

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Агабекян Раиса Левонровна

Должность: ректор

Дата подписания: 18.11.2023 06:04

Уникальный программный ключ:

4237c7ccb9b9e111bbaf174fcd9201d015c4dbaa123ff774747507b9b9fbcbe

**ПЕРОСУДАРСТВЕННОЕ АККРЕДИТОВАННОЕ НЕКОММЕРЧЕСКОЕ
ЧАСТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО
ОБРАЗОВАНИЯ «АКАДЕМИЯ МАРКЕТИНГА И СОЦИАЛЬНО-
ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ – ИМСИТ»**

(г. Краснодар)

ИНСТИТУТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И ИННОВАЦИЙ

КАФЕДРА МАТЕМАТИКИ И ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ

Рассмотрено и одобрено на заседании
кафедры математики и
вычислительной техники Академии
ИМСИТ, протокол №3 от 13 октября
2023 года,
зав. кафедрой МиВТ, доцент

_____ Н.П. Исикова

УТВЕРЖДАЮ

Проректор по учебной работе,
доцент

_____ Н.И. Севрюгина
20 ноября 2023 г.

**МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ВЫПОЛНЕНИЮ
КУРСОВОЙ РАБОТЫ**

по дисциплине «Безопасность систем баз данных»

для обучающихся направления подготовки
10.03.01 Информационная безопасность
направленность (профиль) образовательной программы
«Безопасность автоматизированных систем (по отрасли или в сфере
профессиональной деятельности)»

Квалификация выпускника
«Бакалавр»

Краснодар
2023

Методические указания по выполнению курсовой работы по дисциплине «Безопасность систем баз данных» для обучающихся всех форм обучения направления подготовки 10.03.01 Информационная безопасность направленность (профиль) образовательной программы «Безопасность автоматизированных систем (по отрасли или в сфере профессиональной деятельности)» – Краснодар: Академия маркетинга и социально-информационных технологий – ИМСИТ (г. Краснодар).

Методические указания по выполнению курсовой работы содержат требования к составу и содержанию, рекомендации по выполнению и защите курсовой работы по дисциплине «Безопасность систем баз данных».

Методические указания составлены в соответствии с требованиями Федерального государственного образовательного стандарта высшего образования (ФГОС ВО) по направлению подготовки 10.03.01 Информационная безопасность, утвержденного приказом Министерства образования и науки Российской Федерации от «17» ноября 2020 г. № 1427 направленность (профиль) образовательной программы «Безопасность автоматизированных систем (по отрасли или в сфере профессиональной деятельности)».

Составитель, канд. тех. наук, доцент

С.А. Капустин

Методические указания рассмотрены и рекомендованы на заседании кафедры математики и вычислительной техники от 13.11.2023 г., протокол № 3.

Зав. кафедрой, доцент

Н.П. Исикова

Рабочая программа утверждена на заседании Научно-методического совета Академии ИМСИТ протокол № 3 от 20 ноября 2023 г.

Председатель НМС академии, профессор
20.11.2023 г., протокол № 3

Н.Н. Павелко

Согласовано:

Проректор по учебной работе, доцент

Н.И. Севрюгина

Проректор по качеству образования, доцент

К.В. Писаренко

Рецензенты:

Видовский Л.А., д.т.н., профессор, профессор кафедры информационных систем и программирования КубГТУ

Глебов О.В., директор АО «ЮГ-СИСТЕМА ПЛЮС»

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	4
1 Структура и оформление курсовой работы.....	6
1.1 Структура курсовой работы.....	6
1.2 Оформление текста курсовой работы.....	6
1.4 Оформление таблиц в курсовой работе.....	13
2 Тематика курсовых работ.....	15
2.1 Порядок выполнения варианта курсовой работы.....	30
3 Организация выполнения курсовой работы.....	84
3.1 Выбор темы курсовой работы.....	84
3.2 Контроль выполнения курсовой работы.....	84
3.3 Подведение итогов и защита курсовой работы. Подготовка презентации ...	85
3.4 Порядок размещения в ЭБС и автоматизированной (компьютерной) проверке на объем и характер заимствования курсовой работы.....	91
4 Условия обучения лиц с ограниченными возможностями здоровья.....	92
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	93
ПРИЛОЖЕНИЕ А.....	96
ПРИЛОЖЕНИЕ Б.....	97
ПРИЛОЖЕНИЕ В.....	98
ПРИЛОЖЕНИЕ Г.....	100
ПРИЛОЖЕНИЕ Д.....	101
ПРИЛОЖЕНИЕ Е.....	103

ВВЕДЕНИЕ

В соответствии с федеральным государственным образовательным стандартом высшего образования по направлению подготовки 10.03.01 Информационная безопасность, утвержденного приказом Министерства образования и науки Российской Федерации от «17» ноября 2020 г. № 1427 и разработанным на его основе учебным планом Академии маркетинга и социально-информационных технологий – ИМСИТ для студентов всех форм обучения предусмотрены выполнение и защита курсового проекта по дисциплине «Проектирование защищенных автоматизированных систем».

Курсовая работа по дисциплине «Безопасность систем баз данных» представляет комплексную проектную практическую внедренческую работу студента, предшествующую выполнению выпускной квалификационной работы и вобравшую в себя совокупность ранее выполненных проектных и практических наработок по дисциплинам кафедры. В их числе Основы информационной безопасности, Организационное и правовое обеспечение информационной безопасности.

Таким образом, интегрируя результаты учебно-творческого процесса за несколько лет напряженного труда студента настоящую курсовую работу, выстраивает их в систему, на базе которой реализуется современное наукоемкое проектирование по специальности.

Для выполнения курсовой работы требуется предварительное изучение курсов: "Основы информационной безопасности", " Организационное и правовое обеспечение информационной безопасности ".

Процесс выполнения работы на формирование следующих компетенций:

УК-1: Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач

УК-3: Способен осуществлять социальное взаимодействие и реализовывать свою роль в команде

ОПК-4.2: Способен администрировать операционные системы, системы управления базами данных, вычислительные сети

Цель настоящих методических указаний – оказать помощь обучающимся в выборе темы курсовой работы, определении ее содержания и организации процесса ее написания и защиты. Знание методики написания курсовых работ необходимо обучающимся не только для успешного освоения основной образовательной программы направления подготовки 10.03.01 Информационная безопасность, но и будущей профессиональной деятельности.

Выполнение курсовой работы (КР) является одним из основных этапов в изучении дисциплины «Безопасность систем баз данных», и имеет целью закрепление, углубление и обобщение знаний, полученных студентами в процессе изучения лекционного курса, а также умений и навыков, полученных

при выполнении практических и лабораторных работ и применение этих знаний, умений и навыков к решению конкретных инженерных задач, развитие навыков работы со специальной литературой и навыков инженерного проектирования.

Методические указания по выполнению курсовой работы содержат требования к составу и содержанию, рекомендации по выполнению курсовой работы по дисциплине «Безопасность систем баз данных» и являются обязательными для студентов всех форм обучения направления подготовки 10.03.01 Информационная безопасность.

1 Структура и оформление курсовой работы

1.1 Структура курсовой работы

Материал курсовой работы должен быть изложен чётко и логически последовательно с конкретным описанием результатов научно-технического исследования и выводов.

План курсовой работы студент составляет самостоятельно и затем согласует с ведущим преподавателем учебной дисциплины, научным руководителем курсовой работы.

Рекомендуемая структура курсовой работы выглядит следующим образом:

1. Титульный лист (см. приложение 1).
2. Задание на курсовой проект (см. приложение 2).
3. Реферат (приложение 8)
4. Содержание
5. Введение
6. Обоснование выбранного направления исследований и общую методику достижения поставленной цели;
7. Теоретические и (или) экспериментальные исследования;
8. Обобщение и оценку результатов исследований
9. Заключение.
10. Список использованных источников.
11. Приложения (при необходимости).

Структура работы согласовывается с руководителем курсовой работы и может отличаться от рекомендуемой.

Компетенции формируемые в процессе выполнения курсовой работы.

Индекс	Разделы курсового проекта
УК-1, УК-3	Обоснование выбранного направления исследований и общую методику достижения поставленной цели
ПК-4.2	Теоретические и (или) экспериментальные исследования
УК-1,ПК-4.2	Обобщение и оценку результатов исследований

1.2 Оформление текста курсовой работы

Реферат должен содержать:

- сведения об общем объеме работы, иллюстраций, таблиц, использованных источников, приложений;
- перечень ключевых слов;

- текст реферата.

Перечень ключевых слов должен включать от 5 до 15 слов или словосочетаний из текста работы, которые в наибольшей мере характеризуют его содержание и обеспечивают возможность информационного поиска.

Текст реферата должен отражать:

- объект исследования или разработки;
- цель работы;
- методы или методологию проведения работы;
- результаты работы и их новизну;
- область применения результатов;
- рекомендации по внедрению или итоги внедрения результатов работы;
- экономическую эффективность или значимость работы;
- прогнозные предположения о развитии объекта исследования.

Если работа не содержит сведений по какой-либо из перечисленных структурных частей реферата, то в тексте реферата она опускается, при этом последовательность изложения сохраняется.

Оптимальный объем текста реферата - 850 печатных знаков, но не более одной страницы машинописного текста.

Содержание включает введение, наименование всех разделов и подразделов, пунктов (если они имеют наименование), заключение, список использованных источников и наименования приложений с указанием номеров страниц, с которых начинаются эти элементы курсового проекта.

В элементе "СОДЕРЖАНИЕ" приводят наименования структурных элементов работы, порядковые номера и заголовки разделов, подразделов (при необходимости - пунктов) основной части работы, обозначения и заголовки ее приложений (при наличии приложений). После заголовка каждого элемента ставят отточие и приводят номер страницы работы, на которой начинается данный структурный элемент.

Обозначения подразделов приводят после абзацного отступа, равного двум знакам, относительно обозначения разделов. Обозначения пунктов приводят после абзацного отступа, равного четырем знакам относительно обозначения разделов.

При необходимости продолжение записи заголовка раздела, подраздела или пункта на второй (последующей) строке выполняют, начиная от уровня начала этого заголовка на первой строке, а продолжение записи заголовка приложения - от уровня записи обозначения этого приложения.

Введение должно содержать оценку современного состояния решаемой научно-технической проблемы, основание и исходные данные для разработки

темы, обоснование необходимости проведения курсового проекта, сведения о планируемом научно-техническом уровне разработки, о патентных исследованиях и выводы из них, сведения о метрологическом обеспечении работы. Во введении должны быть отражены актуальность и новизна темы, связь данной работы с другими научно-исследовательскими работами.

В основной части работы приводят данные, отражающие сущность, методику и основные результаты выполненной работы.

Основная часть должна содержать:

- выбор направления исследований, включающий обоснование направления исследования, методы решения задач и их сравнительную оценку, описание выбранной общей методики проведения исследований и реализации проекта;

- процесс теоретических и (или) экспериментальных исследований, включая определение характера и содержания теоретических исследований, методы исследований, методы расчета, обоснование необходимости проведения экспериментальных работ, принципы действия разработанных объектов, их характеристики;

- обобщение и оценку результатов работы, включающих оценку полноты решения поставленной задачи и предложения по дальнейшим направлениям работ, оценку достоверности полученных результатов и технико-экономической эффективности их внедрения и их сравнение с аналогичными результатами отечественных и зарубежных работ, обоснование необходимости проведения дополнительных исследований, отрицательные результаты, приводящие к необходимости прекращения дальнейших исследований.

Она должна содержать:

• обоснование проектных решений по автоматизации информационных процессов;

• структурную схему подсистемы защиты информации;

• обоснование проектных решений по защите информации;

• структура ресурсов, таблица разграничения доступа;

• протоколы и отчеты экспериментальных исследований;

Заключение должно содержать:

- краткие выводы по результатам выполненной работе или отдельных ее этапов;

- оценку полноты решений поставленных задач;

- разработку рекомендаций и исходных данных по конкретному использованию результатов работы;

- результаты оценки эффективности внедрения;

- результаты оценки научно-технического уровня выполненной работы в сравнении с лучшими достижениями в этой области.

Страницы текста курсового проекта и включенные в нее иллюстрации и таблицы должны соответствовать формату А4 по ГОСТ 9327. Допускается применение формата А3 при наличии большого количества таблиц и иллюстраций данного формата.

Работа должна быть выполнена любым печатным способом на одной стороне листа белой бумаги формата А4 через полтора интервала.

Цвет шрифта должен быть черным, размер шрифта - не менее 12 пт (рекомендуется использовать 14 пт). Рекомендуемый тип шрифта для основного текста работы - Times New Roman. Полужирный шрифт применяют только для заголовков разделов и подразделов, заголовков структурных элементов. Использование курсива допускается для обозначения объектов (биология, геология, медицина, нанотехнологии, генная инженерия и др.) и написания терминов (например, *in vivo*, *in vitro*) и иных объектов и терминов на латыни.

Для акцентирования внимания может применяться выделение текста с помощью шрифта иного начертания, чем шрифт основного текста, но того же кегля и гарнитуры. Разрешается для написания определенных терминов, формул, теорем применять шрифты разной гарнитуры.

Текст работы следует печатать, соблюдая следующие размеры полей: левое - 30 мм, правое - 15 мм, верхнее и нижнее - 20 мм. Абзацный отступ должен быть одинаковым по всему тексту работы и равен 1,25 см.

Вне зависимости от способа выполнения работы качество напечатанного текста и оформления иллюстраций, таблиц, распечаток программ должно удовлетворять требованию их четкого воспроизведения.

При выполнении работы необходимо соблюдать равномерную плотность и четкость изображения по всей работе. Все линии, буквы, цифры и знаки должны иметь одинаковую контрастность по всему тексту работы.

Фамилии, наименования учреждений, организаций, фирм, наименования изделий и другие имена собственные в работе приводят на языке оригинала. Допускается транслитерировать имена собственные и приводить наименования организаций в переводе на язык работы с добавлением (при первом упоминании) оригинального названия по ГОСТ 7.79.

Сокращения слов и словосочетаний на русском, белорусском и иностранных европейских языках оформляют в соответствии с требованиями ГОСТ 7.11, ГОСТ 7.12.

Наименования структурных элементов работы: "СПИСОК ИСПОЛНИТЕЛЕЙ", "РЕФЕРАТ", "СОДЕРЖАНИЕ", "ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ", "ПЕРЕЧЕНЬ СОКРАЩЕНИЙ И ОБОЗНАЧЕНИЙ", "ВВЕДЕНИЕ", "ЗАКЛЮЧЕНИЕ", "СПИСОК ИСПОЛЬЗОВАННЫХ

ИСТОЧНИКОВ", "ПРИЛОЖЕНИЕ" служат заголовками структурных элементов работы.

Заголовки структурных элементов следует располагать в середине строки без точки в конце, прописными буквами, не подчеркивая. Каждый структурный элемент и каждый раздел основной части работы начинают с новой страницы.

Основную часть работы следует делить на разделы, подразделы и пункты. Пункты при необходимости могут делиться на подпункты. Разделы и подразделы работы должны иметь заголовки. Пункты и подпункты могут не иметь заголовков.

Заголовки разделов и подразделов основной части работы следует начинать с абзацного отступа и размещать после порядкового номера, печатать с прописной буквы, полужирным шрифтом, не подчеркивать, без точки в конце. Пункты и подпункты могут иметь только порядковый номер без заголовка, начинающийся с абзацного отступа, а могут иметь заголовок после порядкового номера, печатать с прописной буквы, обычным шрифтом, не подчеркивать, без точки в конце.

Если заголовок включает несколько предложений, их разделяют точками.

Переносы слов в заголовках не допускаются.

Страницы работы следует нумеровать арабскими цифрами, соблюдая сквозную нумерацию по всему тексту работы, включая приложения. Номер страницы проставляется в центре нижней части страницы без точки. Приложения, которые приведены в работе и имеющие собственную нумерацию, допускается не перенумеровать.

Титульный лист включают в общую нумерацию страниц работы. Номер страницы на титульном листе не проставляют.

Иллюстрации и таблицы, расположенные на отдельных листах, включают в общую нумерацию страниц работы. Иллюстрации и таблицы на листе формата А3 учитывают как одну страницу.

Разделы должны иметь порядковые номера в пределах всей работы, обозначенные арабскими цифрами без точки и расположенные с абзацного отступа. Подразделы должны иметь нумерацию в пределах каждого раздела. Номер подраздела состоит из номеров раздела и подраздела, разделенных точкой. В конце номера подраздела точка не ставится. Разделы, как и подразделы, могут состоять из одного или нескольких пунктов.

Если работа не имеет подразделов, то нумерация пунктов в нем должна быть в пределах каждого раздела и номер пункта должен состоять из номеров раздела и пункта, разделенных точкой. В конце номера пункта точка не ставится.

Если работа имеет подразделы, то нумерация пунктов должна быть в пределах подраздела и номер пункта должен состоять из номеров раздела, подраздела и пункта, разделенных точками.

Пример - Приведен фрагмент нумерации раздела, подраздела и пунктов работы:

3 Принципы, методы и результаты разработки и ведения классификационных систем ВИНТИ

3.1 Рубрикатор ВИНТИ

3.1.1 Структура и функции рубрикатора

3.1.2 Соотношение Рубрикатора ВИНТИ и ГРНТИ

3.1.3 Место рубрикатора отрасли знания в рубрикационной системе ВИНТИ

Если раздел или подраздел состоит из одного пункта, то пункт не нумеруется.

Если текст работы подразделяется только на пункты, они нумеруются порядковыми номерами в пределах работы.

Пункты при необходимости могут быть разбиты на подпункты, которые должны иметь порядковую нумерацию в пределах каждого пункта: 4.2.1.1, 4.2.1.2, 4.2.1.3 и т.д.

Внутри пунктов или подпунктов могут быть приведены перечисления. Перед каждым элементом перечисления следует ставить тире. При необходимости ссылки в тексте работы на один из элементов перечисления вместо тире ставят строчные буквы русского алфавита со скобкой, начиная с буквы "а" (за исключением букв е, з, й, о, ч, ь, ы, ь). Простые перечисления отделяются запятой, сложные - точкой с запятой.

При наличии конкретного числа перечислений допускается перед каждым элементом перечисления ставить арабские цифры, после которых ставится скобка.

Перечисления приводятся с абзацного отступа в столбик.

Пример 1

Информационно-сервисная служба для обслуживания удаленных пользователей включает следующие модули:

- удаленный заказ,
- виртуальная справочная служба,
- виртуальный читальный зал.

Пример 2

Работа по оцифровке включала следующие технологические этапы:

- а) первичный осмотр и структурирование исходных материалов,
- б) сканирование документов,
- в) обработка и проверка полученных образов,

- г) структурирование оцифрованного массива,
- д) выходной контроль качества массивов графических образов.

Пример 3

8.2.3 Камеральные и лабораторные исследования включали разделение всего выявленного видового состава растений на четыре группы по степени использования их копытными:

- 1) случайный корм,
- 2) второстепенный корм,
- 3) дополнительный корм,
- 4) основной корм.

Пример 4

7.6.4 Разрабатываемое сверхмощное устройство можно будет применять в различных отраслях реального сектора экономики:

- в машиностроении:

- 1) для очистки отливок от формовочной смеси;
- 2) для очистки лопаток турбин авиационных двигателей;
- 3) для холодной штамповки из листа;

- в ремонте техники:

- 1) устранение наслоений на внутренних стенках труб;
- 2) очистка каналов и отверстий небольшого диаметра от грязи.

Заголовки должны четко и кратко отражать содержание разделов, подразделов. Если заголовок состоит из двух предложений, их разделяют точкой.

В работе рекомендуется приводить ссылки на использованные источники. При нумерации ссылок на документы, использованные при составлении работы, приводится сплошная нумерация для всего текста работы в целом или для отдельных разделов. Порядковый номер ссылки (отсылки) приводят арабскими цифрами в квадратных скобках в конце текста ссылки. Порядковый номер библиографического описания источника в списке использованных источников соответствует номеру ссылки.

Ссылаться следует на документ в целом или на его разделы и приложения.

При ссылках на стандарты и технические условия указывают их обозначение, при этом допускается не указывать год их утверждения при условии полного описания стандарта и технических условий в списке использованных источников в соответствии с ГОСТ 7.1.

Примеры

1 приведено в работах [1] - [4].

2 по ГОСТ 29029.

3 в работе [9], раздел 5.

1.3 Оформление иллюстраций курсового проекта

Иллюстрации (чертежи, графики, схемы, компьютерные распечатки, диаграммы, фотоснимки) следует располагать в работе непосредственно после текста, где они упоминаются впервые, или на следующей странице (по возможности ближе к соответствующим частям текста работы). На все иллюстрации в работе должны быть даны ссылки. При ссылке необходимо писать слово "рисунок" и его номер, например: "в соответствии с рисунком 2" и т.д.

Чертежи, графики, диаграммы, схемы, помещаемые в работе, должны соответствовать требованиям стандартов Единой системы конструкторской документации (ЕСКД).

Количество иллюстраций должно быть достаточным для пояснения излагаемого текста работы. Не рекомендуется в Курсовому проекту приводить объемные рисунки.

Иллюстрации, за исключением иллюстраций, приведенных в приложениях, следует нумеровать арабскими цифрами сквозной нумерацией. Если рисунок один, то он обозначается: Рисунок 1.

Пример - Рисунок 1 - Схема прибора

Иллюстрации каждого приложения обозначают отдельной нумерацией арабскими цифрами с добавлением перед цифрой обозначения приложения: Рисунок А.3.

Допускается нумеровать иллюстрации в пределах раздела работы. В этом случае номер иллюстрации состоит из номера раздела и порядкового номера иллюстрации, разделенных точкой: Рисунок 2.1.

Иллюстрации при необходимости могут иметь наименование и пояснительные данные (подрисуночный текст). Слово "Рисунок", его номер и через тире наименование помещают после пояснительных данных и располагают в центре под рисунком без точки в конце.

Пример - Рисунок 2 - Оформление таблицы

Если наименование рисунка состоит из нескольких строк, то его следует записывать через один межстрочный интервал. Наименование рисунка приводят с прописной буквы без точки в конце. Перенос слов в наименовании графического материала не допускается.

1.4 Оформление таблиц в курсовой работе

Цифровой материал должен оформляться в виде таблиц. Таблицы применяют для наглядности и удобства сравнения показателей. Таблицу

следует располагать непосредственно после текста, в котором она упоминается впервые, или на следующей странице. все таблицы в работе должны быть ссылки. При ссылке следует печатать слово "таблица" с указанием ее номера.

Наименование таблицы, при ее наличии, должно отражать ее содержание, быть точным, кратким. Наименование следует помещать над таблицей слева, без абзацного отступа в следующем формате: Таблица Номер таблицы - Наименование таблицы. Наименование таблицы приводят с прописной буквы без точки в конце. Если наименование таблицы занимает две строки и более, то его следует записывать через один межстрочный интервал.

Таблицу с большим количеством строк допускается переносить на другую страницу. При переносе части таблицы на другую страницу слово "Таблица", ее номер и наименование указывают один раз слева над первой частью таблицы, а над другими частями также слева пишут слова "Продолжение таблицы" и указывают номер таблицы.

При делении таблицы на части допускается ее головку или боковик заменять соответственно номерами граф и строк. При этом нумеруют арабскими цифрами графы и (или) строки первой части таблицы. Таблица оформляется в соответствии с таблицей 1.

Таблица 1 – Заголовок таблицы

Таблица _____ - _____

		номер	наименование таблицы			
Головка {						} Заголовки граф
						} Подзаголовки граф
						Строки
						} (горизонтальные ряды)

Боковик (графа для заголовков) Графы (колонки)

Таблицы, за исключением таблиц приложений, следует нумеровать арабскими цифрами сквозной нумерацией.

Допускается нумеровать таблицы в пределах раздела при большом объеме работы. В этом случае номер таблицы состоит из номера раздела и порядкового номера таблицы, разделенных точкой: Таблица 2.3.

Заголовки граф и строк таблицы следует печатать с прописной буквы, а подзаголовки граф - со строчной буквы, если они составляют одно предложение с заголовком, или с прописной буквы, если они имеют

самостоятельное значение. В конце заголовков и подзаголовков таблиц точки не ставятся. Названия заголовков и подзаголовков таблиц указывают в единственном числе.

Таблицы слева, справа, сверху и снизу ограничивают линиями. Разделять заголовки и подзаголовки боковика и граф диагональными линиями не допускается. Заголовки граф выравнивают по центру, а заголовки строк - по левому краю.

Горизонтальные и вертикальные линии, разграничивающие строки таблицы, допускается не проводить, если их отсутствие не затрудняет пользование таблицей.

Текст, повторяющийся в строках одной и той же графы и состоящий из одиночных слов, заменяют кавычками. Ставить кавычки вместо повторяющихся цифр, буквенно-цифровых обозначений, знаков и символов не допускается.

Если текст повторяется, то при первом повторении его заменяют словами «то же», а далее кавычками. В таблице допускается применять размер шрифта меньше, чем в тексте работы.

2 Тематика курсовых работ

Тематика курсового проекта должна соответствовать основным разделам программы дисциплины «Безопасность систем баз данных». Теоретическая часть курсовой работы должна базироваться на лекционном материале дисциплины и определяется практическими потребностями предприятий в области информационных технологий. Курсовая работа должна содержать углубленную разработку вопросов проектирования защищённых баз данных.

Тематика курсовых работ определяется преподавателем, рассматривается на заседании кафедры и утверждается научно-методическим советом академии. При этом выбор основывается как на государственном стандарте, так и на направлениях научно-исследовательской и учебно-методической работы, актуальных направлениях работы других организаций, деятельность которых связана с разработкой математического, информационного и программного обеспечения ЭВМ. Студенту предоставляется право выбора одной из предложенных тем или предложения своей темы с обоснованием целесообразности ее разработки. Темой курсового проекта может быть любая проблема из организационной, технической или экономической области, с которой сталкиваются в практической деятельности предприятия.

Примерный список тем:

1. Коллекция видеофильмов

Коллекция видеофильмов MPEG4 разбросана по компьютерам, соединенным в сеть. Написать базу, в которой хранятся названия фильмов, жанр (комедия, боевик и др.), режиссер, основные артисты, краткая аннотация, сведения о наградах, год выпуска фильма, продолжительность и путь к файлу

(файлам). База предназначена для поиска фильмов по названию, автору, режиссерам, наградам, жанрам, артистам, ключевому слову. Предусмотреть возможность просмотра выбранного фильма по сети. Клиентская программа может быть запущена с любого компьютера.

2. Электронный архив

База предназначена для хранения систематизированной информации о документах электронного архива. Архив распределенный, располагается на нескольких компьютерах сети. Архив включает в себя текстовые документы, графические файлы (растровые и векторные), звуко- и видеозаписи. Документы располагаются по темам проектов. Информация о теме: шифр, наименование, руководитель, дата начала и окончания. Информация о документах: архивный номер, тема проекта, наименование документа, номер версии, дата помещения в архив, тип документа, отдел-разработчик, ФИО исполнителя.

Клиентская часть может быть запущена с любого компьютера сети, и предназначена для поиска документа(-ов) по всем перечисленным признакам, просмотра и/или копирования выбранных документов на компьютер пользователя. Все обращения к документам регистрируются в базе данных.

