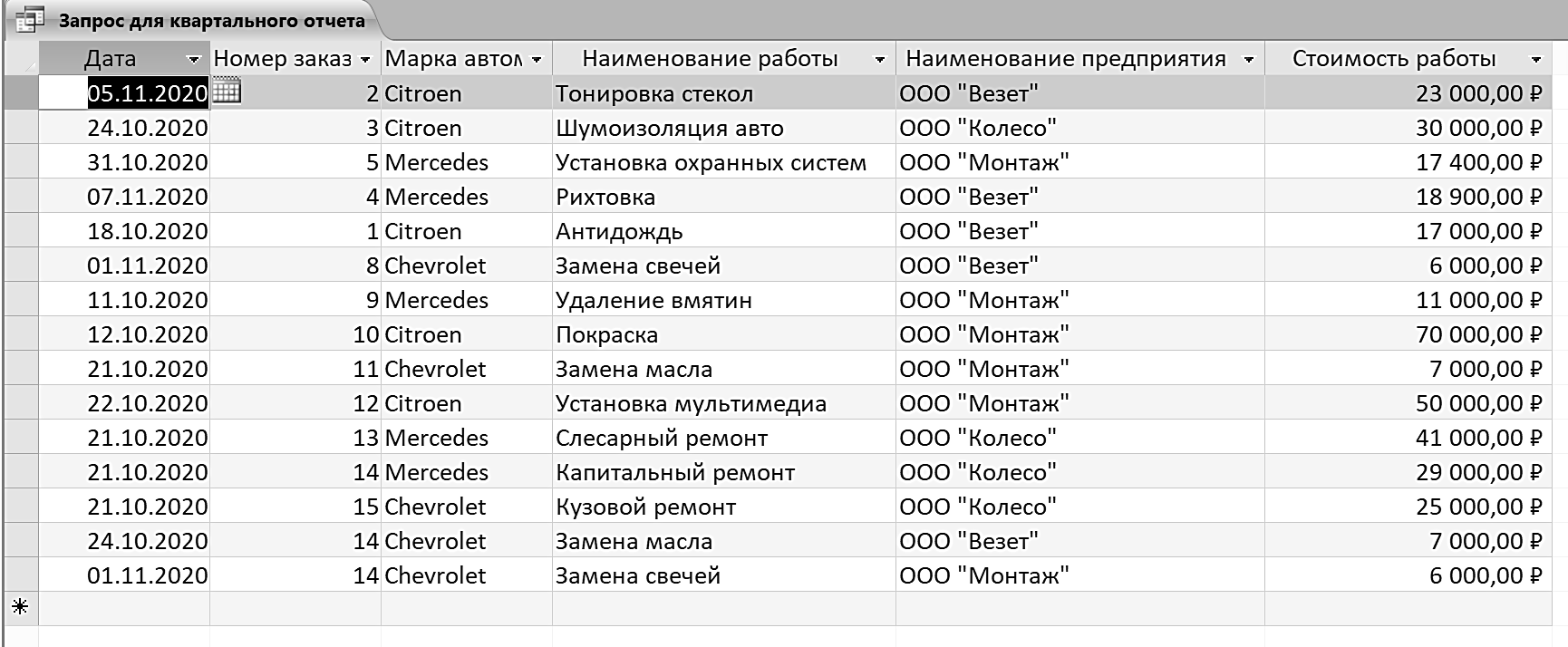


Рисунок 7 — *Запрос для квартального отчета* в режиме таблицы

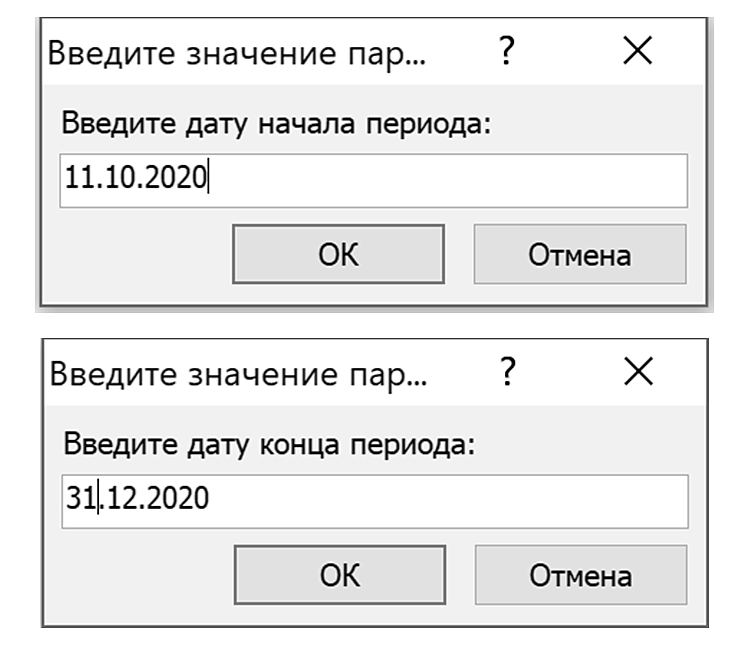


Рисунок 8 — Условия выбора периода

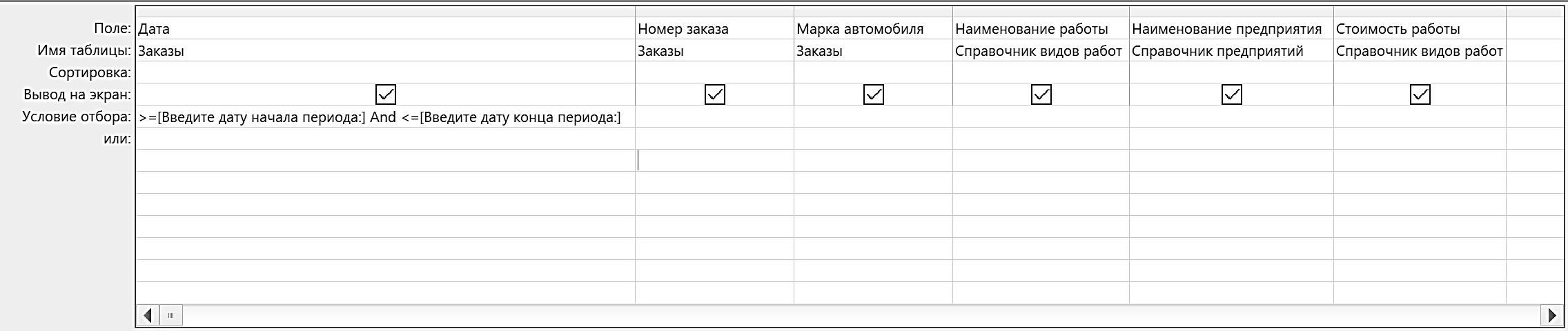


Рисунок 9 — *Запрос для квартального отчета* в режиме конструктора

### 3.2.4 Разработка пользовательских форм

Форма в БД – это структурированное окно, которое можно представить так, чтобы оно повторяло форму бланка. Формы создаются из набора отдельных элементов управления. Источником данных для формы являются записи таблицы или запроса.

Форма предоставляет возможности для:

1. Ввода и просмотра информации базы данных

2. Изменения данных

3. Печати

4. Создания сообщений.

В данной БД представлены следующие формы:

1. *Заказы* – предоставляет возможность внесения информации о новом заказе в сущность заказы, а так же внесение изменений в существующие заказы (рис. 10, 11).



Рисунок 10 — форма *Заказы*



Рисунок 11 — форма *Заказы* в режиме конструктора

1. *Справочник исполнителей работ* – предоставляет пользователю информацию о выполненных работах исполнителями работ и сумму заказов (рис. 12, 13).

