

ФЕДЕРАЛЬНОЕ АГЕНТСТВО СВЯЗИ
Северо-Кавказский филиал ордена Трудового Красного Знамени федерального
государственного бюджетного образовательного учреждения высшего образования
«Московский технический университет связи и информатики»



Методические указания
по выполнению курсового проекта

по дисциплине

ТЕХНОЛОГИИ БАЗ ДАННЫХ

направление подготовки 09.03.01 Информатика и вычислительная техника

Ростов-на-Дону
2019

УДК 681.3.06 (076)
ББК 32.07

Технологии баз данных. Методические указания по выполнению курсового проекта. Ростов-на-Дону: Северо-Кавказский филиал МТУСИ, 2019.- 54с.

В пособии изложены цели и задачи дисциплины, структура дисциплины и краткое содержание каждого раздела. Кроме того, приведено задание на курсовой проект, правила оформления и примеры выполнения заданий курсового проекта.

Дисциплина "Технологии баз данных" имеет целью изучение студентами систематических знаний о базах данных и технологиях, используемых при их разработке. Основное внимание уделяется приобретению знаний и умений, необходимых для работы с системами управления базами данных.

Методические указания предназначены для студентов, обучающихся по направлению 09.03.01 "Информатика и вычислительная техника" профиль "Вычислительные машины, комплексы, системы и сети", "Программное обеспечение и интеллектуальные системы" очной и заочной форм обучения. Пособие может быть использовано также в процессе самостоятельной работы.

Составители: А.Н. Чикалов, доцент кафедры ИВТ
Е.О. Ткачук, доцент кафедры ИВТ

Рецензент: зав. кафедрой ИВТ СКФ МТУСИ, д.т.н. проф. Соколов С.В.

Издание рассмотрено и одобрено на заседании кафедры ИВТ
Протокол от 26 августа 2019 г. № 1

Содержание

1.	Цели дисциплины	4
	Задачи дисциплины	4
	Трудоемкость дисциплины	5
2.	Содержание дисциплины	5
3.	Лабораторные работы и практические занятия	7
4.	Список литературы	8
5.	Задание на курсовой проект	9
6.	Требования к оформлению курсового проекта	20
7.	Пример оформления курсового проекта	21

1. Цели дисциплины

Целью освоения дисциплины является изучение студентами систематических знаний о базах данных и технологиях, используемых при их разработке. Основное внимание уделяется приобретению знаний и умений, необходимых для работы с системами управления базами данных.

Задачи дисциплины

Бакалавр по направлению подготовки 09.03.01 должен решать задачи в сфере *проектной* профессиональной деятельности.

Результатом освоения дисциплины являются сформированные у выпускника следующие компетенции.

ОПК-9: способен осваивать методики использования программных средств для решения практических задач;

В рамках этой компетенции выпускник должен:

знать:

- основные определения и понятия БД; принципы построения, работы, возможности реляционных баз данных;
- основные принципы использования программных средств для построения и управления БД;
- основные принципы построения и функционирования клиент-серверных приложений;

уметь:

- выполнять сбор и структурирование данных, реализовывать принципы нормализации данных, разрабатывать схемы данных, таблицы данных и основные типовые запросы к БД, конструировать основные формы и отчеты БД;
- систематизировать и выполнять нормализацию данных, разрабатывать сложные схемы данных, конструировать сложные запросы, формы и отчеты, пользоваться инструментами решения нестандартных задач при работе с БД;
- выполнять построение таблиц данных и разработку типовых запросов в распределенных БД, управлять доступом к данным, настраивать оборудование;

владеть:

- основными приемами проектирования и управления БД при решении профессиональных задач с использованием средств визуального проектирования;

- приемами проектирования и управления БД при решении нетиповых профессиональных задач;
- основными приемами проектирования и управления БД на основе клиент-серверных технологий.

ПК-1: способен производить разработку и отладку программного кода, интегрировать программные модули и компоненты, проектировать программное обеспечение

знать:

- компоненты программно-технических архитектур, существующие приложения и интерфейсы взаимодействия с БД;
- методы и приемы формализации, алгоритмизации, программирования и оформления программного кода БД;
- подходы к оценке эффективности различных технологий разработки БД;

уметь:

- разрабатывать типовые процедуры создания таблиц, типовые запросы к БД, конструировать основные формы и отчеты БД;
- выбирать методы и алгоритмы решения основных задач по разработке и управлению БД, манипулирования данными, а также находить способы их оценивания;
- выбирать методы и алгоритмы решения основных задач по разработке и управлению распределенных БД, а также находить способы их оценивания;

владеть:

- основными приемами проектирования СУБД при решении профессиональных задач с использованием средств визуального проектирования;
- методами проектирования и управления БД при решении нетиповых профессиональных задач на языке SQL;
- основными методами и средствами проектирования, интеграции модулей и компонент БД на основе клиент-серверных технологий.

Трудоемкость дисциплины

Общая трудоемкость дисциплины 180 часов, в том числе аудиторных занятий:

- заочная форма обучения – 18 часов;
- очная форма обучения -56 часов.

2. Содержание дисциплины

1. Основы проектирования БД

1.1. Модели и типы данных. БД и СУБД. Этапы проектирования БД. Модели структур данных. Основные элементы реляционной модели. Типы взаимосвязей различных сущностей.

1.2. Метод нормальных форм. Основы реляционной алгебры. Нормализация. Технология приведения к нормальным формам. Средства ускоренного доступа к данным. Индексирование.

Вопросы и упражнения для самоконтроля:

1. Каковы этапы разработки проекта?
2. Что называется концептуальной моделью данных?
3. В чем отличие концептуальной и логической модели?
4. Какие типы связей предполагаются в БД?
5. В чем суть метода нормальных форм при проектировании БД?
6. Сформулируйте суть трех первых нормальных форм.
7. В каком порядке выполняются нормальные формы?
8. Какой материал является основой для проектирования?

2. СУБД реляционных БД

2.1. Архитектура СУБД. Основные объекты СУБД. Задачи СУБД. Создание таблиц в различных режимах. Операции с таблицами. Виды соединений в таблицах. операции с пакетом. Создание таблиц с помощью конструктора. Типы данных. Режимы разработки таблиц. Связывание таблиц, установка типов связей. Сортировка и поиск данных. Создание простого и составного первичного ключа. Вторичный ключ. Контроль правильности ввода данных, добавление условий на значение поля

2.2. Манипуляции с данными в БД. Конструирование запросов. Типы запросов. Формы и отчеты. Технология разработки. Средства ускоренного доступа к данным. Сортировка и фильтрация. Изменение данных средствами запроса

2.3. Разработка приложений пользователей. Среда VBA. Переменные, типы данных и константы. Стандартные функции и выражения. Инструкции VBA. Операторы языка. Модули. Макросы. Операторы присваивания, цикла, безусловного перехода. Управление выполнением программы

2.4. Обеспечение информационной безопасности приложения. Принципы управления доступом. Формы для управления безопасностью

Вопросы и упражнения для самоконтроля:

1. Что такое таблица БД?
2. Каковы задачи СУБД?
3. Какие типы данных может содержать БД?
4. Что такое ключ и для каких целей он создается?
5. Какие связи образуются в БД?
6. Какова технология импорта данных из электронных таблиц?

7. Какова технология импорта данных из текстового файла?
8. Что такое запрос к БД?
9. Что означают групповые операции?
10. Как реализуется сортировка данных?
11. Для каких целей используется запрос?
12. Каковы преимущества запроса с параметром?
13. Что такое VBA? Для каких целей он применяется?
14. Что такое форма? Для чего они создаются?
15. Какие типы форм используются?
16. Что такое отчет?
17. Какие средства редактирования отчетов доступны?

3. Распределенная обработка данных

3.1. Структурированный язык запросов. Основы SQL. Создание и редактирование таблиц. Операторы и функции языка SQL. Создание простых запросов. Фильтрация данных. Операторы SELECT, INSERT, UPDATE, DELETE. Взаимодействие с языками программирования. Применение предложений FROM, WHERE, ORDERBY, HAVING, GROUPBY, предикатов DISTINCT, TOP, манипулирование данными

3.2. Модели распределенных систем. Совместный доступ к БД с помощью SQL Server. Режимы работы с БД в сети. Распределение функций в архитектуре "клиент-сервер". Установка связи с источником данных посредством интерфейса OLE DB Создание проекта с использованием существующей базы данных. Средства языка Transact-SQL. Организация транзакций, их свойства. Форматы транзакций. Инструкции SQLBEGINTRANSACTION, COMMIT, предложение ROLLBACK. Механизм хранимых процедур. Инструкция CREATE PROC. Создание процедур в БД. Средства языка Transact-SQL. Вызов сохраненных процедур

3.3. Разработка БД для Интернета. Язык HTML. Преобразование объектов БД в Web-страницы. Создание страниц доступа к БД. Добавление на страницу элементов ActiveX. Публикация данных с помощью Web-страниц. Структура документа HTML. Основные теги: HTML, HEAD, BODY, TITLE, <H1>, <P>, ALIGN, , <TABLE>, CAPTION, BORDER, ALIGN, WIDTH, BGCOLOR, CELLPADDING, CELLSPACING, <TR>, <A>, NAME

Вопросы и упражнения для самоконтроля:

1. Какие задачи реализуются с помощью языка SQL?
2. В чем заключается назначение механизма ODBC?
3. Какие операции над множествами может реализовать оператор SELECT?
4. Что такое индексы БД? Как они создаются?
5. Что такое транзакция?
6. Какие свойства имеют транзакции?

7. Чем отличаются динамические Интернет страницы от статических?
 8. Укажите причины, достоинства и недостатки обработки данных XML-документа на сервере.
 9. Поясните особенности клиент-серверной технологии?
 10. Какие параметры могут быть использованы в операторе SELECT?
- 4. Курсовое проектирование.** Разработка БД по вариантам

3. Лабораторные работы и практические занятия

Практическое занятие №1. Проектирование БД. Анализ предметной области. Определение последовательности выполнения задач. Построение концептуальной и логической модели.

Практическое занятие №2. Разработка таблиц БД. Создание таблиц с помощью конструктора. Задание типов данных. Ввод и редактирование данных. Организация связей между таблицами.

Практическое занятие №3. Экспорт и импорт данных в БД. Импорт электронных таблиц. Импорт и присоединение текстовых файлов. Исправление ошибок после экспорта.

Лабораторная работа №1. Создание простых запросов. Создание запроса в режиме конструктора. Запросы с параметром. Вычисляемые поля.

Лабораторная работа №2. Создание сложных запросов. Конструирование сложных условий отбора. Создание вычисляемых полей. Использование функций в вычисляемых полях. Запросы к связанным таблицам.

Лабораторная работа №3. Разработка форм. Создание форм с помощью мастера и автоматически. Режим конструктора. Редактирование формы. Оформление формы.

Лабораторная работа №4. Разработка отчетов. Создание отчетов автоматическое и с помощью мастера. Редактирование и форматирование отчета. Экспорт отчетов. Преобразование форматов.

Практическое занятие №4. Управление БД с помощью VBA. Написание процедур создания БД и ее заполнение, выполнения обработки данных, создание простых запросов.

Лабораторная работа №5. Работа с БД на языке SQL. Разработка БД и запросов к ней на языке структурированных запросов.

Лабораторная работа №6. Разработка структуры данных на сервере. Создание клиентского соединения. Создание структуры данных. Нормализация. Разработка структуры индексов.

Лабораторная работа №7. Гипертекстовые БД. Структура документа HTML. Разработка структуры гипертекстовой БД. Исследование и модернизация программы получения данных из БД.

