

Министерство Науки и высшего образования Российской Федерации
Филиал федерального государственного бюджетного образовательного
учреждения высшего образования «Ростовский государственный
экономический университет (РИНХ)»
в городе Кисловодске Ставропольского края

Методические указания по выполнению и защите курсового проекта
по программе профессионального модуля

ПМ.02 РАЗРАБОТКА И АДМИНИСТРИРОВАНИЕ БАЗ ДАННЫХ

по МДК 02.02 РАЗРАБОТКА И АДМИНИСТРИРОВАНИЕ БАЗ ДАННЫХ

СПЕЦИАЛЬНОСТИ

09.02.03 ПРОГРАММИРОВАНИЕ В КОМПЬЮТЕРНЫХ СИСТЕМАХ

ВВЕДЕНИЕ

В соответствии с федеральным государственным образовательным стандартом среднего профессионального образования III поколения к минимуму содержания и уровню подготовки выпускников по специальности 09.02.03 «Программирование в компьютерных системах» в ходе освоения профессионального модуля **должен:**

иметь практический опыт:

- работы с объектами базы данных (БД) в конкретной системе управления базами данных;
- использования средств заполнения базы данных;
- использования стандартных методов защиты объектов базы данных;

уметь:

- создавать объекты баз данных в современных системах управления базами данных и управлять доступом к этим объектам;
- работать с современными case-средствами проектирования баз данных;
- формировать и настраивать схему базы данных;
- разрабатывать прикладные программы с использованием языка SQL;
- создавать хранимые процедуры и триггеры на базах данных;
- применять стандартные методы для защиты объектов базы данных;

знать:

- основные положения теории баз данных, хранилищ данных, баз знаний;
- основные принципы построения концептуальной, логической и физической модели данных;
- современные инструментальные средства разработки схемы базы данных;
- методы описания схем баз данных в современных системах управления базами данных (СУБД);

- структуры данных СУБД, общий подход к организации представлений, таблиц, индексов и кластеров;
- методы организации целостности данных;
- способы контроля доступа к данным и управления привилегиями;
- основные методы и средства защиты данных в базах данных;
- модели и структуры информационных систем;
- основные типы сетевых топологий, приемы работы в компьютерных сетях;
- информационные ресурсы компьютерных сетей;
- технологии передачи и обмена данными в компьютерных сетях;
- основы разработки приложений баз данных

Курсовое проектирование является завершающим этапом изучения междисциплинарного курса «Технология разработки и защиты баз данных», в ходе которого осуществляется обучение применению полученных знаний и умений при решении комплексных задач, связанных с профессиональной деятельностью будущих специалистов.

Данные методические указания содержат общие требования и рекомендации к курсовому проектированию для студентов специальности 09.02.03 «Программирование в компьютерных системах».

1. Цели и задачи курсового проектирования

Самостоятельная разработка курсового проекта подготавливает студента к успешному выполнению других курсовых проектов, дипломного проекта и является важным этапом в формировании конкурентоспособного специалиста.

Выполнение студентом курсового проекта проводится с **целью**:

- систематизации и закрепления полученных теоретических знаний и практических умений по общепрофессиональным и специальным дисциплинам специальности «Программирование в компьютерных системах»;
- углубления теоретических знаний в соответствии с заданной темой;
- формирования умений применять теоретические знания при решении поставленных вопросов;

- формирования умений использовать справочную, нормативную и правовую документацию;
- развития творческой инициативы, самостоятельности, ответственности и организованности;
- развития навыков работы в коллективе.

Задачами курсового проекта являются:

- изучение особенностей конкретной предметной области, относящихся к теме курсового проекта (работы);
- анализ возможных подходов и методов решения с обоснованием выбранного метода;
- выбор или разработка модели (математической, структурной, информационной и т.д.), необходимой для достижения цели;
- выбор эффективных алгоритмов с учётом их точности, устойчивости, сходимости и т.д.;
- разработка базы данных и приложений к ней;
- анализ полученных результатов работы БД;
- разработка программной и эксплуатационной документации.

2. Выбор темы

Тематика курсовых работ (проектов) разрабатывается преподавателями учебного заведения, рассматривается и принимается на заседании ЦМК специальности, утверждается директором филиала. Тема курсовой работы (проекта) может быть предложена студентом в соответствии с его интересами и возможностями раскрытия предлагаемой проблемы, при условии обоснования им ее целесообразности.

Допускается выполнение курсовой работы (проекта) по одной теме группой студентов.

Курсовая работа (проект) может стать составной частью – разделом, главой - дипломной работы студента.

3. Темы курсовых проектов

Основным направлением курсовых проектов является «Проектирование баз данных для конкретных предметных областей». Ниже приведены примеры возможных тем:

1. Проектирование БД для работника склада (варианты: склад торговой организации, занимающейся сбытом как продукции собственного производства, так и продукции внешних поставщиков; склад оптовой торговой организации; склад готовой продукции; склад сырья и материалов и др.)-
2. Проектирование БД для контроля выполнения нагрузки преподавателей ВУЗа.
3. Проектирование БД для контроля сессионной успеваемости студентов ВУЗа.
4. Проектирование БД для учета контингента студентов ВУЗа.
5. Проектирование БД для организации дипломного проектирования в ВУЗе.
6. Проектирование БД для организации курсового проектирования.
7. Проектирование БД для профкома ВУЗа.
8. Проектирование БД для начисления стипендии в ВУЗе.
9. Проектирование БД для библиотеки ВУЗа.
10. Проектирование БД для управления работой компьютерных аудиторий учебного заведения.
11. Проектирование БД для управления работой класса свободного доступа.
12. Проектирование БД для начисления заработной платы преподавателей.
13. Проектирование базы данных Ученого совета по защите диссертаций.
14. Проектирование базы данных Отдела аспирантуры.
15. Проектирование БД для контроля успеваемости школьников.
16. Проектирование БД детского сада.
17. Проектирование БД спортивной школы.
18. Проектирование БД центра детского творчества.
19. Проектирование БД партнеров софтверной фирмы.
20. Проектирование БД коммерческого учебного центра.
21. Проектирование БД для учета домашних финансов.
22. Проектирование БД для домашней библиотеки.