3. Прокат автомобилей

Предприятие занимается прокатом автомобилей гражданам и организациям. База предназначена для хранения сведений об автомобилях, клиентах и договорах проката. Пользователями базы являются владелец проката и клиенты. Владелец хранит информацию об имеющихся автомобилях, обратившихся к нему клиентах и историю проката машин. В базе зарегистрированы имеющиеся автомобили: модель, цвет, номер, фотография автомобиля, год выпуска, номер двигателя, номер кузова, дата приобретения, дата прохождения последнего техосмотра, сведения о ремонтах и др. При оформлении договоров проката фиксируется: если клиентом является физическое лицо – Ф.И.О. клиента, паспортные данные, контактный телефон, адрес; если клиентом является юридическое лицо – название организации, ИНН, контактный телефон, юридический адрес; а также: номер договора, дата заключения договора, период проката, номер автомобиля, стоимость услуг.

Клиенты с помощью базы могут выбрать понравившуюся модель, просмотрев характеристики и фотографии имеющихся автомашин, найти среди них те, которые в данный момент свободны и не находятся в ремонте, для остальных определить срок их возврата в пункт проката.

4. Должники

Персональная база для регистрации своих и чужих долгов. В базе хранится информация о личных вещах, отданных в пользование знакомым, и чужих вещей, которые надо вернуть. Вещи группируются в базе по типам (книги, диски и т.д.). Для каждой вещи хранится информацию, кому и когда она была отдана (или от кого взята) и на какой срок.

При добавлении и удалении записей необходимо проверять корректность данных, чтобы не получилось так, что по ошибке одна вещь одновременно отдана двум разным людям.

5. Автовокзал

Через автовокзал проходят пригородные и междугородние, в том числе транзитные маршруты. Автовокзал осуществляет продажу билетов на рейсы в ближайшие три дня.

Разработать базу данных автовокзала. База хранит расписание рейсов, сведения о проданных местах, и служит для оперативного выбора рейса и заказа билетов. Пользователями базы являются: администратор (редактирует расписание и маршруты, анализирует статистику рейсов); кассиры (продают билеты на рейсы на ближайшие три дня) и пассажиры (просматривают расписание, через справочную систему выбирают оптимальный рейс до нужного им населенного пункта, смотрят количество свободных мест).

В базе данных также хранится информация обо всех автобусах парка: марка, сведения о водителях, количество посадочных мест, пригоден ли автобус в данный момент к эксплуатации или находится на ремонте. Расписание составляется в двух вариантах: 1) расписание общее (пункт назначения, время и номер маршрута); 2) расписание работы каждого водителя на ближайшие три дня.

Предусмотреть статистическую обработку сведений о проданных билетах с целью определения наиболее и наименее востребованных рейсов по часам, дням недели и месяцам.

6. База данных микросхем

Разработать базу для поиска микросхем и её поставщиков. В базе хранится информация об отечественных и импортных микросхемах, их аналогах, и предприятиях-изготовителях или фирмах-поставщиках этих микросхем. Каждая микросхема в зависимости от предприятия-изготовителя может выпускаться в разном конструктивном исполнении (отличаться корпусом, временными характеристиками, температурным режимом и т.д.). Для каждой микросхемы в базе хранится: маркировка, краткая характеристика, тип корпуса, рисунок с цоколёвкой, перечень аналогов и список адресов предприятий-изготовителей или поставщиков данной микросхемы с ценами. Микросхемы группируются по типам (усилители, регистры, ПЛИС, процессоры и пр.).

Предусмотреть возможность просмотра всех микросхем заданного типа; поиск аналогов микросхемы; вывод всех предприятий, у которых можно приобрести указанную микросхему; для заданного набора микросхем определить предприятия, у которых наиболее выгодно их приобретать (критерием является не только стоимость, но и возможность приобретения как можно большего числа видов микросхем у одного предприятия).

7. Ботанический сад

База предназначена для ознакомления граждан с коллекцией растений ботанического сада и заказа любившихся растений. В базе хранятся сведения о растениях по группам (плодовые, хвойные и декоративные деревья, плодовые и декоративные кустарники, овощи, цветы, вьющиеся растения и пр.). Для каждого растения имеется: название, краткое описание (когда цветет, где применяется и пр.), одна или несколько фотографий, для растений, которые можно приобрести, дополнительно указывается, как оно продается (семенами и/или саженцами), количество, цена и сроки пересылки.

В базе также регистрируются заказы граждан: ФИО получателя, почтовый адрес, электронный адрес, телефон, список заказанных растений с указанием количества и цены, форма оплаты (предоплата или наложенный платеж). Покупатели, приобретающие растения регулярно в течение трех лет, имеют скидку 20%.

Предусмотреть возможность поиска растения по названию, или всех растений заданной группы.

С целью предсказания спроса граждан на растения в будущем посевном сезоне, предусмотреть статистическую обработку заказов: список растений, пользующихся спросом, с указанием количества; наиболее активный регион РФ, из которого сделано больше всего заказов.

8. Прокат видеокассет, CD, DVD

Разработать многопользовательскую базу для пункта проката. Пользователями базы являются операторы проката, регистрирующие выдачу и возврат кассет и дисков; и клиенты, производящие поиск интересующих фильмов и музыки.

Клиенты могут взять напрокат ресурсы двумя способами: 1) оставляя в залог документ или деньги, получают номерок, по которому регистрируется заказ; 2) оформляются по паспорту, получая карточку постоянного клиента, и далее заказ регистрируется на карточку. При прокате в базу заносится: номерок или номер карточки постоянного клиента, список взятых кассет и дисков, срок начала и окончания проката.

Клиентам предоставляется возможность поиска по базе интересующих их фильмов и музыки: название кассет и дисков, наличие на текущий момент (если отсутствуют, то предполагаемый срок возврата), инвентарные номера. Музыка ищется по: исполнителю, автору, жанру, году издания альбома. Видеофильмы ищутся по названию, жанру, фамилии режиссера, фамилиям актеров. Предусмотреть возможность поиска по шаблону и по типу носителя (кассета, CD, DVD). Для каждого фильма, кроме всего прочего, в базе хранится краткая аннотация.

С целью предсказания спроса предусмотреть возможность статистического анализа: какие жанры музыки и фильмов пользуются

наибольшим спросом, вывести десять наиболее популярных фильмов и исполнителей музыки.

9. Супермаркет

Разработать базу данных для супермаркета. Торговый зал магазина и склад товаров связаны локальной сетью. В торговом зале установлены кассовые аппараты на базе ПК. Каждый аппарат оснащен устройством чтения штрих-кодов. Цена товара выбирается по штрих-коду из базы, и может быть изменена в течение дня администратором базы данных. С помощью базы формируется кассовый чек. В магазине действует система скидок. Покупатели, имеющие карточку постоянного клиента, берут товары со скидкой $N\%$ (величину скидки определяет администратор базы).

По запросу оператора склада формируется отчет о товарах, количество которых в торговом зале близко к нулю, или они полностью раскуплены. Также формируется статистика раскупаемости товаров по дням недели и типам товаров.

10. Расписание тренировок спортклуба

Разработать базу данных спортивного клуба, в которой будет храниться информация о тренерах, спортсменах, соревнованиях, наградах спортсменов, расписании тренировок и командировок. О тренерах в базе хранится: ФИО, квалификация, имеет ли право на судейскую деятельность; список спортсменов, которых он тренирует, контактный телефон и адрес. О спортсменах хранится: ФИО, разряд, дата рождения, телефон, адрес, список наград, расписание тренировок и участия в соревнованиях, и пр.

База используется:

- для составления личного расписания тренеров (тренировки, судейство, выезд в командировки со своими подопечными);
- для составления личного расписания спортсменов;
- для вывода биографии и послужного списка спортсмена;
- для поиска судей, которые находятся на момент проведения соревнований в городе, и в соревнованиях не участвуют их подопечные.

11. Больница

Разработать базу данных для учета больных, находящихся на излечении. Больница состоит из нескольких отделений. Каждое отделение включает в себя: больничные палаты, процедурные кабинеты и кабинеты врачей. Также имеется ряд общебольничных процедурных кабинетов, не входящих в состав ни одного из отделений. Поступающие больные регистрируются (паспортные данные, номер мед. полиса) и размещаются в отделение, соответствующее основному типу заболевания.

К каждому больному прикрепляется лечащий врач, и, если есть заболевания иного профиля, врачи-консультанты из других отделений.

В базе данных должны учитываться сведения о поступлении, переводах и выписках больных; истории болезни, назначенные процедуры, лекарства и диеты. Пользователями базы данных являются: приемное отделение (регистрация, переводы, выписка больных, поиск больных по фамилии), столовая (ежедневно получает заказ на приготовление диет.питания), врачи (ведение истории болезни, назначение лекарств и диет).

12. Электронная библиотека

На сервере хранятся файлы с документацией различного характера. Документация группируется по категориям. Для каждого документа хранится название, авторы, язык, краткое содержание, список ключевых слов. Разработать базу данных для поиска документации, клиентская часть базы может быть запущена с любого компьютера локальной сети. Поиск может вестись по автору, названию, категории, ключевым словам, дате создания документа. По запросу выбранный файл копируется на машину пользователя. Предусмотреть возможность просмотра списка из 20 самых популярных документов.

13. База данных УВД

Разработать базу данных УВД, в которой хранится информация о гражданах, состоящих на учете в милиции и история их преступлений. Для каждого человека хранится: ФИО, дата рождения, краткая биография, список псевдонимов, фотографии по годам, отпечатки пальцев, информация о судимостях, номера уголовных дел, ссылки на электронные документы. Доступ к некоторым документам может ограничиваться паролем. Также в базе хранится список правонарушений, совершенных в контролируемом районе.

Предусмотреть возможность поиска людей по имени, псевдониму, номерам уголовных дел и статьям УК РФ.

Выходные документы:

- список не раскрытых дел;
- список раскрытых дел;
- информация о лицах, проходящих по делу № X.

14. Банк-магазин

Банк и магазины связаны договором, согласно которому клиенты банка могут расплачиваться за покупки с помощью кредитной карты. Каждый такой магазин имеет в банке счет. Деньги, снятые с кредитной карточки клиента, автоматически переводятся на счет магазина. Гражданин может открыть в банке несколько счетов "до востребования" и получить кредитную карту. Когда карта ему выдается, то он отмечает, с каких счетов будут сниматься деньги по этой карте. Если в процессе оплаты деньги с одного счета сняты полностью, далее они снимаются со следующего. Когда все счета исчерпаны, покупка в магазине по карте невозможна.

По требованию клиента (например, в случае потери кредитной карты), расплата кредитной картой блокируется.

Разработать базу данных для проведения банковских операций начисления денег на счет, снятия денег со счета, выдачи и блокировки кредитной карты, оплаты в магазинах кредитной картой.

Выходные документы:

- ежедневная распечатка доходов для каждого магазина: время, сумма, номер кредитной карты;

- распечатка для клиента: в каких магазинах, сколько он потратил, в какое время, с какого счета сняты средства;

- для конкретного счета: распечатка о дате, времени и сумме начислений и расходов;

- список помесечных доходов магазинов.

15. Товары – почтой

База данных установлена в фирме, которая производит поставку товаров почтой в разные регионы страны. В фирме работает несколько сотрудников. Каждый сотрудник курирует несколько заказов. Заказ включает в себя перечень товаров с указанием их количества. Заказы могут оплачиваться наложенным платежом после получения заказа, или предварительно банковским переводом. Если в течение 14 дней со дня регистрации заказа банковский перевод не пришел, заказ считается аннулированным. Сумма пересылки рассчитывается в процентах от стоимости заказа, и зависит от региона и формы оплаты.

База сетевая. Сотрудники вносят в базу новые заказы; отмечают, когда пришел денежный перевод, или заказчик внес наложенный платеж, и отмечают, если посылка отправлена, и заказ выполнен. Сотрудники не имеют права просматривать и редактировать заказы, которые обслуживаются их коллеги. При регистрации заказов в базу вносятся следующие сведения о заказчике: ФИО или название организации, почтовый адрес и телефон, наличие e-mail. Заказчик, который приобрел товаров на сумму более 5 тысяч руб., переходит в разряд постоянных клиентов и имеет 10%-ную скидку от стоимости.

Заказы наложенным платежом не принимаются от заказчиков, которые трижды аннулировали свой заказ.

16. Каталог статей в периодических изданиях

Электронный каталог предназначен для учета и поиска статей в периодических изданиях, которые выписывает библиотека. Учет проводится по нескольким категориям областей знаний. Одна статья может относиться к нескольким категориям. Для каждой статьи хранятся выходные данные (автор, название, название журнала/газеты, год, номер, начальная страница), список ключевых слов и, по возможности, ее краткое содержание.

Поиск статей производят посетители библиотеки. Поиск может вестись по категориям, автору, названию и по ключевым словам. Поисковая система в результате формирует список статей с полными выходными данными и кратким содержанием. Второй тип поиска: вывести список журналов и газет, которые выписывала библиотека в заданный период времени. В списке указываются номера и годы выпуска. Третий тип поиска: для выбранного журнала и газеты (название может быть задано по маске) вывести список номеров, имеющихся в библиотеке.

17. Хлебозавод

Хлебозавод включает в себя: склад сырья и три цеха по выпечке хлебобулочных изделий: хлеб и батоны, сдобная продукция, торты и пирожные. У каждого цеха свой ассортимент. На каждое изделие установлен план выпуска (штук в день) и рецептура, в которой указан расход ингредиентов в килограммах на одно изделие. Ингредиенты хранятся на складе. Хлебозавод принимает заказы на производство нестандартной продукции по рецепту заказчика.

В начале рабочего дня планируется выпуск изделий по цехам. Для каждого цеха определяется перечень изделий и план выпуска, согласно которому на склад направляется план отгрузки сырья по цехам. Если оставшегося на складе сырья недостаточно для бесперебойной работы завода в течение трех дней (с текущим планом загрузки цехов), то на склад направляется требование на закупку сырья с указанием количества, достаточного на 10 дней работы завода.

База данных предназначена для координации работы цехов и отдела поставок сырья. В соответствии с планом выпуска база формирует выходные документы:

- для каждого цеха по выпечке план выпуска изделий. Если изделие по рецепту заказчика, то к плану прикладывается рецептура;
- для склада – план отгрузки по цехам;
- для склада – план закупок сырья.

18. Театральная касса

Организация занимается распространением билетов на представления в цирке, театрах и концертных залах города. Стоимость билетов зависит от «посадочной зоны» конкретного зала. Продажа билетов производится в нескольких кассах (филиалах). Кассы и офис связаны сетью.

Разработать базу данных для продажи билетов. Поисковая система базы определяет наличие свободных мест на представления, и исключает места согласно купленным билетам. Офис может получить статистику раскупаемости билетов через филиалы.

19. Проектная организация

В проектной организации работают инженеры-конструкторы, инженеры-схемотехники и инженеры-программисты. Руководители проектов планируют нагрузку работников с использованием базы данных. Руководитель может одновременно вести несколько проектов. Инженеры-исполнители в каждый момент времени участвуют не более чем в одном проекте. При открытии проекта руководитель планирует квалификацию и число исполнителей и определяет каждому работнику сроки начала и окончания работ. Для выполнения работ руководитель может привлекать инженеров, которые на заданный период времени не участвуют в других проектах.

Выходные документы:

- каждый инженер может просмотреть график своего участия в различных проектах;
- руководитель получает план работ с указанием исполнителей и сроков их участия в проекте;
- руководитель может просмотреть все свои проекты.
- поиск инженеров, не участвующих ни в одном проекте в указанный промежуток времени.

20. Расписание занятий

База данных учебного отдела содержит расписание занятий по аудиториям: время, группа, дисциплина, аудитория, преподаватель. Для каждой аудитории хранится число посадочных мест, назначение (лекционный зал, практические занятия, лаборатория, дисплейный класс, наличие доски – признаки могут комбинироваться). В расписании учитываются верхние и нижние недели.

База дополнена поисковой системой. Она применяется для поиска свободных аудиторий. Запрашивается тип аудитории, количество человек, день недели и номер пары.

Выходные документы:

- для группы вывести ее расписание на неделю;
- для преподавателя вывести его личное расписание;
- для указанной аудитории вывести перечень групп, которые там занимаются.

21. Отдел кадров

Разработать базу данных для отдела кадров. О сотрудниках хранится следующая информация: ФИО, фотография, дата рождения, паспортные данные, семейное положение, сведения о детях, образование (среднее, средне-специальное, высшее) профессия (по диплому), номер и дата выдачи диплома;

"история" занимаемых на этом предприятии должностей: должность, ставка, номер приказа, дата поступления и окончания; сведения о наградах.

Выходные документы:

- список руководящих работников предприятия с указанием отдела и занимаемой должности;
- список всех сотрудников указанного отдела с должностями и ставками;
- список всех сотрудников пенсионного возраста (55 лет для женщин и 60 для мужчин);
- список сотрудников, имеющих детей до 18 лет;
- для указанного сотрудника вывести послужной список и перечень наград.

22. Цех

Для сборки изделий на заводе применяется ряд унифицированных деталей. Детали изготавливаются в механическом цехе, затем направляются в цех лакокрасочных и гальванических покрытий, а после покрытия поступают на сборочный конвейер. Диспетчер производства цеха покрытий каждый день должен запускать в покрытие нужное количество каждой детали, чтобы сборочный конвейер не простаивал. План выпуска изделия каждого типа устанавливается ежедневно.

Разработать базу данных, связывающую механический цех, цех покрытий и сборочный цех. На основании плана выпуска изделий сформировать отчеты:

- 1) для механического цеха – план выпуска деталей;
- 2) для цеха покрытий – перечень деталей с указанием типа покрытия (хромирование, никелирование, лак, краска) и количества;
- 3) для сборочного цеха – план сборки. В плане сборки указывается число изделий, для каждого типа изделий – перечень деталей.

В базе должен храниться план выпуска изделий за последние 30 дней.

23. Компьютерная фирма

Компьютерная фирма состоит из нескольких филиалов. Каждый филиал оснащен складом, на котором хранятся комплектующие и собранные их ПК. Имеется несколько "стандартных" моделей ПК и возможна сборка под заказ. Филиалы связаны сетью. База данных предназначена для учета товаров на складах, выбора покупателями комплектующих и оборудования. Для каждого вида комплектующих хранится его характеристика, цена и размещение по складам филиалов. Через базу идет учет покупок и продаж комплектующих и готовых моделей.

Выходные документы:

- прайс-листы по типам комплектующих;

- состав "стандартных" моделей ПК;
- для указанного вида товара список, на каком складе сколько единиц товара находится;
- пользователь определяет состав устройств ПК и предел цены: вывести все возможные варианты сборки ПК из комплектующих, имеющихся на складах.

Предусмотреть возможность просмотра статистики комплектующих и моделей, пользующихся повышенным спросом.

24. Результаты сессии

В деканате учитываются результаты сессий за все семестры. Для каждой специальности хранится учебный план (список дисциплин по семестрам, количество часов, формы контроля: экзамен, зачет, курсовой проект/работа). Учебные планы отличаются на 4-5 курсах дисциплинами специализации.

В базу заносятся все факты пересдачи по допускам. Итоговой считается последняя оценка. Она идет в выходные документы: выписку из зачетной ведомости; список студентов и дисциплин, которые им необходимо досдать к началу текущей сессии (в том числе имеющие неудовлетворительные оценки); список студентов, окончивших текущую сессию на "4" и "5" – для начисления стипендии, и только на "5" – для начисления повышенной стипендии; список студентов 5 курса (проучившихся 9 семестров) – кандидатов на красный диплом (оценки только "4" и "5", оценок "4" не более 25%).

25. Междугороднее сообщение

База данных представляет собой справочную систему, используемую для выбора оптимального способа добраться до заданного пункта назначения. База хранит информация обо всех железнодорожных, автобусных и авиационных рейсах, которыми можно добраться из нашего города в другие населенные пункты. Для каждого рейса хранится: № рейса, вид транспорта, пункт назначения, время отбытия, время прибытия. Поиск может вестись по нескольким признакам (по отдельности или смешанно): пункт назначения, вид транспорта, время отбытия, время прибытия, день недели (дата).

База многопользовательская. Пользователями являются операторы, осуществляющие ввод и редактирование данных, и граждане, только просматривающие результаты поиска.

26. Биржа труда

База данных биржи труда предназначена для регистрации заявок с предприятий и анкет граждан, ставших на учёт. Пользователями базы являются работники биржи и безработные граждане. Работники биржи регистрируют заявки от предприятий и граждан, результаты устройства на работу (удачное или неудачное). Граждане используют базу для поиска подходящих вакансий и просмотра статистики востребованных профессий.

В заявке от предприятия указывается: наименование предприятия, требуемая квалификация работника, заработная плата и срок, в течение которого будут рассматриваться кандидаты. Заявка удаляется, если истёк срок ее действия, или предприятие сообщило о приеме на работу.

Анкета работника, ставшего на учёт, включает: ФИО, образование (неполное среднее, среднее, среднее специальное, высшее) и список профессий, которыми он владеет (список всевозможных профессий стандартизован). Гражданин, поступивший на работу, снимается с учёта. Также снимаются с учёта граждане, которым по одной и той же профессии предлагались вакансии три раза, но эти предложения их не устроили; впоследствии такие граждане имеют право стать на учет не ранее чем через три месяца по той же профессии, или немедленно по другой профессии.

База данных должна автоматически формировать направления для устройства на работу и список вакансий с заработной платой не менее указанного минимума (перечень профессий задается).

27. Учёт коммунальных платежей

Разработать базу данных для учета коммунальных платежей граждан и расчета пени. В базе хранятся адреса, информация о квартиросъемщиках и членах семьи, прописанных на этой площади (ФИО, паспортные данные), информация о льготах квартиросъемщиков, сумма ежемесячной оплаты (начисленная и фактически уплаченная), пени. База применяется для выписки счетов-квитанций на оплату. В счет-квитанцию входят следующие статьи:

- ремонт и содержание жилья (= общая площадь Ч стоимость по тарифу за м²);
- отопление (= общая площадь Ч стоимость по тарифу за м²; в частных домах и квартирах, оснащенных водогрейными котлами, не начисляется);
- горячая вода (= число проживающих Ч стоимость по тарифу на 1 чел.; в частных домах и квартирах, оснащенных водогрейными котлами, не начисляется);
- холодная вода (= число проживающих Ч стоимость по тарифу на 1 чел.);
- газ (= число проживающих Ч стоимость по тарифу на 1 чел.; для частного сектора и квартир с водогрейными котлами дополнительно прибавляется: общая площадь Ч стоимость по тарифу за м²);
- прочие отчисления (вывоз мусора, антенна, лифт) – ставка фиксированная в расчете на 1 чел., эти статьи начисляются в зависимости от места проживания.

Расчет платежей на проживающих в квартире и имеющих льготы, производится со скидкой 50%, на остальных – 100%-ные начисления.

Оплата за месяц должна производиться не позднее 10 числа следующего месяца включительно. За каждый просроченный день начисляется пени в размере 0,5% от суммы оплаты.

Предусмотреть автоматизированное составление следующих выходных документов:

1. счета-квитанции по квартирам;
2. список квартир, имеющих задолженности:
 - 2.1. с суммой долга, превышающей заданный предел, руб.;
 - 2.2. с задолженностью по оплате 3 и более месяцев;
 - 2.3. всех задолжников.

28. Библиотека

Разработать электронный каталог библиотеки с поисковой системой. Существует алфавитный и систематический каталог, при этом одна книга может соответствовать нескольким разделам систематического каталога. Для каждой книги хранятся выходные данные: шифр, авторы, название, место и год издания, число страниц, количество экземпляров и их инвентарные номера, местонахождение каждого экземпляра (читальный зал или абонемент). Поиск книг может проводиться по разделу систематического каталога, шифру, автору и названию (можно по маске). Также в базе данных учитываются все факты выдачи книг читателям. В результатах поиска должно отражаться, сколько книг в настоящий момент находится в фондах библиотеки, и сколько выдано на руки. Для библиотекарей предусмотреть вывод списка книг, к которым не было обращений в течение последних пяти лет.

29. Зоопарк

База данных предназначена для координации работы отдела поставок, склада продуктов и кухни, а также для хранения сведений о животных, содержащихся в зоопарке и историй болезни. Соответственно пользоваться базой будут четыре отдела: отдел поставок, склад продуктов, кухня, ветеринарный отдел. Состав продуктов, входящих в рацион конкретного животного, определяет ветеринарный отдел в зависимости от вида животного, возраста и состояния здоровья. В рацион включаются овощи, фрукты, каши, мясные и молочные продукты. Все продукты хранятся на складе. Нормативный срок использования мясных и молочных продуктов – 1-3 дня, овощей – 10-30 дней, фруктов – 5-20 дней, круп – 120 дней. Ежедневно в отдел поставок направляется заказ на закупку продуктов, которые скоро закончатся, и заказ на лекарства. База данных предназначена для автоматизированного составления запросов в отдел поставок и выписки рецептов для кухни на приготовление рационов животных. Выходные документы:

- перечень продуктов и лекарств, требующих закупки (по категориям);

- перечень продуктов для выдачи со склада на кухню, с указанием количества;
- среднегодовой расход продуктов по категориям.

30. Автопарк

Автотранспортное хозяйство осуществляет перевозки людей и грузов на заказ. Имеющийся транспорт: легковые автомобили, грузовики и автобусы. Каждый из водителей (категорий С, D, E) закреплен за конкретным транспортным средством, но может замещать отсутствующего коллегу в случае, если собственный транспорт находится на ремонте. Все перевозки регистрируются в базе данных: заказчик, время и дата заказа (с точностью до половины дня), тип перевозок, пункт назначения, транспортное средство, водитель. Сроки ремонтов транспорта также протоколируются. Выходные документы:

- "история" перевозок транспортного средства в заданный промежуток времени: куда ездило, кто был водитель, кто заказчик, и пр.;
- список транспортных средств определенного типа (например, автобусов с числом посадочных мест не менее 20), свободных на заданный интервал времени;
- списки водителей с указанием категории, которые на текущий день остались без работы по причине ремонта их транспорта.

31. Издательство

База данных издательства хранит информацию о статьях, опубликованных журналистами в журнале, персональных сведениях о журналистах, а также реальное содержимое журнала. О журналистах хранятся: ФИО, контактный телефон, адрес постоянного проживания, образование, стаж, штатный/нештатный корреспондент. О статьях: название; год, номер и страницы журнала; число знаков текста, авторы (может быть несколько), жанр (аналитический обзор, интервью, исторический очерк, новости и т.д.), рейтинг (числовая мера качества статьи). Файл с текстом журнала (в формате издательской системы) также помещается в базу данных. База данных применяется для поиска статей по автору и жанру; содержащиеся в ней сведения также могут использоваться поиска областей, в которых наиболее полно раскроется творческий потенциал работника. Выходные документы:

- для заданного журналиста вывести перечень его статей в журналах;
- для каждого работника определить три жанра, статьи в которых обладают наивысшим рейтингом;
- перечень журналистов в порядке убывания их среднего рейтинга.

32. Склад товаров

Однопользовательская база данных оптового склада для учета прихода/расхода товаров. В базе хранятся приходные ордера (номер ордера,

дата, список товаров с указанием упаковки, количества, закупочной цены, адрес и название фирмы-поставщика) и расходные ордера (содержание аналогично). База должна автоматически формировать следующие документы:

- прайс-лист с указанием цены товара и минимального объема закупок;
- расходный ордер;
- месячная прибыль по каждому типу товаров.