4. Список литературы

1. Гурвиц Г.А. Microsoft® Access 2010. Разработка приложений на реаль-

- ном примере. — СПб.: БХВ-Петербург, 2010. — 496 с.
2. Голицына О.Л., Максимов Н.В., Попов И. И. Базы данных: учеб. пособие (Высшее образование). - М.: ФОРУМ: ИНФРА-М, 2019. — 400 с.
 3. Сенное А. Access 2010. Учебный курс. - СПб.: Питер, 2006. - 288 с.
 4. Фролов А.В., Фролов Г.В. Базы данных в Интернете: практическое руководство по созданию Web-приложений с базами данных. - Изд. 2-ое, испр. М.: Издательско-торговый дом "Русская редакция", 2000. -448с.
 5. Кузнецов М.В. MySQL 5. СПб.: БХВ-Петербург, 2010. - 1024 с.
 6. ГОСТ 2.102-95 Основные требования к текстовым документам.
 7. Положение о контрольных и курсовых работах (проектах) СКФ МТУСИ, 2017.
 8. Жуковский А.Г., Манин А.А. Руководство по подготовке курсовых работ (проектов) и выпускных квалификационных работ. Ростов-на-Дону: СКФ МТУСИ, 2019. - 61с.
 9. Чикалов А.Н. Технологии баз данных. Проектирование таблиц БД. Методические указания к лабораторным и практическим занятиям. Ростов-на-Дону: СКФ МТУСИ, 2016 г.
 10. Чикалов А.Н. Технологии баз данных. Запросы. Формы. Отчеты. Методические указания к лабораторным и практическим занятиям. Ростов-на-Дону: СКФ МТУСИ, 2016 г.
 11. Уроки программирования. Базы данных. <http://www.life-prog.ru>
 12. Специальность Разработчик баз данных. <http://edu.cbsystematics.com/ru/education/speciality/databasedeveloper>.
 13. Форум специалистов БД <http://www.cyberforum.ru/database/>.
 14. Уроки программирования. Базы данных. <http://www.life-prog.ru>

5. Задание на курсовой проект

Разработать базу данных по индивидуальному варианту. Перечень работ, подлежащих выполнению определяется заданием, утвержденным ведущим кафедрой. Типовой вариант предполагает:

- обоснование концептуальной модели базы данных;
- выполнение нормализации базы данных;
- конструирование таблиц базы данных и заполнение полей;
- оформление схемы данных;
- разработку запросов к базе данных. Запросы должны быть следующих типов:

- по одному условию;
- по двум и более условиям;
- с использованием оператора Like для поиска по фрагменту данных;
- с параметром;

-с групповой операцией.

Должно быть разработано **не менее пяти запросов**. Конкретные запросы необходимо выбрать из примеров индивидуального задания, а при отсутствии запросов требуемых типов - разработать самостоятельно. Желательно в отдельных запросах применить функции времени. В случае необходимости для организации требуемых запросов разработчику следует изменить состав информации, хранящейся в базе данных;

- разработку форм базы данных. Форма должна предусматривать компактное размещение материала на основе элемента управления "Вкладки". Вкладок необходимо создать не менее трех на основе подчиненных форм. Отредактировать на них все необходимые надписи, использовать графические элементы (фото) из разработанных таблиц хранения данных;

- разработку отчета. Отчет должен быть приближен к реальным документам (по теме индивидуального задания). В отчете необходимо использовать уровни группировки данных по выбранным полям, текущую дату и подписи должностных лиц.

Работа может быть выполнена в доступной для автора среде. В качестве инструмента разработки допустимо использовать СУБД MSAccess, OpenOfficeBase, MSSQLServer, MySQL, PostgreSQL и пр.

Вариант индивидуального задания выбирается по последним двум цифрам зачетной книжки (табл.1). Для определения варианта в диапазоне значений 31-60 предварительно необходимо вычесть 30, в диапазоне 61-90 предварительно вычесть 60, для остальных - вычесть 90.

Таблица 1. Индивидуальные варианты заданий на курсовой проект

Вариант и тема	Обязательный перечень форм и запросов
1. Модель «Деканат» должна содержать информацию о списках групп, начисленных стипендиях, зачетах, оценках, полученных на экзаменах	<ul style="list-style-type: none">- получить полную информацию обо всех сданных экзаменах и зачетах в выбранной группе;- получить список всех студентов, не сдавших зачет по выбранной дисциплине;- получить список студентов, не сдавших зачет более чем по 2-м предметам;- получить список студентов, не получивших стипендию;- получить список студентов, получивших ту же оценку на экзамене по выбранному предмету, что и выбранный студент;- получить список предметов, по которым получено одинаковое количество зачетов;- получить список групп, в которых студенты получили заданное количество оценок «пять»

	по выбранному предмету
2. Модель «Страховая компания» должна содержать информацию о клиентах, договорах страховки, объектах страховки, страховых случаях	<ul style="list-style-type: none"> - получить список клиентов с заданным объектом страховки; - получить список клиентов с заданным объектом страховки и одинаковой страховой суммой; - получить список всех договоров, по которым произошел заданный страховой случай; - получить список всех страховых случаев, происшедших с даты по дату; - получить список клиентов, у которых тот же объект страховки, что и у заданного клиента; - для заданного объекта страховки получить пары (имя клиента и номер договора), по которым произошел страховой случай
3. Модель «Поликлиника» должна содержать информацию о пациентах, докторах, диагнозах	<ul style="list-style-type: none"> - получить список всех пациентов с заданным диагнозом; - получить список пациентов, которые обслуживаются тем же доктором, что и заданный пациент; - получить список докторов-однофамильцев с заданным пациентом; - получить список докторов, у которых лечатся пациенты не меньше заданного количества с заданным диагнозом; - получить список пациентов, имеющих те же симптомы болезни, что и у заданного пациента; - для заданного доктора «А» получить список его пациентов, имеющих тот же диагноз, что и выбранный пациент, лечащийся у доктора «В»

<p>4. Модель «Отдел кадров» должна содержать информацию о сотрудниках, подразделениях организации, должностях, официальных документах (паспорт, диплом, военный билет и т.п.)</p>	<ul style="list-style-type: none"> - получить список всех сотрудников, работающих в заданном подразделении; - получить список сотрудников, работающих под руководством того же начальника, что и у выбранного сотрудника; - получить список всех однофамильцев заданного начальника подразделения; - для заданного подразделения получить пары (Ф.И.О. сотрудника, № военного билета), сотрудников, работающих в заданной должности; - получить список всех подразделений, в которых работают сотрудники с ученой степенью (на основании диплома); - получить список сотрудников, состоящий из троек (Ф.И.О. сотрудника, № паспорта, должность), пенсионного возраста. (Обратить внимание на то, что у мужчин и женщин пенсионный возраст разный)
<p>5. Модель «Фильмотека» должна содержать информацию о кинофильмах, актерах, киностудиях и контрактах между актерами и киностудиями</p>	<ul style="list-style-type: none"> - получить список всех актеров, снимающихся на заданной киностудии; - получить список кинофильмов, в которых снимаются те же актеры, что и в заданном кинофильме; - получить пары (Ф.И.О. актера, № контракта), занятых в фильмах, выпущенных на заданной киностудии; - получить список киностудий, в которых были сняты фильмы по заданной тематике и в заданном году; - получить список актеров, у которых имеется контракт с киностудией, расположенной в том же городе, в каком проживают эти актеры; - получить список кинофильмов, в которых заданный актер не снялся ни разу
<p>6. Модель «Роддом» должна содержать информацию о родившихся детях, их матерях, докторам, курирующих ребенка</p>	<ul style="list-style-type: none"> - получить список всех матерей, у которых родились девочки; - получить список матерей – однофамилиц, с заданным врачом; - выбрать докторов, у которых пациентами являются дети, родившиеся в тот же день, что и у выбранной матери; - получить список докторов, у которых пациентами являются близнецы; - получить список матерей, у которых роди-

	<p>лись мальчики, на определенную дату;</p> <ul style="list-style-type: none"> - для заданного доктора получить пары (мать, ребенок), у которых пол совпадает
<p>7. Модель «Заказ билетов» должна содержать информацию о пассажире, билете, рейсе самолета</p>	<ul style="list-style-type: none"> - получить список всех пассажиров заданного рейса; - получить список пассажиров, вылетающих рейсами до заданного города и имеющих билет на заданное место; - получить список рейсов, на которых вылетали пассажиры с заданными Ф.И.О. и датой вылета; - получить список, состоящий из пар (№ рейса, № места, на которые не проданы билеты); - получить список пассажиров, вылетавших до заданного города в заданном интервале дат; - получить список типов самолетов, летающих по тому же маршруту, что и самолет, на котором летел пассажир с заданным номером билета
<p>8. Модель «Склад» должна содержать информацию о складе, товаре, кладовщике, документе (накладная на прием товара, акт на отпуск товара)</p>	<ul style="list-style-type: none"> - получить список всех товаров, находящихся на всех складах; - получить список кладовщиков, принимавших заданный товар; - получить пары (название товара, № акта), отпущенные с заданного склада в заданный день товаров; - получить список складов, на которые был осуществлен прием тех же товаров, что и на заданный склад; - получить номера всех накладных на прием тех же товаров, что и принимал заданный кладовщик; - получить список (наименование товара, номер склада, Ф.И.О. кладовщика), принятых по заданной накладной

<p>9. Модель «Предприятие» должна содержать информацию о поставщиках, производимых изделиях (составе изделия), потребителях продукции, товарообороте</p>	<ul style="list-style-type: none"> - получить список поставщиков заданной детали; - получить список всех изделий, в которое входит заданная деталь от заданного поставщика; - получить список потребителей изделия, в которые входит та же деталь, что и производится заданным поставщиком; - получить список изделий, у которых поставщик деталей находится в том же городе, что и потребитель; - получить список потребителей изделия, в которое входит одно и то же количество деталей, производимых разными поставщиками; - получить все номера деталей, которые поставляются только одним поставщиком
<p>10. Модель «Издательство» должна содержать информацию об авторах, изданиях, видах рукописей</p>	<ul style="list-style-type: none"> - получить список всех изданий, выпускаемых заданным издательством; - получить список авторов, которые публикуют заданный вид рукописи в заданном издательстве; - получить список изданий, выпущенных в том же издательстве, в котором публиковался автор, проживающий в том же городе, что и автор с заданной фамилией; - получить список издательств и авторов, расположенных в одном и том же городе; - получить список авторов, опубликовавших одну и ту же рукопись в разных издательствах; - получить пары (автор, издательство) для рукописей, изданных за заданный период времени
<p>11. Модель «Плановый отдел» должна содержать информацию о материальных ценностях, подразделениях организации, заявках, поставщиках материальных ценностей</p>	<ul style="list-style-type: none"> - получить список всех мат.ценностей (МЦ), требуемых заданному подразделению; - получить список всех подразделений, которым требуется та же МЦ, что и заданному подразделению; - получить список пар (МЦ, поставщик) таких, у которых стоимость МЦ не больше заданной; - получить список пар (МЦ, количество), требуемых подразделениям организации, находящимся в том же городе, что и заданное подразделение; - получить номера заявок от подразделений, которым требуется заданная МЦ;

	<ul style="list-style-type: none"> - получить список поставщиков МЦ, находящихся в той же стране, что и подразделение организации, которой эта МЦ требуется
12. Модель «Канцелярия» должна содержать информацию о типах документов (входящие, исходящие), собственно о документе, адресате и корреспонденте	<ul style="list-style-type: none"> - получить список всех исходящих документов за период; - получить список всех корреспондентов, которым пришли письма в тот же день, что и заданному корреспонденту; - получить номера входящих документов от корреспондентов, проживающих в том же городе, что и адресат; - получить пары (номер входящего документа, номер исходящего), у которых темы одинаковы; - получить список всех адресатов, которым были направлены письма по той же теме, что и заданному адресату; - получить список (корреспонденты, номер входящего письма), ответы которым должны быть подготовлены к заданной дате.
13. Модель «Налоговая инспекция» должна содержать информацию о зарегистрированных налогоплательщиках, видах налогов, начисленных и перечисленных суммах налогов	<ul style="list-style-type: none"> - получить список всех налогоплательщиков, начисляющих заданный вид налога; - получить список налогоплательщиков, переплативших по заданному виду налога; - получить список налогов, по которым произошла недоплата; - получить суммы начисленных налогов налогоплательщиками, которые перечислили деньги в тот же день, что и заданный налогоплательщик; - получить список налогоплательщиков, не перечисливших деньги к заданной дате; - получить список налогоплательщиков, у которых начисленная сумма по заданному виду налога лежит в заданном диапазоне.
14. Модель «Филателистическое агентство» должна содержать информацию о видах филателистической продукции, управлениях федеральной почтовой связи (УФПС), собственно о филателистической продукции, заявках от УФПС	<ul style="list-style-type: none"> - получить список всех УФПС, которым необходим заданный вид филат.продукции; - получить список УФПС, которые заказали ту же фил.продукцию, что и заданное УФПС; - получить номера заявок от УФПС, которым требуется заданное количество заданной фил.продукции; - получить пары (вид фил.продукции, количество) для УФПС, расположенного в заданном