23. Проектирование БД для расчета заработной платы (варианты: преподавателей ВУЗа, всех сотрудников ВУЗа, предприятий /организаций с разными системами оплаты труда).
24. Проектирование БД для районной библиотеки.
25. Проектирование БД для домашней видеотеки.
26. Проектирование БД для пункта проката видеофильмов.
27. Проектирование БД кинотеатра.
28. Проектирование БД драматического театра.
29. Проектирование БД для домашней аудиотеки.
30. Проектирование БД тренера спортивной команды.
31. Проектирование БД агентства по аренде квартир.
32. Проектирование БД риэлтерского агентства.
33. Проектирование БД для учета услуг, оказываемых юридической консультационной фирмой.
34. Проектирование БД для автосервисной фирмы.
35. Проектирование БД для автозаправочной станции.
36. Проектирование БД центра по продаже автомобилей.
37. Проектирование БД таксомоторного парка.
38. Проектирование БД по подсистеме «Кадры» (варианты: для ВУЗа, школы, промышленного предприятия, торговой фирмы, софтверной фирмы и т.п.).
39. Проектирование БД службы знакомств.
40. Проектирование базы данных туристического агентства.
41. Проектирование базы данных туристического оператора.
42. Проектирование базы данных туристического клуба.
43. Проектирование БД районной поликлиники. Подсистема «Работа с пациентами».
44. Проектирование БД районной поликлиники. Подсистема «Учет льготных лекарств».
45. Проектирование БД районной поликлиники. Подсистема «Планирование и учет работы медицинского персонала».

46. Проектирование БД районной поликлиники. Подсистема «Учет пациентов».
47. Проектирование БД больницы. Подсистема «Работа с пациентами».
48. Проектирование БД больницы. Подсистема «Лекарственное обеспечение».
49. Проектирование базы данных аптеки.
50. Проектирование БД гостиницы. Подсистема «Работа с клиентами».
51. Проектирование базы данных дачного кооператива.
52. Проектирование базы данных Издательства. Подсистема «Работа с авторами».
53. Проектирование БД Издательства. Подсистема «Служба маркетинга».
54. Проектирование базы данных Учета расчетов с клиентами в банке.
55. Проектирование базы данных строительной фирмы.
56. Проектирование базы данных городской телефонной сети. Подсистема «Учет расчетов с клиентами».
57. Проектирование базы данных торговой организации.
58. Проектирование базы данных аэропорта.
59. Проектирование базы данных ГИБДД.
60. Проектирование базы данных фотоцентра.

Помимо выше приведенных тем студенты могут предложить свою предметную область.

4. Содержание и этапы выполнения курсового проекта

Курсовые проекты по любой теме обязательно должны включать проектную часть, которая состоит из разработки базы данных и создания программного приложения в визуальной среде разработки. Для выполнения курсового проекта может быть выбрана любая СУБД (например, MS Access, SQL Server) и любая среда разработки (например, Borland Delphi).

Основной тематикой курсовых проектов является «Проектирование баз данных для конкретных предметных областей». Проекты должны содержать следующие разделы:

1. Описание предметной области. Постановка задачи.
2. Разработка базы данных.
 - 2.1. Построение инфологической (концептуальной) модели предметной области.
 - 2.2. Выбор средств/методологии проектирования. Реализация проекта в среде конкретной СУБД.
3. Разработка программного приложения.
 - 3.1. Обоснование выбора среды разработки.
 - 3.2. Разработка и организация подключения к среде разработки.
 - 3.3. Разработка модулей интерфейса.
 - 3.4. Реализация запросов, получение отчетов.
4. Описание готовой программы.

Раздел «Описание предметной области. Постановка задачи» должен содержать всю необходимую и достаточную информацию для проектирования Базы данных. Прежде всего, должен быть определен круг лиц, который будет иметь доступ к базе данных, их права и обязанности, описаны бизнес-процессы, происходящие в предметной области, приведены формы всех входных и выходных документов, описаны регламентированные запросы, определена периодичность решения всех задач. Также должны быть описаны алгоритмы

получения промежуточных и результатных показателей, изображен граф взаимосвязей показателей.

Предметная область должна быть описана с такой степенью подробности, чтобы можно было определить характер связи между объектами. Так, например, если предметной областью является ВУЗ, для которого разрешен экстернат, то студент может не относиться ни к одной из студенческих групп. Кроме того, некоторые вузы бывают полностью заочными. Также надо определить, может ли один и тот же студент одновременно получать несколько специальностей одновременно в одном и том же вузе. В вузе могут быть дистанционные формы обучения. Необходимо уточнить, могут ли студенты, обучающиеся полностью дистанционно, обучаться на бюджетной основе или это не предусмотрено современным законодательством. Нюансов подобного рода очень много, и все они должны быть выявлены и описаны.

ВУЗ может быть чисто коммерческой структурой, где все студенты обучаются на платной основе, а может финансироваться на смешенной основе. В последнем случае надо знать, как формируются группы.

Описывая любую организацию, надо оговорить, допустимо или нет внутреннее совместительство.

Описывая предметную область, надо знать действующее законодательство. Эта информация позволит определить, какие свойства необходимо определить для объекта, а также выявить ограничения целостности. Так, например, на дневное отделение в вузы можно принимать лиц, не достигших 35 лет. Стипендию можно выплачивать, если студент получает первое образование. Есть ограничения на минимально допустимую зарплату, на минимальный возраст, с которого можно принимать на работу.

В разделе «Построение инфологической (концептуальной) модели предметной области» надо построить ER-модель. Рекомендуется сначала построить базовую ER-модель, а затем ER-модель в среде любой CASE-системы.

В разделе «Выбор средств/методологии проектирования. Выбор СУБД» должна быть выбрана методика проектирования базы данных, которая будет

зависеть от выбранной СУБД. СУБД может быть выбрана любая. Степень подробности описания тех средств, которые рассматриваются при выборе, зависит от того, насколько средства, используемые при анализе, распространены. Если они известны и охарактеризованы в используемой в учебном процессе литературе, то подробно описывать их не надо, необходимо просто обосновать свой выбор. В противном случае описание должно быть более подробным.

В разделе «Разработка модулей интерфейса» должны быть разработаны экранные формы ввода данных. Организация корректировки БД может потребовать разработку специальных форм для выполнения тех или иных видов корректировки. Результатом выполнения раздела «Разработка модулей интерфейса» должна быть разработанная и реализованная система меню, отражающая весь функционал системы. Система меню должна отражать те функции, которые выполняют пользователи базы данных.