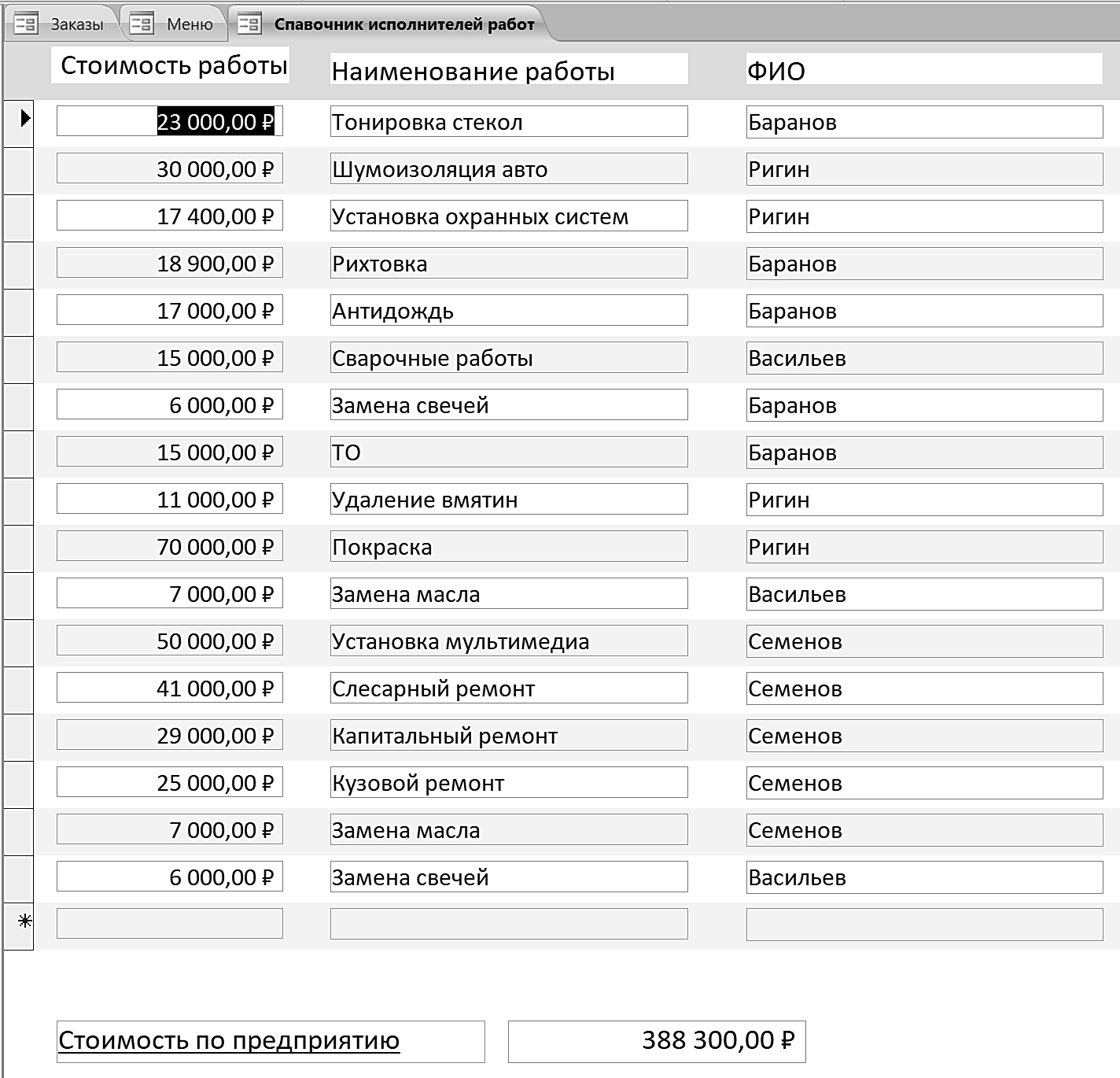


Рисунок 12 — форма *Справочник исполнителей работ*

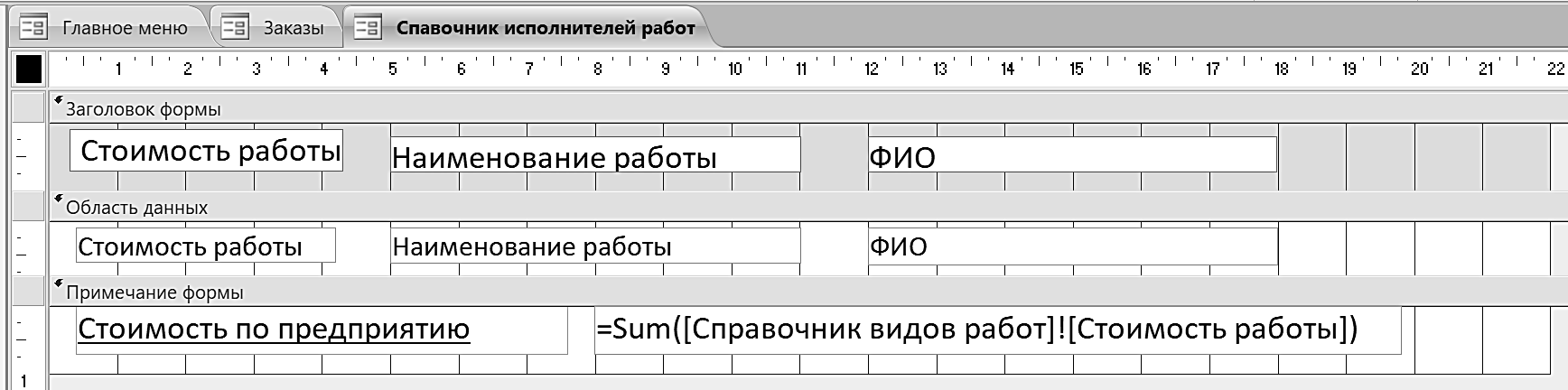


Рисунок 13 — форма *Справочник исполнителей работ* в режиме конструктора

*Справочник предприятий* – предоставляет пользователю возможность выбора предприятия – одного из трёх филиалов (ООО « Везёт», ООО «Колесо», ООО «Монтаж») и просмотра информации о выполненных работах, а так же доходность по предприятию, с помощью которого осуществляется создание отчета по предприятию. Форма создается на основе формы: справочник исполнителей работ (рис. 14, 15).