	регионе
15. Модель «Завод» должна содержать информацию о поставщиках, деталях и проектах	<ul style="list-style-type: none"> - получить все номера деталей, поставляемых поставщиком из заданного города; - получить номера деталей, поставляемых для всех проектов, обеспечиваемых поставщиком из того же города, где размещен проект; - получить список всех поставщиков, поставляющих одну и ту же деталь для всех проектов; - получить список проектов, использующих, по крайней мере, одну деталь, производимую заданным поставщиком; - получить список троек (номер детали, количество, поставщик), аналогичных деталям, поставляемым заданным поставщиком; - получить список проектов, которые обеспечиваются деталями от одного поставщика
16. Модель «Театр» должна содержать информацию о театре, спектакле, ведущих артистах в спектакле	<ul style="list-style-type: none"> - получить список ведущих артистов всех театров; - получить список спектаклей, в которых занят заданный артист; - получить список театров, в которых играют однофамильцы; - получить пары (название театра, город), в котором идут те же спектакли, что и спектакль, в котором занят заданный артист; - получить список спектаклей, в которых участвуют артисты, живущие в том же городе, что и театр; - получить пары (название театра, спектакль), в которых играют артисты с той же фамилией, что и автор пьесы

<p>17. Модель «Собачий питомник» должна содержать информацию о собаке, ее родителях, ее хозяевах</p>	<ul style="list-style-type: none"> - получить список собак заданной породы; - получить адреса хозяев собаки с заданными родителями; - получить список хозяев собак заданной породы и заданного окраса; - получить пары (имя хозяев, телефон) всех собак женского пола, у которых «папа» был тем же самым, что и у заданной собаки; - получить список пород собак, хозяева которых проживают на одной улице; - получить список имен собак заданной породы, у которых родители были чемпионами в заданном году
<p>18. Модель «Швейная компания» должна содержать информацию о моделях одежды, об используемых тканях, фурнитуре, о поставщиках ткани и фурнитуры</p>	<ul style="list-style-type: none"> - получить список всех поставщиков, производящих заданную ткань; - получить список моделей одежды, у которых поставщики ткани и фурнитуры расположены в одном городе; - получить пары(номер модели, номер изделия фурнитуры), входящих в модель одежды, сшитую из заданной ткани; - получить список всех поставщиков, поставляющих ткань и фурнитуру для моделей заданного сезона; - получить пары (наименование фурнитуры, количество), входящих в модели заданного сезона и заданной ткани; - получить список номеров моделей одежды, сшитых из ткани заданного цвета
<p>19. Модель «Торговая компания» должна содержать информацию о товарах, магазинах, поставщиках, покупателях</p>	<ul style="list-style-type: none"> - получить список всех товаров, продаваемых в заданном магазине; - получить список товаров, поставщик которых находится в том же городе, что и магазин, их продающий; - получить список всех потребителей, купивших товары стоимостью не больше заданной; - получить список поставщиков, которые поставили заданный товар в магазин с заданным адресом; - получить пары (номер товара, название поставщика), продаваемых через магазины, в которых обслуживается заданный потребитель; - получить список пар (магазин, адрес), в которых продается заданный товар от заданного

	поставщика
20. Модель «ГАИ» должна содержать информацию о зарегистрированных автомобилях, их владельцах, выданных документах на автомобиль	<ul style="list-style-type: none"> - получить список всех автомобилей, зарегистрированных в заданном году; - получить список владельцев автомобиля заданной марки; - получить список номеров свидетельств о регистрации АМТС той же марки и того же цвета, что и автомобиль, принадлежащий заданному владельцу; - получить список пар (марка автомобиля, гос.номер), принадлежащих заданному владельцу и зарегистрированному не позднее заданного года; - получить список троек (Ф.И.О. владельца, марка автомобиля, гос.номер) автомашин, выпущенных после заданного года и зарегистрированных в заданном отделении ГАИ; - получить список адресов владельцев автомобилей, зарегистрированных в том же отделении ГАИ и в том же году, что и автомобиль заданного владельца
21. Модель «Политклуб» должна содержать информацию о членах клуба; образовании; членстве в общественных организациях и представительных органах гос.власти и местного самоуправления; мероприятиях клуба; выполнении персональных поручений	<ul style="list-style-type: none"> - получить список членов политклуба по определённой форме; - получить список членов клуба, участвующих в заданном исполнительном органе государственной власти; - получить список членов клуба, имеющих ту же награду, что и заданный член клуба; - определить общую сумму взносов за заданный период времени; - получить список невыполненных поручений и ответственных за них

<p>22. Модель «Салон мобильных телефонов» должна содержать информацию о товарах и складах</p>	<ul style="list-style-type: none"> - список товаров на складе в заданном районе; - список товаров на складе заданном ценовом диапазоне; - список товаров на складе, которые будут в наличии в течение 5 дней; - список складов с минимальной ценой определенного товара; - список складов с максимальной ценой определенного товара; - список товаров в магазине, приобретенных на определенном складе
<p>23. Модель «Сервис» должна содержать информацию об услугах и клиентах</p>	<ul style="list-style-type: none"> - перечень имеющихся услуг в заданном ценовом диапазоне; - перечень оказанных услуг в определённом районе (городе); - список клиентов, заказавших определённую услугу; - список услуг, которые оказываются в настоящее время; - минимальное и максимальное количество оказанных услуг в определённый промежуток времени; - список услуг, предоставленных в заданный период времени
<p>24. Модель «Заводы» должна содержать информацию о заводе, основном виде деятельности по Общероссийскому классификатору видов экономической деятельности(ОКВЭД), продукции, покупателях</p>	<ul style="list-style-type: none"> - список заводов в заданном районе; - список заводов по ОКВЭД; - список товаров, приобретенных на определенном заводе; - список организаций-покупателей, покупающих определенные товары; - список заводов, производящих продукцию на экспорт
<p>25. Модель «Отель» должна содержать информацию о номерах, стоимости, клиентах, периоде проживания</p>	<ul style="list-style-type: none"> - перечень свободных номеров, в том числе по классам; - список гостей, проживающих в номерах VIP класса; - список гостей, приехавших из определенной страны/города; - список и количество номеров, выдаваемых каждый день; - минимальная и максимальная количество свободных/не свободных номеров в определенный период времени;

	<ul style="list-style-type: none"> - стоимость проживания в номере в зависимости от класса
26. Модель «Кабельное телевидение» должна содержать информацию об услуге, клиентах, сотрудниках, стоимости	<ul style="list-style-type: none"> - перечень имеющихся услуг в заданном ценовом диапазоне; - перечень услуг в заданном районе; - список клиентов о сопровождающих сотрудниках; - доход за определенный период; - популярность отдельных услуг
27. Модель «Больница» должна содержать информацию о пациентах, докторах, диагнозах, загруженности стационара	<ul style="list-style-type: none"> - список всех пациентов с заданным диагнозом; - загруженность стационара по отделениям; - список пациентов, которые находятся на лечении больше заданного времени; - список лечащих врачей, закрепленных за пациентами с заданными диагнозами; - список докторов, у которых лечатся пациенты больше заданного времени; - список пациентов, имеющих те же симптомы болезни, что и у заданного пациента; - для заданного доктора «А» список его пациентов, имеющих тот же диагноз, что и выбранный пациент, лечащийся у доктора «В»
28. Модель "Строительно-монтажное предприятие" должна содержать информацию о работах, заказчиках, сроках работ, бригадах, нормативах, стоимости работ	<ul style="list-style-type: none"> - продолжительность работ на конкретном объекте; - задержки в выполнении работ; - занятости бригад на конкретном объекте; - перечень бригад, выполнивших работы за период; - затратах стройматериалов; - доход по конкретному объекту
29. Модель "Ресторан" должна содержать сведения о расчете затрат приготовления блюд из ингредиентов, прибыли, финансовом мониторинге	<ul style="list-style-type: none"> - - заказы, которые были выполнены или выполняются сегодня; - расход продуктов по всем выполненным заказам; - затраты на продукты по всем выполненным заказам за период времени; - общая стоимость всех блюд в продажах; - прибавочная стоимость без вычета средств на зарплату, накладных расходов и налогов и другие запросы

30. Модель "Нефтепереработка" должна содержать сведения о цехах, продукции, персонале, выработанной продукции, выбросах	<ul style="list-style-type: none"> - выработанной продукции за период; - количестве полученного сырья; - проданной продукции; - доходе за период; - пропорции продукции по наименованиям; - начальнике смены, который ответственен за конкретную партию топлива
---	---

6. Требования к оформлению курсового проекта

Работа должна быть оформлена в соответствии с требованиями Положения о контрольных и курсовых работах (проектах) СКФ МТУСИ и положений ГОСТ 2.102-95 Основные требования к текстовым документам.

В соответствии с Положением (Раздел 2) пояснительная записка должна содержать 30-40 стр. машинописного текста размером 14 с полуторным интервалом с выравниванием по ширине.

Каждый лист пояснительной записки должен иметь рамку и чертежный штамп.

На защиту студент представляет пояснительную записку на бумажном и электронном носителе, файл разработанной базы данных. Пояснительная записка должна быть размещена в портфолио студента.

В процессе защиты автору предстоит обосновать принятые решения и продемонстрировать работающую базу данных. Автор для доклада может воспользоваться подготовленной презентацией.

Пояснительная записка должна содержать следующие элементы:

1. Титульный лист (приложение А Положения);
2. Задание (приложение Б Положения);
3. СОДЕРЖАНИЕ;
4. ВВЕДЕНИЕ (содержит описание состояния проблемы, актуальность цели и задачи проекта);
5. Разделы и подразделы;
6. ЗАКЛЮЧЕНИЕ (включает выводы и рекомендации);
7. СПИСОК ЛИТЕРАТУРЫ;
8. ПЕРЕЧЕНЬ СОКРАЩЕНИЙ (при необходимости);
9. Приложения (при необходимости).

Записка должна быть подписана автором с указанием даты.

7. Пример оформления курсового проекта

Федеральное агентство связи
Северо-Кавказский филиал ордена Трудового Красного Знамени
федерального государственного бюджетного образовательного учреждения
высшего образования
«Московский технический университет связи и информатики»

Кафедра Информатики и вычислительной техники

Допустить к защите
«__» _____ 20__ г.

(подпись, Фамилия, И.О.)

Курсовой проект

по дисциплине: Технологии баз данных

по теме: Аэропорт

Студент Капустина А.М.

Направление Информатика и ВТ

Группа ВМ-41 Курс 4

Шифр ст. билета 13227

Вариант 27

Дата «__» _____ 20__ г.

Подпись _____

Ростов-на-Дону

2019

ФЕДЕРАЛЬНОЕ АГЕНТСТВО СВЯЗИ
СЕВЕРО-КАВКАЗСКИЙ ФИЛИАЛ
ОРДЕНА ТРУДОВОГО КРАСНОГО ЗНАМЕНИ
ФЕДЕРАЛЬНОГО ГОСУДАРСТВЕННОГО БЮДЖЕТНОГО ОБРАЗОВАТЕЛЬНОГО УЧРЕЖДЕНИЯ
ВЫСШЕГО ОБРАЗОВАНИЯ
«МОСКОВСКИЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ СВЯЗИ И ИНФОРМАТИКИ»

Кафедра "Информатики и вычислительной техники"

Утверждаю
Зав. кафедрой ИВТ СКФ МТУСИ
_____ С.В. Орлов
« » _____ 20 ____ г.