В разделе «Реализация запросов, получение отчетов» необходимо реализовать запросы и получить отчеты, описанные в разделе «Описание предметной области».

Работа спроектированной системы должна быть опробована на контрольном примере, который может быть описан в разделе «Описание готовой программы».

5. Структура курсового проекта

По содержанию курсовой проект должен носить практический или опытно-экспериментальный характер.

Курсовой проект (работа) должен включать оттестированную базу данных с приложением и пояснительную записку (не менее 20 листов формата А4). В зависимости от курсового проекта (работы) может быть включена графическая часть.

Пояснительная записка курсового проекта (работы) должна иметь следующую **структуру**.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

1 ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ. ПОСТАНОВКА ЗАДАЧИ

2 РАЗРАБОТКА БАЗЫ ДАННЫХ

2.1 Построение инфологической модели предметной области

2.2 Реализация проекта в среде конкретной СУБД

3 РАЗРАБОТКА ПРОГРАММНОГО ПРИЛОЖЕНИЯ

3.1 Обоснование выбора среды разработки

3.2 Разработка и организация подключения к среде разработки

3.3 Разработка модулей интерфейса

3.4 Реализация запросов, получение отчетов

4 ОПИСАНИЕ ГОТОВОЙ ПРОГРАММЫ

ЗАКЛЮЧЕНИЕ

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

ПРИЛОЖЕНИЯ

Во **введении** кратко раскрывается актуальность и значение темы, формулируется цель и задачи проекта (работы).

Заключение включает основные выводы и перспективы дальнейшего развития защищаемого ПО.

В **список использованных источников**, с указанием библиографических данных, включается литература, используемая автором курсового проекта при разработке ПО. Если в работе сделаны ссылки на научную информацию, позволяющую принять конкретное решение, включение в список литературы первоисточника обязательно. При использовании формул, взятых из библиографического источника, ссылка на источник обязательна. Список литературы составляется в алфавитном порядке авторов.

В **приложения** могут включаться следующие материалы:

- таблицы БД;
- схемы межтабличных связей, структур баз данных и т.п.;
- интерфейсные формы к БД, образцы отчетов, запросов, используемых в БД;
- программный код.

В **графической** части курсового проекта (работы) могут быть представлены следующие результаты:

- структурная схема разрабатываемой предметной области;
- концептуальная схема БД;
- логическая схема БД, ER – модель;
- физическая схема БД.

Объем графической части может составлять 3 – 5 листов формата А1 для проекта (работы).

6. Оформление пояснительной записки к курсовому проекту

По объему курсовой проект должен быть не менее 20 страниц печатного текста. Шрифт текста – Times New Roman, размер шрифта – 14 пт, междустрочный интервал – полуторный. В списке литературы не менее 5 источников.

Пояснительная записка к курсовому проекту печатается на принтере на листах писчей бумаги формата А4 (210 × 297 мм). Для разворотных таблиц и рисунков допускается формат А3 (297 × 420 мм). Заголовки таблиц, названия схем допускается печатать через одинарный интервал.

Напечатанный текст должен иметь поля следующих размеров:

- верхние и нижние – 20 мм;
- правые - 10 мм;
- левые – 30 мм.

Абзацный отступ («красная строка») равен 1,25 мм. Глава начинается с новой страницы. Заголовки глав отделяются от параграфов двойным интервалом (т.е. двумя пустыми строками). Заголовки параграфов отделяются от текста одинарным интервалом (т.е. одной пустой строкой).

Основной текст печатается строчными (маленькими) буквами, заглавными буквами (прописными, большими) печатаются аббревиатуры, а также названия глав, слова «ВВЕДЕНИЕ», «ЗАКЛЮЧЕНИЕ» и «ПРИЛОЖЕНИЕ», которые

располагаются по центру листа. Названия параграфов печатаются с «красной строки» строчными буквами (кроме первой буквы).

В тексте должна быть соблюдена соподчинённость глав, параграфов и пунктов. Нумерация глав и параграфов выполняется арабскими цифрами, которые отделяются от названий пробелом; номер параграфа состоит из цифры, обозначающей номер главы, и цифры, обозначающей его порядковый номер в составе главы, отделённых друг от друга точкой. Знак § не ставится. Если параграфы состоят из нумерованных пунктов, их нумерация состоит из трёх разделённых точками цифр.

Нумерация таблиц и рисунков сквозная или разбитая по главам (локальная, номер рисунка - последняя цифра, первая цифра - номер главы, если же глава разбита на разделы, то номер главы и через точку, номер раздела).

Следует обратить внимание на положение на странице названий таблиц (сверху – слева, с абзацного отступа) и рисунков (снизу - посередине), причём перед названием после номера обязательно ставится тире и название печатается с заглавной буквы.

Каждая глава пояснительной записки к курсовому проекту начинается с новой страницы

Страницы пояснительной записки нумеруются от титульного листа и до последнего, цифра 1 на титульном листе не ставится. Нумерация страниц выполняется арабскими цифрами по центру страницы размером шрифта 10 пт.

Приложения нумеруются арабскими цифрами (без значка №) и должны иметь названия.

Пояснительная записка к курсовому проекту должна быть заключена в папку.

7. Защита курсового проекта

После полного завершения курсового проекта происходит защита курсового проекта.

Курсовая работа защищается перед преподавателем, который определяет уровень теоретических знаний и практических навыков студента, соответствие работы предоставляемым к ней требованиям.

Если курсовая работа является неудовлетворительной, то после исправления она представляется на повторное оценивание. При выявлении серьезных отклонений от предъявляемых требований к курсовому проекту студенту предлагается устранить недостатки или разработать новую тему курсового проекта.

Критериями оценки курсового проекта являются:

- степень разработки темы;
- полнота охвата научной литературы;
- творческий подход к написанию курсового проекта;
- правильность и научная обоснованность выводов;
- аккуратность и правильное оформление курсового проекта.

Окончательная оценка курсовой работы выставляется по итогам защиты и качеству работы.

Студенты, не сдавшие курсовые проекты или получившие на защите неудовлетворительные оценки, не допускаются к очередным экзаменам.