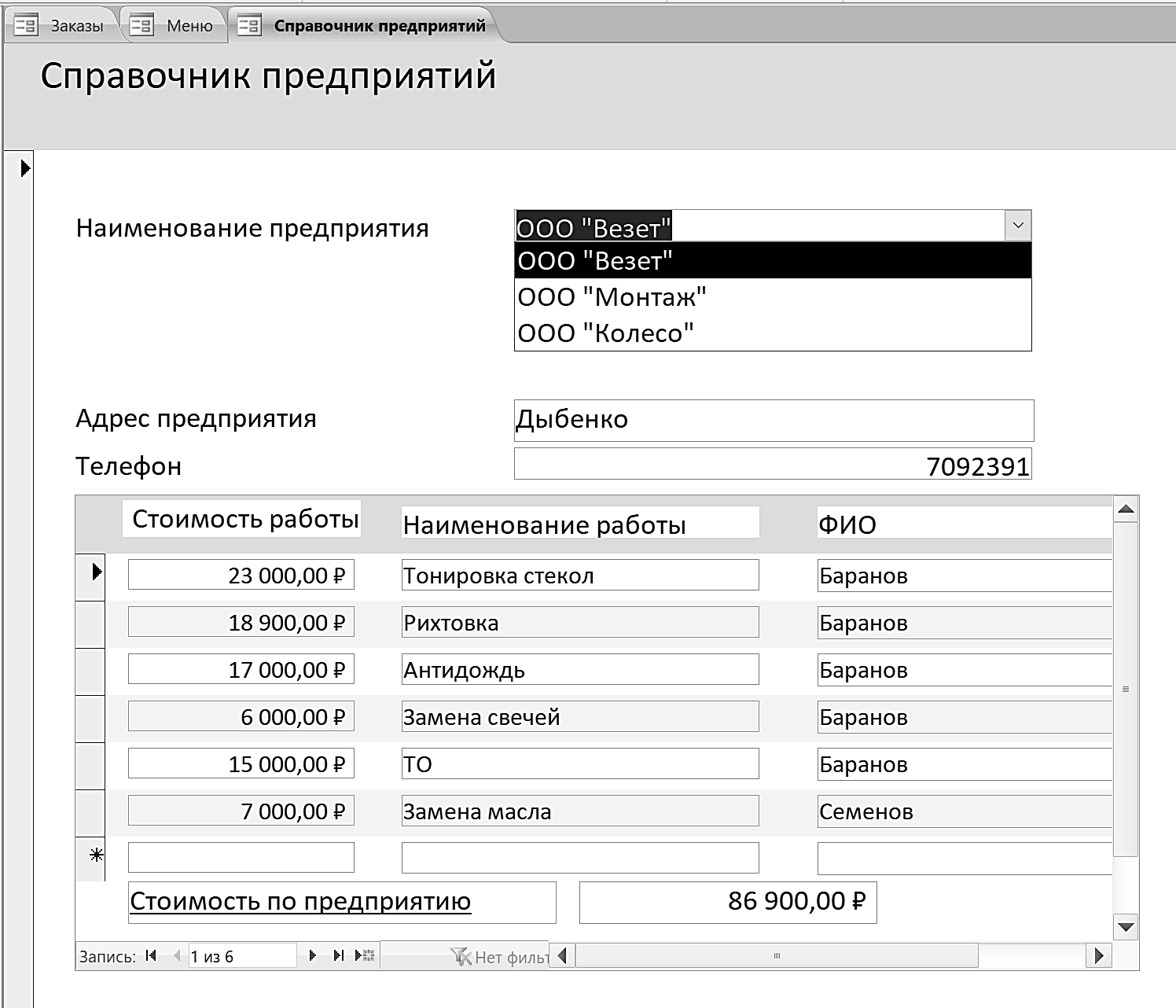


Рисунок 14 — форма: *Справочник предприятий*

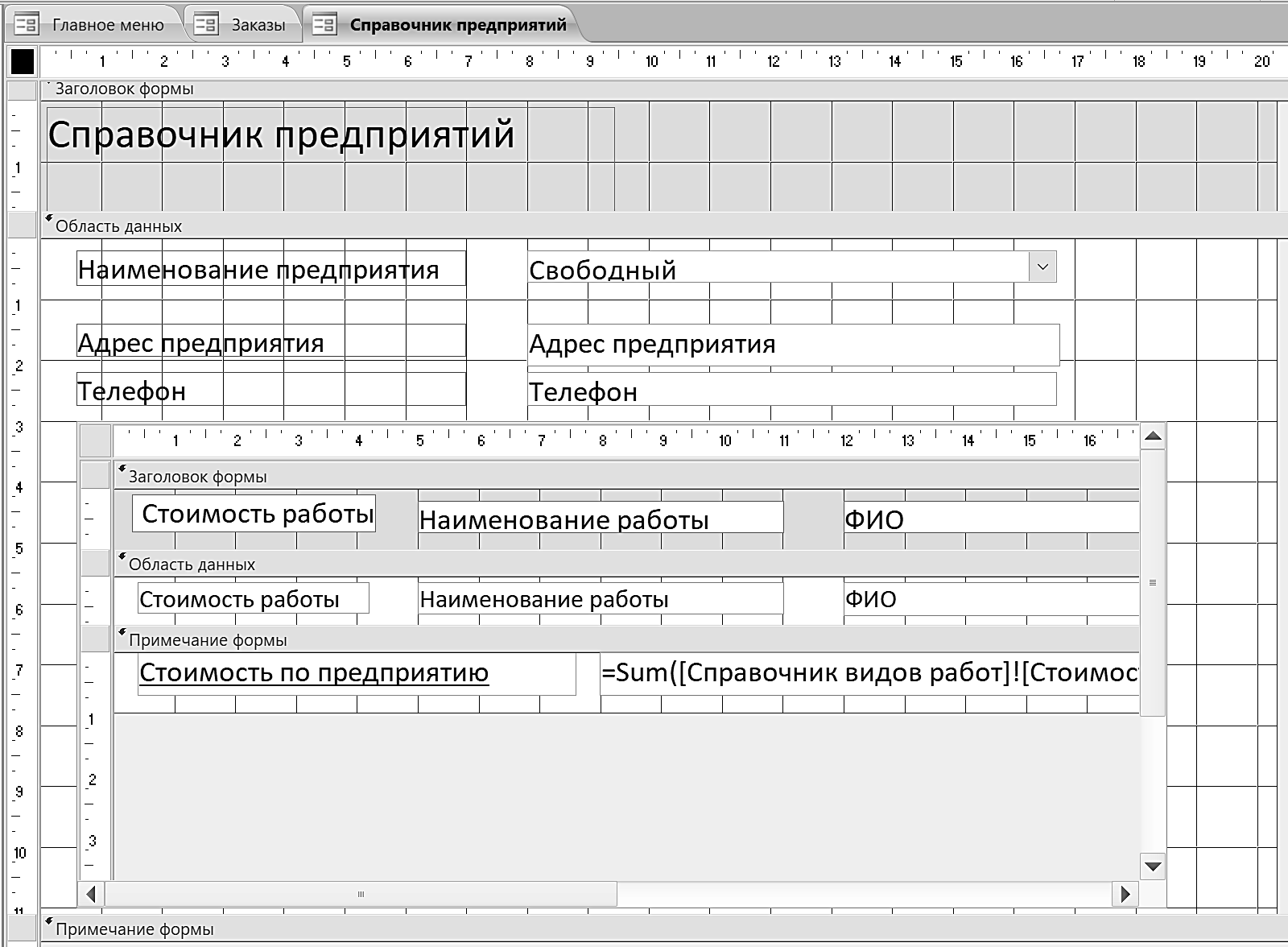


Рисунок 15 — форма *Справочник предприятий* в режиме конструктора

### 3.2.5 Разработка отчетов

Отчет– это объект базы данных, который предназначен для вывода информации из баз данных, прежде всего на печать.

Отчеты позволяют выбрать из баз данных нужную пользователю информацию, оформить ее в виде документа, перед выводом на печать просмотреть на экране.

Источником данных для отчета может служить таблица или запрос. Кроме данных, полученных из таблиц, в отчете могут отображаться вычисляемые поля, например, итоговые суммы.

В данной БД представлены следующие отчеты:

1) *Отчет о стоимости выполнения работ по предприятиям* – предлагает пользователю ввести дату создания отчета, его фамилию и выводит отчет о доходности каждого предприятия, а так же суммарную доходность (рис. 16, 17).