ЗАДАНИЕ
НА КУРСОВОЙ ПРОЕКТ

Студенту Капустину Андрею Макаровичу

группы ВМ-41

Направление 09.03.01 Информатика и вычислительная техника

Профиль Вычислительные машины, комплексы, системы и сети

Тема курсового проекта: **База данных "Аэропорт"**

утверждена приказом директора филиала _____

Содержание пояснительной записки
Исходные данные: Разработать базу данных в соответствии с вариантом 27 и требований раздела 6 Методических указаний.
Введение 1. Обоснование концептуальной модели базы данных 2. Выполнение нормализации базы данных 3. Конструирование таблиц базы данных и заполнение полей 4. Оформление схемы данных 5. Разработка запросов к базе данных 6. Разработка форм базы данных 7. Разработка отчета Заключение

Перечень графического материала, выносимого на защиту:

1. Схема данных;
2. Пример структуры и заполнения таблицы;
3. Запросы в режиме конструктора с заполненными бланками запроса (5 запросов);
4. Таблицы записей с результатами выполнения запросов;
5. Внешний вид формы;
6. Вариант отчета.

Срок сдачи студентом курсового проекта 01.09.2019

Дата выдачи задания 30.04.2019

Руководитель Молчанов А.Н., доцент кафедры ИВТ, к.т.н.

(фамилия и инициалы, должность, ученая степень)

(подпись руководителя)

Задание принял к исполнению _____
(дата и подпись студента)

СОДЕРЖАНИЕ

Введение	26
1. Обоснование концептуальной модели базы данных	27
2. Выполнение нормализации базы данных	31
3. Конструирование таблиц базы данных и заполнение полей	34
4. Оформление схемы данных	40
5. Конструирование запросов к базе данных	41
6. Разработка форм базы данных	47
7. Разработка отчета	52
Заключение	55
Список литературы	56

					<i>СКФ МТУСИ 09.03.01 13227 ПЗ</i>			
Изм	Лист	№ документа	Подп.	Дата	База данных "АЭРОПОРТ"			
Разработал	Капустин А.М.							
Проверил	Капустин А.М.							
Рецензент	Молчанов А.Н.							
Н. контр								
Утвердил	Орлов С.В.							
					Лит.		Лист	Листов
							4	23
					<i>каф.ИВТ гр. ВМ-41</i>			

ВВЕДЕНИЕ

В современном информационном обществе человеку все сложнее отследить все важные изменения и своевременно реагировать на сложившуюся обстановку. Именно поэтому сегодня, как никогда раньше, нашу жизнь определяют механизмы распределения данных и знаний, средства автоматизации обработки и представления информации. Именно базы данных являются одним из таких средств, обеспечивающих эффективную работу персонала при управлении аэропортом.

Достаточно простым и довольно мощным инструментом для разработки информационных систем является системы управления базами данных (СУБД) MicrosoftAccess. Ее можно использовать как для разработки приложений (в этом случае она является средой разработки), так и для запуска приложений (как пользовательская среда)

В данном проекте предложена созданная в среде MicrosoftAccess система анализа географии полётов судов аэропорта, задействованных компаниях и лётных экипажах, предоставления информации об авиалайнерах, обслуживающих конкретные маршруты.

					СКФ МТУСИ 09.03.01 13227 ПЗ	Лист
Изм	Лист	№докум.	Подпись	Дата		24

1 Обоснование концептуальной модели базы данных

Построение концептуальной модели предполагает последовательную реализацию ряда этапов. Первый из них это анализ предметной области, которая задается заданием на проектирование.

В соответствии с заданием [1] в информационной системе аэропорта должна быть представлена следующая информация:

- о сотрудниках, из которых формируются летные экипажи;
- о медосмотрах сотрудников;
- об экипажах
- о самолётах и их моделях;
- о технических осмотрах самолётов;
- о странах и городах;
- об авиакомпаниях;
- о маршрутах;
- о рейсах;
- о расписании полетов.

В рамках предметной области все эти объекты могут стать элементами инфологической модели и образовать собой сущности. В процессе функционирования аэропорта сотрудники проходят предполетный медицинский осмотр, характеризующийся датой проверки, частотой сердечных сокращений, давлением, и т.д. Из этих сотрудников формируются летные экипажи, которые выполняют конкретные рейсы по конкретным маршрутам. Рейсы выполняются в соответствии с расписанием. Для выполнения рейсов задействуются конкретные самолеты, которые предварительно проходят технический осмотр. Ответственность за готовность конкретного самолета несет компания-владелец. Рейс выполняется до пункта назначения, который характеризуется географической привязкой, расстоянием до пункта назначения, особенностями взлетно-посадочной полосы и т.д.

					СКФ МТУСИ 09.03.01 13227 ПЗ	Лист
Изм	Лист	№докум.	Подпись	Дата		25

В процессе осуществления полетов информация об объектах изменяется: самолеты могут оказаться задействованными, сотрудники могут выполнять очередные рейсы, экипажи могут изменяться по объективным причинам (отпуск, болезнь, увольнение), накапливается количество пунктов назначения, изменяются работающие компании, растет количество выполненных рейсов по конкретным маршрутам.

С точки зрения пользователей вопросы к базе данных могут быть следующими:

1. По каким маршрутам выполняли рейсы конкретные экипажи;
2. Какие самолеты выполняли рейсы по пунктам назначения;
3. Какие компании выполняют рейсы по конкретным пунктам назначения;
4. Показать список направлений, по которым осуществляются полеты из этого аэропорта;
5. Сколько рейсов по направлениям совершил конкретный экипаж;
6. Кто из членов экипажей проживает в пунктах назначения;
7. Показать список экипажей, выполнивших рейсы в указанные аэропорты за конкретный период времени и т.д.

На основе динамики предметной среды, состава запрашиваемой информации предлагается создать следующие информационные объекты (сущности). Они характеризуются именем и набором свойств (характеристик). Спецификации сущностей представлены в таблице 1.

Таблица 1 - Спецификация сущностей

Название сущности	Атрибуты сущности
Сотрудники	ФИО, должность, адрес, дата рождения, Дата прие-

	ма на работу, паспортные данные, фото, заметки
Экипаж	Состав экипажа по должностям
Медосмотр	Дата, врач, норма/не норма, уточнение причины
Расписание	Дата, пункт назначения, вылет, прилет, время в пути, компания, тип самолет
Самолет	Тип самолета, компания, техническое состояние, бортовой номер, занятость, вместимость, начало эксплуатации
Техосмотр	Дата, кто проводил, норма/не норма, Детали осмотра
Компания	Название, ФИО директора, телефон, адрес
Город	Страна, город, название аэропорта, расстояние, особенности ВПП

На этом этапе можно выполнить также уточнение типа атрибута, но удобнее это сделать при разработке инфологической модели, т.к. в этом случае будет детализированы все атрибуты и выявлены особенности их использования.

Одной из распространенных видов концептуальной модели является модель "сущность-связь" (или ER-модель) [5]. Для рассматриваемой системы необходимые связи представлены в таблице 2.

Таблица 2 - Спецификация связей ER-модели

Название связи	Связываемые сущности	Тип связи
Тех. осмотр	Техосмотр-Самолет	$\infty - 1$
Принадлежит	Самолет-Компания	$\infty - 1$
Используется	Самолет-Расписание	$1 - \infty$
Выполняет рейс	Расписание-Экипаж	$\infty - \infty$

Включает	Экипаж-Сотрудники	1 - ∞
Мед.осмотр	Сотрудники-Медосмотр	1 - ∞
Находится	Город-Расписание	1 - ∞

Графически диаграмма Ег-модели показана на рисунке 1. Знаком "бесконечность" обозначено окончание связи с признаком "ко многим". Второй конец связи, если он без обозначения - соответствует типу окончания "один".



Рисунок 1- Типы связей между сущностями

2 Выполнение нормализации базы данных

Выполнение нормализации относится к этапу разработки инфологической модели и является ее первым шагом. Она выполняется в предположении, что СУБД ориентирована на реляционную модель данных. Цель нормализации - устранение избыточности. Нормализация осуществляется путем последовательного удовлетворения несколькими правилами (нормальным формам). Это может потребовать изменения количества объектов (сущностей, таблиц) и состава атрибутов [2,3].

1 нормальная форма.

Она называется формой "Уникальных полей" и предполагает, что каждое поле таблицы должно представлять уникальный тип информации. Это означает, что должны отсутствовать повторяющиеся по смыслу поля в одной записи, и каждое поле должно быть атомарным.

В соответствии с этим правилом в таблице Сотрудники необходимо поле ФИО разделить на несколько составляющих: на Фамилию, Имя и Отчество, т.е. сделать атомарными. То же самое потребует и поле Адрес: в противном случае невозможен будет отдельный поиск по названию улицы, номеру дома и т.д. Однако такая задача в этой работе не решается, и этим делением можно пренебречь.

Каждый экипаж по составу должен быть стабильным для обеспечения слаженности и взаимозаменяемости. Поэтому каждый сотрудник может принадлежать только одному экипажу. Это потребовало ввести таблицу Екипаж, которая сама несет мало информации, но позволяет компоновать экипажи из произвольного количества членов экипажа, мобильно актуализировать состав экипажа.

					СКФ МТУСИ 09.03.01 13227 ПЗ	Лист
Изм	Лист	№докум.	Подпись	Дата		29

2 нормальная форма.

Эта форма носит название "Первичный ключ" и предполагает, что каждая таблица должна иметь уникальный идентификатор - первичный ключ. В этой работе ключ будет формироваться специальным счетчиком, т.к. содержательная сторона этого поля не имеет значения. При этом уникальность этого поля будет контролировать сама СУБД. Имя каждого ключевого поля будет формироваться из символов ID и имени соответствующей таблицы. Эти же имена будут использованы в качестве вторичных ключей в других таблицах при организации связей. Эти цифровые коды имеют малый размер (тип - двойное целое), что сокращает объем памяти и ускоряет поиск. Все таблицы этой форме удовлетворяют.

3 нормальная форма.

Название этой формы - "Функциональная зависимость". Это означает, что для каждого значения первичного ключа (т.е. в каждой записи конкретной таблицы) в полях данные должны относиться только к объекту этой таблицы и полностью его описывать. Данные, относящиеся к другой сущности (таблице), должны находиться в другой таблице со своим ключом. В предлагаемой модели, например, таблица Расписание должна иметь данные о Самолете, о Городе, об Экипаже, но они организованы в своих одноименных таблицах, имеют свои первичные ключи. Таблица Расписание содержать должна только ссылки на эти дополнительные таблицы. Поэтому предложенная модель (см. рисунок 1) этой форме удовлетворяет.

4. нормальная форма.

Эта форма - "Независимость полей" - обеспечивает возможность изменения значения любого поля, кроме ключевого, без воздействия на данные других полей.

В предложенной инфологической схеме самолет принадлежит конкретной компании. Если название компании изменяется, то в данные о самолетах

					СКФ МТУСИ 09.03.01 13227 ПЗ	Лист
Изм	Лист	№докум.	Подпись	Дата		30

требуется изменить все поля с названием компании. Для удовлетворения требованиям 4 нормальной формы вместо названия компании и ее характеристик используется ссылка на отдельную сущность - Компания. Это позволяет изменить только название в одной записи таблицы Компания без дополнительных изменений еще чего-либо.

Однако в таблице Сотрудники поле Должность может быть изменено при организационно-штатных мероприятиях. Поэтому для этой таблицы необходимо ввести дополнительную таблицу Должность с характеристиками конкретных должностей.

Кроме того, связь сущностей Расписание и Экипаж реализуется по типу "многие ко многим". Такие связи напрямую в СУБД реализовывать недопустимо, потому, что при этом нарушается корректность выполнения операций реляционной алгебры. Поэтому эту связь необходимо представить в виде двух типа "один ко многим" через дополнительную таблицу Рейс с уточнением даты вылета. Предполагается, что экипаж за сутки может выполнить не более одного рейса, хотя введенные связи допускают и несколько рейсов.