Защищенные курсовые проекты студентам не возвращаются и хранятся в архиве учебного заведения.

8. Пример реализации базы данных для написания курсового проекта

«Проектирование БД «Отдел кадров»

1 ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ. ПОСТАНОВКА ЗАДАЧИ

Кадровая политика включает разработку общих принципов и определение приоритетов целей в области управления персоналом; планирование потребности в трудовых ресурсах и путей формирования штатов; создание поддержку системы движения кадровой информации; определение принципов распределения средств; обеспечение эффективной системы стимулирования труда; политику развития персонала.

К функциям отдела кадров относится принятие и увольнение сотрудников, составление различных приказов и табеля рабочего времени, оформление, текущей учет документации по движению кадров, нормативный контроль по КЗОТу правильных формулировок приказов и распоряжений по отделу кадров, получение и ведение дополнительной информации о работнике, не нормативный по отделу кадров, но на сегодняшний день закрепленный за ним (это данные о пенсионерах, инвалидах, и т.п.) и т.д.

Принимая сотрудника на работу, фирма заключает с ним трудовой договор, в котором содержится информация различного рода, например: какого числа он был принят на работу, в качестве кого, размер его заработной платы, табельный номер сотрудника, условия работы, информация о том сколько, раз он может уходить в отпуск, оплачиваемый ли больничный или отпуск и т.д.

После официального принятия на работу на сотрудника заводится личная карточка. Личная карточка может содержать фамилию, имя и отчество сотрудника, его дату рождения, паспортные данные, место прописки и проживания, состав семьи и т.д.

Увольнение сотрудника происходит либо на основании личного заявления сотрудника, либо из-за нарушения правил трудового договора. После

официального подписания приказа сотрудник обязан отработать на данном предприятии еще две недели.

В процессе обследования предметной области было установлено, что в обязанности отдела кадров входит:

- ведение учета сотрудников предприятия;
- удаление уволившегося или добавление в базу нового сотрудника;
- изменение должности определенного сотрудника.
- изменение сведений о сотруднике;
- изменение сведений о подразделениях предприятия.

Назначение разработки БД:

Автоматизировать обработку информации при следующих операциях:

- прием нового сотрудника;
- коррекция данных о сотруднике и его личных данных;
- предоставление сведений о сотрудниках и подразделениях.

Предметной областью данного курсового проекта является информационная система отдела кадров предприятия. Какие данные на сотрудника нам понадобятся? Прежде всего, фамилия, имя и отчество. Затем укажем пол (мужской или женский), семейное положение (холост или женат/замужем), количество детей. Также понадобятся дата рождения и дата поступления на работу. Стаж работы в годах. Образование. Военнообязанный сотрудник, или нет. Телефоны, по которым можно связаться с сотрудником в любое время. Должность сотрудника и отдел (если есть), в котором он числится. А также его домашний адрес. При этом учитываем, что сотрудник не обязательно является жителем города, где он работает. Следовательно, придется вводить и страну, и город.

2 РАЗРАБОТКА БАЗЫ ДАННЫХ

2.1 Построение инфологической модели предметной области

Одним из наиболее удобных инструментов унифицированного представления данных, независимого от реализующего его программного обеспечения, является модель «сущность-связь» (entity - relationship model, ER - model). Графически такое представление предметной области можно представить с помощью диаграммы, в которой сущности изображены прямоугольниками, атрибуты – овалами, соединенными с сущностями ненаправленными дугами, связи – ромбами. Такая система обозначений была предложена Питером Ченом.

В результате обследования предметной области получим концептуальную (инфологическую) схему (рисунок 1). Для удобства некоторые несущественные атрибуты опущены.

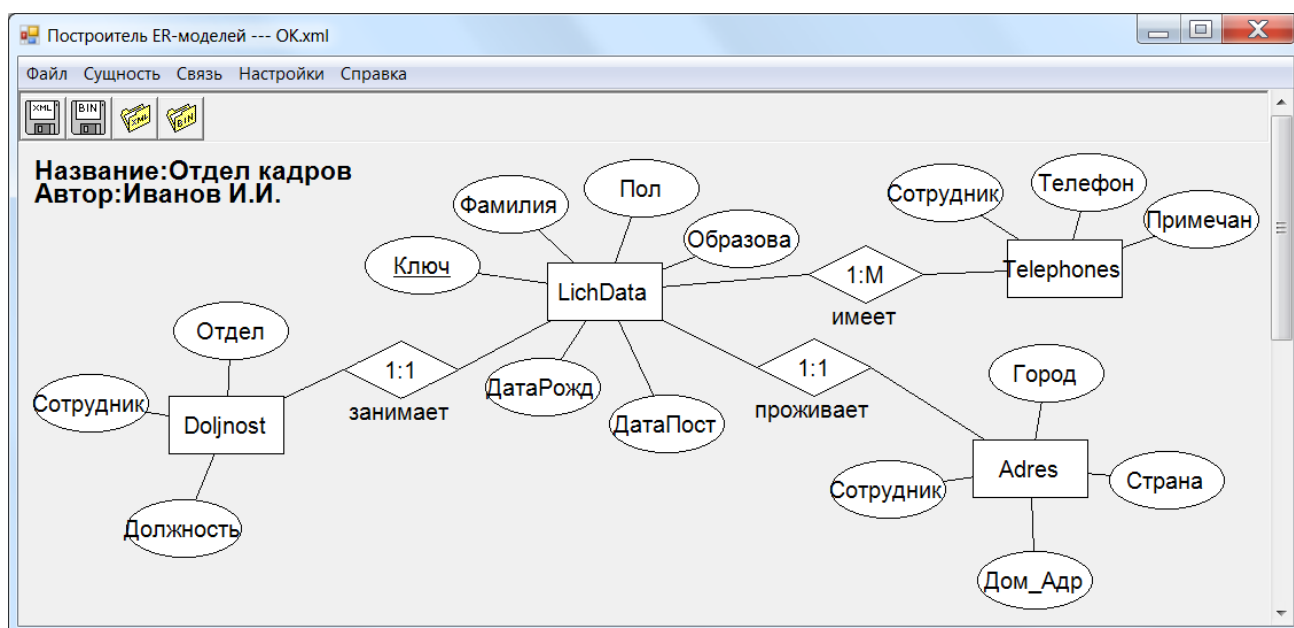


Рисунок 1 – Инфологическая модель БД «Отдел кадров»

2.2 Реализация проекта в СУБД MS Access

При преобразовании инфологической модели в реляционную её можно представить как совокупность таблиц, связанных между собой.