Название фирм-поставщиков и отображать на экране по убыванию суммы закупленного у нее товара.

33. Спрос и предложение

Составить базу данных для учета спроса и предложения товаров и услуг, и помощи в поиске партнеров. В базу заносится информация о гражданах, предприятиях и организациях: ФИО (название организации), контактная информация (телефон и/или адрес), вид услуг, спрос или предложение, срок, прочая информация (минимальный объем, цена, условия реализации и т.п.).

Выходные документы:

- для заданного типа спроса вывести все имеющиеся предложения, действительные на текущий момент;
- для заданного типа предложения вывести все заявки на спрос;
- найти и вывести пары: спрос-предложение.

34. Экологические технологии

База данных предназначена для поиска технологий обезвреживания и использования отходов, вырабатываемых промышленными предприятиями страны. В базе хранится информация о предприятиях: название, адрес, контактные телефоны, ФИО директора или иных руководителей; вырабатываемой продукции, побочных продуктах производства, полуфабрикатах, отходах производства, исходном сырье и ингредиентах (с указанием количества). Всё вышеперечисленное определяется применяемой технологией. Список технологий, применяемых на каждом предприятии, также хранится в базе данных.

База предназначена для выявления таких сырьевых связей между предприятиями, в которых побочные продукты и отходы производства одного предприятия могут служить сырьем для другого предприятия. Желательно, чтобы объемы производства и потребления были соизмеримы.

Выходные документы:

- для заданной технологии вывести перечень предприятий, на которых данная технология применяется (выводить адреса предприятий и прочую контактную информацию);

- для заданного ингредиента вывести список предприятий, на которых он используется как сырье, и на которых он является продуктом/отходом производства (с указанием годового объема);

- вывести список предприятий, которые могут быть вовлечены в цепочку производитель-потребитель, с указанием вида продукта.

35. База данных ГИБДД

В базе данных хранится информация об автомобилях и водительских удостоверениях граждан. Для автомобиля хранится: модель, цвет, текущие номера, дата постановки на учет, дата снятия с учета, номера двигателя и кузова, владелец, срок последнего техосмотра. Также имеется список автомобилей, находящихся в розыске. При постановке и снятии с учета проверяется, чтобы автомобиль не был в розыске.

При выдаче водительского удостоверения в базу заносится: номер удостоверения, дата выдачи, категория, ФИО, дата рождения, паспортные данные, прописка, фотография, вождение в очках. Также в базе хранится информация обо всех дорожных авариях, вовлеченных в них автомобилях и наложенных штрафах.

Выходные документы:

- для указанного владельца машины вывести все дорожные происшествия, произошедшие с его машинами (машиной) за последние N месяцев;

- список автомобилей с просроченной датой техосмотра.

2.1 Порядок выполнения варианта курсовой работы

ПРОЕКТИРОВАНИЕ БАЗЫ ДАННЫХ.

РАБОТА С ТАБЛИЦАМИ. СОЗДАНИЕ ДИАГРАММЫ.

ЦЕЛЬ РАБОТЫ

Научиться проектировать базу данных по индивидуальному заданию. Построить уточненную концептуальную модель в виде ER-диаграммы. Разработать текст SQL-запросов для создания базы данных, доменов, таблиц, ограничений целостности и при необходимости других объектов БД.

Получить представление базы данных в среде Microsoft SQL Server 2008 R2. Создать таблицы, диаграмму и связи между таблицами.

ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

Терминология, используемая в области баз данных очень широка и неоднозначна, будем последовательно вводить термины и определения,

описывать особенности существующих технологий для всех этапов работы с базами данных.

Предметная область (ПО) - часть реального мира (например: банк, институт, спортклуб, деканат, студенческая группа, предприятие, бухгалтерия, цех и т.д.). Конкретное состояние ПО фиксируется в данных. ПО описывается с какой-то степенью точности, следовательно, она является моделью, т.е. создается модель предметной области (МПО). ПО моделируется при помощи информационных объектов и связей между ними, следовательно, МПО - информационная модель.

Объект - это нечто существующее и различное. Иногда вместо слова "объект" употребляют слово "сущность". Каждый объект имеет уникальное имя.

При идентификации (выделении) ПО теория рекомендует составить описание объекта, т.е. краткое информативное утверждение, которое позволяет установить: является ли реальный предмет экземпляром объекта.

Каждый объект может быть задан (выделен среди других) при помощи наборов свойств. Свойства, которые справедливы для всех возможных экземпляров объекта, т.е. являются общими, называются атрибутами объекта. Т.к. объект может иметь несколько свойств, то он задается набором своих атрибутов.

Элемент данных (ЭД) - то, при помощи чего определяется или задается свойство данных объекта. ЭД имеет уникальное имя и совокупность значений, связанных с этим именем. Это множество называется доменом.

Типичные домены - это множество целых или рациональных чисел, множество символьных строк.

Иногда те значения, которые принимают элементы данных, называются данными, которые хранятся в предметной области.

Пример:

Объект	Элементы данных	Домен
Личность	паспорт Ф.И.О. адрес	38 04 124674 Газаев С.Е.
Счет	тип номер баланс	расходный 367 27.8

Если речь идет о конкретном объекте, то среди множества его атрибутов существует некоторое его подмножество, значения атрибутов которого однозначно определяют каждый экземпляр данного объекта. Это подмножество называется ключом.

Ключ (ключевой элемент данных) - минимальное множество элементов данных, значение которых однозначно определяет запись об объекте.

Суперключ - любое множество атрибутов, включающее ключ.

Элементы данных, не являющиеся ключами, называются атрибутами. Ключей может быть несколько для одного объекта. Ключ, который выбирают для идентификации записи, называются первичными (основными). Если ключ содержит несколько элементов данных, он называется составным, сцепленным, сложным.

Запись - совокупность значений элементов данных, которые описывают конкретный экземпляр объекта.

Пример:

ЛИЧНОСТЬ=Номер паспорта+Фамилия+Имя+Отчество+Адрес

Если два элемента данных двух конкретных информационных объектов могут использоваться вместе, тогда говорят, что между такими объектами есть связь.

Связь – отношение между элементами данных или объектами, которые могут использоваться совместно в информационной модели. Связь можно выявить в предметной области на основе анализа взаимодействий элементов данных или объектов. Также она возникает, когда есть соотношение "часть - целое", "класс - подкласс", "вид - подвид".

Свойства баз данных (требования, предъявляемые к файлам в данной предметной области).

1. **Неизбыточность** в базах данных (минимально необходимая избыточность). Если данные об одном и том же объекте или связи объекта, без остаточных на то оснований, хранятся в нескольких файлах данных, то такая ситуация называется избыточностью. Избыточность приводит к противоречивости данных (например, в одном файле сведения об объекте поменяли, а в другом - забыли). Избыточность в наборах файлов минимальная, если при удалении какого-нибудь файла теряются сведения об объекте данных или о его связях. Файлы данных интегрируют в одну систему. Выделяют в этих файлах одинаковые части.

2. **Независимость данных и программ.** Все данные хранятся в файлах. Для файла различают логическую и физическую организацию. Программисту необходимо знать структуру записи и порядок их следования. Программа физически не зависит от данных, если она не зависит от того, как реально на диске расположены данные. Под логической независимостью данных и программ подразумеваются два свойства:

- Возможность создания и ведения (модификация, удаление, добавление) файлов данных с помощью универсальной системы управления данными независимо от прикладных программ.

- Изменение в файлах данных (например, добавление еще одного поля данных) не должно приводить к изменению прикладных программ.

3. **Взаимосвязь данных и их структурируемость** (файлы должны быть формализованы).

4. Способность набора данных и системы данных отвечать на непредвиденные запросы, наращивать (расширять) предметную область без переделки существующих файлов.

Для добавления новых или модификации существующих данных, а также для осуществления поиска данных, применяется общий управляющий способ. Программно-технические средства, реализующие общий управляющий способ и поддерживающие все свойства (т.е. которая позволяет реализовать) жизненный цикл) БД, называют системой управления БД (СУБД). Конкретная СУБД с конкретной заполненной БД называется банком данных. Иногда сюда включаются прикладные программы. Жизненный цикл БД - это проектирование, заполнение и обслуживание.

Модель Сущность-Связь (ER-модель) (англ. entity-relationship model или entity-relationship diagram) — это модель данных, позволяющая описывать концептуальные схемы. Она предоставляет графическую нотацию, основанную на блоках и соединяющих их линиях, с помощью которых можно описывать объекты и отношения между ними какой-либо другой модели данных. В этом смысле ER-модель является мета-моделью данных, то есть средством описания моделей данных.

ER-модель удобна при прототипировании (проектировании) информационных систем, баз данных, архитектур компьютерных приложений, и других систем (далее, моделей). С её помощью можно выделить ключевые сущности, присутствующие в модели, и обозначить отношения, которые могут устанавливаться между этими сущностями. Важно отметить что сами отношения также являются сущностями (выделяются в отдельные графические блоки), что позволяет устанавливать отношения на множестве самих отношений.

ER-модель является одной из самых простых визуальных моделей данных (графических нотаций). Она позволяет обозначить структуру «крупными мазками», в общих чертах. Это общее описание структуры называется ER-диаграммой или онтологией выбранной предметной области (area of interest).

На этапе перехода к реализации данной ER-диаграммы в виде реальной информационной системы или программы, происходит отображение ER-модели в более детальную модель данных реляционной (объектной, сетевой, логической, или др.) базы данных, которая называется физической моделью данных по отношению к исходной ER-диаграмме.

Существует несколько графических нотаций описания ER-диаграмм (несколько похожих ER-моделей данных). Классический вариант описывается во второй части данной статьи. Есть несколько типичных примеров использования ER-модели данных: IDEF1x (ICAM DEFinition Language) и dimensional modelling.

Тип данных определяет формат, в котором хранится значение столбца. Стандарт SQL92 предусматривает набор базовых типов данных. Реальные СУБД, как правило, дополняют этот набор своими, специфическими типами данных с целью повышения функциональности системы. В таблице 1 приведены типы данных, поддерживаемые СУБД Microsoft SQL Server 2000, MySQL 3.23, InterBase 6.0.

Рассмотрим типы данных. При выборе типа данных для столбца следует отдавать предпочтение типу, который позволит хранить любые возможные для этого столбца значения и занимать при этом минимальное место на диске. Типы данных в MS SQL Server можно разделить на восемь категорий:

Целочисленные данные

`bit` (1 байт). Может хранить только значения 0, 1 или `null` (пустое значение, сообщающее об отсутствии данных). Его удобно использовать в качестве индикатора состояния – включено/выключено, да/нет, истина/ложь.

`tinyint` (1 байт). Целые значения от 0 до 255.

`smallint` (2 байта). Диапазон значений от -215 (-32768) до 215 (32767).

`int` (4 байта). Может содержать целочисленные данные от -2147483648 до 2147483647.

`bigint` (8 байт). Включает в себя данные от -263 (9223372036854775808) до 263 (9223372036854775807). Удобен для хранения очень больших чисел, не помещающихся в типе данных `int`.

Текстовые данные

`char`. Содержит символьные не Unicode - данные фиксированной длины до 8000 знаков.

`varchar`. Содержит символьные не Unicode - данные переменной длины до 8000 знаков.

`nchar`. Содержит данные Unicode фиксированной длины до 4000 символов. Подобно всем типам данных Unicode его удобно использовать для хранения небольших фрагментов текста, которые будут считываться разноязычными клиентами.

`nvarchar`. Содержит данные Unicode переменной длины до 4000 символов.

Десятичные данные

`decimal`. Содержит числа с фиксированной точностью от $-1038-1$ до $1038-1$. Он использует два параметра: точность и степень. Точностью называется общее количество знаков, хранящееся в поле, а степень – это количество знаков справа от десятичной запятой.

`numeric`. Это синоним типа данных `decimal` – они идентичны.

Денежные типы данных

money (8 байт). Содержит денежные значения от -263 до 263 с десятичной точностью от денежной единицы.

smallmoney (4 байта). Содержит значения от -214748,3648 до 214748,3647 с десятичной точностью.

Данные с плавающей точкой

float. Содержит числа с плавающей запятой от -1,79E+38 до 1,79E+38.

real. Содержит числа с плавающей запятой от -3,40E+38 до 3,40E+38.

Типы данных даты и времени

datetime (8 байт). Содержит дату и время в диапазоне от 1 января 1753 года до 31 декабря 9999 года с точностью 3,33 мс.

smalldatetime (4 байта). Содержит дату и время, начиная от 1 января 1900 года и заканчивая 6 июнем 2079, с точностью до 1 минуты.

Двоичные типы данных

binary. Содержит двоичные данные фиксированной длины до 8000 байт.

varbinary. Содержит двоичные данные переменной длины до 8000 байт.

Специализированные типы данных

sql_variant. Используется для хранения значения с различными типами данных.

timestamp. Используется для установки временных меток записей при вставке, которые соответствующим образом обновляются.

uniqueidentifier. Глобальный уникальный идентификатор.

xml. Используется для хранения целых документов или фрагментов XML.

Окно Management Studio имеет следующую структуру:

Оконное меню – содержит полный набор команд для управления сервером и выполнения различных операций.

Панель инструментов – содержит кнопки для выполнения наиболее часто производимых операций. Внешний вид данной панели зависит от выполняемой операции.

Панель «Обозреватель объектов». Это панель с древовидной структурой, отображающая все объекты сервера, а также позволяющая производить различные операции, как с самим сервером, так и с его базами данных и их объектами. Обозреватель объектов является основным инструментом для разработки.

Рабочая область. В рабочей области производятся все действия с базой данных, а также отображается её содержимое.

Прежде чем перейти к созданию своих собственных рабочих баз данных рассмотрим служебные базы данных SQL Server, которые создаются

автоматически в процессе его установки. Если мы раскроем узел «Базы данных – Системные базы данных» в обозревателе объектов, то увидим следующий набор служебных баз данных:

1. master. Главная служебная база данных всего сервера. В ней хранится общая служебная информация сервера: настройки его работы, список баз данных на сервере с информацией о настройках каждой базы данных и ее файлах, информация об учетных записях пользователей, серверных ролях и т.п.

2. msdb. Эта база данных в основном используется для хранения информации службы SQL Server Agent (пакетных заданий, предупреждений и т.п.), но в нее записывается и другая служебная информация (например, история резервного копирования).

3. model. Эта база данных является шаблоном для создания новых баз данных в SQL Server. Если внести в нее изменения, например, создать набор таблиц, то эти таблицы будут присутствовать во всех создаваемых базах данных.

4. tempdb. Эта база данных предназначена для временных таблиц и хранимых процедур, создаваемых пользователями и самим SQL Server. Эта база данных создается заново при каждом запуске SQL Server.

Таблицы представляют собой объекты базы данных, используемые непосредственно для хранения всех данных. Одним из самых главных правил организации баз данных является то, что в одной таблице должны храниться данные лишь об одном конкретном типе сущности (например, книги, читатели, движение книги и т. п.).

Панель инструментов – содержит кнопки для выполнения наиболее часто производимых операций. Внешний вид данной панели зависит от выполняемой операции.

Панель «Обозреватель объектов». Это панель с древовидной структурой, отображающая все объекты сервера, а также позволяющая производить различные операции, как с самим сервером, так и с его базами данных и их объектами. Обозреватель объектов является основным инструментом для разработки.

Рабочая область. В рабочей области производятся все действия с базой данных, а также отображается её содержимое.

Прежде чем перейти к созданию своих собственных рабочих баз данных рассмотрим служебные базы данных SQL Server, которые создаются автоматически в процессе его установки. Если мы раскроем узел «Базы данных – Системные базы данных» в обозревателе объектов, то увидим следующий набор служебных баз данных:

1. master. Главная служебная база данных всего сервера. В ней хранится общая служебная информация сервера: настройки его работы, список баз

данных на сервере с информацией о настройках каждой базы данных и ее файлах, информация об учетных записях пользователей, серверных ролях и т.п.

2. msdb. Эта база данных в основном используется для хранения информации службы SQL Server Agent (пакетных заданий, предупреждений и т.п.), но в нее записывается и другая служебная информация (например, история резервного копирования).

3. model. Эта база данных является шаблоном для создания новых баз данных в SQL Server. Если внести в нее изменения, например, создать набор таблиц, то эти таблицы будут присутствовать во всех создаваемых базах данных.

4. tempdb. Эта база данных предназначена для временных таблиц и хранимых процедур, создаваемых пользователями и самим SQL Server. Эта база данных создается заново при каждом запуске SQL Server.

Таблицы представляют собой объекты базы данных, используемые непосредственно для хранения всех данных. Одним из самых главных правил организации баз данных является то, что в одной таблице должны храниться данные лишь об одном конкретном типе сущности (например, книги, читатели, движение книги и т. п.).

Данные в таблицах организованы по полям и записям. Поля (или столбцы таблицы) содержат определенный тип информации, например, фамилию, адрес, место работы читателя. Запись (или строка таблицы) - группа связанных полей, содержащих информацию об отдельном экземпляре сущности.

ВЫПОЛНЕНИЕ РАБОТЫ

ЗАДАНИЕ НА РАЗРАБОТКУ БАЗЫ ДАННЫХ:

Разработать базу данных предметов инвентаризации, в которой будет храниться информация о кафедрах, ответственных, компьютерах, компьютерных комплектующих. О кафедрах в базе хранится: название кафедры, декан, телефон кафедры. О компьютерах хранится: наименование, описание, прочее. Об ответственных хранится: ФИО, телефон, e-mail, прочее. О комплектующих в базе хранится: тип, имя, серийный номер, описание.

База используется:

- Для инвентаризации материальных ценностей Академии ИМСИТ.
- Для поиска ответственных лиц за материальные ценности.
- Для формирования отчетов.

Область включает в себя:

1. Кафедры – основное учебно-научное подразделение, осуществляющее учебную, методическую и научно-исследовательскую работу по одной или нескольким родственным дисциплинам, воспитательную работу среди студентов, а также подготовку научно-педагогических кадров, повышение квалификации специалистов.

2. Аудитории – помещения для проведения лекционных, семинарских, практических и иных занятий.

3. Компьютеры – ЭВМ (электронная вычислительная машина) — машина для проведения вычислений, а также приёма, переработки, хранения и выдачи информации по заранее определённом алгоритму (компьютерной программе).

4. Комплектующие – составляющие части компьютеров.

5. Ответственные – люди, отвечающие за состояние и сохранность компьютеров и комплектующих в данной аудитории.

Таблица 1 – Объект «Comps»

Meaning	Designation	Example	Type
По уникальному идентификатору имеем возможность найти объект.	id_comps	1	INT, NOT NULL, PRIMARY KEY
У каждой аудитории имеется свой номер.	id_audit	100	INT, NOT NULL
Качественные признаки системного блока.	comps_name	P4/100Gb/1024 Mb/LAN	VARCHAR(50), NOT NULL
Отличительные признаки.	comps_desc	Notebook	VARCHAR(50), NULL
Дополнительная информация.	comps_other		VARCHAR(255) NULL

Таблица 2 – Объект «Kaf»

Meaning	Designation	Example	Type
По уникальному идентификатору имеем возможность найти объект.	id_kaf	1	INT, NOT NULL, PRIMARY KEY
У каждой кафедры имеется свое наименование.	kaf_name	МИВТ	VARCHAR(50), NOT NULL
ФИО зав кафедры.	kaf_dekan	Исикова Н.П.	VARCHAR(50), NOT NULL
Телефон кафедры.	kaf_tel	88612534406	VARCHAR(50), NULL

Таблица 3 - Объект «Otvetst»

Meaning	Designation	Example	Type
У каждой аудитории имеется свой номер.	id_audit	1	INT, NOT NULL, PRIMARY KEY

По уникальному идентификатору имеем возможность найти объект.	id_kaf	100	INT, NOT NULL
Наименование корпуса.	otvetst_korpus	Главный	VARCHAR(50), NULL
Номер аудитории, закрепленной за ответственным лицом.	otvetst_audit	A400	VARCHAR(50), NOT NULL
ФИО ответственного.	otvetst_fio	Петров Петр	VARCHAR(150) NOT NULL
Телефонный номер ответственного.	otvetst_tel	29938	VARCHAR(50) NULL
Email ответственного.	otvetst_email	1@1.ru	VARCHAR(50), NULL
Прочее.	otvetst_other	После 17 не звонить.	VARCHAR (255) NULL

Таблица 4 – Объект «Hardware»

Meaning	Designation	Example	Type
По уникальному идентификатору имеем возможность найти объект.	id_hardware	1	INT, NOT NULL, PRIMARY KEY
Уникальный идентификатор компьютера.	id_comps	100	INT, NOT NULL
Инвентаризационный номер	id_invent	NZ324	VARCHAR(25), NOT NULL
Отличительные признаки.	comps_desc	Notebook	VARCHAR(50), NULL
Дополнительная информация.	hardware_type	3	SMALLINT, NOT NULL
Наименование комплектующих.	hardware_name	BENQ 17"	VARCHAR(100) NOT NULL
Серийный номер.	hardware_name	Ahtgur6t875	VARCHAR(50), NULL
Описание	hardware_desc	Без Сидиромы.	VARCHAR(200) NULL

СИСТЕМА ФУНКЦИОНАЛЬНЫХ ЗАВИСИМОСТЕЙ

Функциональная зависимость (functional dependency) – такая логическая связь между атрибутами отношения, при которой по известному значению одного атрибута можно найти (или вычислить) значение другого атрибута.

Нормальные формы

Здесь изложены несколько правил, относящихся к нормализации. Все эти правила представляют собой частные случаи только что описанного процесса нормализации.

В 70-х годах XX века теоретики реляционных баз данных обнаруживали различные типы аномалий модификации, вызванные структурой отношений. Классы отношений, лишенные аномалий определенного типа, называются нормальными формами (normal forms). Известно семь нормальных форм: первая, вторая, третья, четвертая, пятая нормальные формы (1НФ, 2НФ, 3НФ, 4НФ, 5НФ), нормальная форма Бойса-Кодда (НФБК) и доменно-ключевая нормальная форма (ДКНФ). Нормальные формы являются вложенными друг в друга (рис.11). То есть отношение во второй нормальной форме является отношением в первой нормальной форме, а отношение в 5НФ одновременно находится в 4НФ, НФБК, 3НФ, 2НФ, 1НФ.

Первая нормальная форма (first normal form) – 1НФ

Таблица находится в 1НФ, если она удовлетворяет определению отношения. Таблица, находящаяся в 1НФ может быть подвержена аномалиям. Более старшие нормальные формы позволяют избежать определенных типов аномалий.

Вторая нормальная форма (second normal form) – 2НФ

Известно, что по значению первичного ключа можно однозначно определить значения остальных ячеек этой строки. Следовательно, все неключевые атрибуты функционально зависят от первичного ключа.

Определение: таблица находится во второй нормальной форме (2НФ), если она удовлетворяет определению 1НФ и все ее атрибуты, не входящие в первичный ключ, функционально зависят от первичного ключа и не зависят от части первичного ключа. Отсюда следует вывод: все таблицы с простым первичным ключом находятся во 2НФ.

Третья нормальная форма (third normal form) – 3НФ

Определение: таблица находится в третьей нормальной форме (3НФ), если она удовлетворяет определению 2НФ и не существует функциональных зависимостей между не ключевыми атрибутами.

Нормальная форма Бойса-Кодда (Boyce-Codd normal form, BC/NF) – НФБК

Определение: таблица находится в нормальной форме Бойса-Кодда (НФБК), если не ключевые атрибуты функционально зависят только от возможных ключей, и не зависят от частей этих потенциальных ключей.

Четвертая нормальная форма (fourth normal form) – 4НФ

В отношениях возможны другие виды аномалий, связанные с наличием многозначных зависимостей (multivalued dependency) между атрибутами. По определению, атрибут А многозначно определяет атрибут В той же таблицы,

если для каждого значения атрибута А существует хорошо определенное множество соответствующих значений В.

Пятая нормальная форма (fifth normal form) – 5НФ

Пятая нормальная форма затрагивает отношения, которые имеют несколько многозначных атрибутов, и эти атрибуты зависимы между собой.

Доменно-ключевая нормальная форма (domain/key normal form) – ДКНФ.

Определение: отношение находится в доменно-ключевой нормальной форме, если каждое ограничение целостности, накладываемое на это отношение, является логическим следствием определения доменов и ключей.

Доказано, что таблицы, находящиеся в ДКНФ, лишены каких бы то ни было аномалий модификации. К сожалению, общего подхода, позволяющего привести таблицу к ДКНФ, пока не существует.

При составлении модели "сущность-связь", а затем реляционной модели данных, следует планировать данные так, чтобы каждая таблица содержала ровно одну тему. Это поможет избежать аномалий в таблицах.

Далее каждую таблицу необходимо проверить на соответствие нормальным формам в следующем порядке (и при необходимости разбить на более мелкие таблицы):

1НФ -> 2НФ -> НФБК -> 4НФ -> 5НФ -> ДКНФ

Нормализация таблиц имеет свои плюсы и минусы. Существенным плюсом является то, что пропадают аномалии модификации, избыточность, хранение противоречивой информации. Минус нормализации проявляется в замедленной выборке данных. После нормализации количество таблиц возрастает; информация, которая раньше лежала в одной таблице, теперь разбросана по нескольким. Чтобы составить комплексный отчет, приходится просматривать несколько таблиц, что занимает больше времени, чем поиск в одной ненормализованной таблице. В некоторых базах данных это замедление поиска оказывается столь существенным, что выгоднее пренебречь нормализацией, зато выиграть в производительности. Тогда выполняют обратный процесс – денормализацию – намеренное соединение нормализованных таблиц в не нормализованные. Логика работы прикладных программ, обрабатывающих базу данных, дополняют процедурами дополнительной поддержки целостности и непротиворечивости ненормализованных данных.

Пример СИСТЕМЫ ФУНКЦИОНАЛЬНЫХ ЗАВИСИМОСТЕЙ:

- (1) Объяснить почему был выбран первичный ключ.
- (2) F1- Описание таблицы, первым выделяются первичные ключевые атрибуты таблицы, после знака -> идёт описание остальных атрибутов.
- (3) K1- Только первичные ключи.

«otvetst»:

(1) Не может быть двух ответственных за одну аудиторию.

(2) F1: id_audit -> id_kaf, otvetst_korpus, otvetst_audit, otvetst_fio, otvetst_tel, otvetst_email, otvetst_other

(3) K1 = id_audit.

«kaf»:

Не может быть двух кафедр с одинаковыми идентификаторами.

F2: id_kaf -> kaf_name, kaf_dekan, kaf_tel

K2 = id_kaf

«hardware»:

Не может существовать несколько единиц комплектующих с одинаковым уникальным идентификатором.

F3: id_hardware -> id_comps, id_invent, hardware_type, hardware_name, hardware_sn, hardware_desc

K3 = id_hardware

«comps»:

Не может существовать несколько компьютеров с одинаковыми уникальными идентификаторами.

F4: id_comps -> id_audit, comps_name, comps_description, comps_other

K4 = id_comps

Рассмотрим концептуальную модель (ER-диаграмма). С помощью программы PLATINUM ERwin ERX 3.5.2 составляем ER-диаграмму ("сущность-связь" Entity-Relationship) нашей базы данных.