Таким образом, указанные требования четырех форм нормализации позволяют избежать нерациональных трудозатрат, уменьшить вероятность ошибок ввода данных, обеспечить непротиворечивость и целостность данных. Нормальных форм существует больше, но на практике достаточно удовлетворить не все. Некоторые формы даже искусственно нарушают в целях увеличения скорости выполнения операций.

					СКФ МТУСИ 09.03.01 13227 ПЗ	Лист
Изм	Лист	№докум.	Подпись	Дата		31

3 Конструирование таблиц базы данных и заполнение полей

С учетом сформулированных требований по нормализации таблиц осуществлена разработка таблиц инфологической модели. В последующих таблицах представлены спецификации таблиц инфологической модели базы данных Аэропорт.

Таблица 3 - Спецификация таблицы Dolgnost (Должность)

Имя поля	Тип данных	Свойства поля
ID_Dolgnost	Счетчик	Ключевое, индексированное, повторения не допускаются
Name_Dolgnost	Короткий текст	Размер - 20, обязательное, из списка

Таблица 4 - Спецификация таблицы Medosmotr (Мед.осмотр)

Имя поля	Тип данных	Свойства поля
ID_Medosmotr	Счетчик	Ключевое, индексированное, повторения не допускаются
Date_Med	Дата/Время	Формат поля – краткий формат даты; маска ввода — 00.00.0000; обязательное
Vrach_Name	Короткий текст	Размер - 50, обязательное
Gotov_Med	Логический	Обязательное, по умолчанию 1
Diagnoz	Текст	Текстовое, Не обязательное
ID_Sotrudnik	Длинное целое	Индексированное, повторения допускаются, автоподстановка

Таблица 5 - Спецификация таблицы Sotrudnik (Сотрудник)

Имя поля	Тип данных	Свойства поля
----------	------------	---------------

ID_Sotrudnik	Счетчик	Ключевое, индексированное, повторения не допускаются
ID_Ekipag	Длинное целое	Индексированное, повторения допускаются, автоподстановка
Fam	Короткий текст	Размер - 50, обязательное
Name_Sotr	Короткий текст	Размер - 50, обязательное
Date_Rogd	Дата/Время	Формат поля – краткий формат даты; маска ввода — 00.00.0000; обязательное
Stag_Sotr	Дата/Время	Формат поля – краткий формат даты; маска ввода — 00.00.0000; обязательное
Adres_Sotr	Короткий текст	Размер - 100, обязательное
ID_Medosmotr	Длинное целое	Индексированное, повторения допускаются, автоподстановка
ID_Dolgnost	Длинное целое	Индексированное, повторения допускаются, автоподстановка
Foto_Sotr	Вложение	Цифровое фото
Gotov	Логический	Обязательное

Таблица 6 - Спецификация таблицы Ekipag (Экипаж)

Имя поля	Тип данных	Свойства поля
ID_Ekipag	Счетчик	Ключевое, индексированное, повторения не допускаются
Nomer_Ekipag	Целое	Размер поля — целое; обязательное. Формальный регистрационный номер экипажа

Таблица 7 - Спецификация таблицы Reys (Рейс)

Имя поля	Тип данных	Свойства поля
ID_Reys	Счетчик	Ключевое, индексированное, повторения не допускаются
ID_Ekipag	Длинное целое	Индексированное, повторения допускаются, автоподстановка
Date_Reys	Дата/Время	Формат поля – краткий формат даты; маска ввода — 00.00.0000; обязательное

Таблица 8 - Спецификация таблицы Raspisanie (Расписание)

Имя поля	Тип данных	Свойства поля
ID_Raspisanie	Счетчик	Ключевое, индексированное, повторения не допускаются
ID_Gorod	Длинное целое	Индексированное, повторения допускаются, автоподстановка
ID_Samolet	Длинное целое	Индексированное, повторения допускаются, автоподстановка
ID_Reys	Длинное целое	Индексированное, повторения допускаются, автоподстановка
Vylet	Дата/Время	Формат поля – краткий формат времени; маска ввода — 00:00; обязательное
Pribytie	Дата/Время	Формат поля – краткий формат времени; маска ввода — 00:00; обязательное

Таблица 9 - Спецификация таблицы Gorod (Город)

Имя поля	Тип данных	Свойства поля
ID_Gorod	Счетчик	Ключевое, индексированное, повторения не допускаются

Aeroport	Короткий текст	Размер - 20, обязательное
Gorod	Короткий текст	Размер - 50, обязательное
Long_km	Числовой	Размер поля —целое; обязательное
Foto_Gorod	Вложение	Цифровое фото

Таблица 10 - Спецификация таблицы Samolet (Самолет)

Имя поля	Тип данных	Свойства поля
ID_Samolet	Счетчик	Ключевое, индексированное, повторения не допускаются
Tip_Samolet	Короткий текст	Размер поля – 15; обязательное
Nomer_Bort	Числовой	Размер поля — целое; обязательное
ID_Kompania	Длинное целое	Индексированное, повторения допускаются, автоподстановка
ID_TO	Длинное целое	Индексированное, повторения допускаются, автоподстановка
Spid	Числовой	Размер поля —целое; обязательное
Foto_Samolet	Вложение	Цифровое фото
Vmestimost	Числовой	Размер поля — целое; обязательное
Nach_Ekspl	Дата/Время	Формат поля – краткий формат даты; маска ввода — 00.00.0000; обязательное

Таблица 11 - Спецификация таблицы Kompania (Компания)

Имя поля	Тип данных	Свойства поля
ID_Kompania	Счетчик	Ключевое, индексированное, повторения не допускаются
Name_Komp	Короткий текст	Размер поля – 30; обязательное

Fam_Komp	Короткий текст	Размер поля – 30; обязательное
Adres_Komp	Короткий текст	Размер - 100, обязательное
Tel_Komp	Числовой	Размер поля —целое; обязательное
Foto_Komp	Вложение	Цифровое фото
Email_Komp	Гиперссылка	Не обязательное

Таблица 12 - Спецификация таблицы ТО (Тех. осмотр)

Имя поля	Тип данных	Свойства поля
ID_TO	Счетчик	Ключевое, индексированное, повторения не допускаются
Date_TO	Дата/Время	Формат поля – краткий формат даты; маска ввода — 00.00.0000; обязательное
Fam_TO	Короткий текст	Размер поля – 50; обязательное
Gotov_TO	Логический	Обязательное, по умолчанию 1
Diagnoz_TO	Текст	Не обязательное

На основе спецификаций (таблицы 3-12) разработаны таблицы с помощью СУБД [4]. Использовались режимы Конструктора и Таблицы. Один из вариантов таблицы представлен на рисунке 2.

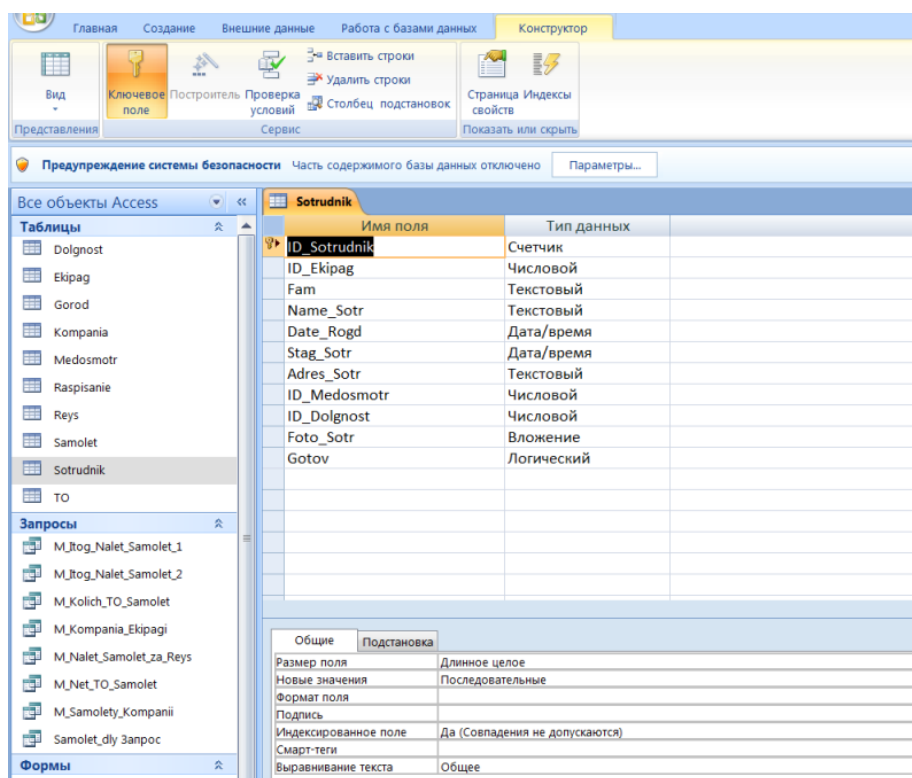


Рисунок 2 - Создание таблицы Sotrudnik в режиме конструктора

Заполнение полей осуществлялось с учетом возможных запросов для осуществления их тестирования. Пример первичного заполнения таблицы Sotrudnik представлен на рисунке 3.

Режим таблицы										
Часть содержимого базы данных отключено										
ID_Sotrudnik	ID_Ekipag	Fam	Name_Sotr	ID_Dolgnost	Date_Rogd	Stag_Sotr	Adres_Sotr	ID_Medosmotr	Голос	Gotov
1	1	Сидоров	Иван	Командир	20.04.1985	01.02.2010	Ростов, Садовая, 10, кв. 27	12.03.2017	0(1)	✓
2	1	Парамед	Александр	2 пилот	15.05.1975	10.02.2005	Ростов, Зорге 20, кв. 15	02.03.2017	0(1)	✓
3	2	Конев	Федор	1 стюарт	25.01.1995	20.07.2015	Азов, Петровская 10	02.03.2017	0(1)	✓
4	2	Гец	Ольга	2 стюарт	15.02.1996	20.07.2015	Азов, Калугина 20, кв. 17	02.03.2017	0(1)	✓
5	2	Порохов	Марат	Командир	16.02.1980	20.02.2007	Шахты, Алмазова 10	02.03.2017	0(1)	✓
6	2	Кузин	Андрей	2 пилот	15.02.1987	20.04.2005	Шахты, Аманды 20	02.03.2017	0(1)	✓
7	1	Борзова	Анна	1 стюарт	15.07.1993	20.02.2011	Азов, петровская 10	12.03.2017	0(1)	✓
8	1	Кашина	Мария	2 стюарт	11.01.1992	12.02.2010	Новгород, Пановой 15	02.03.2017	0(1)	✓
9	3	Пец	Клавдия	1 стюарт	12.03.1991	15.04.2010	Новгород, Стадионная 27	01.03.2017	0(1)	✓
10	3	Кваша	Михаил	Командир	13.01.1985	10.10.2011	Козельск, героев 18	12.03.2017	0(1)	✓
11	3	Морозова	Юлия	2 стюарт	15.04.1990	05.06.2007	Ковров, Иванова 70	02.03.2017	0(1)	✓
12	3	Махов	Петр	2 пилот	10.02.1986	02.02.2006	Москва, Ордынка 20	12.03.2017	0(1)	✓
14	4	Полов	Валентин	Командир	11.12.1989	15.01.2008	Москва, Казанская, 15	02.03.2017	0(0)	✓
15	4	Марков	Андрей	2 пилот	12.09.1995	13.04.2009	Иваново, Вятская, 18	01.03.2017	0(0)	✓
16	4	Воронова	Иветта	1 стюарт	15.09.1999	13.04.2010	Казань, Хибинская 10	17.03.2017	0(0)	✓
17	4	Морошка	Вера	2 стюарт	10.10.1990	13.04.2010	Казань, мартаова, 18	02.03.2017	0(0)	✓
(№)									0(0)	✓

Рисунок 3 - Заполнение таблицы Sotrudnik

4 Оформление схемы данных

По сути, разработка инфологической модели уже последовательно осуществлялась в ходе реализации последних двух разделов. Однако графический образ логической модели может быть получен только на этапе создания схемы данных с помощью СУБД. После создания всех таблиц в СУБД схема данных представлена на рисунке 4.