В результате уточнения модели предметной области было решено, что проектируемая база данных будет состоять из 4 таблиц (сущностей), связанных между собой в соответствии с семантикой ПО.

Сведения о сотрудниках будут храниться в таблице «LichData».

Сведения о телефонах будут храниться в таблице «Telephones».

Сведения о должностях будут храниться в таблице «Doljnost».

Сведения об адресах будут храниться в таблице «Adres».

Главной будет сущность LichData, которая содержит основные данные о сотруднике. Она имеет связи с другими сущностями.

Главная таблица поддерживает связь один-к-одному с таблицами Doljnost и Adres, и связь один-ко-многим с таблицей Telephones, ведь у сотрудника наверняка есть и домашний, и рабочий телефоны.

Для реализации базы данных была выбрана СУБД MS Access, которая имеет простой и понятный интерфейс. Система управления базами данных Microsoft Access является общеизвестной. Еще одним важным моментом является невысокая стоимость. Цена пакета окажется очень привлекательной, если покупать MS Access в составе полного пакета Microsoft Office. Access предоставляет возможность создания пароля к своей базе данных.

Для программистов при использовании СУБД MS Access можно выделить еще одно значительное преимущество. Оно состоит в том, что ядро Access JET 4 встроено в операционные системы Windows начиная с версии Windows 98. Это означает, что разработанное приложение, которое в своей работе использует базу данных, созданную в MS Access, без проблем можно переносить на любой компьютер с установленной операционной системой Windows. Оно будет работать, не требуя при этом установки СУБД. Возможно, потребуется только установка RUNTIME-версии, которую бесплатно можно скачать с официального

сайта корпорации Microsoft. Для разработки моей базы данных преимущества от использования MS Access очевидны.

В процессе проектирования была создана база данных следующей структуры (рисунок 2).

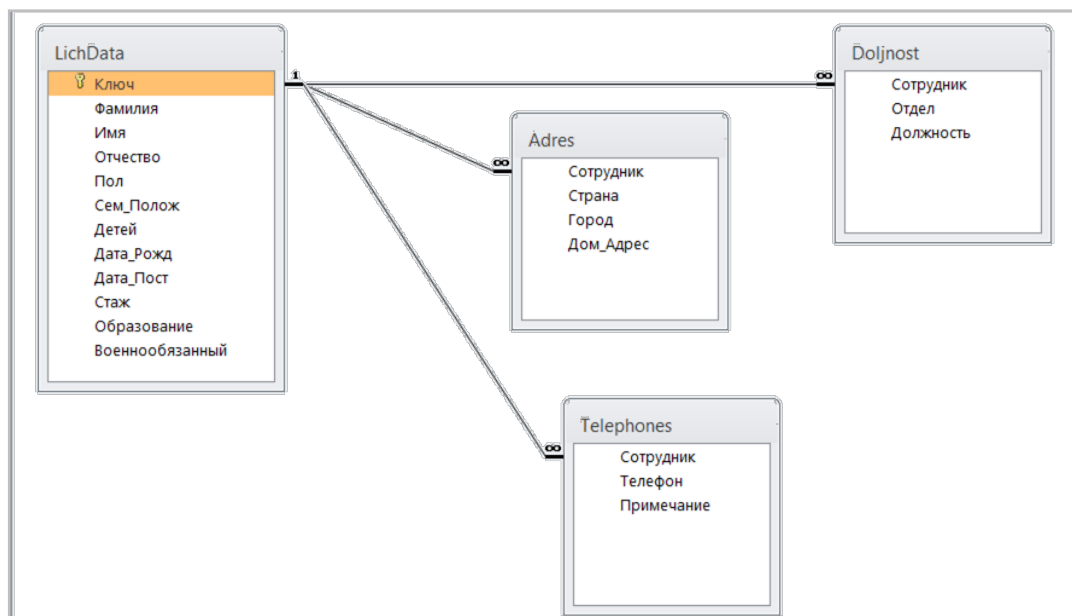


Рисунок 2 – Схема данных БД «Отдел кадров»

В результате анализа зависимостей между атрибутами были спроектированы таблицы следующей структуры.

LichData		
	Имя поля	Тип данных
Ключ	Ключ	Счетчик
Фамилия	Фамилия	Текстовый
Имя	Имя	Текстовый
Отчество	Отчество	Текстовый
Пол	Пол	Текстовый
Сем_Полож	Сем_Полож	Логический
Детей	Детей	Числовой
Дата_Рожд	Дата_Рожд	Дата/время
Дата_Пост	Дата_Пост	Дата/время
Стаж	Стаж	Числовой
Образование	Образование	Текстовый
Военнообязанный	Военнообязанный	Логический

Рисунок 3 – Таблица «LichData» в режиме Конструктора

Doljnost		
	Имя поля	Тип данных
Сотрудник	Сотрудник	Числовой
Отдел	Отдел	Текстовый
Должность	Должность	Текстовый

Рисунок 4 – Таблица «Doljnost» в режиме Конструктора

LichData Doljnost Adres Telephones	
Имя поля	Тип данных
Сотрудник	Числовой
Страна	Текстовый
Город	Текстовый
Дом_Адрес	Текстовый

Рисунок 5 – Таблица «Adres» в режиме Конструктора

LichData Doljnost Adres Telephones	
Имя поля	Тип данных
Сотрудник	Числовой
Телефон	Текстовый
Примечание	Текстовый

Рисунок 6 – Таблица «Telephones» в режиме Конструктора

3 РАЗРАБОТКА ПРОГРАММНОГО ПРИЛОЖЕНИЯ

3.1 Обоснование выбора среды разработки

Для создания проекта будем использовать среду программирования Borland Delphi 7.

Delphi – оптимальный инструмент для создания приложений для баз данных. Оптимальный, т.к. поддерживает технологию визуальной разработки, которая позволяет существенно сократить время разработки при сохранении хорошего качества и надежности программного продукта. Delphi в структуре языка позволяет избегать скрытых ошибок.