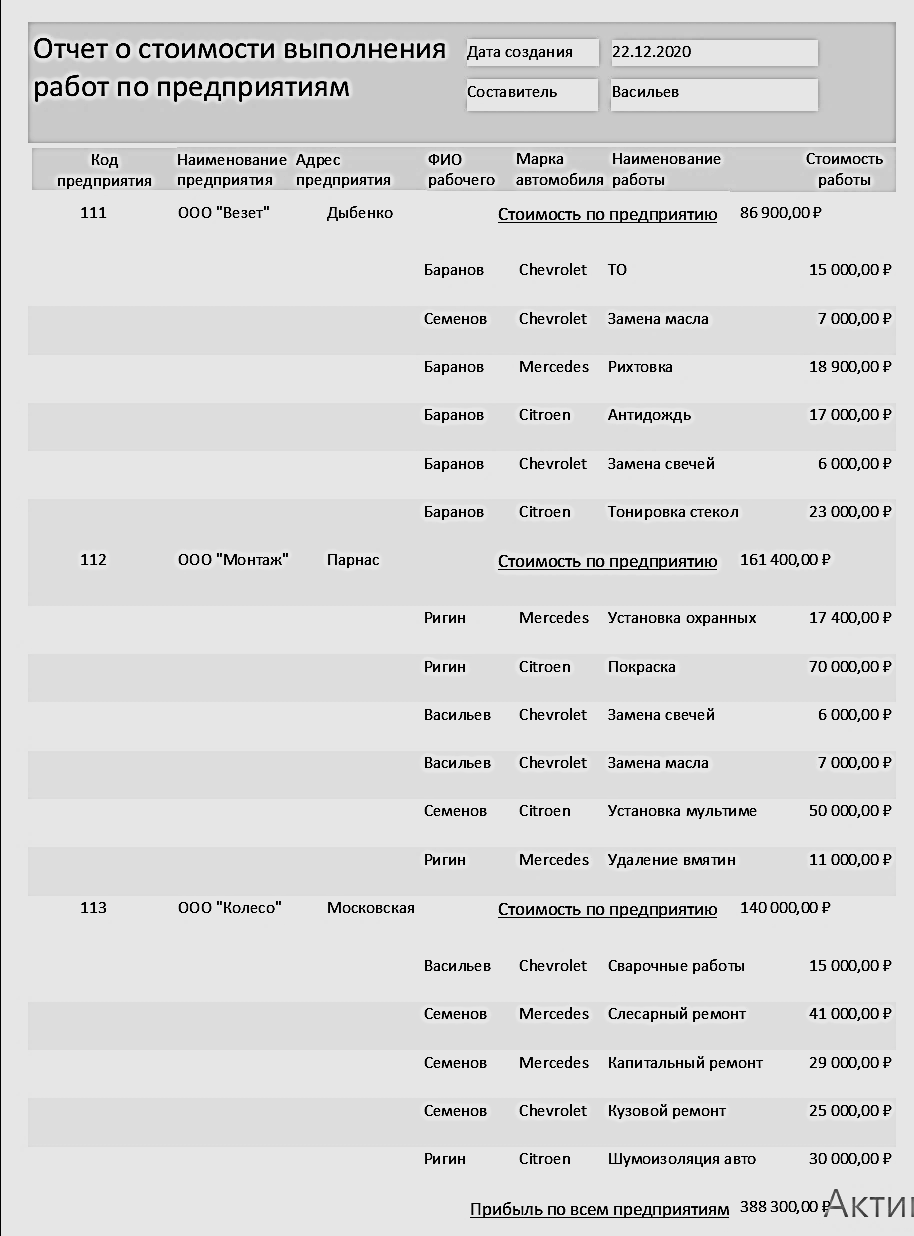


Рисунок 16 — *Отчет о стоимости выполнения работ по предприятию*

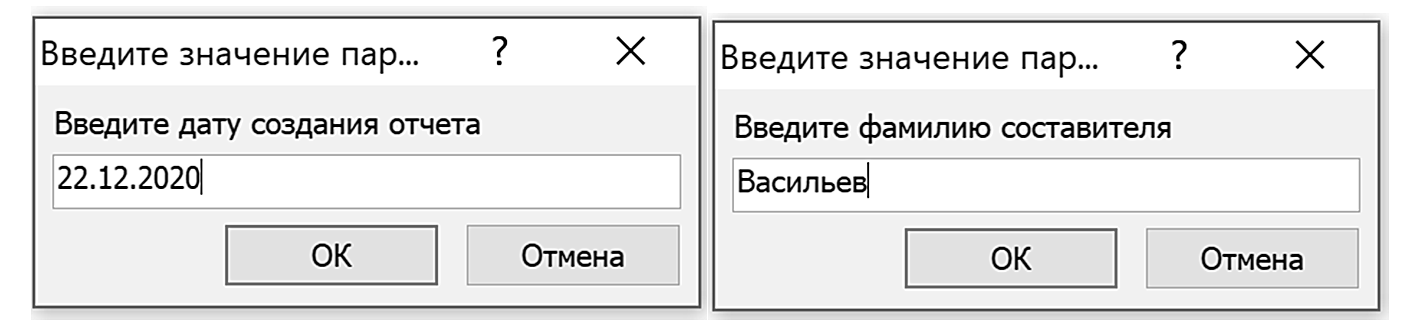


Рисунок 17 — Ввод параметров даты и составителя отчета

2) *Квартальный отчет* – позволяет пользователю ввести начало и конец интересующего периода и создать отчет по всем предприятиям (рис. 18).