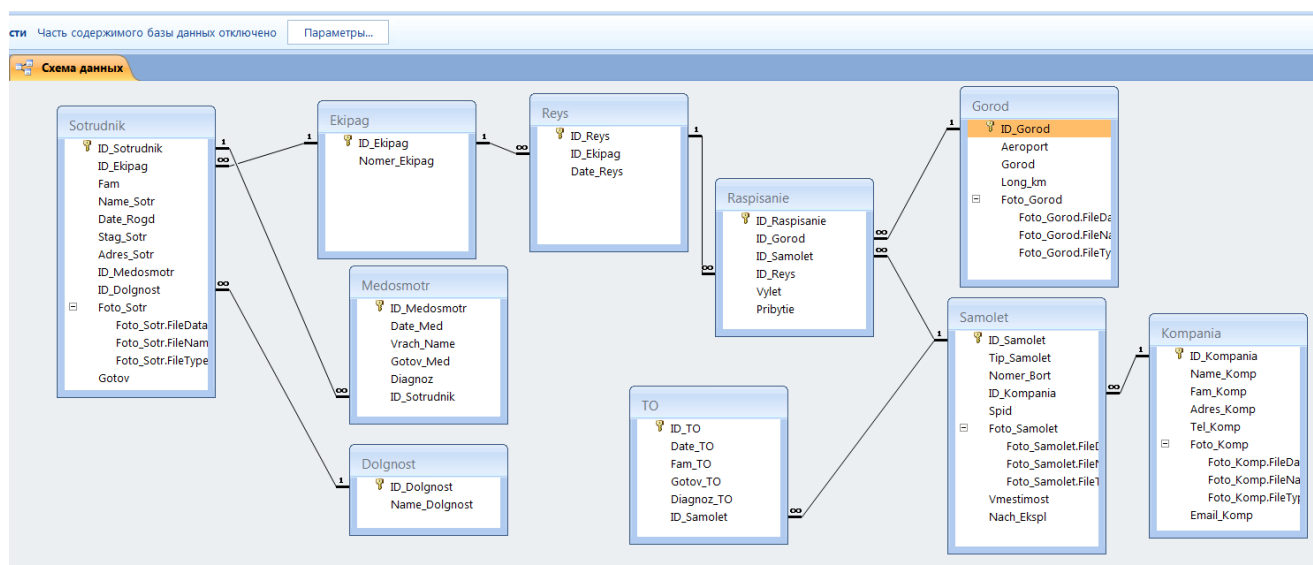


Рисунок 4 - Схема данных БД Аэропорт

5Конструирование запросов к базе данных

Состав возможных типовых запросов к базе данных был предложен в разделе 1. Он может значительно расширяться при необходимости, если состав имеющихся полей это допускает. В противном случае потребуется изменить структуру таблиц. Кроме того состав запросов диктуется необходимой информацией для организации требуемых форм. Поэтому до полной разработки всех объектов базы данных список запросов завершенным считать сложно. Поэтому были реализованы запросы из имеющегося списка раздела 1 в количестве, определенном заданием [4, 5].

Запрос 1. Показать самолеты, совершавшие рейсы в конкретный город (запрос с параметром)

Этот запрос является многотабличным: привлекаются таблицы Gorod, Samolet, Raspisanie. Это простой запрос на выборку из нескольких таблиц с параметром. Параметр задается по названию города. На рисунке 5 представлен бланк запроса Gorod_Samolet и результат его выполнения.

The screenshot displays a database management system interface. At the top, there are tabs for 'Схема данных', 'S_Gorod_Vrema_v_Polete', and 'S_Gorod_Samolet'. The main area shows a schema diagram with three tables: 'Gorod', 'Raspisanie', and 'Samolet'. 'Gorod' has fields: ID_Gorod, Aeroport, Gorod, Long_km, Foto_Gorod, and Foto_Gorod.Fil. 'Raspisanie' has fields: ID_Raspisanie, ID_Gorod, ID_Samolet, ID_Reys, Vylet, and Pribytie. 'Samolet' has fields: ID_Samolet, Tip_Samolet, Nomer_Bort, ID_Kompania, Spid, Foto_Samolet, Foto_Samolet.File1, Foto_Samolet.File2, Foto_Samolet.File3, Vmestimost, and Nach_Ekspl. Below the schema, a query window is open with the following fields: 'Город', 'Tip_Samolet', 'Nomer_Bort', 'Vmestimost', and 'Nach_Ekspl'. The 'Город' field is selected, and the query is executed. The results are shown in a table with columns: 'Город', 'Tip_Samolet', 'Nomer_Bort', 'Vmestimost', 'Nach_Ekspl', and a link icon. The results are as follows:

Город	Tip_Samolet	Nomer_Bort	Vmestimost	Nach_Ekspl	Link
Ленинград	A-380	88888	250	10.01.2000	🔗(1)
Ленинград	A-380	55884	250	31.01.2002	🔗(1)
Ленинград	ИЛ-62	23456	102	15.07.2010	🔗(1)
					🔗(0)

Рисунок 5 - Бланк запроса Gorod_Samolet и результаты его выполнения

В этом запросе в качестве параметра введена строка "Ленинград".

Запрос 2. Показать экипажи, которые совершали рейсы в конкретные города (запрос с одним условием).

В этом запросе участвуют довольно много таблиц: Gorod, Samolet, Raspisanie, Reys, Ekipag, Sotrydnik, Dolgnost. Все они связывают город с конкретным командиром экипажа, который выполнял рейс в этот город. Без всяких условий такой запрос выведет всех членов экипажей с привязкой к названию города. Чтобы сократить число выводимых записей и, учитывая, что экипажи стабильны по составу, можно ввести условие: выводить только фамилию командира экипажа - "Командир". Кроме того, экипаж мог совершать полеты в этот город многократно. Поэтому для вывода только одного упоминания о совершенном рейсе в этот город следует ввести признак групповой операции и добавить функцию подсчета числа записей (рейсов в этот город) в поле ID_Gorod. Бланк запроса и результат выполнения запроса Goro_Ekipagi представлены на рисунке 6.

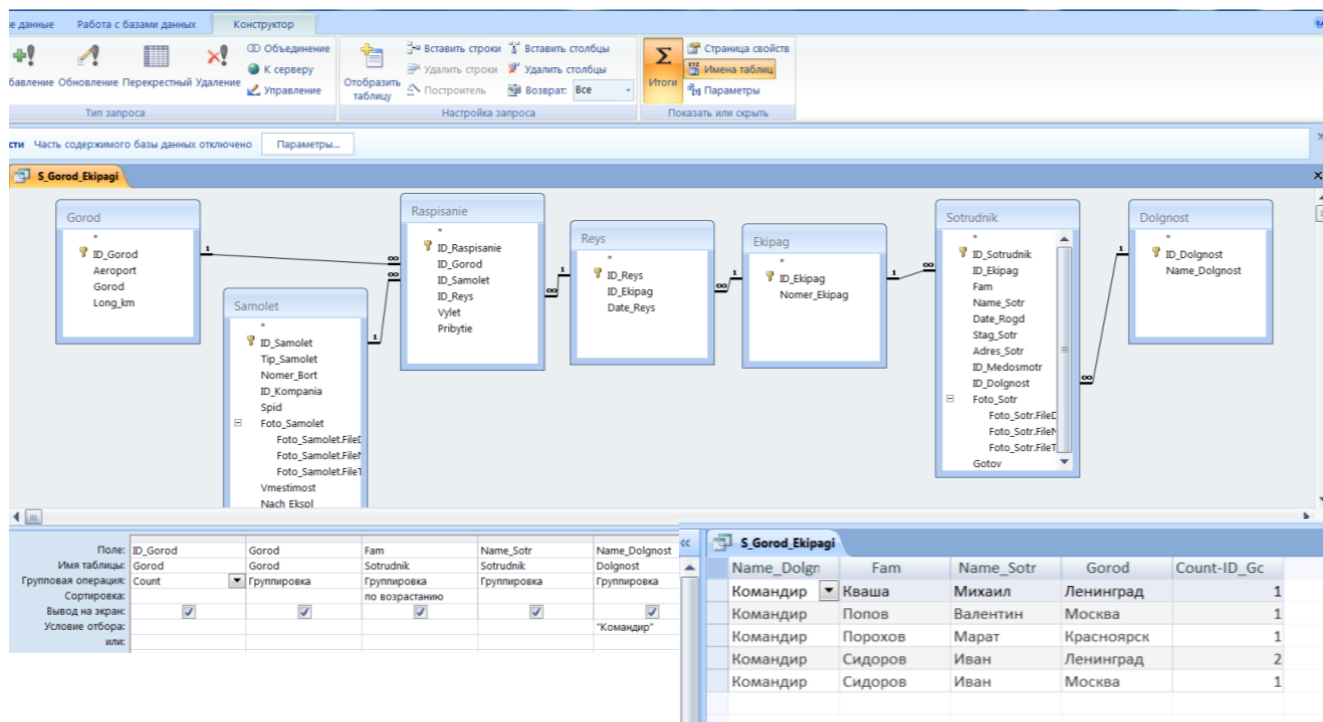


Рисунок 6 - Бланк запроса Gorod_Ekipagi и результаты его выполнения

Запрос 3. Показать компании, обслуживающие рейсы по направлениям (запрос с групповой операцией).

Этот запрос использует таблицы Kompania, Samolet, Raspisanie, Gorod. Для исключения многократных записей об одной и той же компании, которая совершала рейсы в один и тот же город, следует ввести групповую операцию. Поле Gorod необходимо сортировать по алфавиту.

Бланк запроса и результат его выполнения представлены на рисунке 7.

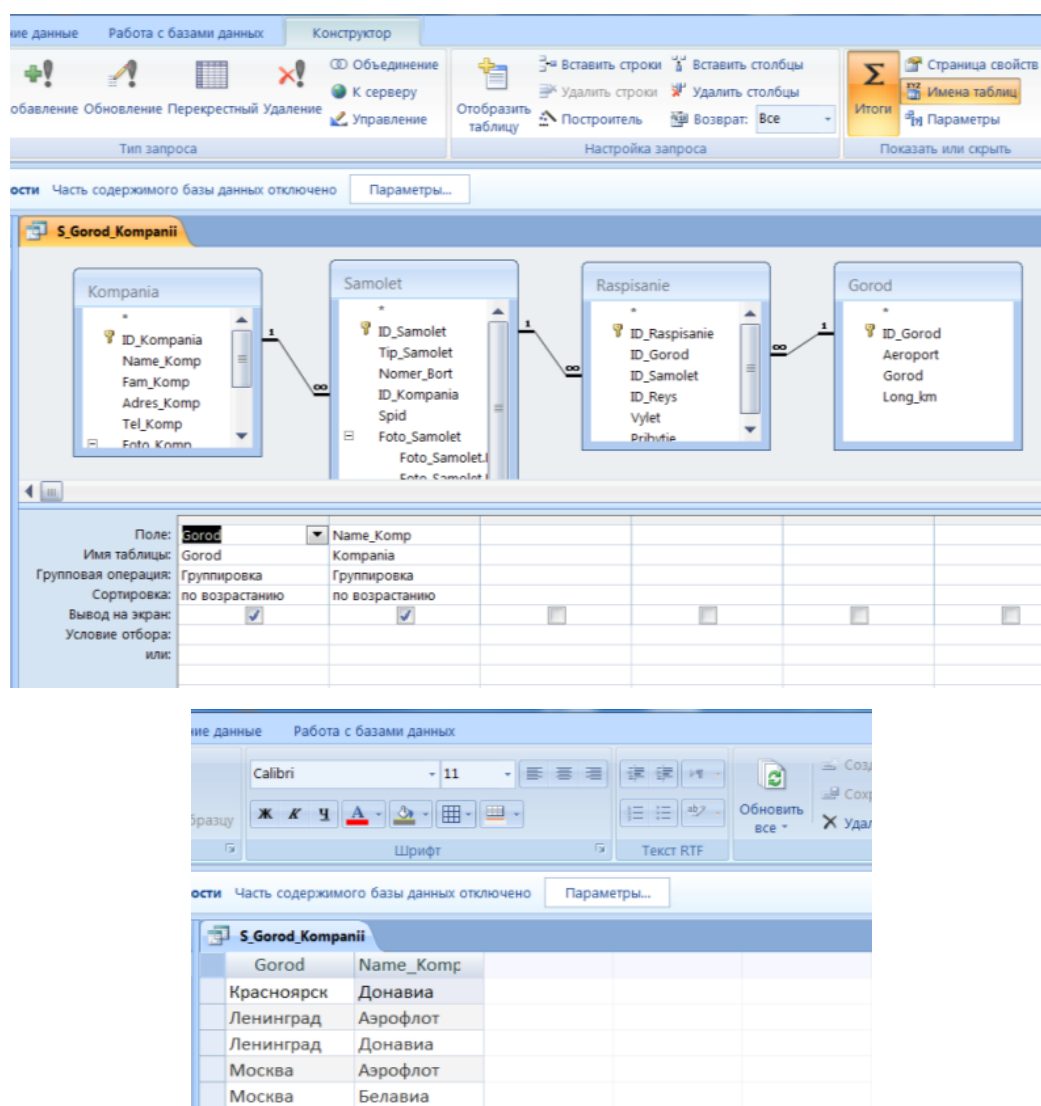


Рисунок 7 - Бланк и результаты запроса Gorod_Kompanii

Запрос 4. Показать список экипажей, выполнивших рейсы в аэропорты за конкретный период времени (запрос с двумя условиями и групповой операцией).