1. Открываем PLATINUM ERwin ERX 3.5.2

2. Настраиваем программу -> Вкладка Server -> Выбираем Target Server -> устанавливаем SQL Server и версию 7.x.

3. Для того чтобы создать таблицу выбираем вкладку Windows -> нажимаем Erwin Toolbox или ctrl+t, появится инструментарий Erwin -> выбираем Independent Table и щёлкаем на рабочем столе программы, создаётся таблица и переходим к шагу заполнения значениями.

4. Даём название таблице Sportsmens -> далее начинаем заполнять таблицу значениями -> щёлкнув на ней два раза появится зона в которой можно редактировать атрибуты нашей таблицы -> в вкладке General выбираем тип атрибута (Default, Blob, Datetime, Number, String) переходим к вкладке SQL Server, выбираем нужный нам тип атрибута и задаём его длину.

5. Далее начинаем расставлять связи в нашей диаграмме -> в инструментарии выбираем соответствующую связь и связываем таблицы.

Текст SQL-запросов на создание объектов базы данных.

```

CREATE DATABASE inv
USE inv
CREATE TABLE [dbo].[kaf](
[id_kaf] [int] IDENTITY(1,1) NOT NULL,
[kaf_name] [varchar](50) COLLATE Cyrillic_General_CI_AS
NOT NULL,
[kaf_dekan] [varchar](50) COLLATE Cyrillic_General_CI_AS
NULL,
[kaf_tel] [varchar](50) COLLATE Cyrillic_General_CI_AS
NULL,
CONSTRAINT [PK_kaf] PRIMARY KEY CLUSTERED )
GO
CREATE TABLE [dbo].[otvetst](
[id_audit] [int] IDENTITY(1,1) NOT NULL,
[id_kaf] [int] NOT NULL,
[otvetst_korpus] [varchar](50) COLLATE
Cyrillic_General_CI_AS NULL,
[otvetst_audit] [varchar](50) COLLATE Cyrillic_General_CI_AS
NOT NULL,
[otvetst_fio] [varchar](150) COLLATE Cyrillic_General_CI_AS
NOT NULL,
[otvetst_tel] [varchar](50) COLLATE Cyrillic_General_CI_AS
NULL,
[otvetst_email] [varchar](50) COLLATE Cyrillic_General_CI_AS
NULL,
[otvetst_other] [varchar](255) COLLATE
Cyrillic_General_CI_AS NULL,
CONSTRAINT [PK_otvetst] PRIMARY KEY CLUSTERED )
GO
CREATE TABLE [dbo].[hardware](
[id_hardware] [int] IDENTITY(1,1) NOT NULL,
[id_comps] [int] NOT NULL,
[id_invent] [varchar](25) COLLATE Cyrillic_General_CI_AS

```

```

NOT NULL,
[hardware_type] [smallint] NOT NULL,
[hardware_name] [varchar](100) COLLATE
Cyrillic_General_CI_AS NOT NULL,
[hardware_sn] [varchar](50) COLLATE Cyrillic_General_CI_AS
NULL,
[hardware_desc] [varchar](200) COLLATE
Cyrillic_General_CI_AS NULL)
GO
CREATE TABLE [dbo]. [comps](
[id_comps] [int] IDENTITY (1,1) NOT NULL,
[id_audit] [int] NOT NULL,
[comps_name] [varchar](50) COLLATE Cyrillic_General_CI_AS
NOT NULL,
[comps_description] [varchar](50) COLLATE
Cyrillic_General_CI_AS NULL,
[comps_other] [varchar](255) COLLATE
Cyrillic_General_CI_AS NULL,
CONSTRAINT [PK_comps] PRIMARY KEY CLUSTERED )

```

Откройте вашу Erwin-диаграмму. В появившемся окне после выбора пункта главного меню File -> New задать раздел Blank Diagram.

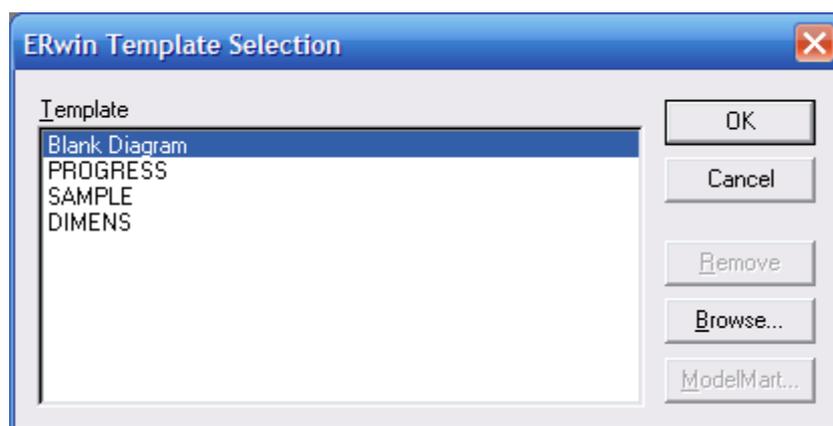


Рисунок 1 - Окно создания новой диаграммы

Для отображения символов русского языка, в появившемся диалоговом окне задайте по умолчанию шрифт, поддерживающий символы кириллицы (@Arial Unicode MS) как показано на рисунке 2. Применить данный шрифт для всех объектов (Apply settings to All objects).

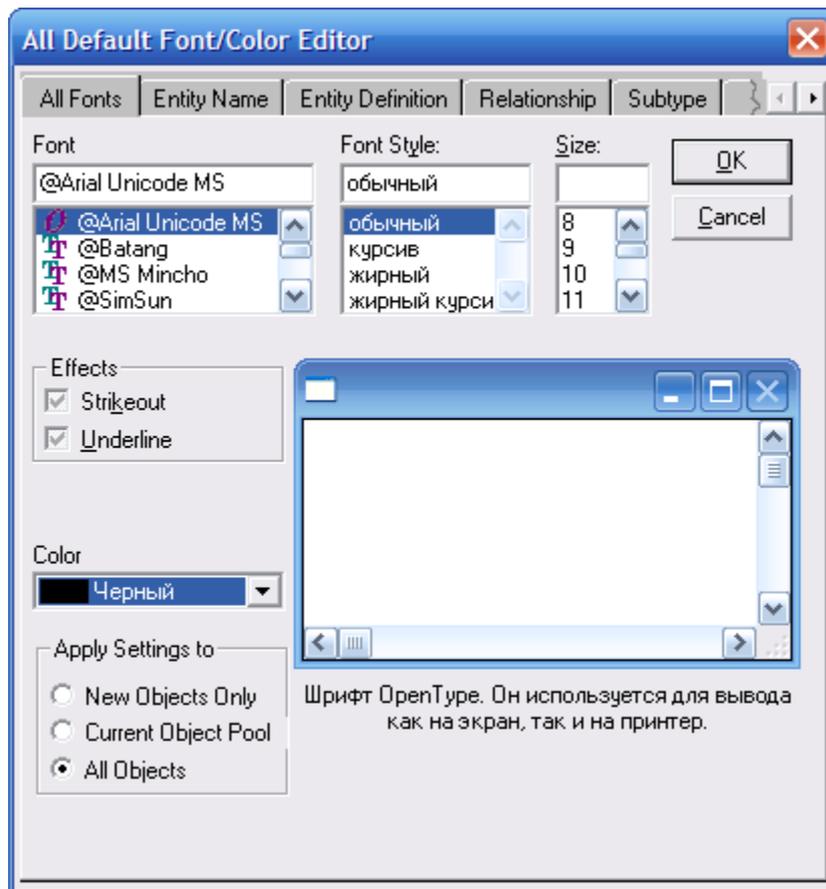


Рисунок 2 - Окно настройки шрифта

Задать отображение физической модели и выбрать сервер-назначение MS SQL Server 7

В итоге получаем примерно такую физическую модель базы данных, рисунок 3.

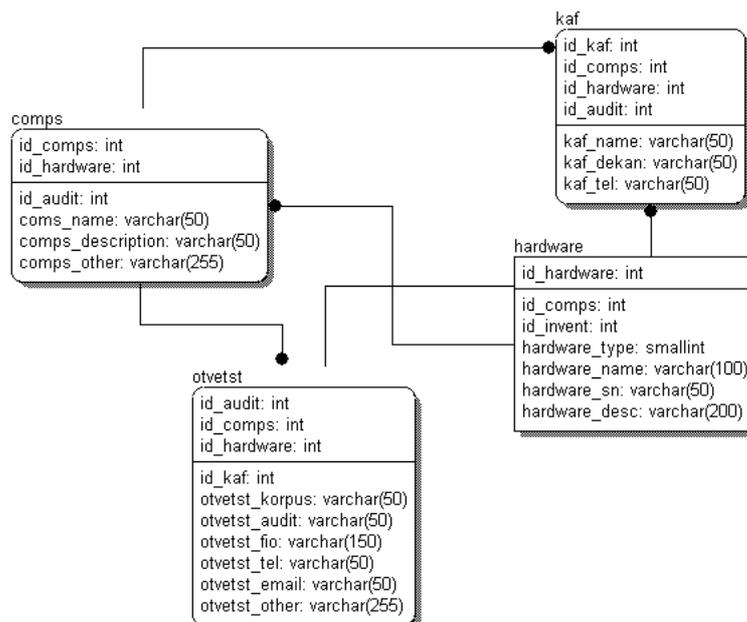


Рисунок 3 - Физическая модель БД

Выбираем меню типа сервера (SQL Server 7x), рисунок 4.

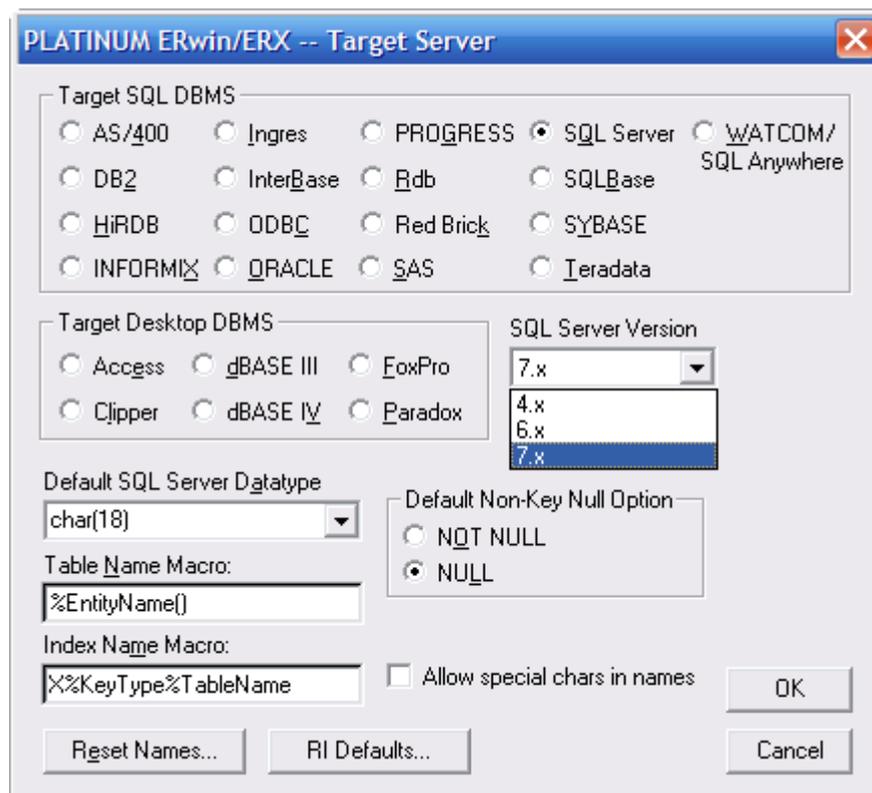


Рисунок 4 - Окно выбора типа SQL Server

Задаем сущности. Для этого на панели инструментов выбираем второй инструмент.

После двойного нажатия на добавленной сущности появится окно, показанное на рисунке 5. Для добавления атрибута нажмите кнопку New.

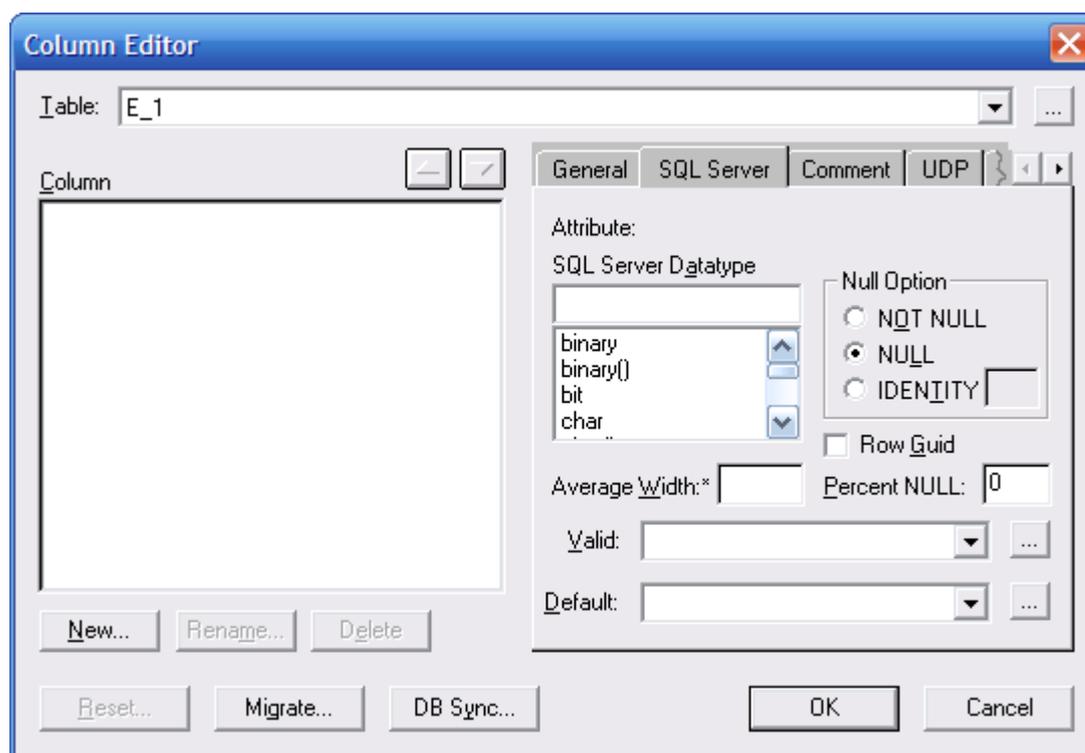


Рисунок 5 - Окно добавления атрибута



Рисунок 6 - Окно свойств атрибута

В появившемся окне необходимо задать тип атрибута, его логическое и физическое имена. Логическое имя может содержать произвольные символы. Физическое представление должно начинаться с латинского символа и содержать латинские буквы, цифры и знаки подчеркивания. Желательно не использовать длинные физическое имена (более 12 символов), рисунок 6. В результате добавления атрибутов логическое и физическое представления сущности:



В верхней части таблицы располагаются ключевые поля. Для того, чтобы атрибут стал первичным ключом, необходимо в редакторе атрибутов (Attribute Editor) на закладке General установить свойство Primary Key.

Следующим этапом является переопределение типов (в случае если типы не указаны точно при изначальном задании атрибутов) в соответствии с набором типов сервера-назначения. Также здесь указываются значения по умолчанию, множество допустимых значений атрибута и признак обязательного заполнения, рисунок 7.

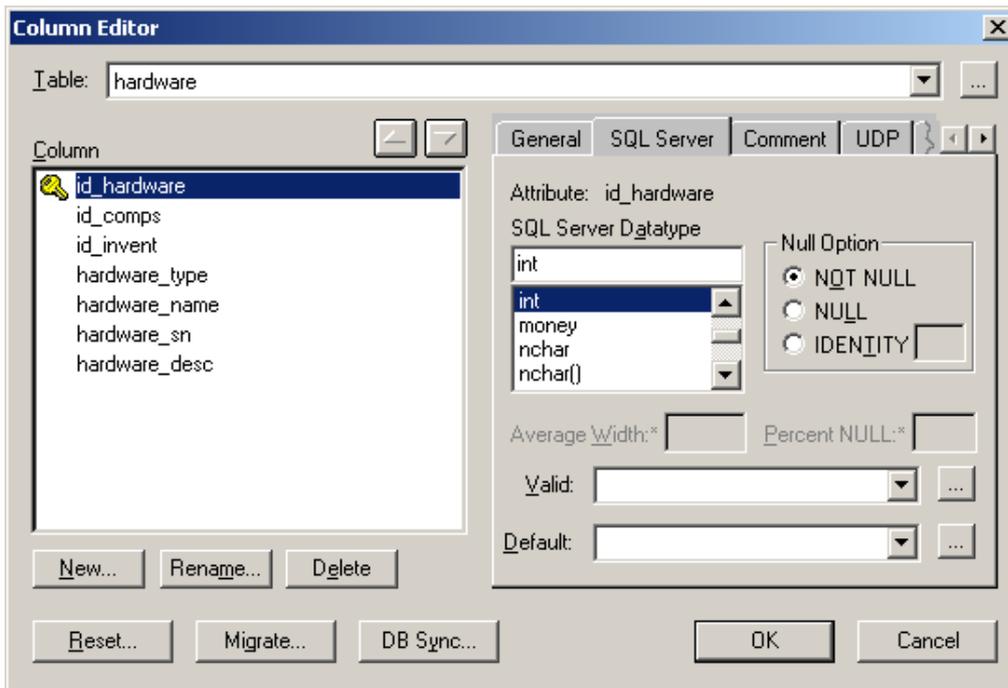


Рисунок 7 - Окно переопределения типов

Valid – Правило, определяющее множество значений атрибута (используется в случае задания домена)

Default – Значение по умолчанию.

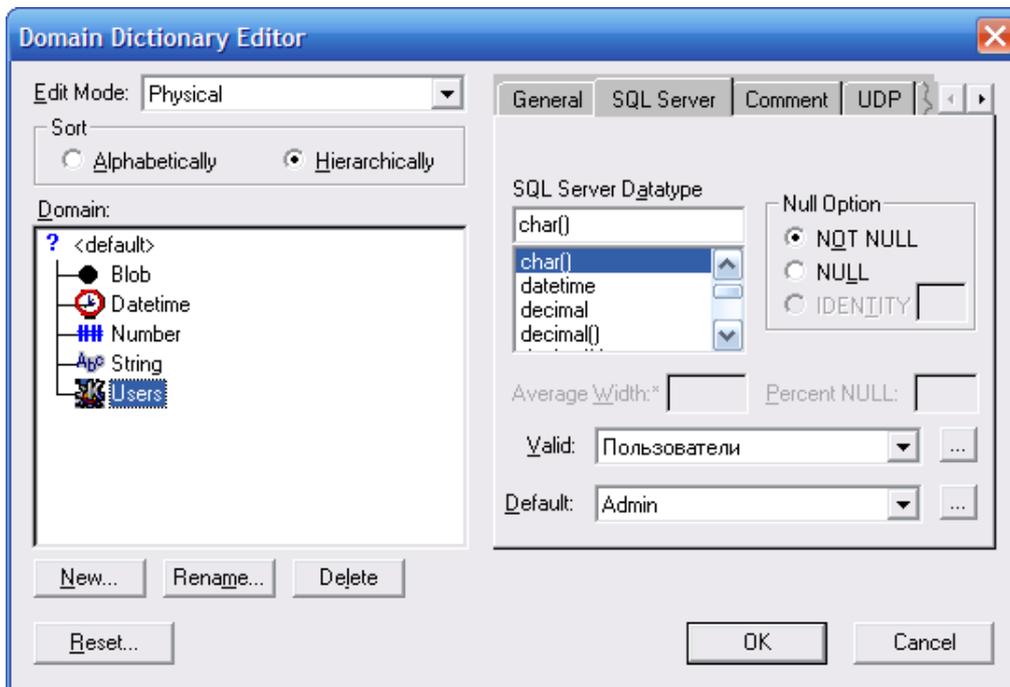


Рисунок 8 - Выбор типа данных

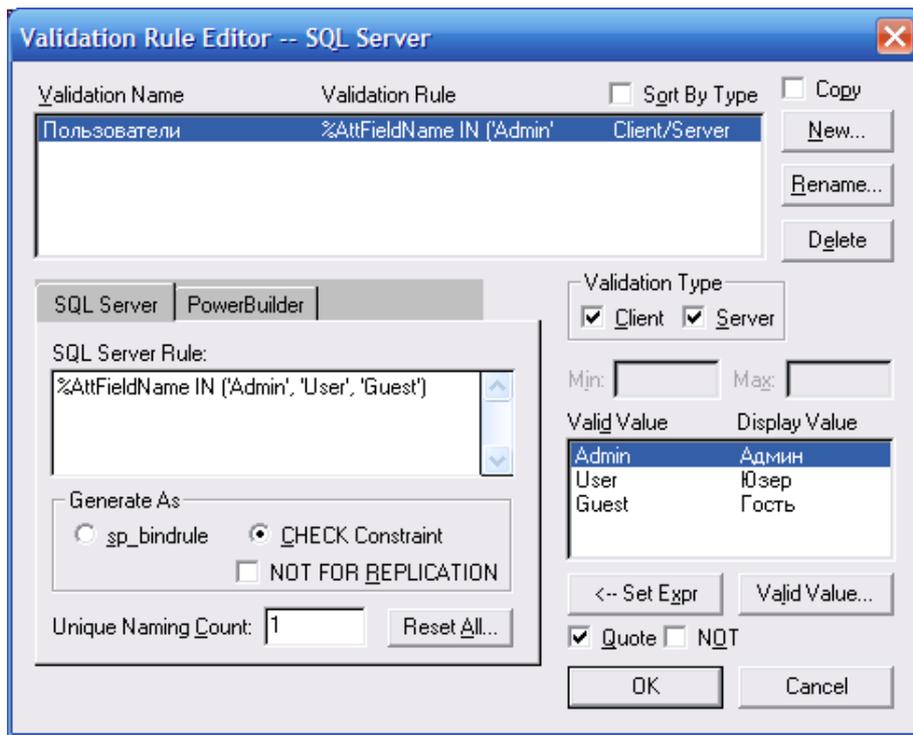


Рисунок 9 - Назначение пользователей

После того, как все сущности заданы, установите связи между ними.

Для представления физической модели на языке SQL воспользуйтесь функцией Forward Engineer.

В появившемся окне выставите нужные опции (вид, таблицы, индексные поля, ограничения целостности данных, триггеры и др.) и проверьте схему на закладке Summary, рисунок 10.

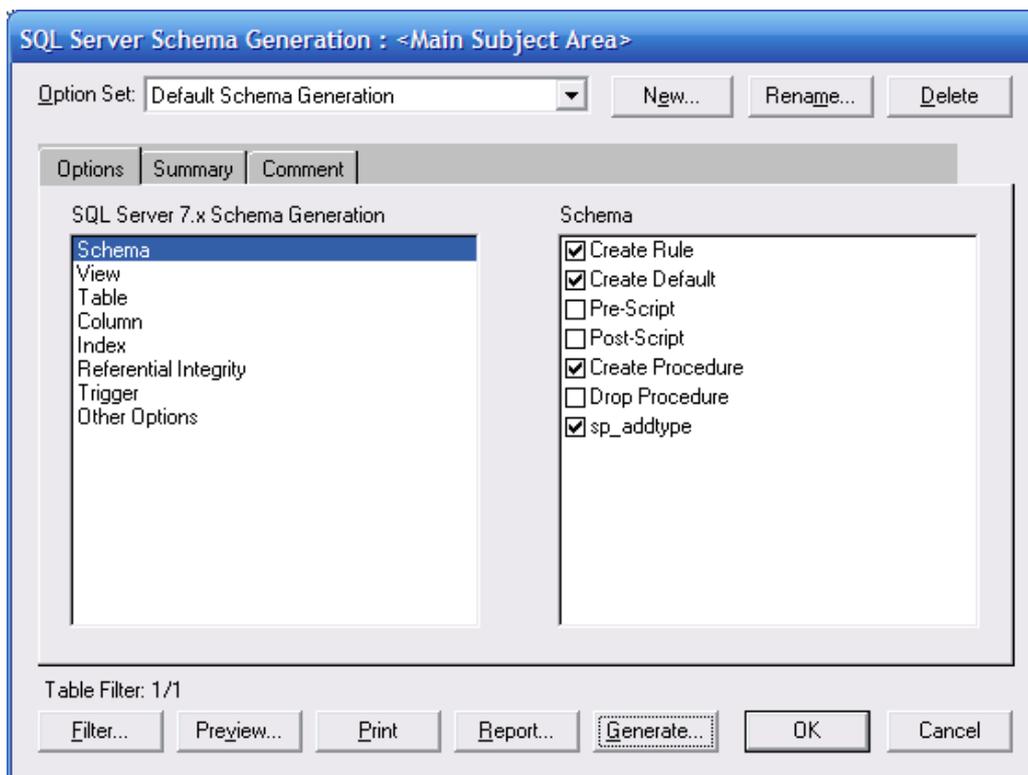


Рисунок 10 - Окно настройки схемы

Нажмите кнопку Preview и скопируйте сформированный SQL-скрипт:

```
CREATE TABLE kaf(  
id_kaf int IDENTITY(1,1) NOT NULL,  
kaf_name varchar(50) NOT NULL,  
kaf_dekan varchar(50) NULL,  
kaf_tel varchar(50) NULL,  
CONSTRAINT PK_kaf PRIMARY KEY CLUSTERED)  
GO  
CREATE TABLE otvetst(  
id_audit int IDENTITY(1,1) NOT NULL,  
id_kaf int NOT NULL,  
otvetst_korpus varchar(50) NULL,  
otvetst_audit varchar(50) NOT NULL,  
otvetst_fio varchar(150) NOT NULL,  
otvetst_tel varchar(50) NULL,  
otvetst_email varchar(50) NULL,  
otvetst_other varchar(255) NULL,  
CONSTRAINT PK_otvetst PRIMARY KEY CLUSTERED)  
GO  
CREATE TABLE hardware(  
id_hardware int IDENTITY(1,1) NOT NULL,  
id_comps int NOT NULL,  
id_invent varchar(25) NOT NULL,  
hardware_type smallint NOT NULL,  
hardware_name varchar(100) NOT NULL,  
hardware_sn varchar(50) NULL,  
hardware_desc varchar(200) NULL)  
GO  
CREATE TABLE comps(  
id_comps int IDENTITY(1,1) NOT NULL,  
id_audit int NOT NULL,  
comps_name varchar(50) NOT NULL,  
comps_description varchar(50) NULL,
```

comps_other varchar(255) NULL,

CONSTRAINT PK_comps PRIMARY KEY CLUSTERED)

Запустите приложение SQL Server Management Studio Express. В меню дерева сервера выберите пункт “New Database”, рисунок 11.