Разработка приложений баз данных является одной из наиболее востребованных возможностей среды программирования Delphi. Библиотека объектов содержит набор визуальных компонент, значительно упрощающих разработку приложений для СУБД с архитектурой клиент-сервер.

Предусмотрены специальные наборы компонент, отвечающих за доступ к данным, и компонент, отображающих данные. Компоненты доступа к данным позволяют осуществлять соединения с БД, производить выборку, копирование данных, и т.п.

Компоненты визуализации данных позволяют отображать данные виде таблиц, полей, списков. Отображаемые данные могут быть текстового, графического или произвольного формата.

Хотя Delphi не имеет своего формата таблиц БД, она, тем не менее, обеспечивает мощную поддержку большого количества различных СУБД – как локальных, так и промышленных.

В Delphi поддерживается технология ADO (ActiveX Data Objects – объекты данных, построенные как объекты ActiveX), которая усиленно развивается корпорацией Microsoft. На основе этой технологии созданы соответствующие компоненты-наборы TADOTable, TADOQuery, TADOStoredProc, повторяющие в

функциональном отношении компоненты TTable, TQuery, TStoredProc, но не требующие развертывания и настройки BDE на клиентской машине.

Основные особенности использования технологии ADO не зависят от архитектуры БД: эта технология характерна не только для файл-серверных БД, но также и для клиент-серверных и трехзвенных БД.

Основным достоинством технологии ADO является ее естественная ориентация на создание «облегченного» клиента. В рамках этой технологии на машине разработчика БД устанавливаются базовые объекты MS ADO и соответствующие компоненты Delphi, обеспечивающие использование технологии ADO (эти установки осуществляются автоматически при развертывании Delphi). На машине сервера данных (это может быть файловый сервер в рамках файл-серверной технологии или машина с сервером данных – в технологии клиент-сервер) устанавливается так называемый провайдер данных – некоторая надстройка над специальной технологией OLE DB, «понимающая» запросы объектов ADO и «умеющая» переводить эти запросы в нужные действия с данными. Взаимодействие компонентов ADO и провайдера осуществляется на основе универсальной для Windows технологии ActiveX, причем провайдер реализуется как COM-сервер, а ADO-компоненты – как COM-клиенты.

3.2 Разработка и организация подключения к среде разработки

Загружаем Delphi, делаем новый проект. Создадим модуль данных. Добавляем в модуль компонент ADOConnection с вкладки ADO палитры компонентов. Этот компонент обеспечит связь других компонентов с базой данных при помощи механизма ADO. Связь обеспечивается свойством компонента ConnectionString. У таких компонентов, как ADOTable, тоже есть это свойство, однако придется устанавливать связь для каждой таблицы. Проще единожды соединиться компонентом ADOConnection и использовать его для связи других компонентов.

Щелкнем дважды по свойству `ConnectionString` компонента `ADOConnection`. Откроется окно подключения компонента к ADO (рисунок 7).



Рисунок 7 – Окно подключения к ADO

Нажмем кнопку `Build`. Открывается новое окно, содержащее настройки подключения. Для связи с базой данных MS Access больше всего подходит «Microsoft Jet 4.0 OLE DB Provider». Jet - это название механизма работы с СУБД, встроенного в MS Access (рисунок 8).

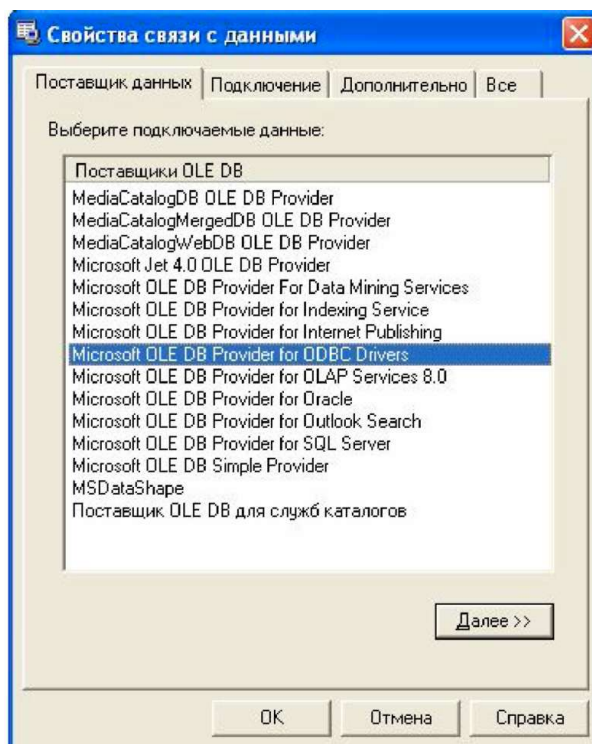


Рисунок 8 – Настройки подключения

Нажимаем на кнопку «Далее», либо переходим к вкладке «Подключение». Здесь нам нужно выбрать или ввести базу данных. Если ввести имя БД вручную, например, как в нашем случае, `ok.mdb`, путь доступа к ней не будет указан явно, и проект при запуске будет искать ее в текущей папке. Теперь свойство `Connected` переведем в `True`. Если не вышло никаких сообщений об ошибке, то подключение состоялось.

3.3 Разработка модулей интерфейса

Среда программирования Delphi позволяет формировать программу, используя стандартные объекты с установкой их свойств или записывая соответствующий текст. При этом сама Delphi предоставляет разработчику заготовки соответствующих фрагментов программы. Полученные результаты сразу отображаются на экране монитора.

При организации подключения к базе данных был создан специальный контейнер – модуль данных (DataModule). Его также будем использовать для компактного хранения всех компонентов для работы с таблицами (TDataSource, TADOTable и др.) (рисунок 9).