Запрос требует использования таблиц Gorod, Raspisanie, Reys, Ekipag, Sotrydник, Dolgnost. Для ограничения числа записей с учетом стабильности экипажей возможно выводить только имя командира. Для этого в условия отбора поля Name_Dolgnost необходимо указать "Командир". Для подсчета количества рейсов за задаваемый период необходимо ввести строку Группировка с указанием функции Count. Период времени формируется в поле Date_Reys с помощью построителя выражений. Расчетная формула представлена на рисунке 8. В данном случае выбираются рейсы за прошлый год.



Рисунок 8 - Формула для задания условия отбора в период прошлого года

Бланк запроса для получения такого списка представлен на рисунке 9.

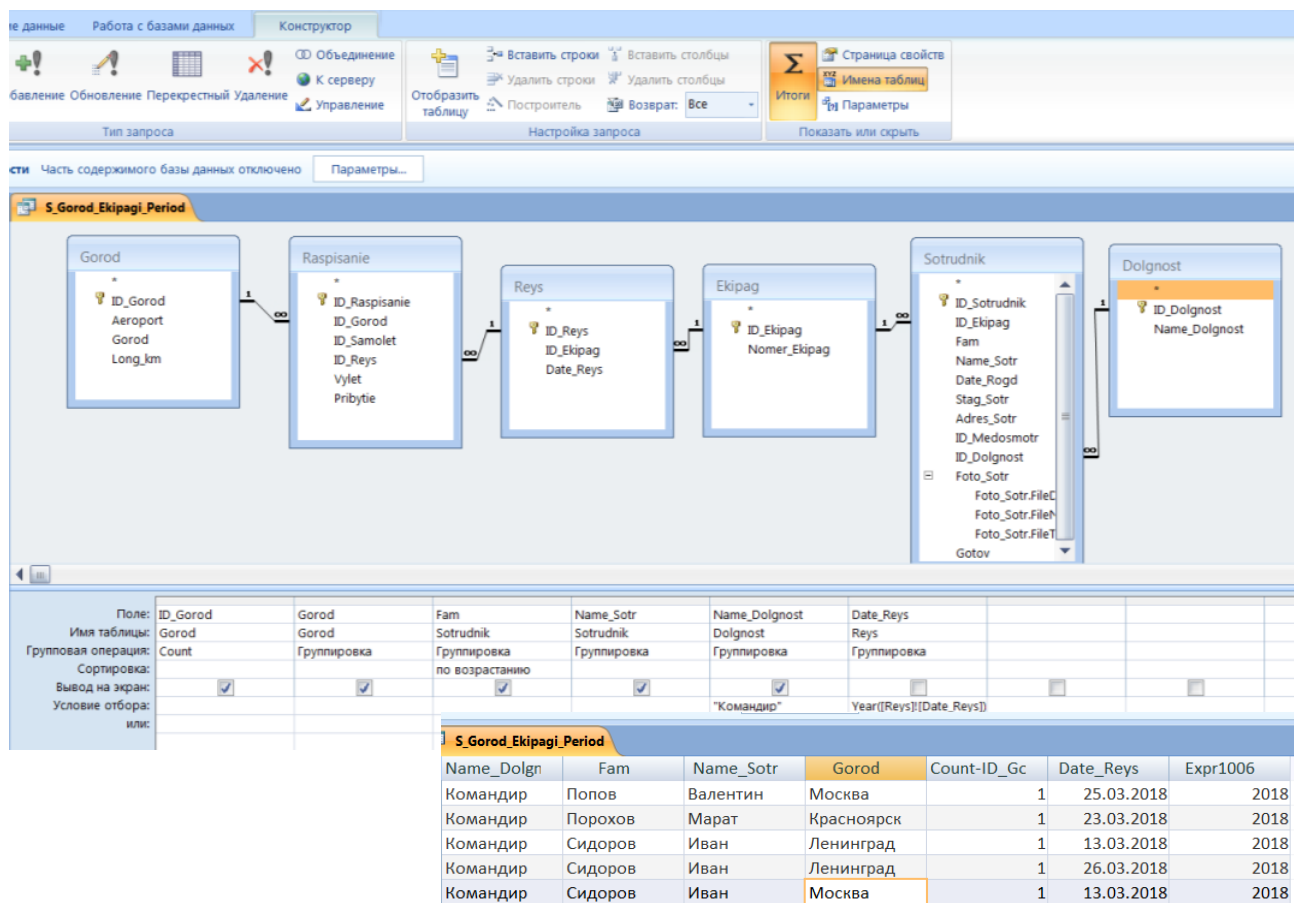


Рисунок 9 - Бланки результат запросаGorod_Ekipagi_Period

Запрос 5. Показать список направлений, по которым совершаются рейсы из аэропорта, тип самолета, время в пути (запрос с оператором Like).

Этот запрос предполагает вычисление времени нахождения в воздухе с учетом расстояния по выполняемому рейсу, скорости того тип самолета, на котором рейс выполняется. Оператор Like позволяет отфильтровать тип конкретного воздушного судна. При этом для поиска значений полей будет использоваться шаблон и допустимо вводить только часть соответствующих значений. В данном случае не всю строку типа судна: только одну букву и одну цифру.

Запрос предполагает построение запроса с вычисляемым полем для расчета времени полета каждого самолета по каждому рейсу. Названия городов

следует сортировать по алфавиту. Для исключения повторов строк необходимо использовать групповые операции. Бланк такого запроса представлен на рисунке 10. Расчетная формула для вычисляемого поля представлена на рисунке 11.

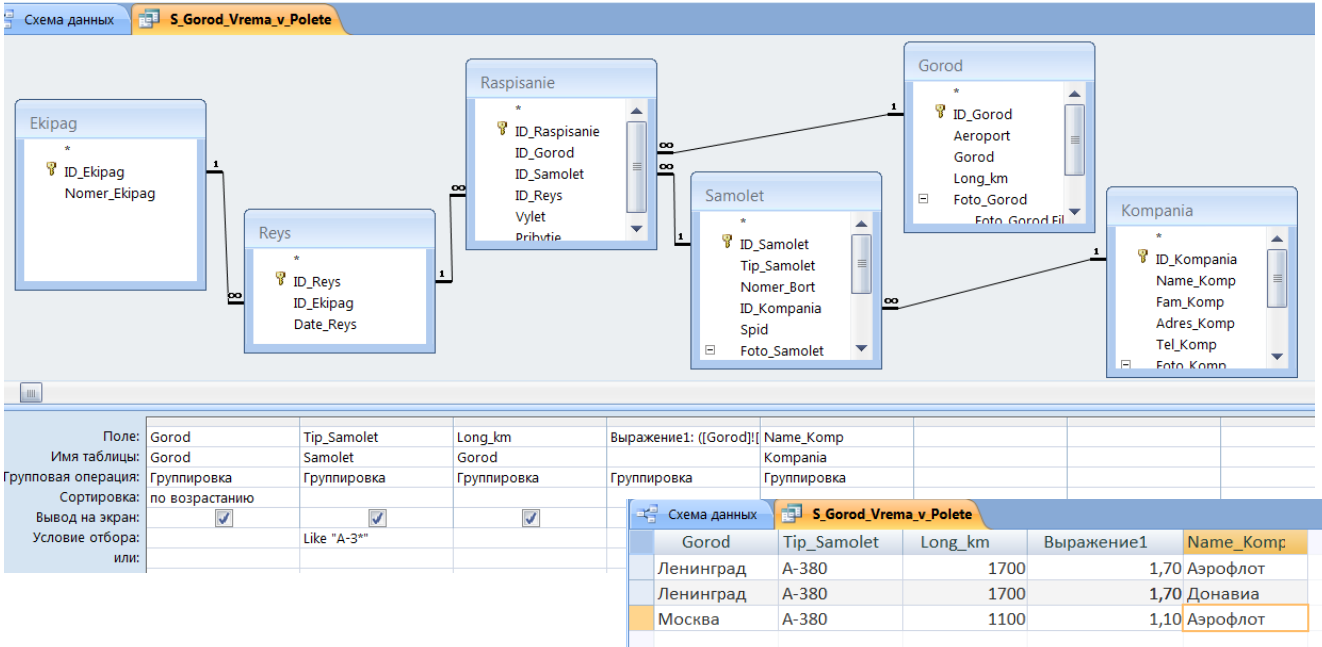


Рисунок 10 - Бланк запроса и результаты для отображения рейсов, времени в полете и компании

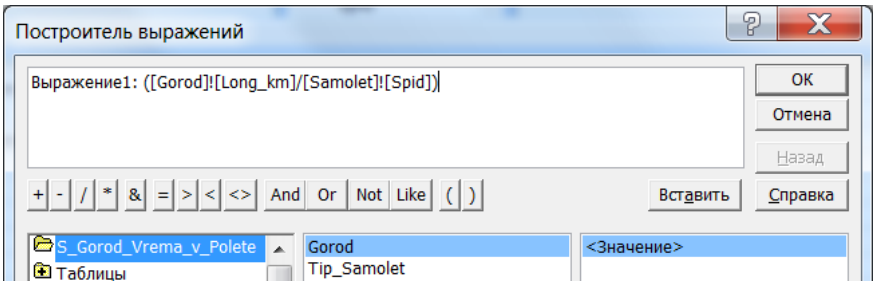


Рисунок 11 - Формула для расчета времени полета

6 Разработка форм базы данных

Работа с базой данных непосредственно в форме таблиц или запросов сложна. Их заменяют на объекты, которые называются формами. Они позволяют организовать удобный доступ к данным, выполнять все допустимые операции с самими данными, организовать понятный и наглядный интерфейс.

Используя формы, можно осуществить первоначальный ввод данных в таблицы, просмотр и редактирование записей в привычном для пользователя виде, напоминающем обычный документ, при этом выполнение многих операций упрощается, а присутствие на экране только нужной информации помогает не отвлекаться от сути операций.

В том случае, когда данных слишком много, чтобы размещать их в обозримом пространстве без использования движков, используют многостраничные формы. Для их создания удобно использовать набор вкладок. Допустим также разрыв страницы, но это удобно больше для печати отчетов.

Для создания формы Аэропорты на вкладке Создание в группе Формы выбрать элемент Конструктор форм. В окне Список полей выбрать и перенести на форму поля таблицы Gorod: ID_Gorod, Aeroport, Gorod, Long_km, Foto_Gorod. Расположить их друг под другом. Эти данные будут отображаться по каждому аэропорту по мере продвижения по списку с помощью стрелок.

Если список полей при конструировании не отображается, то его можно вызвать, выполнив Alt+F8.