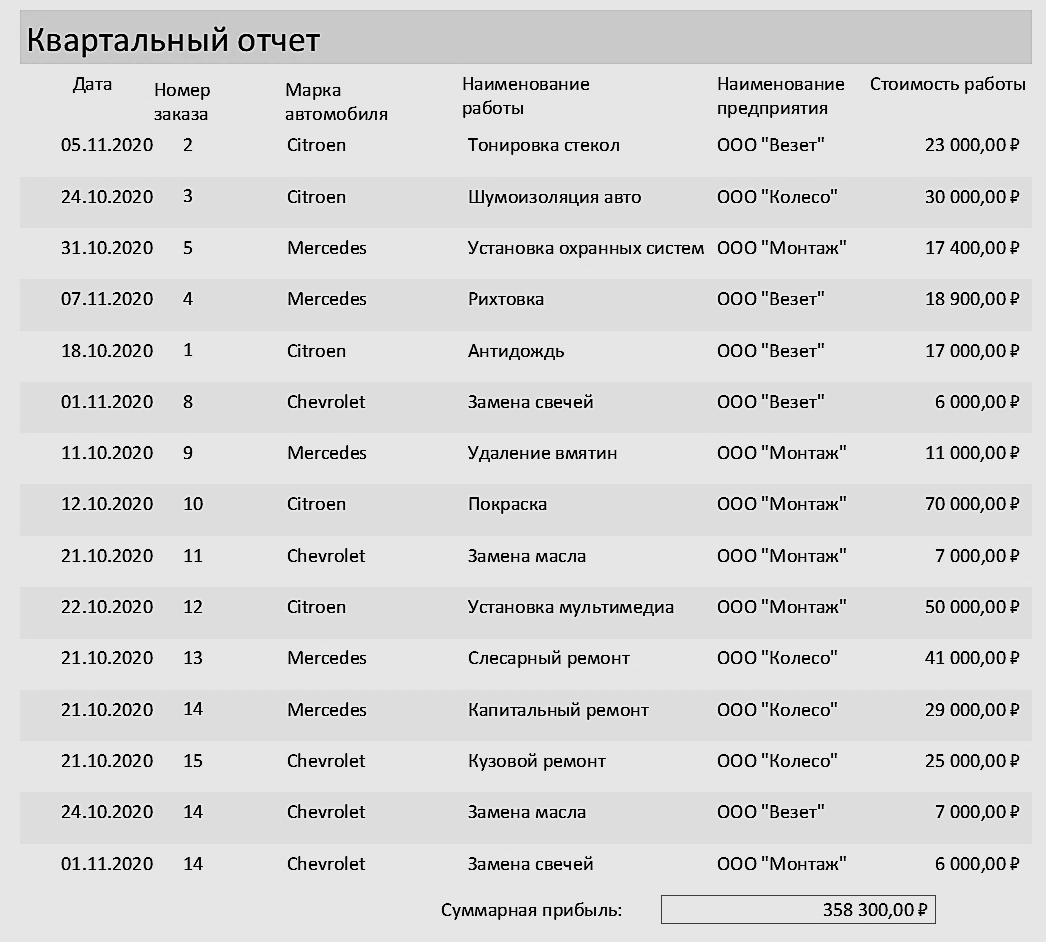


Рисунок 18 — *Квартальный отчет*

3) *Счет на оплату заказа* – позволяет пользователю выбрать интересующий заказ и на его основе создать отчет - счет на оплату заказа (рис. 19, 20).

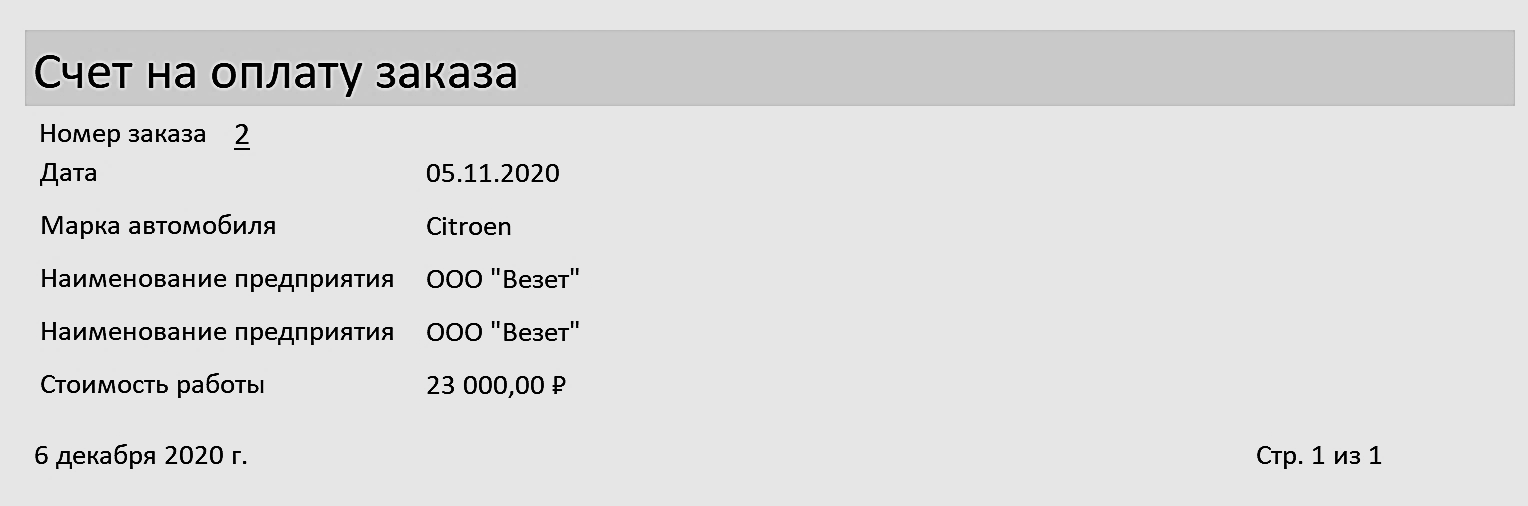


Рисунок 19 — отчет: *счет на оплату заказа*

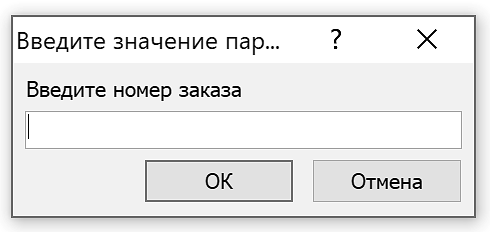


Рисунок 20 — Ввод параметра: *номер заказа*

### 3.2.3.4 Разработка макросов

Макрос – это программа, состоящая из последовательности макрокоманд.

Макрокоманда – это инструкция, ориентированная на выполнение определенного действия. Например, макрокомандой можно открыть форму, отчет, запустить на выполнение запрос, применить фильтр, присвоить значение и пр.). Язык макросов обеспечивает возможность решения большинства задач пользователя, не прибегая к программированию на языке Visual Basic.

Язык макросов является средством программирования, которое позволяет реализовать задачи пользователя, выполняя необходимые действия над объектами БД и их элементами.

В данной БД представлены следующие макросы:

1. *Макрос на закрытие программы* (рис. 12).