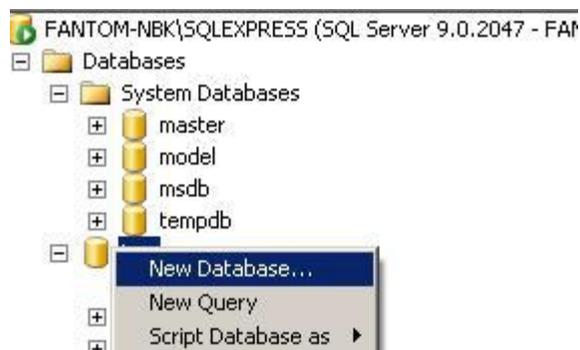


Рисунок 11 - Обзор объектов sql

В появившемся окне введите имя новой БД.

После сообщения об успешном создании БД выполните SQL-скрипт, полученный в результате выгрузки модели из среды ErWin.

Примечание: Первой командой, предшествующей полученному запросу необходимо использовать USE (имя базы данных).

После успешного выполнения запроса (при возникновении ошибок и неточностей внести необходимые коррективы) переходим к созданию диаграммы БД в SQL Server Enterprise Manager.

Связи должны отобразиться автоматически (в противном случае допущена ошибка на более раннем этапе – вернитесь на более ранний этап выполнения работы!).

Переходим к созданию диаграммы БД, для этого в SQL Server Management Studio Express выбираем базу данных `inv` -> Database Diagram, рисунок 12.



Рисунок 12 - Создание диаграммы БД

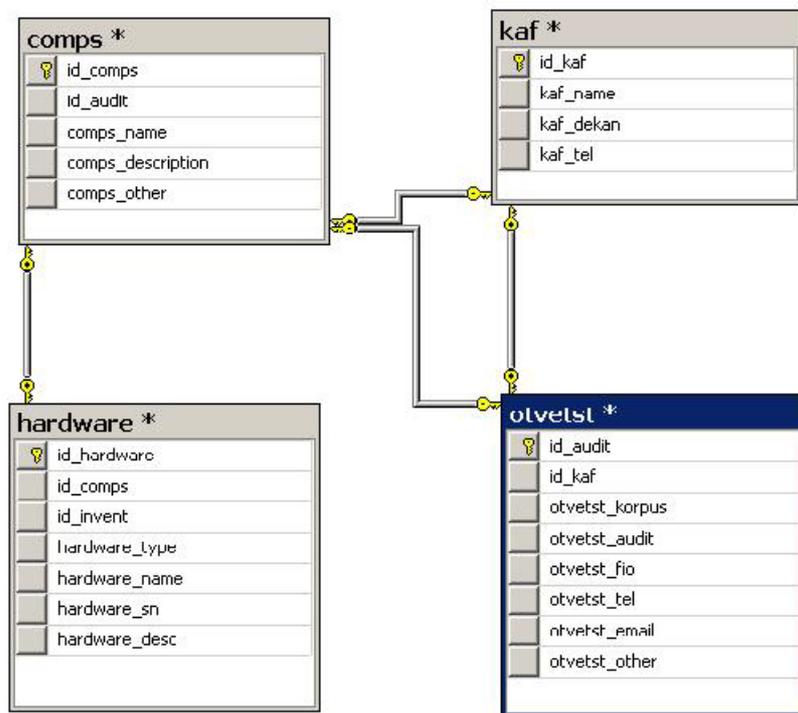


Рисунок 13 - Диаграмма полученной БД

Шифрование - процесс кодирования конфиденциальных данных с использованием ключа или пароля. Шифрование надежно защищает данные и сокращает вероятность несанкционированного раскрытия конфиденциальной информации, так как без соответствующего ключа или пароля данные бесполезны. SQL Server располагает многими режимами шифрования, в том числе на уровне ячеек, базы данных, файлов через Windows и шифрования на транспортном уровне.

Шифрование SQL Server не решает проблему доступности инфраструктуры и баз данных SQL Server, но повышает защищенность данных на уровнях базы данных и операционной системы, даже если нарушена конфиденциальность инфраструктуры или баз данных SQL Server.

Несмотря на то, что шифрование является полезным средством обеспечения безопасности, его не следует применять ко всем данным или соединениям. При решении о внедрении шифрования необходимо проанализировать, как пользователи получают доступ к данным.

Если пользователи получают доступ к данным через открытую сеть, то шифрование может потребоваться для повышения безопасности. Однако если весь доступ осуществляется по безопасной внутренней сети, то шифрование не требуется. Использование шифрования включает политику управления паролями, ключами и сертификатами.

Модель шифрования SQL Server в основном предоставляет функции управления ключами шифрования, соответствующие стандарту ANSI X9.17. В этом стандарте определены несколько уровней ключей шифрования,

использующихся для шифрования других ключей, которые в свою очередь применяются для шифрования собственно данных.

В таблице перечислены уровни ключей шифрования SQL Server и ANSI X9.17.

Таблица	Уровень шифрования ключей SQL Server и ANSI X9.17	
Уровень SQL Server	Уровень ANSI X9.17	Описание
SMK	Главный ключ	SMK – ключ верхнего уровня, используемый для шифрования DMK. SMK шифруется с применением DPAPI.
DMK	Ключ шифрования ключей	DMK – симметричный ключ, используемый для шифрования симметричного ключа, асимметричного ключа и сертификата. Для каждой базы данных может быть определен только один DMK.
Симметричные ключи, асимметричные ключи и сертификаты	Ключ данных	Симметричные ключи, асимметричные ключи и сертификаты используются для шифрования данных.

Главный ключ службы Service master key (SMK) - ключ верхнего уровня и предок всех ключей в SQL Server. SMK - асимметричный ключ, шифруемый с использованием Windows Data Protection API (DPAPI). SMK автоматически создается, когда шифруется какой-нибудь объект, и привязан к учетной записи службы SQL Server. SMK используется для шифрования главного ключа базы данных Database master key (DMK).

Второй уровень иерархии ключей шифрования - DMK. С его помощью шифруются симметричные ключи, асимметричные ключи и сертификаты. Каждая база данных располагает лишь одним DMK.

Следующий уровень содержит симметричные ключи, асимметричные ключи и сертификаты. Симметричные ключи - основное средство шифрования в базе данных. Microsoft рекомендует шифровать данные только с помощью симметричных ключей. Кроме того, в SQL Server 2008 и более новых версиях есть сертификаты уровня сервера и ключи шифрования базы данных для прозрачного шифрования данных.

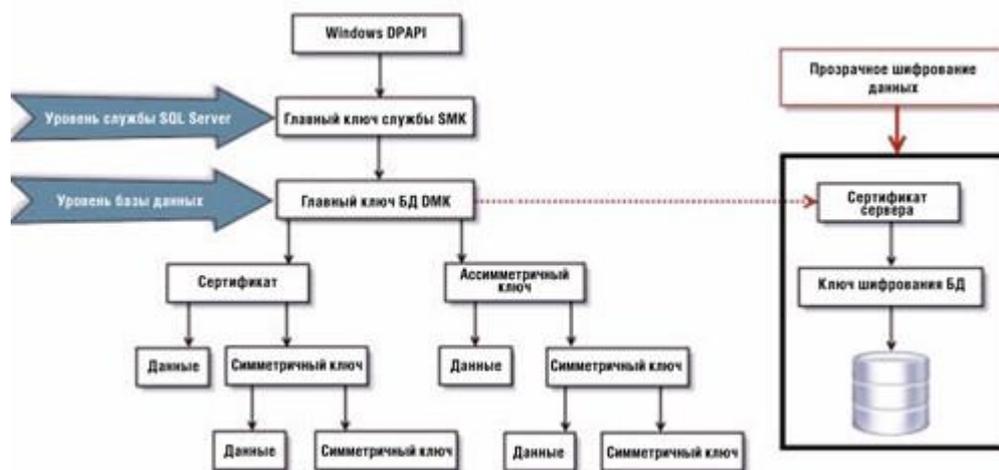


Рисунок 14 - Иерархия ключей шифрования для SQL Server 2008

Следует учитывать следующие основные понятия.

- Для лучшей производительности данные следует шифровать с помощью симметричных ключей, а не с помощью сертификатов и асимметричных ключей.
- Главные ключи базы данных защищены главным ключом службы. Главный ключ службы создается при установке SQL Server и шифруется API-интерфейсом защиты данных Windows (DPAPI).
- Возможны другие иерархии шифрования с дополнительными уровнями.
- Модуль расширенного управления ключами хранит симметричные или асимметричные ключи вне SQL Server.
- Прозрачное шифрование данных (TDE) должно использовать симметричный ключ, который называется ключом шифрования базы данных, защищенный сертификатом, который, в свою очередь защищается главным ключом базы данных master или асимметричным ключом, хранящимся в модуле расширенного управления ключами.
- Главный ключ службы и все главные ключи базы данных являются симметричными ключами.

SQL Server поддерживает следующие механизмы шифрования:

- функции Transact-SQL;
- асимметричные ключи;
- симметричные ключи;
- сертификаты
- прозрачное шифрование данных

Функции Transact-SQL - отдельные элементы можно шифровать по мере того, как они вставляются или обновляются, с помощью функций Transact-SQL.

Сертификат открытого ключа, или просто сертификат, представляет собой подписанную цифровой подписью инструкцию, которая связывает значение открытого ключа с идентификатором пользователя, устройства или службы, имеющей соответствующий закрытый ключ. Сертификаты поставляются и подписываются центром сертификации (certification authority, CA). Сущность, получающая сертификат от центра сертификации, является субъектом этого сертификата. Как правило, сертификаты содержат следующие сведения.

- Открытый ключ субъекта.
- Идентификационные данные субъекта, например имя и адрес электронной почты.
- Срок действия, то есть интервал времени, на протяжении которого сертификат будет считаться действительным.

Сертификат действителен только в течение указанного в нем периода, который задается в каждом сертификате при помощи дат (Valid From и Valid To), определяющих начало и окончание срока действия. По истечении срока действия сертификата его субъект должен запросить новый сертификат.

- Идентификационные данные поставщика сертификата.
- Цифровая подпись поставщика.

Эта подпись подтверждает действительность связи между открытым ключом и идентификационными данными субъекта. (В процессе создания цифровой подписи данные, вместе с некоторыми секретными данными отправителя, преобразуются в тег, называемый подписью.)

Главное преимущество сертификатов в том, что они позволяют не хранить на узлах совокупность паролей отдельных субъектов. Вместо этого узел просто устанавливает доверительные отношения с поставщиком сертификата; после этого поставщик может подписать неограниченное количество сертификатов.

Когда узел (например, защищенный веб-сервер) указывает, что конкретный поставщик сертификата является доверенным корневым центром сертификации, он неявно выражает доверие к политикам, которые поставщик использовал при определении связей для изданных им сертификатов. Таким образом, узел выражает уверенность в том, что поставщик проверил идентификационные данные субъекта сертификата. Узел делает поставщика доверенным корневым центром сертификации, помещая самостоятельно подписанный сертификат поставщика, содержащий открытый ключ поставщика, в хранилище сертификатов доверенного корневого центра сертификации на главном компьютере. Промежуточные или подчиненные

центры сертификации являются доверенными только в том случае, если к ним ведет правильный путь от доверенного корневого центра сертификации.

Поставщик может отозвать сертификат до истечения срока его действия. При этом отменяется связь открытого ключа с идентификационными данными, указанными в сертификате. Каждый поставщик ведет список отозванных сертификатов, который можно использовать для проверки конкретного сертификата.

Самостоятельно подписанные сертификаты, созданные SQL Server, соответствуют стандарту X.509 и поддерживают поля X.509 v1.

Асимметричный ключ состоит из закрытого ключа и соответствующего открытого ключа. Каждый из этих ключей позволяет дешифровать данные, зашифрованные другим ключом. На выполнение асимметричных операций шифрования и дешифрования требуется сравнительно много ресурсов, но они обеспечивают более надежную защиту, чем симметричное шифрование. Асимметричный ключ можно использовать для шифрования симметричного ключа перед его сохранением в базе данных.

Симметричный ключ – это ключ, используемый и для шифрования, и для дешифрования данных. Данные при использовании симметричного ключа шифруются и дешифруются быстро, и он вполне подходит для повседневной защиты конфиденциальных данных, хранящихся в базе данных.

Прозрачное шифрование данных (TDE) является особым случаем шифрования с использованием симметричного ключа. TDE шифрует всю базу данных, используя симметричный ключ, который называется ключом шифрования базы данных. Функция прозрачного шифрования данных (TDE) выполняет в реальном времени шифрование и дешифрование файлов данных и журналов в операциях ввода-вывода. При шифровании используется ключ шифрования базы данных (DEK), который хранится в загрузочной записи базы данных для доступности при восстановлении.

Ключ шифрования базы данных является симметричным ключом, защищенным сертификатом, который хранится в базе данных master на сервере, или асимметричным ключом, защищенным модулем расширенного управления ключами. Функция прозрачного шифрования данных защищает «неактивные» данные, то есть файлы данных и журналов.

Основное правило прозрачного шифрования, это то, что шифрование файла базы данных проводится на уровне страниц. Страницы в зашифрованной базе данных шифруются до записи на диск и дешифруются при чтении в память. Прозрачное шифрование данных не увеличивает размер зашифрованной базы данных. При этом не стоит забывать, что прозрачное шифрование не шифрует данные при передаче через каналы связи, для этого к примеру, можно включить шифрование соединения. Так же надо учитывать, что шифрование\дешифрование данных это нагрузка на CPU, следовательно, при проектировании необходимо учесть запас мощностей процессоров.

При включении функции прозрачного шифрования данных необходимо немедленно создать резервную копию сертификата и закрытого ключа, связанного с этим сертификатом. В случае если сертификат окажется недоступен или понадобится восстановить базу данных на другом сервере либо присоединить ее к другому серверу, необходимо иметь копии сертификата и закрытого ключа. В противном случае будет невозможно открыть базу данных. Сертификат шифрования следует сохранить, даже если функция прозрачного шифрования в базе данных будет отключена. Даже если база данных не зашифрована, части журнала транзакций могут по-прежнему оставаться защищенными, а для некоторых операций будет требоваться сертификат до выполнения полного резервного копирования базы данных. Сертификат, который превысил свой срок хранения, все еще может быть использован для шифрования и расшифровки данных с помощью прозрачного шифрования данных.

Рассмотрим шифрование с помощью ассиметричного ключа. Основные функции для управления ассиметричными ключами:

- CREATE ASYMMETRIC KEY.

Создает ассиметричный ключ в базе данных. Ассиметричный ключ является защищаемой сущностью на уровне базы данных.

В его форме по умолчанию эта сущность содержит как открытый, так и закрытый ключ. CREATE ASYMMETRIC KEY при выполнении без предложения FROM формирует новую пару ключей. CREATE ASYMMETRIC KEY при выполнении с предложением FROM импортирует пару ключей из файла или открытый ключ из сборки.

По умолчанию закрытый ключ защищается с помощью главного ключа базы данных. Для защиты закрытого ключа необходим пароль, если не был создан главный ключ базы данных. Если главный ключ базы данных существует, пароль необязателен. Закрытый ключ может быть длиной 512, 1024 или 2048 бит.

- ALTER ASYMMETRIC KEY.

Изменяет свойства ассиметричного ключа:

- Изменение пароля закрытого ключа. В следующем примере изменяется пароль, используемый для защиты закрытого ключа ассиметричного ключа PacificSales09. Новым паролем будет <enterStrongPasswordHere>.

```
ALTER ASYMMETRIC KEY PacificSales09
WITH PRIVATE KEY (
DECRYPTION BY PASSWORD = '<oldPassword>',
ENCRYPTION BY PASSWORD = '<enterStrongPasswordHere>');
GO
```

- Удаление закрытого ключа из асимметричного ключа. В следующем примере закрытый ключ удаляется из асимметричного ключа PacificSales19, в котором остается только открытый ключ.

```
ALTER ASYMMETRIC KEY PacificSales19  
REMOVE PRIVATE KEY;  
GO
```

- Снятие парольной защиты с закрытого ключа. В следующем примере снимается парольная защита закрытого ключа и устанавливается его защита главным ключом базы данных.

```
OPEN MASTER KEY;  
ALTER ASYMMETRIC KEY PacificSales09  
WITH PRIVATE KEY (  
DECRYPTION BY PASSWORD = '<enterStrongPasswordHere>');  
GO
```

- DROP ASYMMETRIC KEY.

Удаляет асимметричный ключ из текущей базы данных. Нельзя удалять асимметричный ключ, которым в базе данных зашифрован симметричный ключ или с которым сопоставлено имя входа.

Прежде чем удалять такой ключ, следует сначала удалить пользователя или имя входа, с которым этот ключ сопоставлен, либо удалить или перешифровать симметричный ключ, который зашифрован данным асимметричным ключом. Для удаления шифрования, выполненного асимметричным ключом, предназначен параметр DROP ENCRYPTION инструкции ALTER SYMMETRIC KEY.

Метаданные асимметричных ключей доступны через представление каталога sys.asymmetric_keys. Сами ключи для просмотра из базы данных недоступны.

Если асимметричный ключ сопоставлен ключу поставщика расширенного управления ключами на устройстве поставщика, а параметр REMOVE PROVIDER KEY не указан, то ключ будет удален из базы данных, но не с устройства. Будет выдано предупреждение.

- ENCRYPTBYASYMKEY.

Шифрует данные при помощи асимметричного ключа. Шифрование/дешифрование с помощью асимметричного ключа является очень дорогостоящим по сравнению с шифрованием/дешифрованием с помощью симметричного ключа. Рекомендуется не шифровать асимметричным ключом большие наборы данных, например таблицы с пользовательскими данными.

Вместо этого данные следует шифровать с помощью сильного симметричного ключа, а симметричный ключ шифровать с помощью асимметричного.

EncryptByAsymKey возвращает значение NULL, если ввод превышает определенное число байтов, в зависимости от алгоритма. Ограничения: ключ RSA на 512 бит может шифровать до 53 байт, ключ на 1024 бита может шифровать до 117 байт, ключ на 2048 бит может шифровать до 245 байт. (Обратите внимание, что в SQL Server и сертификаты, и асимметричные ключи служат оболочками для ключей RSA.)

- DECRYPTBYASYMKEY.

Дешифрует данные асимметричным ключом. Шифрование и дешифрование асимметричным ключом являются очень дорогостоящими по сравнению с шифрованием и дешифрованием симметричным ключом. Рекомендуется использовать асимметричный ключ при работе с большими наборами данных, например, с данными пользователей в таблицах.

Переходим непосредственно к шифрованию нашей базы данных. Рассмотрим пример шифрования с помощью асимметричного ключа:

1. Войдём в нашу базу данных (“Библиотека”) и создадим для неё новый запрос. Чтобы создать асимметричный ключ, напишем следующий код:

```
CREATE ASYMMETRIC KEY SampleAsymKey  
WITH ALGORITHM = RSA_2048  
ENCRYPTION BY PASSWORD = '<Password>'
```

Если всё введено верно вы увидите сообщение об успешном выполнении команд. Во вкладке Безопасность => Асимметричные ключи нашей базы данных появится созданный нами асимметричный ключ, рисунок 15.

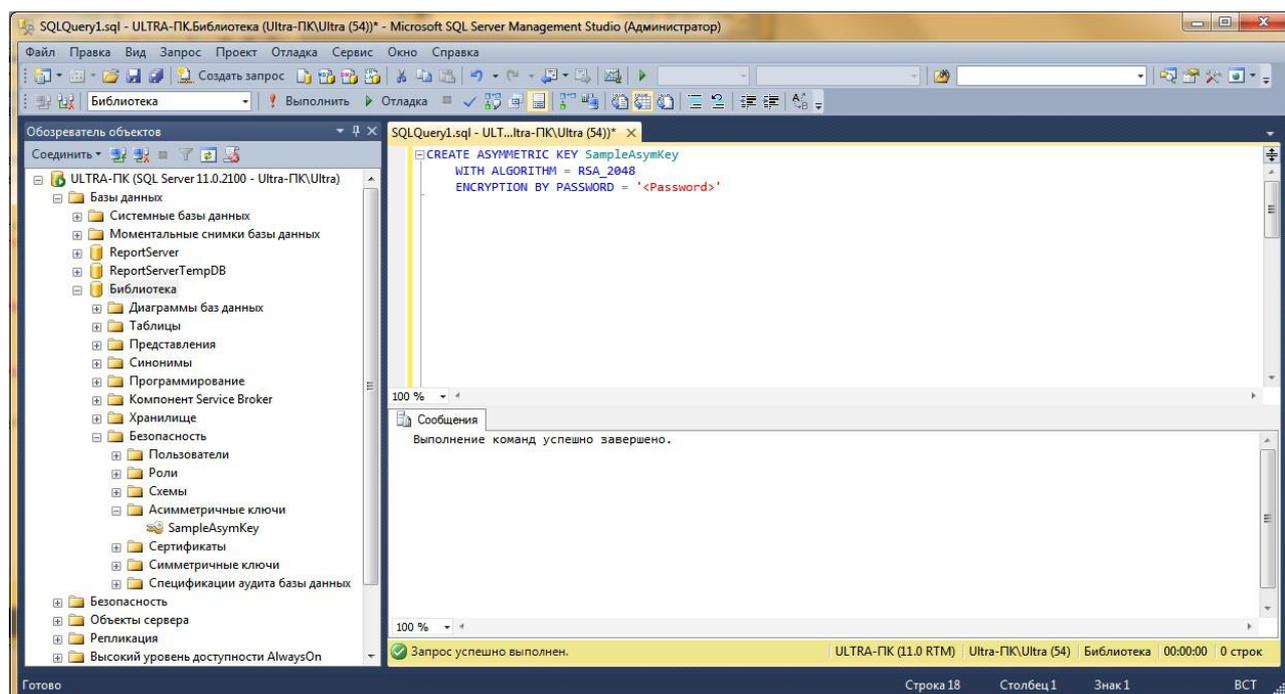


Рисунок 15 - Создание асимметричного ключа

2. Следующий шаг – объявление переменных. Объявим две переменные (текстовая и бинарная). В текстовой записываем некий текст и выведем его на экран. Зашифруем его с помощью функции ENCRYPTBYASYMKEY и расшифруем с помощью функции DECRYPTBYASYMKEY, рисунок 16:

```
declare @plaintext nvarchar(60)
declare @ciphertext varbinary(256)
--Инициализировать открытый текст
set @plaintext='Это простой текст'
print @plaintext
--Шифруем
set @ciphertext=ENCRYPTBYASYMKEY
(AsymKey_ID('SampleAsymKey'),@plaintext)
print @ciphertext
--Расшифровка
set @ciphertext=DECRYPTBYASYMKEY
(AsymKey_ID('SampleAsymKey'),@ciphertext,N'<Password>')
print cast (@plaintext as nvarchar(max))
```

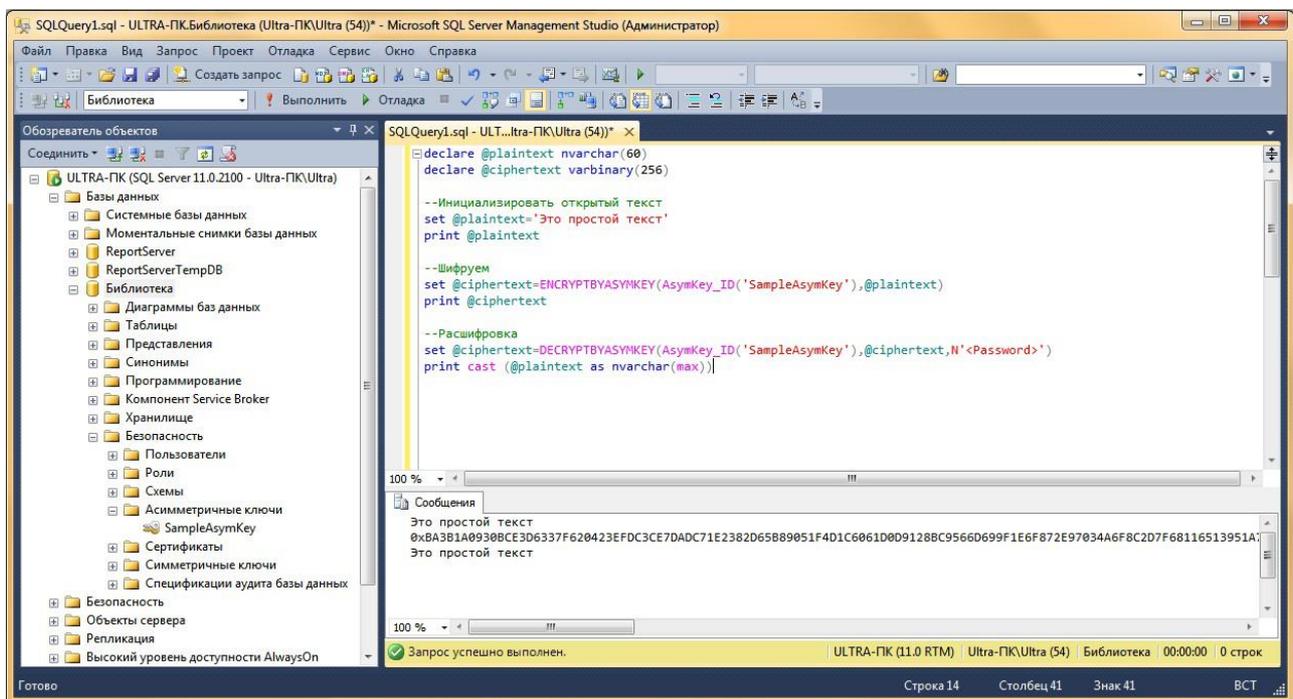


Рисунок 16 - Шифрование и дешифрование текста

3. Чтобы удалить асимметричный ключ введите следующий запрос, рисунок 17):

```
DROP ASYMMETRIC KEY SampleAsymKey
```

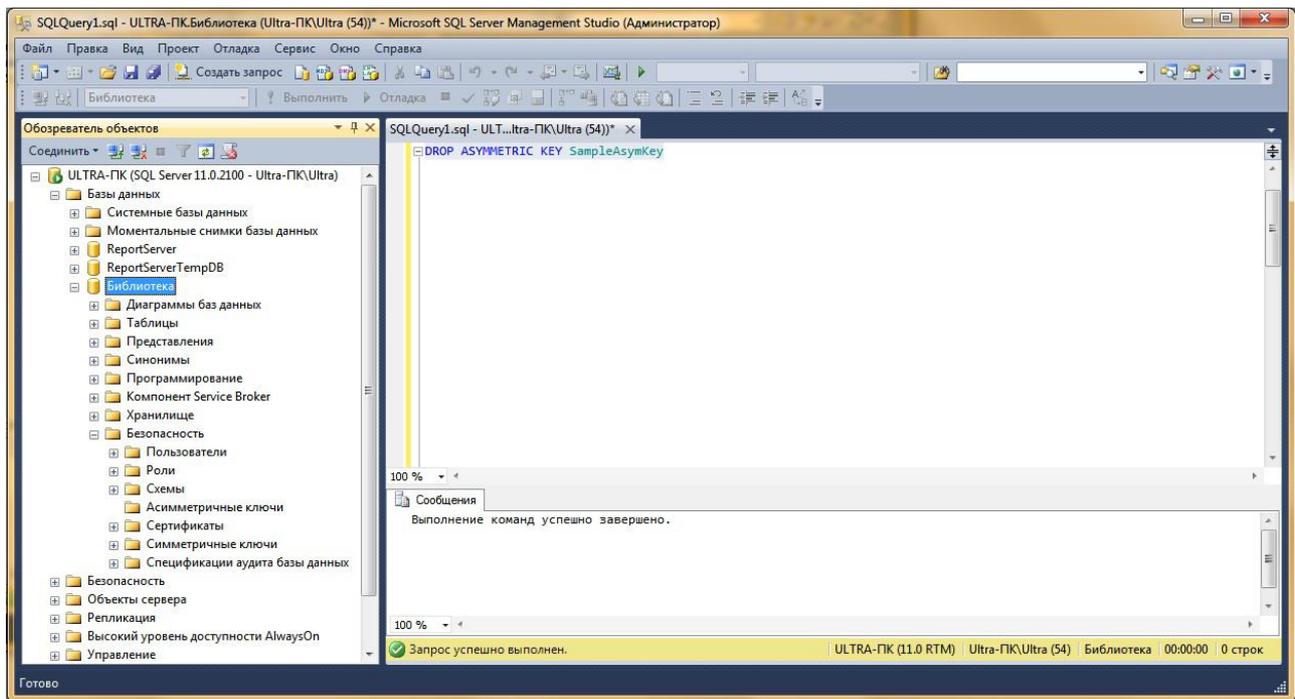


Рисунок 17 - Удаление асимметричного ключа

Рассмотрим шифрование с помощью сертификата. Функции для управления сертификатами:

1. CREATE CERTIFICATE - добавляет сертификат в базу данных SQL Server. Сертификат — это защищаемый объект уровня базы данных, соответствующий стандарту X.509 и поддерживающий поля X.509 V1. Инструкция CREATE CERTIFICATE может загрузить сертификат из файла или сборки. Она также может создать пару ключей и самостоятельно подписанный сертификат.