Форму можно отредактировать по своему усмотрению, разместить на ней стилизованный рисунок, отображающий принадлежность к структурам аэропортов, изменить название формы, которое будет отображаться при ее открытии, изменить фоновый рисунок и т.д. Все эти манипуляции осуществляются в окне свойств соответствующего элемента формы в области заголовка или основных данных. Изменять геометрию выбранных элементов на форме

					СКФ МТУСИ 09.03.01 13227 ПЗ	Лист
Изм	Лист	№докум.	Подпись	Дата		45

рационально в режиме Макета. В этом случае все изменения элементов формы сразу видны на общем изображении.

Для организации вкладок в режиме Конструктор форм следует выбрать элемент Вкладка, поместить его на форму, придать необходимые размеры. В этом случае будет размещено две вкладки. Для добавления последующих вкладок необходимо выбрать элемент управления Вставить вкладку.

В окне свойств на вкладке Макет в поле Подпись внести подписи всех вкладок. Операции выполнены должны быть при выбранных соответствующих объектах Вкладки. Для главной формы Аэропортов подписи вкладок создать: Экипажи, Самолеты, Компании.

Для организации информационного поля соответствующей вкладки требуется указать имена полей, которые должны отобразиться на этой вкладке.

Если информационное поле вкладки образуют поля, которые принадлежат источнику основной формы, то они перемещаются на вкладку непосредственно из списка полей этого источника, например, таблицы.

Если же на этой форме на вкладке требуется отобразить данные из других источников данных, то создается подчиненная форма. Но изначально такая форма должна быть создана на основе таблицы или запроса. Для реализации главной формы Аэропорты (Aeroporty) созданы формы Gorod_Ekipagi, Gorod_SamoletiGorod_Kompanii на основе одноименных запросов.

Для отображения информации на вкладке формы она переводится в режим конструктора. Затем выбирается объект форма Gorod_Kompanii. На вкладке Создание в группе Формы выбирается элемент Подчиненная форма. Образовавшийся курсор переносится на выбранную вкладку и растягивается до необходимых размеров. Кроме того для подчиненной формы в Окне свойств на вкладке Данные формируются следующие записи: В окне Основное поле - ID_Gorod, в окне Подчиненные поля - ID_Gorod в окне Объект источник - форма Gorod_Kompanii. После выполнения всех операций получится форма, представленная на рисунке 13.

					СКФ МТУСИ 09.03.01 13227 ПЗ	Лист
Изм	Лист	№докум.	Подпись	Дата		46

Такая организация форм позволяет, перебирая названия аэропортов с помощью стрелок навигации, иметь возможность при необходимости уточнять данные самолетов, совершающих рейсы в этот аэропорт, либо данные по экипажам, которые выполняли рейсы в этот аэропорт, либо данные о компаниях, которые обеспечивают полеты в эти аэропорты.

В подчиненных формах с заголовками Экипажи и Самолеты (см. рисунок 14 и 15 соответственно) возможно пролистывать записи собственными элементами навигации. Однако все имеющиеся записи будут иметь отношение только к выбранному сейчас названию компании. По мере наращивания количества самолетов и экипажей количество записей может существенно увеличиться, поэтому при необходимости в таблицы можно выводить ограниченное число записей за конкретный период. Для этого придется сформировать иные подчиненные формы, по аналогии запросу, который показывал рейсы самолетов за период времени.

Состав информации по каждой форме может быть расширен, исходя из требований заинтересованных подразделений, отдела кадров и других органов.

					СКФ МТУСИ 09.03.01 13227 ПЗ	Лист
Изм	Лист	№докум.	Подпись	Дата		47

Работа с базами данных

Часть содержимого базы данных отключено

S_Gorod_Kompanii S_AEROPORT


АЭРОПОРТ

Код: 1

Аэропорт: Пулково

Город: Ленинград

Удаленность, км: 1700



Экипажи Самолеты Компании

КОМПАНИИ

Название	Директор	Тел:	Электронный адрес
Аэрофлот	Иванов Иван Иванович	(495)223-55-55	www.aeroflot.ru
Донавиа	Капустин Федор Васильевич	(495)223-55-55	www.aeroflot-don.ru

Запись: 1 из 2 Нет фильтра Поиск

Рисунок 13 - Форма Компании с открытой вкладкой Компании

Работа с базами данных

Часть содержимого базы данных отключено

S_Gorod_Kompanii S_AEROPORT


АЭРОПОРТ

Код: 1

Аэропорт: Пулково

Город: Ленинград

Удаленность, км: 1700



Экипажи Самолеты Компании

ЭКИПАЖИ

Фамилия	Имя	Должность	Кол-во рейсов
Кваша	Михаил	Командир	1
Сидоров	Иван	Командир	2

Запись: 1 из 2 Нет фильтра Поиск

Рисунок 14 - Форма Аэропорты с открытой вкладкой Экипажи


Работа с базами данных

Шрифт Текст RTF Обновить все Записи Создать Итоги Сохранить Орфография Удалить Дополнительно Фильтр Применить фильтр Сортировка и фильтр Найти Перейти Выбрать

Часть содержимого базы данных отключено Параметры...

АЭРОПОРТ

Код: 2
Аэропорт: Шереметьево
Город: Москва
Удаленность, км: 1100



Экипажи Самолеты Компании

САМОЛЕТЫ

Тип	Борт.номер	Вместимость	Начало эксплуатации
A-380	88888	250	10.01.2000
ЯК-40	34774	30	20.05.1999

Записи: 1 из 2 Нет фильтра Поиск

Рисунок 15 - Форма Аэропорты с открытой вкладкой Самолеты

7 Разработка отчета

Отчёт является средством документирования БД. Он позволяет выбрать из базы данных требуемую пользователем информацию и оформить ее в виде документов, которые можно просмотреть и напечатать. Данные для отчетов извлекаются из таблицы или запроса. Кроме этого, в отчете могут отображаться вычисленные по исходным данным значения, например итоговые суммы.

Отчеты и формы имеют много общего. Однако, в отличие от форм, отчеты не предназначены для ввода и правки данных в таблицах. Они позволяют лишь просматривать и печатать данные. В отчете невозможно изменить исходные данные с помощью элементов управления, как это можно сделать с помощью форм. В отчетах можно использовать такие же элементы управления для указания состояния переключателей, флажков и списков.

Отчёт – достаточно гибкий инструмент. Он позволяет не только определить состав полей, выводимых на печать, но и изменять структуру вывода, форматировать документ и добавлять различную служебную информацию, также выводимую на печать. Это могут быть фирменные знаки, служебные надписи, подписи, элементы управления и т.д.

Для отчета о списочном составе самолетов авиапредприятия следует за основу взять запрос `Samolety_dly`.

Для построения отчета необходимо:

1. Выбрать запрос `Samolety_dly`;
2. Выполнить в закладке Создание группы Отчеты инструмент Мастер отчетов;
3. В группе Таблицы и отчеты выбрать Запрос: `Samolety_dly`, если он там не отобразился;
4. Переместить в окно Выбранные поля необходимые данные: `ID_Samolet`, `Tip_Samolet`, `Nomer_Bort`, `Name_Kompania`, `Nach_Ekspl`;

					СКФ МТУСИ 09.03.01 13227 ПЗ	Лист
Изм	Лист	№докум.	Подпись	Дата		50

5. Выбрать Kompania в виде представления данных;

6. Добавить Уровни группировки - Kompania. При отсутствии такой возможности в Мастере группировку возможно установить по выбранным данным в режиме конструктора после окончания процедуры. Для этого следует выбрать параметр группирования, нажать правую кнопку мыши и выбрать в строке Итог требуемую процедуру (Число записей);

7. Порядок сортировки не изменять, макет - ступенчатый, ориентация книжная;

10. Стилль оформления - Модульная;

10. Задать Имя отчета - Kompania;

В порядке редактирования отчета необходимо:

- отредактировать фон, цвет, размер шрифта и размещение заголовка и верхнего колонтитула;

- для оформления отчета следует разместить элемент Рисунок для вставки логотипа или стилизованного рисунка, выделить фон надписей для отделения от содержательного текста (возможно использование элемента Линия для отделения списка сотрудников от шапки отчета),;

- в области нижнего колонтитула разместить подпись ответственного лица с помощью инструмента Надпись;

- отредактировать размер шрифта и размещение полей в области данных;

- в режиме макета по полю Tip_Samolet следует установить порядок сортировки записей;

- расстояние между строчками подобрать экспериментально;

- текущую дату перенести в нижний колонтитул. При ее отсутствии с помощью инструмента Поле задать данные в виде стандартной функции =Now();

- в области нижнего колонтитула при отсутствии разместить поясняющий текст с помощью инструмента Поле. Для него в Окне свойств во вкладке

					СКФ МТУСИ 09.03.01 13227 ПЗ	Лист
Изм	Лист	№докум.	Подпись	Дата		51

Данные построить строку, используя операцию конкатенации и стандартные функции работы со страницами: ="Страница " & [Page] & " из " & [Pages].

После выполнения всех манипуляций получается отчет, который представлен на рисунке 16. Реально при печати подпись начальника будет размещена внизу текста.



Самолеты АЭРОПОРТА

Компания	Код	Тип судна	Бортовой номер	Начало эксплуатации
Аэрофлот				
	9	ТУ-154	22334	15.05.2009
	8	А-380	88888	10.01.2000
	1	ИЛ-62	23456	15.07.2010
	Итого по компании			3
Белавиа				
	2	ЯК-40	34774	20.05.1999
	Итого по компании			1
Донавиа				
	6	А-320	77777	13.03.2005
	5	А-380	55884	31.01.2002
	4	Як-40	44578	15.04.2001
	3	ТУ-154	66577	25.03.2000
	Итого по компании			4
	Всего самолетов			8

20 января 2019 г. Начальник инженерной службы _____ **Ф.Кудолесов**

Страница 1 из 1

Рисунок 16 - Список самолетов аэропорта в виде отчета с группировкой

					СКФ МТУСИ 09.03.01 13227 ПЗ	Лист
						52
Изм	Лист	№докум.	Подпись	Дата		

ЗАКЛЮЧЕНИЕ

Таким образом, на основе анализа информационной предметной среды аэропорта удалось построить инфологическую модель реляционной базы данных, выполнить ее нормализацию и реализовать базу данных в среде Access на основе среды визуального проектирования.

Совокупность использованных таблиц и их связей обеспечили возможность организации необходимых форм, которые представляют полную информацию о составе самолетов аэропорта, данных о техническом сопровождении полетов, выполненных рейсах и общем налете самолетов.

Примеры подготовленных отчетов продемонстрировали достаточно широкие возможности оформления типовых выходных документов на основе хранящейся в базе данных информации.

Имеющиеся возможности информационной модели позволяют значительно расширить объем предоставляемой информации. Форму и способы ее визуализации возможно изменить по требованию заказчика.

Задание на курсовое проектирование выполнено в полном объеме.

					СКФ МТУСИ 09.03.01 13227 ПЗ	Лист
Изм	Лист	№докум.	Подпись	Дата		53

СПИСОК ЛИТЕРАТУРЫ

1. Технологии баз данных. Методические указания по выполнению курсового проекта. Ростов-на-Дону: СКФ МТУСИ, 2019.
2. Гурвиц Г. А. Разработка приложений на реальном примере. СПб.: БХВ-Петербург, 2010. — 496 с.
3. Зрюмов Е. А. Базы данных для инженеров: учебное пособие /Е. А. Зрюмов, А. Г. Зрюмова; Алт. гос. техн. ун-тет. И. И. Ползунова. –Барнаул: Изд-во АлтГТУ, 2010. – 131 с.
4. Сеннов А. Access 2010. Учебный курс. СПб.: Питер, 2010. - 288 с.
5. Бейли Л. Изучаем SQL. СПб.: Питер, 2012. — 592 с.

					<i>СКФ МТУСИ 09.03.01 13227 ПЗ</i>	<i>Лист</i>
<i>Изм</i>	<i>Лист</i>	<i>№докум.</i>	<i>Подпись</i>	<i>Дата</i>		54