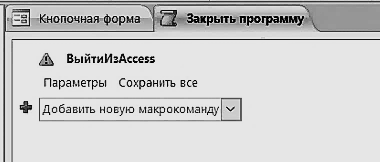


Рисунок 21 — *макрос на закрытие программы*

1. *Макрос для составления отчета* (рис. 22)

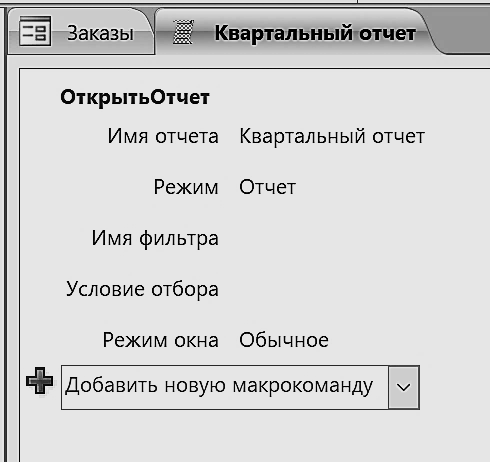


Рисунок 22 — *макрос для составления отчета*

В данной работе макросы используются для ускорения часто выполняемых операций, таких как закрытие программы и составление отчета.

## 3.3 Руководство пользователю базой данных

### 3.3.1 Назначение и возможности базы данных

Разработанная база данных предназначена для формирования и учета заказов на оказание услуг клиентам автосервиса, отслеживание суммарной выручки, полученной от выполнения заказов за любые периоды времени, а также для отслеживания возрастания или убывания клиентского спроса на услуги автосервиса.

### 3.3.2 Правила и порядок работы с базой данных

1) Открыть базу данных

2) В главном меню выбрать нужный пункт (рис. 23, 24):

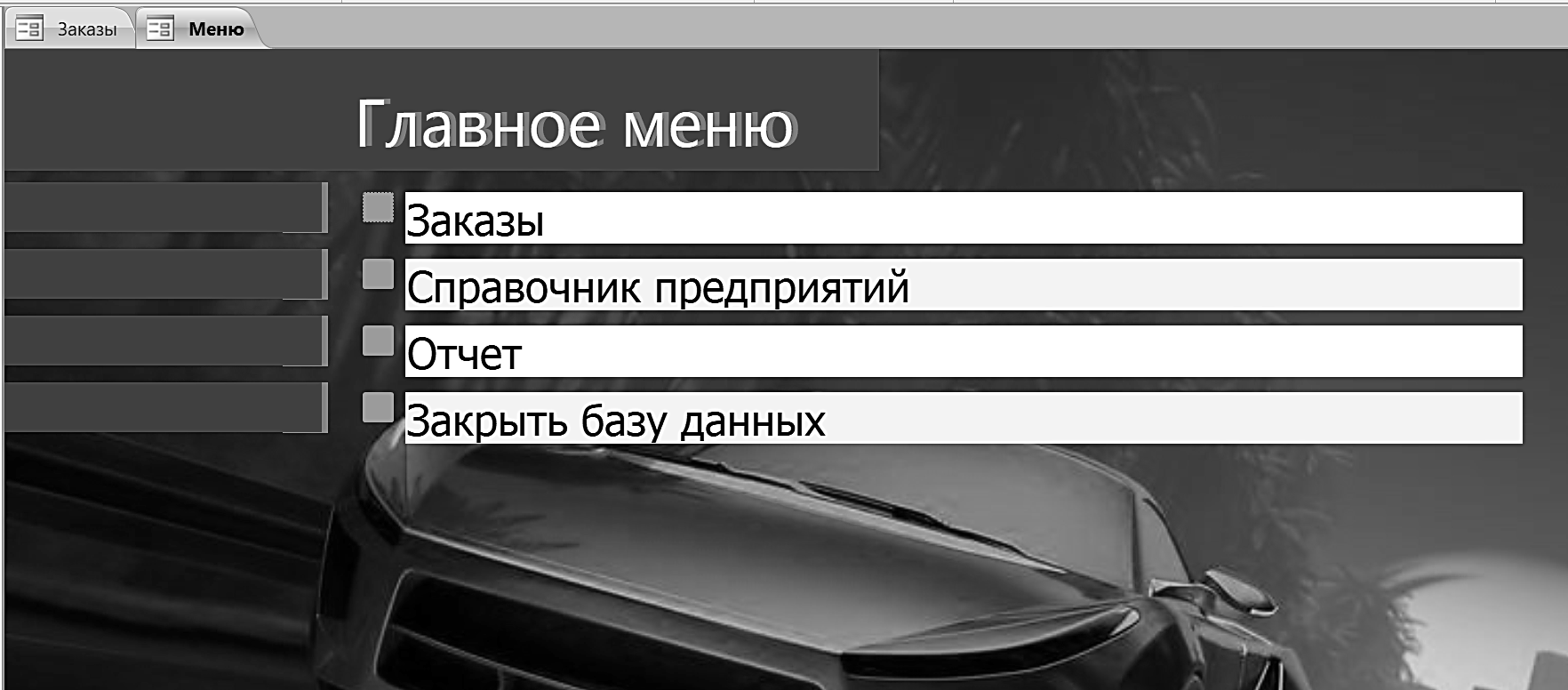


Рисунок 23 — *Главное меню* (кнопочная форма)