Закрытые ключи, создаваемые SQL Server имеют в длину 1024 бит. Закрытые ключи, импортированные из внешнего источника, имеют минимальную длину в 384 бит и максимальную длину в 4096 бит. Длина импортируемого закрытого ключа должна быть кратной 64 бит. Закрытый ключ должен соответствовать открытому ключу, указанному в аргументе certificate_name.

При создании сертификата из контейнера загрузка закрытого ключа необязательна. Однако в случае, когда SQL Server создает самостоятельно подписанный сертификат, закрытый ключ создается всегда. По умолчанию закрытый ключ шифруется главным ключом базы данных. Если главного ключа базы данных не существует, а пароль не указан, произойдет ошибка при выполнении инструкции.

Если закрытый ключ зашифрован главным ключом базы данных, указывать пароль расшифровки не требуется.

2. ALTER CERTIFICATE - изменяет закрытый ключ, используемый для шифрования сертификата, или добавляет закрытый ключ, если он не существует. Изменяет доступность сертификата для компонента Компонент

Service Broker. Закрытый ключ должен соответствовать открытому ключу, заданному в аргументе `certificate_name`. Предложение `DECRYPTION BY PASSWORD` может быть опущено, если пароль в файле защищен паролем, равным `NULL`.

Если закрытый ключ сертификата, уже существующий в базе данных, импортируется из файла, закрытый ключ будет автоматически защищен главным ключом базы данных. Чтобы защитить закрытый ключ паролем, используйте предложение `ENCRYPTION BY PASSWORD`.

Параметр `REMOVE PRIVATE KEY` удалит закрытый ключ сертификата из базы данных. Это можно сделать, если сертификат будет использоваться для проверки подписей или в сценариях компонента Компонент Service Broker, не требующих закрытого ключа. Не удаляйте закрытый ключ сертификата, который защищает симметричный ключ.

Указывать пароль расшифровки не нужно, если закрытый ключ зашифрован с помощью главного ключа базы данных.

- Изменение пароля сертификата:

```
ALTER CERTIFICATE Shipping04
WITH PRIVATE KEY
(DECRYPTION BY PASSWORD = 'pGF$5DGvbd2439587y',
ENCRYPTION BY PASSWORD = '4-329578thlkajdshglXCSgf');
GO
```

- Изменение пароля, используемого для шифрования закрытого ключа:

```
ALTER CERTIFICATE Shipping11
WITH PRIVATE KEY
(ENCRYPTION BY PASSWORD = '34958tosdgtkfkh##38',
DECRYPTION BY PASSWORD = '95hkjdskghFDGGG4%');
GO
```

- Изменение защиты закрытого ключа паролем на защиту главным ключом базы данных:

```
ALTER CERTIFICATE Shipping15
WITH PRIVATE KEY
(DECRYPTION BY PASSWORD = '95hk000eEnvjkjy#F%');
GO
```

3. `DROP CERTIFICATE` - удаляет сертификат из базы данных. Резервная копия сертификата, используемого для шифрования базы данных, должна быть сохранена, даже если она больше не включена в базе данных. Даже если база данных больше не зашифрована, части журнала транзакций могут по-прежнему

оставаться защищенными, а для некоторых операций будет требоваться сертификат до выполнения полного резервного копирования базы данных.

Сертификат также потребуется для восстановления из резервных копий, созданных в то время, когда база данных была зашифрована. Сертификаты можно удалять только при условии, что с ними не связано никаких сущностей.

4. `BACKUP CERTIFICATE` - экспортирует сертификат в файл. Если закрытый ключ зашифрован с паролем в базе данных, необходимо указать пароль для дешифрования.

При создании резервной копии закрытого ключа в файле шифрование является необходимым. Пароль, используемый для защиты резервной копии сертификата, не является тем же ключом, который применялся для шифрования закрытого ключа сертификата.

Чтобы восстановить сертификат из резервной копии, используйте инструкцию `CREATE CERTIFICATE`.

5. `ENCRYPTBYCERT` - шифрует данные на открытом ключе сертификата.

Эта функция шифрует данные при помощи открытого ключа сертификата. Код может быть расшифрован только с помощью соответствующего закрытого ключа. Ассиметричные преобразования гораздо более накладны, чем шифрование и дешифрование с использованием симметричного ключа. Поэтому использование ассиметричного шифрования не рекомендуется при работе с большими объемами данных, например, таблицами пользовательских данных.

6. `DECRYPTBYCERT` - расшифровывает данные при помощи закрытого ключа сертификата.

Эта функция расшифровывает данные при помощи закрытого ключа сертификата. Криптографические преобразования с использованием ассиметричных ключей требуют значительных ресурсов. Поэтому функции `EncryptByCert` и `DecryptByCert` не подходят для операций шифрования пользовательских данных.

Рассмотрим пример использования функций для управления сертификатами:

1. Вначале создадим папку (`Certf`), где будут храниться сертификаты, рисунок 18.

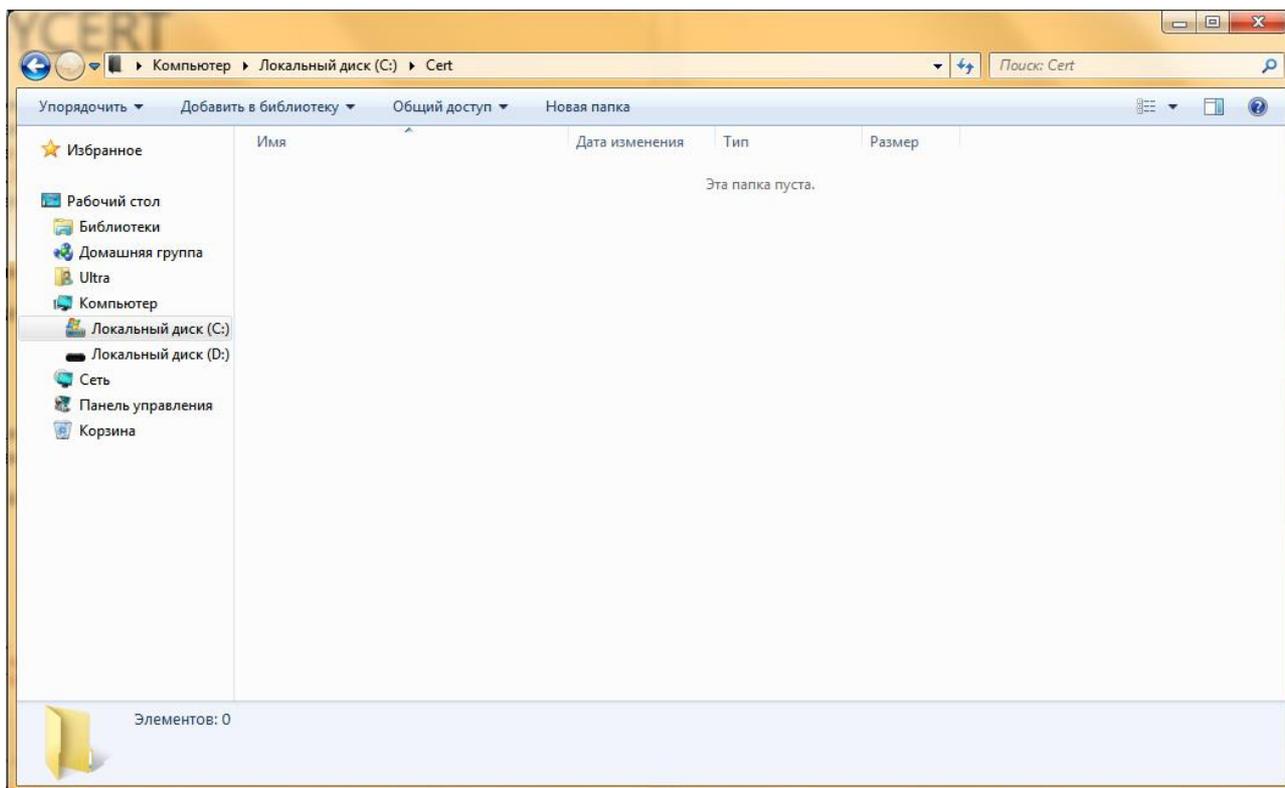


Рисунок 18 - Папка для хранения сертификатов

2. Создаём сертификат в нашей базе данных, который шифруется с помощью пароля, указываем тему сертификата и время действия сертификата. Экспортируем его в файловую систему и указываем приватный ключ. Задаём пароль для шифрации и дешифрации, рисунок 19:

```
USE Библиотека go
--Создаём сертификат
CREATE CERTIFICATE SampleCert
ENCRYPTION BY PASSWORD = 'Password'
WITH SUBJECT = 'Sample certificate',
EXPIRY_DATE = '2026-10-10';
--Сохраняем сертификат в виде файла
BACKUP CERTIFICATE SampleCert
TO FILE='C:\Cert\BackupSampleCert.cer'
WITH PRIVATE KEY ( FILE='C:\Cert\BackupSampleCert.pvk',
ENCRYPTION BY PASSWORD='p@ssword',
DECRYPTION BY PASSWORD='Password')
```

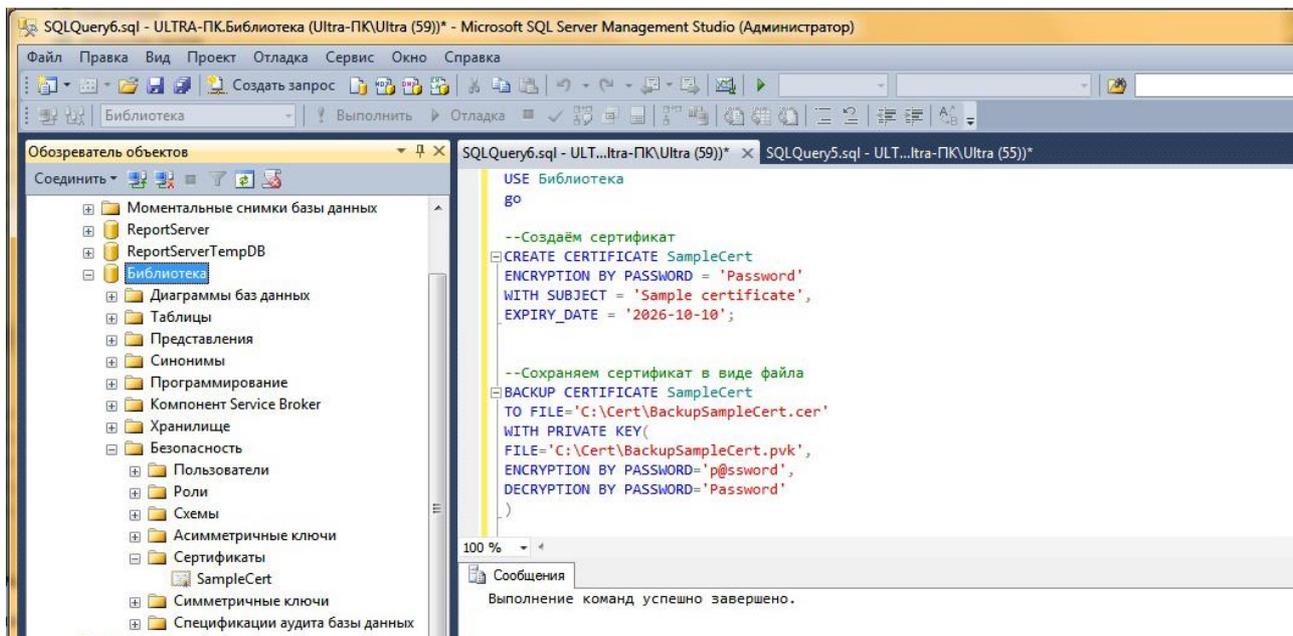


Рисунок 19 - Создание сертификата

4. Чтобы удалить сертификат из sql server введите следующий запрос, рисунок 20:

DROP CERTIFICATE SampleCert

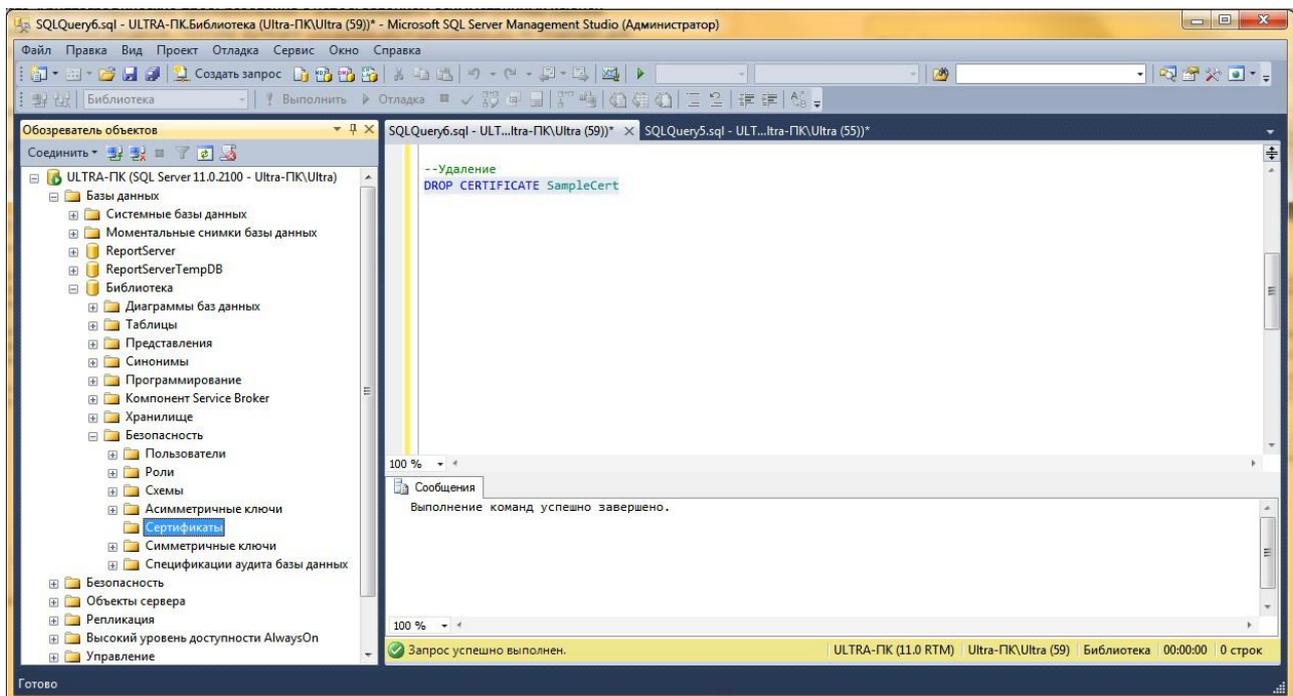


Рисунок 20 - Удаление сертификата

5. Чтобы воссоздать из файла введите следующий вопрос, рисунок 21:

**CREATE CERTIFICATE SampleCert
FROM FILE='C:\Cert\BackupSampleCert.cer'
WITH PRIVATE KEY(
FILE='C:\Cert\BackupSampleCert.pvk',**

ENCRYPTION BY PASSWORD='p@ssword',
DECRYPTION BY PASSWORD='Password')

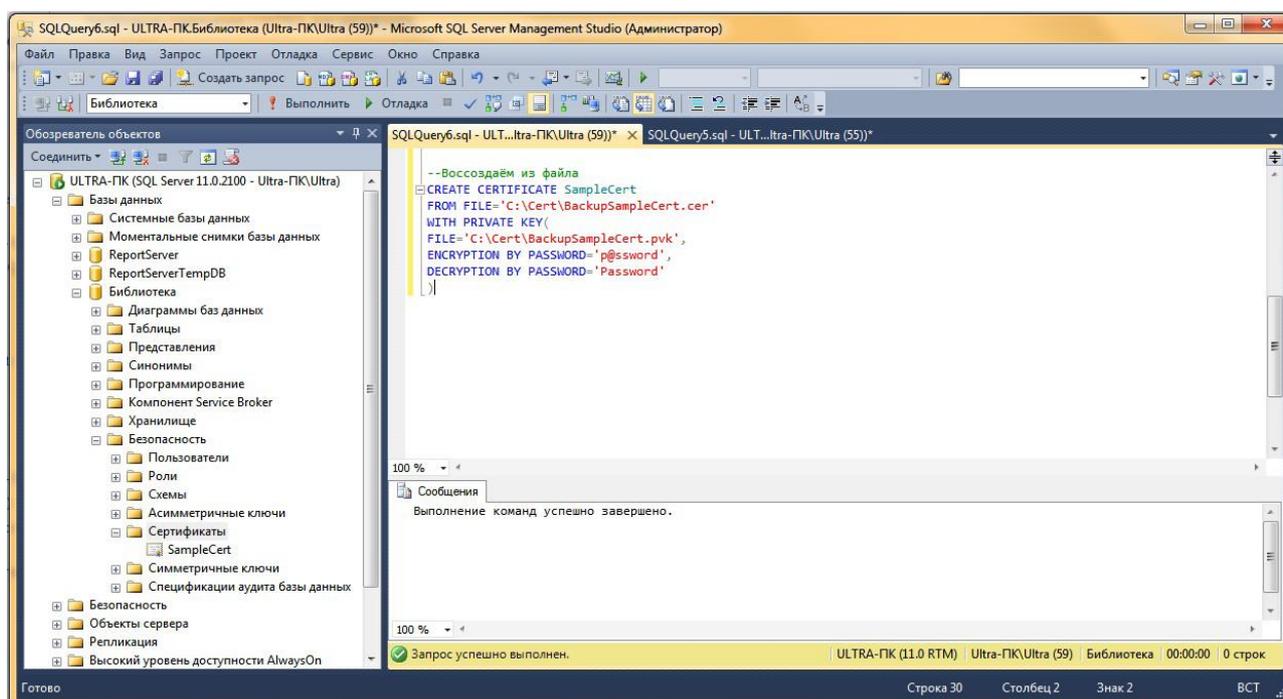


Рисунок 21 - Воссоздаём сертификат из созданного файла

Рассмотрим симметричные ключи. Функции для управления симметричными ключами:

- **CREATE SYMMETRIC KEY** - создает симметричный ключ и указывает его свойства в SQL Server. После создания симметричный ключ должен быть зашифрован с помощью по крайней мере одного из следующих средств: сертификат, пароль, симметричный ключ, асимметричный ключ или PROVIDER. Ключ может быть зашифрован более чем один раз для каждого типа шифрования. Другими словами, один симметричный ключ может быть зашифрован с использованием нескольких сертификатов, симметричных ключей и асимметричных ключей одновременно.

Если симметричный ключ шифруется с использованием пароля вместо открытого ключа главного ключа базы данных, то используется алгоритм шифрования TRIPLE DES. Поэтому ключи, созданные с помощью сильных алгоритмов шифрования, таких как AES, защищены с помощью более слабого алгоритма.

Необязательный пароль можно использовать для шифрования симметричного ключа перед распространением ключа нескольким пользователям.

Владельцем временных ключей является пользователь, который создает их. Временные ключи действительны только в текущем сеансе.

Аргумент **IDENTITY_VALUE** формирует идентификатор GUID, с помощью которого маркируются данные, зашифрованные новым

симметричным ключом. Маркирование может быть использовано для сопоставления ключей шифрованным данным. Идентификатор GUID, формируемый указанной фразой, будет всегда одним и тем же. Фраза, использованная для создания идентификатора GUID, не может быть повторно использована до тех пор, пока активен хотя бы один сеанс, использующий эту фразу. Предложение `IDENTITY_VALUE` является необязательным, но рекомендуется использовать его для хранения данных, зашифрованных с применением временного ключа.

Не существует алгоритма шифрования по умолчанию.

Защищать конфиденциальные данные с помощью потоковых шифров RC4 и RC4_128 не рекомендуется.

Сведения о симметричных ключах доступны в представлении каталога `sys.symmetric_keys`.

Симметричные ключи не могут быть зашифрованы с помощью симметричных ключей, созданных поставщиком шифрования.

Пояснение к алгоритмам DES:

- DESX был именован неправильно. Симметричные ключи, созданные с параметром `ALGORITHM = DESX`, в действительности используют шифр TRIPLE DES с 192-битным ключом. Алгоритм DESX не предоставляется. В будущей версии Microsoft SQL Server этот компонент будет удален. Избегайте использования этого компонента в новых разработках и запланируйте изменение существующих приложений, в которых он применяется.

- Симметричные ключи, созданные с параметром `ALGORITHM = TRIPLE_DES_3KEY`, используют шифр TRIPLE DES с 192-битным ключом.

- Симметричные ключи, созданные с параметром `ALGORITHM = TRIPLE_DES`, используют шифр TRIPLE DES с 128-битным ключом.

Множественное использование одного и того же RC4 или RC4_128 KEY_GUID для различных блоков данных приведет к одному и тому же ключу RC4, так как SQL Server не предоставляет рассеивание автоматически. Повторное использование одного и того же ключа RC4 является типичной ошибкой, становящейся причиной очень слабого шифрования. Таким образом, ключевые слова RC4 и RC4_128 являются устаревшими. В будущей версии Microsoft SQL Server этот компонент будет удален. Не используйте его при работе над новыми приложениями и как можно быстрее измените приложения, в которых он в настоящее время используется.

Алгоритм RC4 поддерживается только в целях обратной совместимости. Когда база данных имеет уровень совместимости 90 или 100, новые материалы могут шифроваться только с помощью алгоритмов RC4 или RC4_128. (Не рекомендуется.) Используйте вместо этого более новые алгоритмы, например AES. В SQL Server 2014 материалы, зашифрованные с помощью алгоритмов RC4 или RC4_128, могут быть расшифрованы на любом уровне совместимости.

2. ALTER SYMMETRIC KEY - изменяет свойства симметричного ключа. Для изменения шифрования симметричного ключа используйте предложения ADD ENCRYPTION и DROP ENCRYPTION. Невозможно, чтобы ключу не соответствовал никакой способ шифрования. Поэтому оптимальным способом изменения метода шифрования является добавление новой формы шифрования и затем удаление старой.

Для изменения владельца симметричного ключа выполните инструкцию ALTER AUTHORIZATION.

3. DROP SYMMETRIC KEY - удаляет симметричный ключ из текущей базы данных. Если ключ является открытым в текущем сеансе, то инструкция завершится с ошибками.

Если асимметричный ключ был сопоставлен с ключом расширенного управления ключами на устройстве расширенного управления ключами, а параметр REMOVE PROVIDER KEY не был указан, то ключ будет удален из базы данных, но не с устройства; также будет выдано предупреждение.

4. OPEN SYMMETRIC KEY - расшифровывает симметричный ключ и делает его доступным для использования. Открытые симметричные ключи привязаны к сеансу, а не к контексту безопасности. Открытый ключ останется доступным, пока не будет явно закрыт или сеанс не будет прерван. Если открыт симметричный ключ, после чего произошло переключение контекста, ключ останется открытым и будет доступным в олицетворенном контексте.

Если симметричный ключ был зашифрован другим ключом, сначала необходимо открыть этот ключ. Если симметричный ключ уже открыт, то запрос является запросом NO_OP. Если пароль, сертификат или ключ для расшифровки симметричного ключа неверен, запрос завершается сбоем.

Симметричные ключи, созданные из поставщиков шифрования, не могут быть открыты. Операции шифрования и расшифровки, для которых используется симметричный ключ этого типа, выполняются успешно без инструкции OPEN, поскольку открытие и закрытие ключа осуществляется поставщиком шифрования.

5. CLOSE SYMMETRIC KEY - закрывает симметричный ключ или все симметричные ключи, открытые в текущем сеансе. Открытые симметричные ключи привязаны к сеансу, а не к контексту безопасности. Открытый ключ останется доступным, пока не будет явно закрыт или сеанс не будет прерван. Инструкция CLOSE ALL SYMMETRIC KEYS закрывает любой главный ключ базы данных, который был открыт в текущем сеансе при помощи инструкции OPEN MASTER KEY.

6. ENCRYPTBYKEY - производит шифрование данных при использовании симметричного ключа. Функция EncryptByKey использует симметричный ключ. Этот ключ должен быть открыт. Если симметричный ключ уже открыт в текущем сеансе, не нужно открывать его снова в контексте запроса.

Структура проверки подлинности позволяет исключить прямую подмену зашифрованных полей.

Если при шифровании данных указана структура проверки подлинности, то для дешифровки данных функцией DecryptByKey потребуются эти же данные структура проверки подлинности. При шифровании хэш данных проверки подлинности шифруется вместе с данными. При дешифровке эта же структура проверки подлинности должны быть переданы функции DecryptByKey. Если эти данные не совпадают, дешифровка завершится ошибкой. Это будет означать, что значение перемещено с другого места со времени шифрования. В качестве структуры проверки подлинности рекомендуется использовать столбец, содержащий уникальное и неизменное значение. Если значение структуры проверки подлинности изменится, можно утратить доступ к данным.

Симметричное шифрование и расшифровка осуществляются относительно быстро и подходит для работы с большими объемами данных.

7.DECRYPTBYKEY - расшифровывает данные с помощью симметричного ключа. В функции DecryptByKey используется симметричный ключ. Этот симметричный ключ должен быть открыт уже в базе данных. Одновременно могут быть открыты несколько ключей. Открывать ключ непосредственно перед раскодированием необязательно.

Симметричное кодирование и декодирование осуществляется относительно быстро и подходит для работы с большими объемами данных.

Рассмотрим пример шифрования с помощью симметричного ключа. Будем создавать два ключа первый будет шифровать данные, а второй будет шифровать первый ключ.