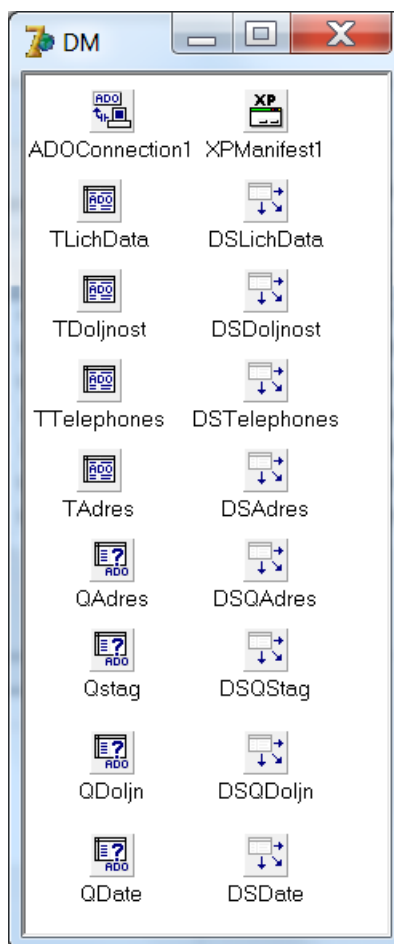


Рисунок 9 – Модуль данных DM (DataModule)

Для доступа к таблицам ADO используется компонент ADOTable, который предоставляет интерфейс взаимодействия с таблицами, подключенными при помощи ADOConnection.

Для предоставления доступа к таблицам и запросам используется графический элемент управления DataSource. Связав DataSource с необходимой таблицей (ADOTable) или запросом (ADOQuery), а затем, связав необходимый графический элемент управления с ним, создается простой интерфейс для вывода данных на экран или принтер.

К графическим элементам относятся DBGrid, DBNavigator, QuickReport. DBGrid позволяет легко выводить данные, предоставляемые DataSource, на экран в виде таблицы и корректировать их. DBNavigator позволяет производить навигацию по таблице, добавлять и удалять записи, корректировать записи, обновлять таблицу, а также «посылать» изменения серверу базы данных.

Для создания интерфейса пользователя были также использованы кнопки Button (командные кнопки), RadioButton, Label, Edit, BitBtn, DBCombobox, DBCheckbox, DBCheckbox, DBEdit (рисунки 10, 11).

Ключ	Фамилия	Имя	Отчество	Пол	Сем_Полож	Детей	Дата_Рожд
2	Петров	Петр	Петрович	муж	True	1	20.01.1975
3	Иваненко	Анатолий	Петрович	Муж	True	2	20.03.1985
4	Аванян	Арсен	Ашотович	Муж	True	2	18.06.1975
5	Антонова	Мария	Александровна	Жен	True	1	18.02.1990
6	Абрамова	Валентина	Ивановна	Жен	True	2	14.08.1985
7	Кулешов	Петр	Петрович	Муж	True	2	16.04.1975
8	Белозеров	Константин	Сергеевич	Муж	False		12.03.1990

Сотрудник	Страна	Город	Дом_Адрес
2	Россия	Пятигорск	ул. Мира, 12

Рисунок 10 – Главная форма приложения в процессе разработки

Рисунок 11 – Фрагмент формы «Ввод/Редактирование данных»
в процессе разработки

3.4 Реализация запросов, получение отчетов

SQL (Structured Query Language) является языком интерактивных запросов, который обеспечивает пользователям немедленный доступ к данным. С помощью SQL пользователь может в интерактивном режиме получить ответы на самые сложные запросы в считанные минуты или секунды, тогда как программисту потребовались бы дни и недели, чтобы написать для пользователя соответствующую программу. Из-за того, что SQL допускает немедленные запросы, данные становятся более доступными, и могут помочь в принятии решений, делая их более обоснованными.

Под запросом к базе данных понимается требование совершить определенное действие с информацией из базы данных. Можно выбрать данные в соответствии с заданным условием и отобразить результат запроса на выборку. Можно также вставить, удалить, обновить определенные записи в базе данных, используя соответствующие запросы. Такие требования формируются на специальных языках программирования QBE и SQL.

Оператор SELECT является чаще всего используемой командой языка SQL.

Общий вид оператора SELECT:

```

SELECT [All или DISTINCT] (<список полей> или *)
FROM <список таблиц>
[WHERE <условие выборки или соединение>]
[GROUP By <список полей результата>]
[HAVING <условие для группы>]
[ORDER By <список полей, по которым упорядочивается выбор>]

```

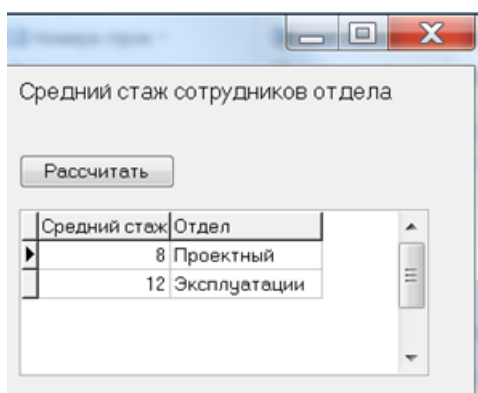
На листинге 1 приводится код SQL-запроса на выборку.

Листинг 1 – SQL-код запроса с группировкой и нахождением среднего значения на выборку данных из двух таблиц для нахождения среднего стажа сотрудников по каждому отделу.

```

SELECT Doljnost.Отдел, Avg(LichData.Стаж) AS [Средний Стаж]
FROM LichData INNER JOIN Doljnost ON LichData.Ключ =
Doljnost.Сотрудник
GROUP BY Doljnost.Отдел;

```



Средний стаж	Отдел
8	Проектный
12	Эксплуатации

Рисунок 12 – Результат выполнения запроса (Листинг 1)

В курсовом проекте реализовано еще три запроса, результаты выполнения которых приводятся в главе 4, а код запросов приводится в приложении.

В Delphi есть встроенные средства для получения отчетов. Это QuickReport и RaveReport. Эти компоненты позволяют построить отчет из базы данных быстро и просто. Можно не только отпечатать отчет, но и просмотреть его на экране, что позволяет пользователю проверить результат без порчи бумаги и экспортировать отчет в другие форматы, такие как простой ASCII текст, текст разделенный запятыми (CSV) и в HTML.

Можно также воспользоваться технологией экспорта в приложение Excel. В данной работе была реализована возможность экспорта в Excel списка сотрудников, отображаемых на главной форме.