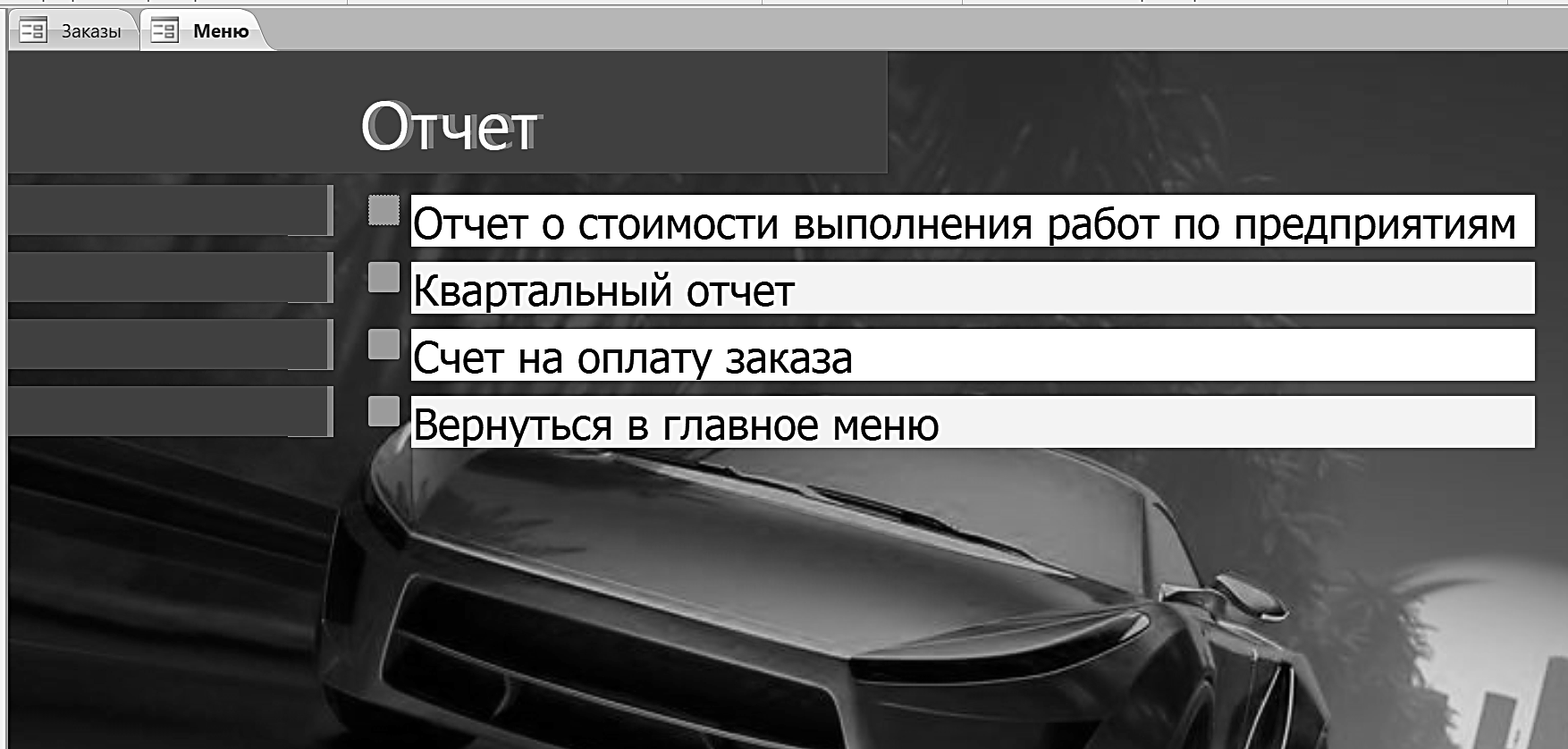


Рисунок 24 — *Отчет*, иерархический подпункт меню

1. Пункты меню :

1-ый уровень:

* Заказы – добавление и редактирование заказа,
* Справочник предприятий – информация о доходности каждого предприятия,
* Отчет – открытие меню для просмотра всех отчетов.

2-ой уровень:

* Отчет о стоимости выполнения работ по предприятиям – создание отчета о доходности всех предприятий,
* Квартальный отчет – создание отчета за определенный промежуток времени,
* Счет на оплату заказа – создание отчета о конкретном заказе,
* Вернуться в главное меню – возвращение в главное меню для выбора пункта.

1. После всей проделанной работы следует нажать пункт меню «Закрыть программу» для сохранения и закрытия базы.

# 4 Выводы по работе

В результате выполнения курсовой работы была создана база данных для формирования и учета заказов на оказание услуг клиентам автосервиса и отслеживания клиентского спроса.

Разработанная база данных предназначена для автоматизации хранения, просмотра и внесения новой информации о заказах, сделанных клиентами, а также для отслеживания изменения клиентского спроса, анализа суммы выручки, полученной в результате выполнения заказов. Содержащаяся в базе данных информации о клиентах позволит клиентской службе оповещать клиентов о новый акциях, напоминать о дате ремонта. Это позволит сохранить существующих клиентов и расширить клиентскую базу. Данные о выручке помогут руководству принимать правильные решения в планировании работы автосервиса, анализировать спрос, проводить необходимые изменения.

В ходе работы получены практические навыки анализа предметной области, постановки задачи, проектирования и разработки реляционной базы данных в среде СУБД Міcrosoft Access.

## Список литературных источников

1. Базы данных: учебное пособие / В. И. Халимон, Г.А. Мамаева, А.Ю. Рогов, В.Н. Чепикова; СПбГТИ(ТУ). Каф. систем. анализа и информ. технологий. - Электрон. текстовые дан. - СПб. : СПбГТИ(ТУ), 2017. - 217 с. (ЭБ).
2. Мамаева, Г.А. Система управления базами данных Microsoft Access : Учебное пособие / Г. А. Мамаева, В. Н. Чепикова ; СПбГТИ(ТУ). Каф. систем. анализа и информ. технологий. - Электрон. текстовые дан. - СПб. СПбГТИ(ТУ), 2018. - 52 с. (ЭБ).
3. Волк, В.К. Базы данных. Проектирование, программирование, управление и администрирование : Учебник / В. К. Волк. - Электрон. текстовые дан. - СПб: Лань, 2019. - 244 с. (ЭБС Лань).
4. Советов, Б.Я. Базы данных. Теория и практика : Учебник для вузов по направ. "Информатика и вычислительная техника" и "Информационные системы" / Б.Я. Советов, В. В. Цехановский, В. Д. Чертовской. - 2 -е изд., стер. - М. : Высш. шк., 2007. - 463 с.