1. Сначала создаём второй ключ, а потом создаём ключ, который шифрует данные, рисунок 22:

```
USE Библиотека
go
--Создаём симметричный ключ для шифрования другого симметричного
ключа
CREATE SYMMETRIC KEY SymKey
WITH ALGORITHM =Triple_DES
ENCRYPTION BY PASSWORD='password'
```

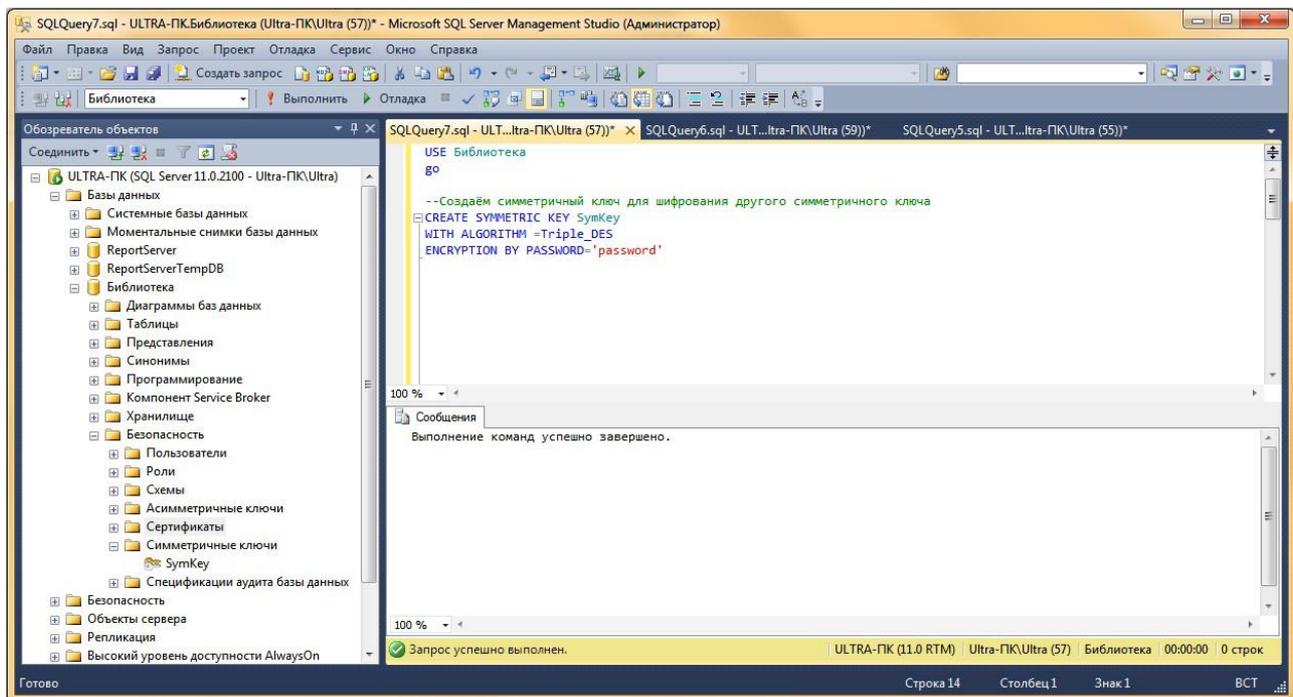


Рисунок 22 - Создание ключа для шифрования другого ключа

2. Открываем этот ключ:

```
OPEN SYMMETRIC KEY SymKey
```

```
DECRYPTION BY PASSWORD='password'
```

3. Создаём и открываем ключ, который шифрует данные, рисунок 23:

```
--Создаём симметричный ключ для шифрования данных
```

```
CREATE SYMMETRIC KEY SymData
```

```
WITH ALGORITHM=Triple_DES
```

```
ENCRYPTION BY SYMMETRIC KEY SymKey
```

```
--Открываем ключ для шифрования данных
```

```
OPEN SYMMETRIC KEY SymData
```

```
DECRYPTION BY SYMMETRIC KEY SymKey
```

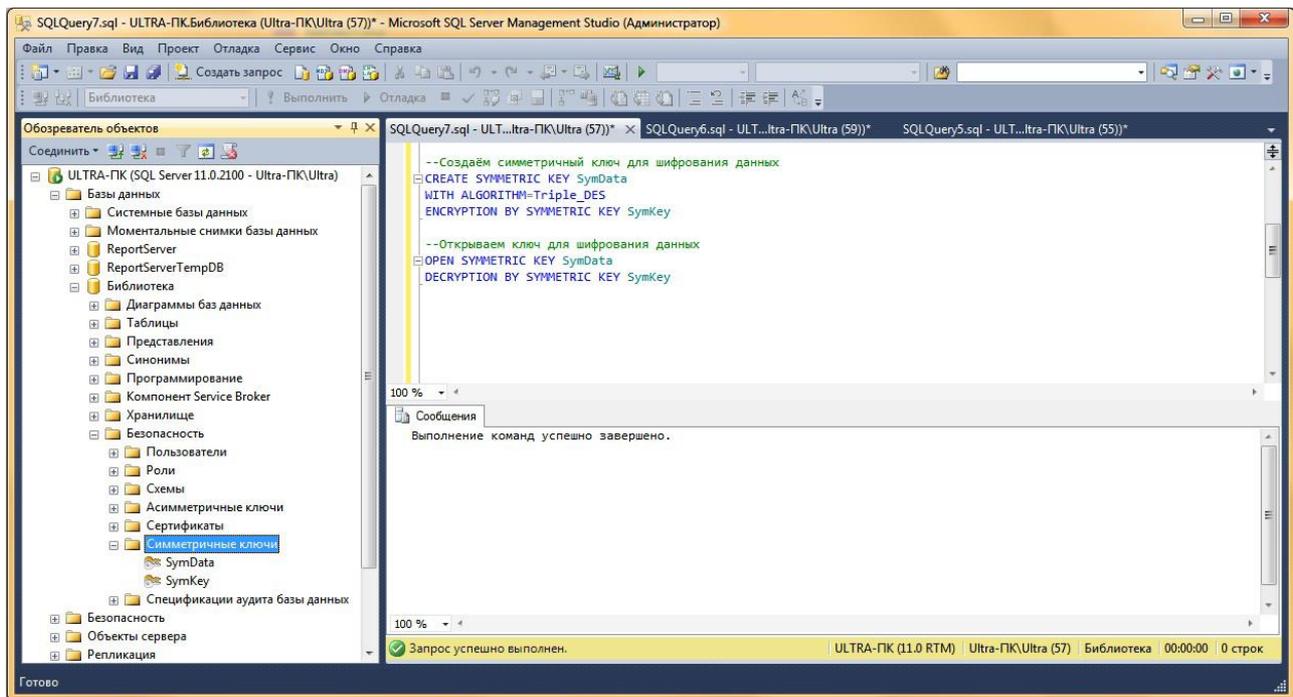


Рисунок 23 - Создание ключ для шифрования данных

4. Инициализируем текст, для шифрования, шифруем его и сразу же расшифровываем, рисунок 24:

--Инициализируем открытый текст

```
DECLARE @plaintext nvarchar(512)
```

```
SET @plaintext='Добрый день'
```

```
print @plaintext
```

--Шифруем данные

```
DECLARE @ciphertext varbinary(1024)
```

```
SET @ciphertext=ENCRYPTBYKEY
```

```
(KEY_GUID('SymData'),@plaintext)
```

```
print @ciphertext
```

--Расшифровываем данные

```
SET @plaintext=CAST
```

```
(DECRYPTBYKEY(@ciphertext) as nvarchar(512))
```

```
print @plaintext
```

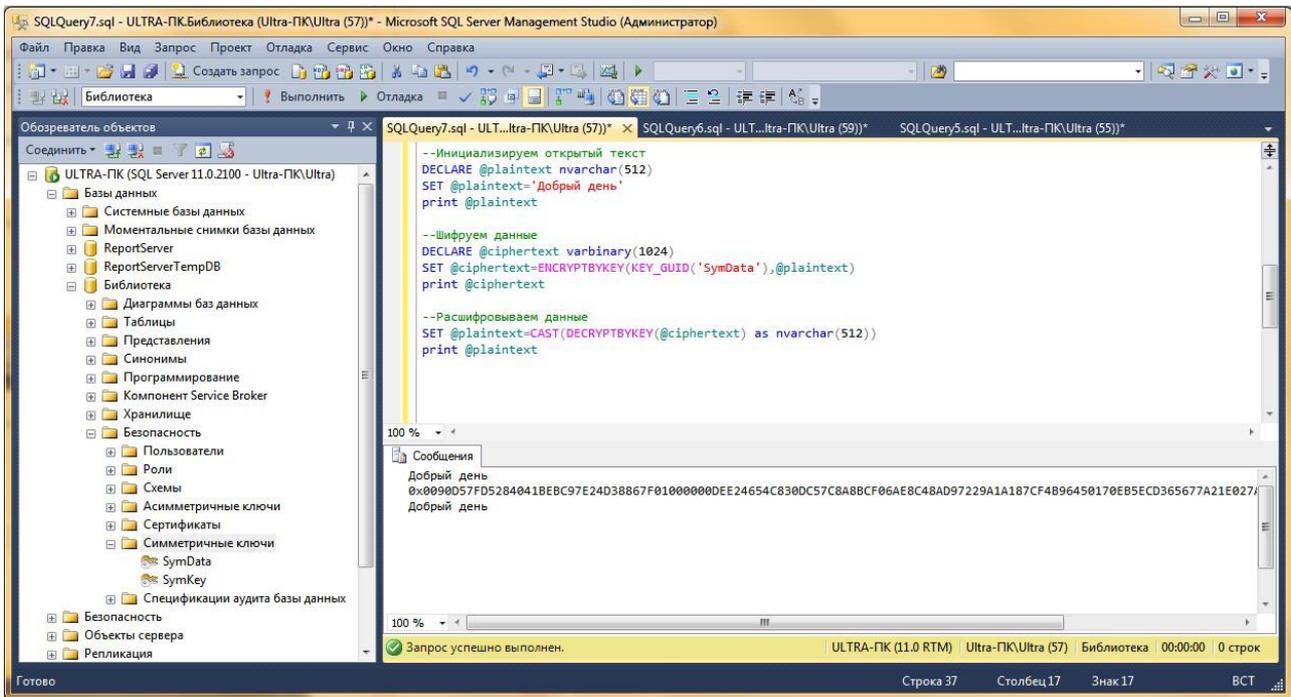


Рисунок 24 - Шифрование и расшифрование текста

5. Закрываем ключ шифрования данных и ключ шифрования ключа, рисунок 25:

```

--Закрываем ключ шифрования данных
CLOSE SYMMETRIC KEY SymData

--Закрываем ключ шифрования ключа
CLOSE SYMMETRIC KEY SymKey

```

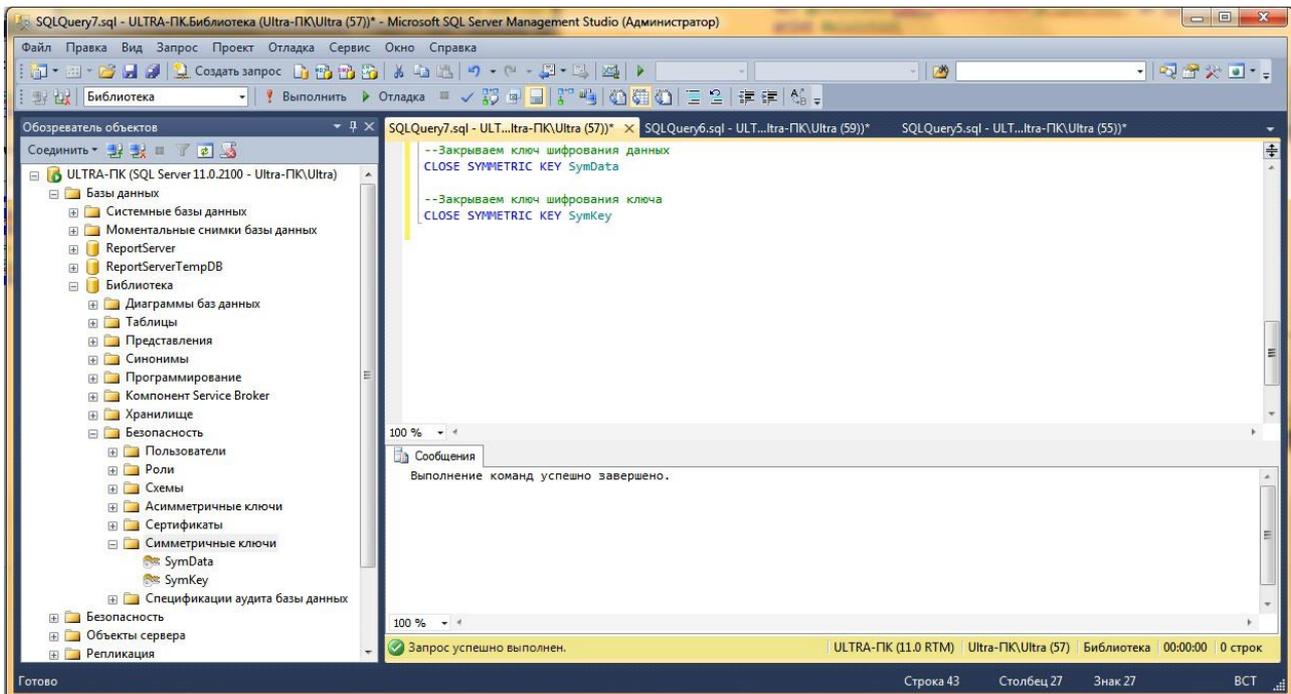


Рисунок 25 - Закрываем симметричные ключи

Функция ENCRYPTBYPASSPHRASE - шифрование данных с помощью парольной фразы с использованием алгоритма TRIPLE DES и 128-битного ключа.

DECRYPTBYPASSPHRASE - расшифровывает данные, зашифрованные с помощью парольной фразы.

Пример (рисунок 26):

```
Declare @plaintext nvarchar(1000), @enc_text varbinary(2000)
SET @plaintext='Я помню чудное мгновенье'
SET @enc_text=ENCRYPTBYPASSPHRASE('LOM',@plaintext)
SELECT 'Оригинальный текст: ',@plaintext
SELECT 'Зашифрованный текст: ',@enc_text
SELECT 'Расшифровка:', CAST (DECRYPTBYPASSPHRASE
('LOM',@enc_text) as nvarchar(1000))
```

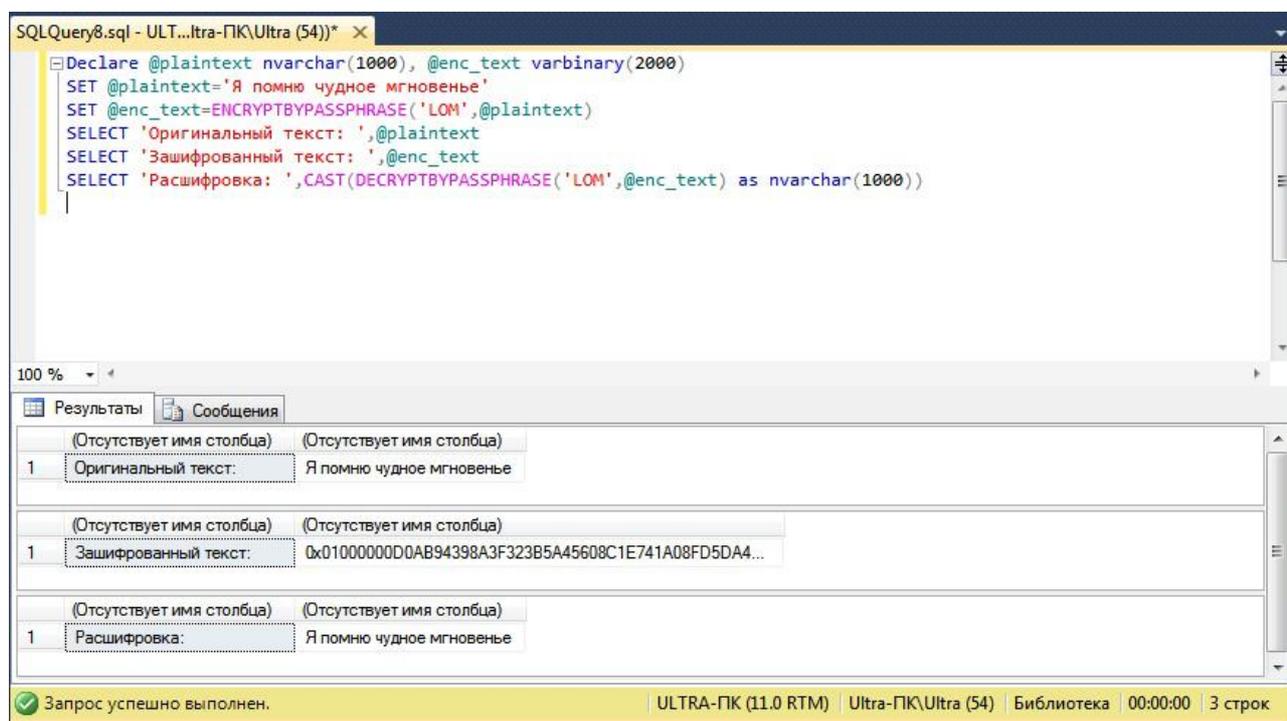


Рисунок 26 - Шифрование с помощью ключевой фразы

Создадим базу данных, данные в которой шифруются:

1. с использованием симметричного ключа, который будет зашифрован с помощью асимметричного ключа.

2. с использованием симметричного ключа, но на этот раз симметричный ключ шифруется сертификатом.

Создание базы данных:

USE [master]

GO

```
CREATE DATABASE [DB]
```

```
GO
```

Запустите код T-SQL для создания таблицы с именем TelephonNumber в базе данных DB:

```
USE [DB]
```

```
GO
```

```
CREATE TABLE [dbo].[ TelephonNumber]
```

```
([PersonID] [int] PRIMARY KEY,
```

```
[TelephonNumber] [varbinary](max))
```

```
GO
```

Эта таблица будет содержать ложную информацию о телефонных номерах. Номера телефонов будут сохранены в столбце двоичных переменных, потому что они будут шифроваться.

Используйте следующий программный код для создания главного ключа DMK базы данных DB, шифруемого с помощью парольной фразы \$str0nGPa\$\$w0rd:

```
USE [DB]
```

```
GO
```

```
CREATE MASTER KEY ENCRYPTION BY PASSWORD =  
'$str0nGPa$$w0rd'
```

```
GO
```

Необходимо создать асимметричный ключ, зашифровать его парольной фразой \$str0nGPa\$\$w0rd, создать симметричный ключ и зашифровать симметричный ключ с помощью только что созданного асимметричного ключа. Выполнить эти задачи можно, запустив программный код:

```
USE [DB]
```

```
GO
```

```
--Создание асимметричного ключа, зашифрованного парольной фразой  
StrongPa$$w0rd!
```

```
CREATE ASYMMETRIC KEY MyAsymmetricKey
```

```
WITH ALGORITHM = RSA_2048
```

```
ENCRYPTION BY PASSWORD = 'StrongPa$$w0rd!'
```

```
GO
```

```
--Создание симметричного ключа, зашифрованного асимметричным  
ключом
```

```
CREATE SYMMETRIC KEY MySymmetricKey
```

```
WITH ALGORITHM = AES_256
```

```
ENCRYPTION BY ASYMMETRIC KEY MyAsymmetricKey
```

```
GO
```

1. Теперь мы можем приступить к шифрованию данных. Для этого необходимо сначала открыть симметричный ключ, только что созданный с помощью команды `OPEN SYMMETRIC KEY`, за которой следует имя симметричного ключа. Затем указать, что нужно расшифровать его с использованием заданного асимметричного ключа. Программный код выглядит следующим образом:

```
USE [DB]
```

```
GO
```

```
OPEN SYMMETRIC KEY MySymmetricKey
```

```
DECRYPTION BY ASYMMETRIC KEY MyAsymmetricKey
```

```
WITH PASSWORD = 'StrongPa$$wOrd!'
```

```
GO
```

После выполнения этого кода направьте запрос в представление `sys.openkeys`, чтобы убедиться, что ключ открыт, рисунок 27:

```
USE [DB]
```

```
GO
```

```
SELECT * FROM [sys].[openkeys]
```

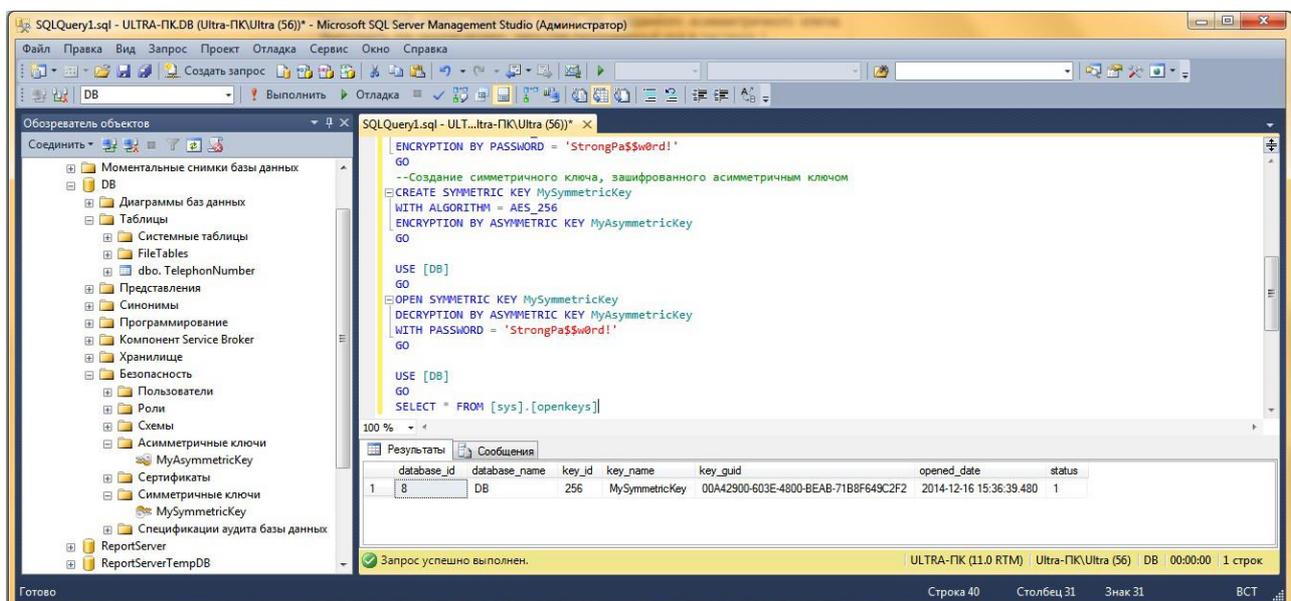


Рисунок 27 - Открытие симметричного ключа

Введем несколько телефонных номеров в таблицу `TelephoneNumber`, запустив код:

```
USE [DB]
```

```

GO
DECLARE @SymmetricKeyGUID AS [uniqueidentifier]
SET @SymmetricKeyGUID = KEY_GUID('MySymmetricKey')
IF (@SymmetricKeyGUID IS NOT NULL)
BEGIN
INSERT INTO [dbo].[ TelephonNumber]
VALUES (01, ENCRYPTBYKEY(@SymmetricKeyGUID,
N'8-761-123-87-63'))
INSERT INTO [dbo].[ TelephonNumber]
VALUES (02, ENCRYPTBYKEY(@SymmetricKeyGUID,
N'8-768-765-87-65'))
INSERT INTO [dbo].[ TelephonNumber]
VALUES (03, ENCRYPTBYKEY(@SymmetricKeyGUID,
N'8-761-234-11-11'))
END
TRUNCATE TABLE [dbo].[TelephonNumber]
Выведем получившуюся таблицу на экран, рисунок 28:
USE [DB]
GO
SELECT * FROM [dbo].[ TelephonNumber]

```

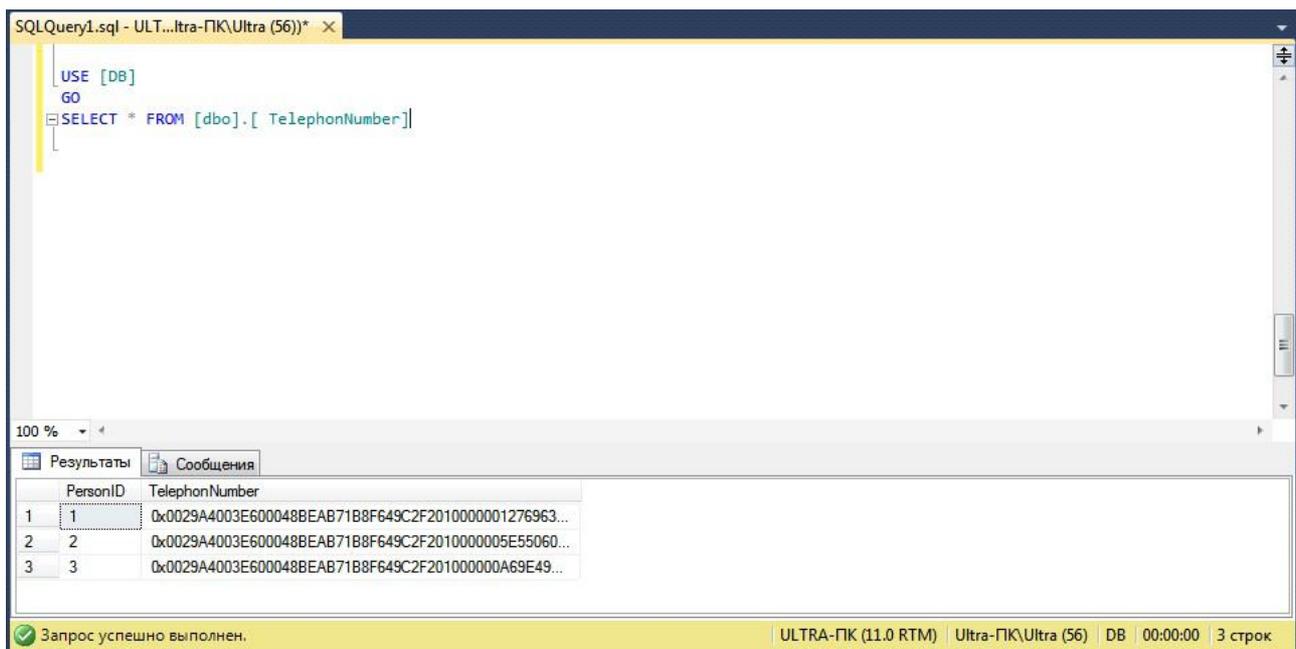


Рисунок 28 - Зашифрованные данные из таблицы TelephoneNumber

Все данные в столбце TelephoneNumber представлены в двоичном формате. С помощью функции DECRYPTBYKEY можно посмотреть зашифрованные данные, рисунок 29:

```
USE [DB]
GO
SELECT [PersonID],
CONVERT([nvarchar](32),
DECRYPTBYKEY(TelephoneNumber))
AS [TelephoneNumber]
FROM [dbo].[ TelephoneNumber]
GO
```