Для этого в папку с проектом помещаем файл spisok.xls. На форму помещаем кнопку для экспорта, и в процедуре нажатия кнопки пишем код (рисунок 13).

```
procedure TForm1.BitBtn2Click(Sender: TObject);  
    var  
    i,j,index: Integer;  
begin  
    Excel:= CreateOleObject('Excel.Application');  
    Excel.Visible := True;  
    // Открываем готовый файл, находящийся в текущей папке  
    Excel.Workbooks.Open(ExtractFilePath(ParamStr(0))+'.\spisok.xls');  
  
    index:=2; // переносим со 2-й строки содержимое DBGrid1  
    DBGrid1.DataSource.DataSet.First;  
    for i:=1 to DBGrid1.DataSource.DataSet.RecordCount do  
    begin  
        for j:=1 to DBGrid1.FieldCount do  
            Excel.ActiveSheet.Cells[index,j]:=DBGrid1.Fields[j-1].AsString;  
            inc(index);  
        DBGrid1.DataSource.DataSet.Next;  
    end;  
end;
```

Рисунок 13 – Процедура экспорта списка сотрудников в Excel

4 ОПИСАНИЕ ГОТОВОЙ ПРОГРАММЫ

После запуска приложения на экран монитора выводится главное окно программы (рисунок 14). Данное окно содержит основные компоненты отображения данных, а также кнопки доступа к запросам.

Ключ	Фамилия	Имя	Отчество	Пол	Сем_Полоз	Детей	Дата_Рожд	Дата_Пост	Стаж	Образование
1	Петров	Петр	Петрович	муж	True	1	20.01.1975	10.02.2010	15	Высшее
3	Иваненко	Анатолий	Петрович	Муж	True	2	20.03.1985	10.05.2008	10	Среднее
4	Аванян	Арсен	Ашотович	Муж	True	2	18.06.1975	20.07.2005	15	Среднее
5	Антонова	Мария	Александровна	Жен	True	1	18.02.1990	10.04.2014	3	Среднее
6	Абрамова	Валентина	Ивановна	Жен	True	2	14.08.1985	20.06.2010	7	Высшее
7	Кулешов	Петр	Петрович	Муж	True	2	16.04.1975	20.05.2008	20	Среднее специальное
8	Белозеров	Константин	Сергеевич	Муж	False		12.03.1990	15.10.2015	2	Высшее

Телефон	Примечание
8(918)-444-55-44	Мобильный
8(979)-373-33-44	Домашний

Рисунок 14 – Главная форма приложения

После выбора сотрудника в нижней части окна отображается адрес, телефоны или его должность в соответствии со сделанным выбором. На этой форме также организован быстрый фильтр для поиска сотрудника по фамилии (рисунок 15). Кнопкой со значком Excel содержимое таблицы экспортируется в Excel (рисунок 16).

Ключ	Фамилия	Имя	Отчество	Пол	Сем_Полоз	Детей	Дата_Рожд	Дата_Пост	Стаж	Образование
4	Аванян	Арсен	Ашотович	Муж	True	2	18.06.1975	20.07.2005	15	Среднее
5	Антонова	Мария	Александровна	Жен	True	1	18.02.1990	10.04.2014	3	Среднее
6	Абрамова	Валентина	Ивановна	Жен	True	2	14.08.1985	20.06.2010	7	Высшее

Телефон	Примечание
8(928)-333-55-86	Мобильный
8(879)-334-77-88	Домашний

Рисунок 15 – Поиск сотрудника по фамилии

	A	B	C	D	E	F	G	H	I	J	K	L
1	№	Фамилия	Имя	Отчество	Пол	Женат/Холост (да/Нет)	Детей	ДатаРожд	ДатаПост	Стаж	Образов	Военнообяз
2	4	Аванян	Арсен	Ашотович	Муж	ИСТИНА	2	18.06.1975	20.07.2005	15	Среднее	ЛОЖЬ
3	5	Антонова	Мария	Александр	Жен	ИСТИНА	1	18.02.1990	10.04.2014	3	Среднее	ЛОЖЬ
4	6	Абрамова	Валентин	Ивановна	Жен	ИСТИНА	2	14.08.1985	20.06.2010	7	Высшее	ЛОЖЬ

Рисунок 16 – Экспорт в Excel

Кнопки «Редактировать» и «Новый сотрудник» открывают форму ввода/редактирования данных сотрудника (рисунок 17).

Рисунок 17 – Форма ввода/ редактирования данных

Кнопка «Дополнительные сведения» (рисунки 14, 15) открывает форму просмотра результатов запросов (рисунок 18).

Поиск данных

Поиск по городу местожительства сотрудников

Введите город
Пятигорск Поиск по городу

Фамилия	Имя	Отчество	Город	Дом_Адрес
Петров	Петр	Петрович	Пятигорск	ул. Мира, 12
Аванян	Арсен	Ашотович	Пятигорск	ул. Кировас, 2

Средний стаж сотрудников отдела

Рассчитать

Средний стаж	Отдел
8	Проектный
12	Эксплуатации

Поиск по должности и стажу

Введите должность
Секретарь Найти

Введите стаж
от 1 до 3

Фамилия	Имя	Отчество	Отдел
Антонова	Мария	Александровна	Эксплуатации

Рисунок 18 – Форма «Поиск данных», основанная на запросах

Кнопка «Дата поступления» (рисунки 14, 15) открывает форму запроса по дате приема сотрудников на работу (рисунок 19).

Дата поступления на работу

Выберите период

с 20.11.2015 по 01.09.2018 Показать

Фамилия	Имя	Дата_Пост	Образование
Вайнштейн	Сигизмунд	12.04.2017	Высшее
Богушевский	Петр	10.08.2017	Высшее

Рисунок 19 – Форма «Дата поступления на работу»

Project1

Измените интервал поиска

OK

Рисунок 20 – Окно сообщения об отсутствии значений в заданном интервале

Образец титульного листа

Министерство Науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Ростовский государственный экономический университет (РИНХ)»
Филиал в г. Кисловодске Ставропольского края

КУРСОВОЙ ПРОЕКТ

по дисциплине МДК 02.02 «Технология разработки и защиты баз данных»
на тему: «**Проектирование базы данных «Отдел кадров»**»

Студента

Иванова Ивана Ивановича

4 курса очной формы обучения
по программе подготовки специалистов
среднего звена на базе основного общего
образования,
обучающегося по специальности 09.02.03 Программирование в компьютерных
системах

Руководитель курсового проекта: преподаватель Шаповалова М.С.

Кисловодск, 2018 г.