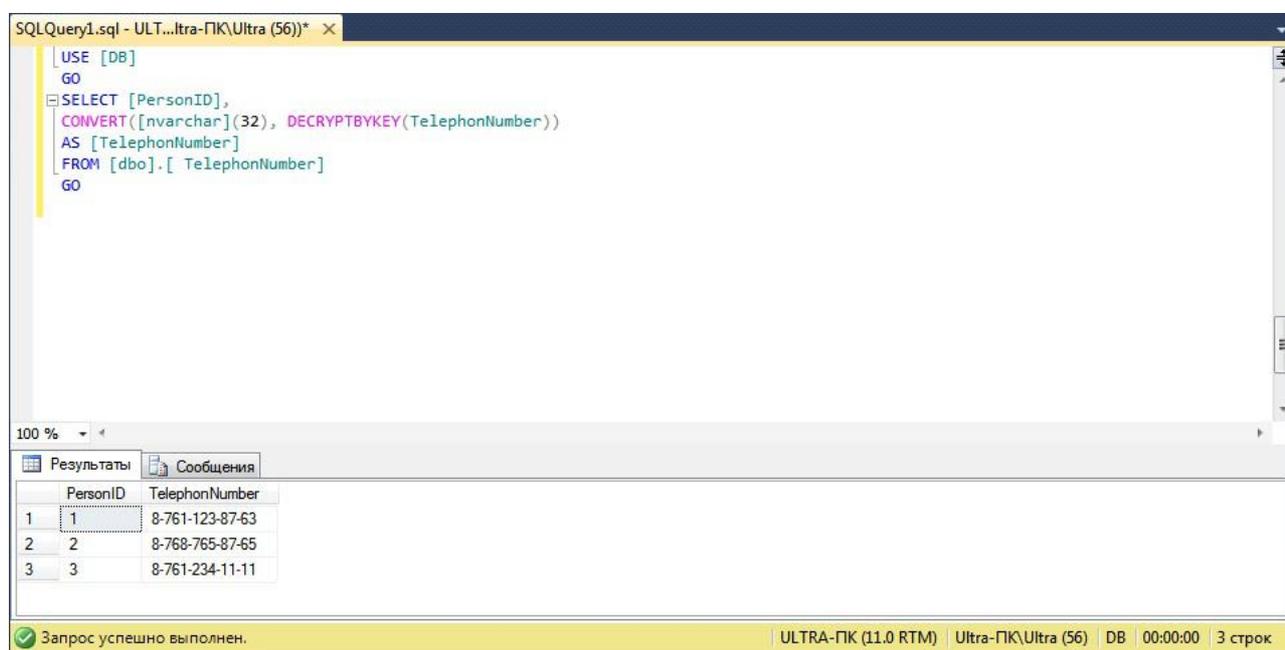


Рисунок 29 - Просмотр зашифрованных данных

2. Создадим сертификат с помощью инструкции CREATE CERTIFICATE. Затем создается симметричный ключ, шифруемый сертификатом. Наконец, открыв симметричный ключ, код вставляет три строки в таблицу TelephoneNumber:

```
USE [DB]
GO
--Создание сертификата
CREATE CERTIFICATE
[CertToEncryptSymmetricKey]
WITH SUBJECT ='Самозаверяющий сертификат
для шифрования симметричного ключа.'
```

```

--Создание симметричного ключа, зашифрованного сертификатом
CREATE SYMMETRIC KEY
[SymmetricKeyEncryptedWithCert]
WITH ALGORITHM = AES_256
ENCRYPTION BY CERTIFICATE
[CertToEncryptSymmetricKey]
--Открытие симметричного ключа
OPEN SYMMETRIC KEY
[SymmetricKeyEncryptedWithCert]
DECRYPTION BY CERTIFICATE
[CertToEncryptSymmetricKey]
--Усечение таблицы TelephonNumber
TRUNCATE TABLE
[dbo].[ TelephonNumber]
--Вставка данных в таблицу
DECLARE @SymmetricKeyGUID
AS [uniqueidentifier]
SET @SymmetricKeyGUID =
KEY_GUID
('SymmetricKeyEncryptedWithCert')
IF (@SymmetricKeyGUID IS NOT NULL)
BEGIN
INSERT INTO [dbo].[ TelephonNumber]
VALUES (01, ENCRYPTBYKEY
(@SymmetricKeyGUID,
N'8-861-123-87-63'))
INSERT INTO [dbo].[ TelephonNumber]
VALUES (02, ENCRYPTBYKEY
(@SymmetricKeyGUID,
N'8-868-765-87-65'))
INSERT INTO [dbo].[ TelephonNumber]
VALUES (03, ENCRYPTBYKEY
(@SymmetricKeyGUID,

```

N'8-861-234-11-11'))

END

Просмотр зашифрованных данных, рисунок 30:

USE [DB]

GO

SELECT * FROM

[dbo].[TelephonNumber]

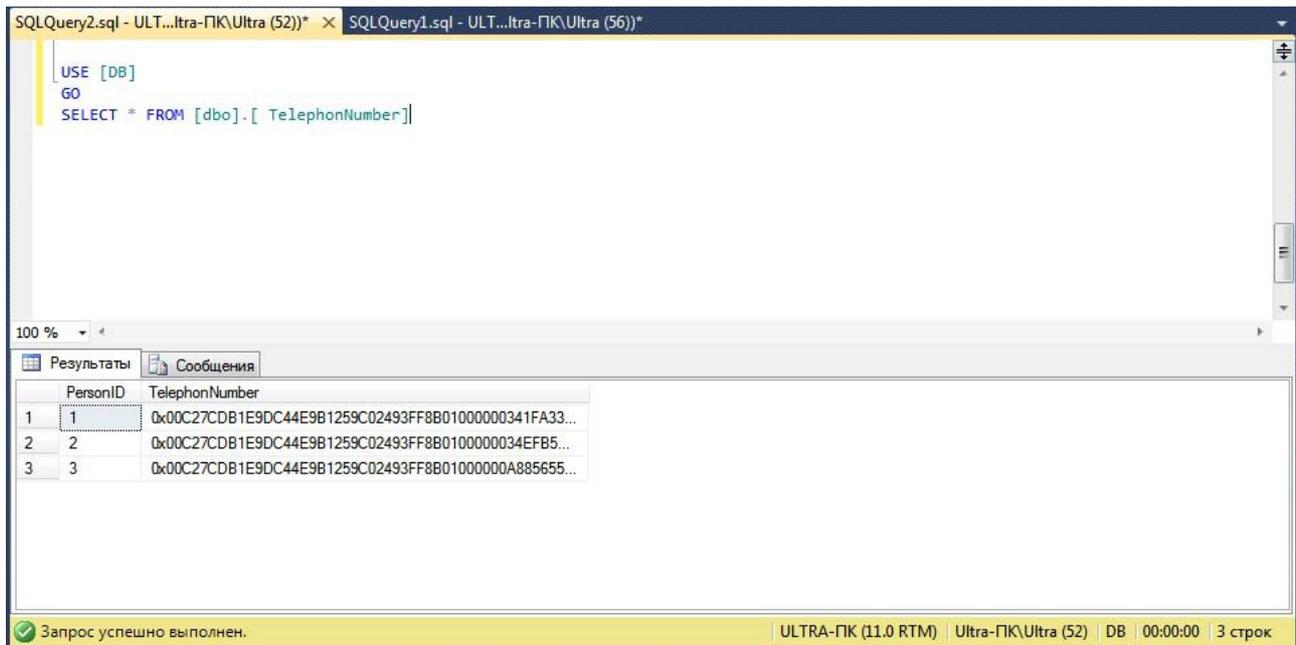


Рисунок 30 - Вывод данных таблицы «TelephonNumber»

Расшифровка, рисунок 31:

USE [DB]

GO

SELECT [PersonID],

CONVERT([nvarchar](32), DECRYPTBYKEY(TelephonNumber))

AS [TelephonNumber]

FROM [dbo].[TelephonNumber]

GO

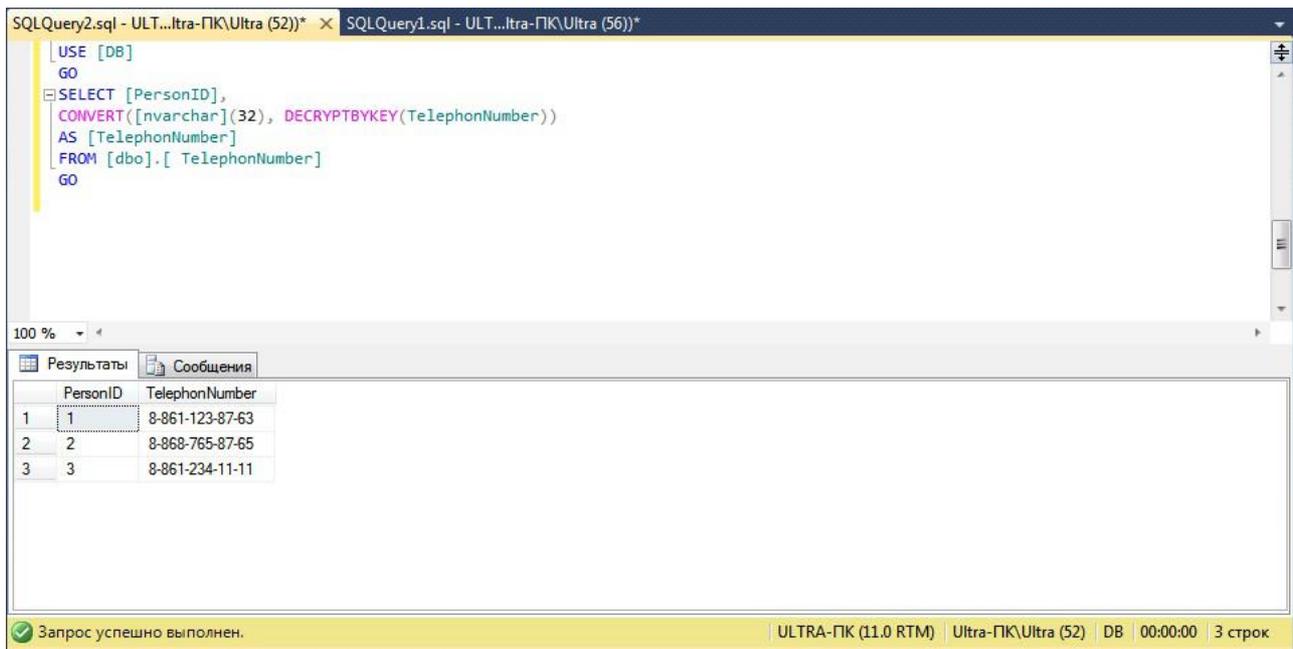


Рисунок 31 - Расшифровка данных таблицы «TelephonNumber»

Рассмотрим пример прозрачного шифрования данных, рисунок 32.

1. Создаем главный ключ шифрования
CREATE MASTER KEY ENCRYPTION
BY PASSWORD = 'StrongPassword#1';

Созданный наш ключ можно увидеть в view:

```
select * from sys.key_encryptions
```

2. Чтобы удалить ключ введите следующий запрос:

```
drop master key
```

3. После того как создали главный ключ, необходимо сделать его резервную копию и поместить резервную копию в надежное место:

```
BACKUP MASTER KEY TO FILE = 'C:\Cert\MasterBackup.bak'  
ENCRYPTION BY PASSWORD = 'Password1'
```

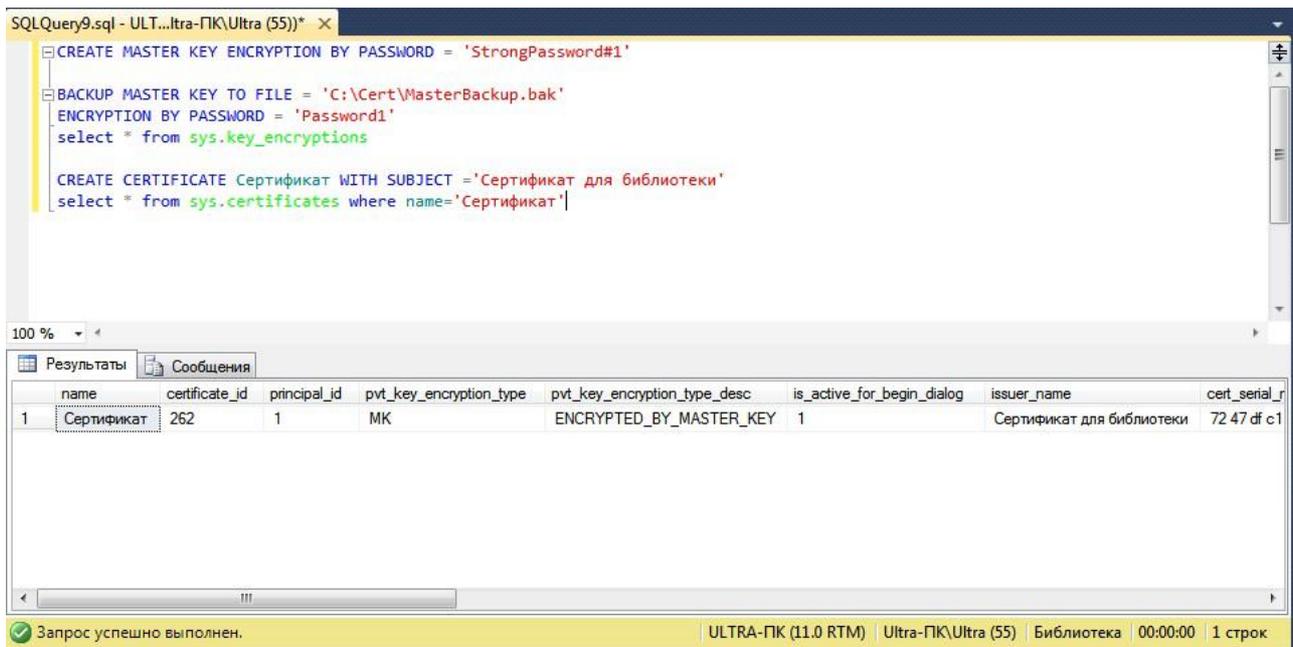


Рисунок 32 - Создание главного ключа и сертификата

4. Создание сертификата осуществляется следующим запросом:

```

CREATE CERTIFICATE Сертификат
WITH SUBJECT = 'Сертификат для библиотеки'

```

Проверка наличия созданного сертификата:

```

select * from sys.certificates where name='Сертификат'

```

5. Создание резервной копии сертификата с закрытым ключом, рисунок 33:

```

BACKUP CERTIFICATE Сертификат
TO FILE = 'C:\Cert\Certif'
WITH PRIVATE KEY
(FILE = 'C:\Cert\PrivateCertif',
ENCRYPTION BY PASSWORD = 'Password#3');

```

Выполнив всё вышеизложенное приступим к созданию ключа шифрования в нашей базе данных с использованием нашего сертификата:

```

USE Библиотека

```

```

go

```

```

CREATE DATABASE ENCRYPTION KEY
WITH ALGORITHM = AES_128

```

```

ENCRYPTION BY SERVER CERTIFICATE Сертификат;

```

Включаем шифрование для нашей базы данных:

```

ALTER DATABASE Библиотека

```

SET ENCRYPTION ON;

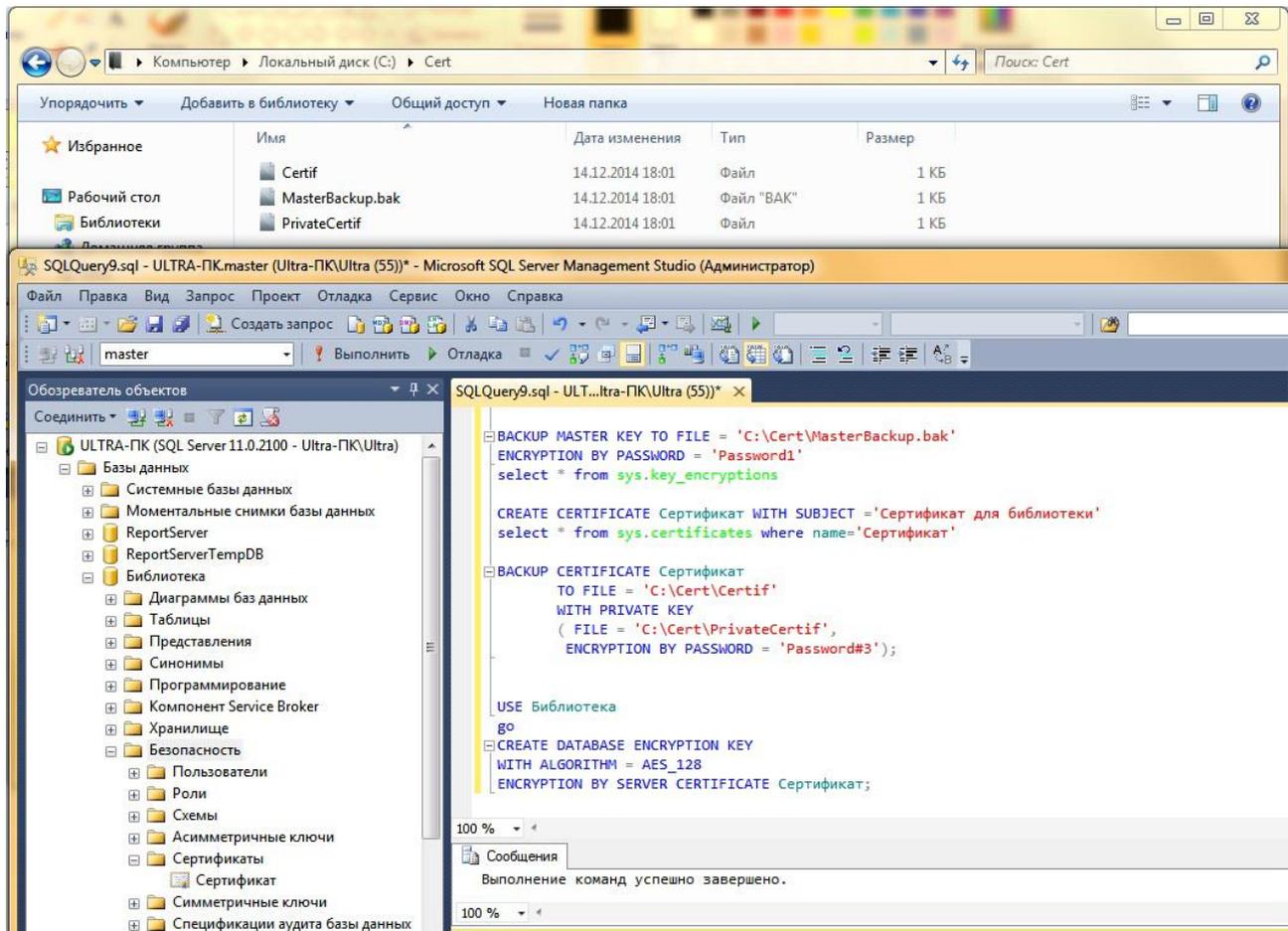


Рисунок 33 - Резервные копии сертификата и главного ключа. Создание ключа шифрования в нашей базе данных

Чтобы посмотреть зашифрованные базы данных, нужно ввести следующий запрос, рисунок 34:

```
USE [master]  
GO  
SELECT db.[name]  
  , db.[is_encrypted]  
  , dm.[encryption_state]  
  , dm.[percent_complete]  
  , dm.[key_algorithm]  
  , dm.[key_length]  
FROM [sys].[databases] db  
LEFT OUTER JOIN [sys].[dm_database_encryption_keys] dm  
ON db.[database_id] = dm.[database_id];  
GO
```

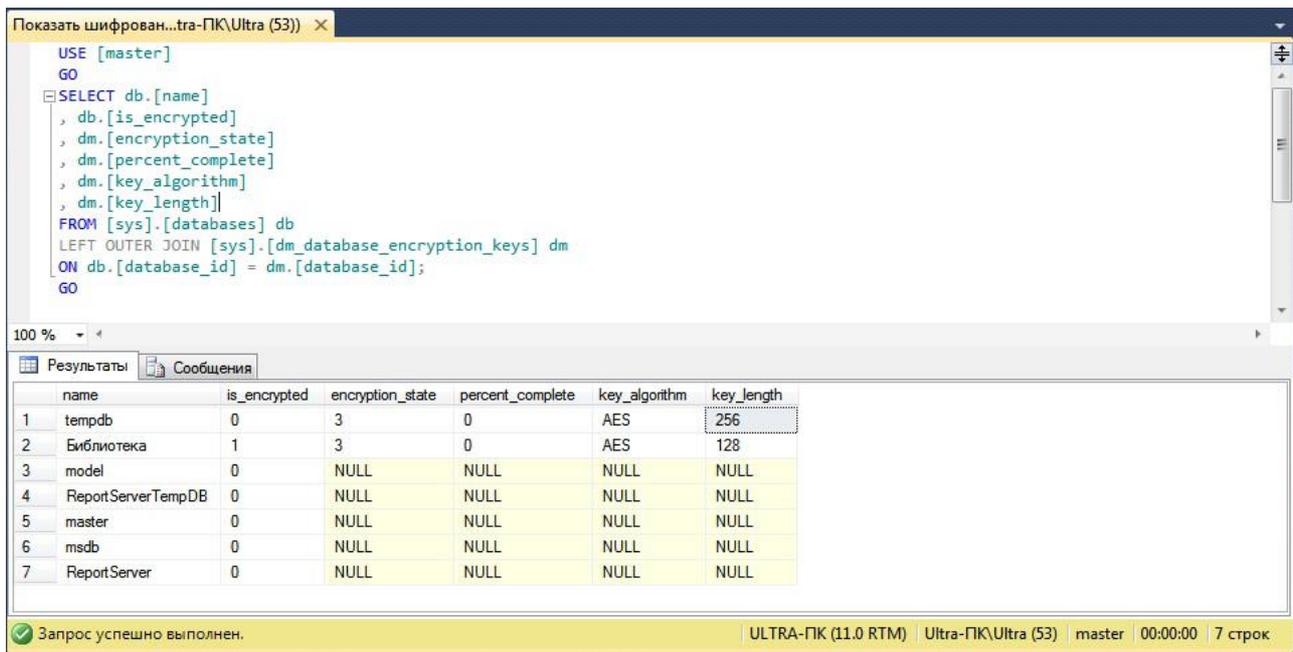


Рисунок 34 - Вывод списка баз данных

Несколько моментов в работе с базой данных с включенным шифрованием:

Если создать резервную копию и попытаться восстановить на другом сервере, то получим ошибку:

Msg 33111, Level 16, State 3, Line 2

Cannot find server certificate with thumbprint '0x5B139FF1F2C5ED9EB3D503E78A63DEF3DD1FD96F'. Msg 3013, Level 16, State 1, Line 2 RESTORE DATABASE is terminating abnormally.

Такую же ошибку получим и при попытке присоединения файлов базы данных. Порядок восстановления базы данных с прозрачным шифрованием на другом экземпляре MS SQL Server:

1. Создать мастер главный ключ шифрования на сервере MS SQL Server.

2. Восстановить из резервной копии сертификат с закрытым ключом:

```

CREATE CERTIFICATE Сертификат
FROM FILE = 'c:\Program Files\Microsoft SQL
Server\MSSQL11.MSSQLSERVER\MSSQL\DATA\Certif'
WITH PRIVATE KEY (FILE = 'c:\Program Files\Microsoft SQL
Server\MSSQL11.MSSQLSERVER\MSSQL\DATA\PrivateCertif',
DECRYPTION BY PASSWORD = 'Password#3');

```

3. Восстановить шифрованную базу данных или присоединить файлы базы данных с включенным шифрованием. БД готова для работы.

3 Организация выполнения курсовой работы

3.1 Выбор темы курсовой работы

На первой неделе срока, выделенного учебным планом для выполнения курсовой работы, до сведения студентов доводится список тем.

Студенту предоставляется право в течение недели выбрать любую тему из предложенного списка. Также студент может сам предложить интересующую его тему, если она соответствует изучаемому предмету и целям курсового проектирования.

При выборе темы рекомендуется консультироваться с преподавателем дисциплины «Безопасность систем баз данных».

Выполнять курсовые работы на одну и ту же тему нескольким студентам из одной учебной группы не рекомендуется.

Выбранная студентом тема утверждается руководителем курсового проектирования, о чем вносится соответствующая запись в бланк задания на подготовку курсового проекта (см. Приложение 2).

Задание на выполнение курсовой работы является нормативным документом, устанавливающим границы и глубину разработки темы, а также сроки представления работы на кафедру в завершённом виде.

3.2 Контроль выполнения курсовой работы

После утверждения темы курсовой работы студент обязан изучить исходные данные к курсовой работе, подобрать и изучить литературу по теме проекта, составить план, регулярно посещать консультации руководителя, дорабатывать отдельные части проекта по замечаниям руководителя, своевременно подготовить и сдать на проверку законченную курсовую работу и защитить его.

На руководителя возлагается ответственность за постоянное наблюдение за разработкой всех разделов курсового проекта в соответствующие сроки и оказание студенту необходимой помощи на всех этапах выполнения проекта.

В процессе проектирования для студентов проводятся в соответствии с утвержденным кафедрой графиком групповые и индивидуальные консультации.

Руководитель обязан:

- установить студенту календарный график выполнения этапов проектирования и занести его в бланк задания на подготовку курсового проекта (см. Приложение Б);

- регулярно проводить консультации в соответствии с графиком при уточнении темы, разработке плана, составлении списка литературы, обобщении материала и т.д.;

- контролировать соблюдение календарных сроков и качество выполнения как отдельных частей, так и проекта в целом. Если при проверке обнаружатся ошибки, неполнота объема, незавершенность проектирования или низкое качество оформления, то проект возвращается студенту для доработки;

- принять (по-возможности, совместно с комиссией) защиту курсового проекта.

3.3 Подведение итогов и защита курсовой работы. Подготовка презентации

В качестве отчетных материалов по проекту студент должен представить пояснительную записку.

Подведение итогов подготовки курсовой работы включает следующие этапы:

- сдача курсовой работы на проверку руководителю;
- доработка курсовой работы с учетом замечаний руководителя;
- сдача готовой курсовой работы на защиту;
- защита курсовой работы.

Срок сдачи готовой курсовой работы определяется заведующим кафедрой. График защиты курсовых работ вывешивается на доске объявлений.

Срок доработки курсовой работы устанавливается руководителем с учетом сущности замечаний и объема необходимой доработки.

Выполненная курсовой работы подписывается студентом и представляется на защиту. Курсовой работ, удовлетворяющая предъявленным требованиям и прошедшая проверку на объем заимствований допускается к защите.

На проверенную курсовую работу руководитель в обязательном порядке пишет отзыв по строго установленной в академии форме (Приложение 9).

В отзыве дается оценка уровня сформированности компетенций, соответствия работы предъявляемым требованиям, содержание и структура работы, степень самостоятельности, теоретическая и практическая значимость выводов и предложений, а также уровень грамотности (общий и специальный). В отзыве отмечаются положительные качества работы и недостатки.

Если, по мнению руководителя, курсовая работа заслуживает неудовлетворительной оценки и подлежит переработке, то в отзыве указываются недостатки, которые следует устранить и/или доработать. После устранения недостатков работа представляется на повторную проверку. Если представляется несколько курсовых работ с идентичным содержанием, что не отражает степень самостоятельности выполнения работы, все эти работы возвращаются исполнителям на переработку и повторное рецензирование, после чего обучающийся приступает к процедуре защиты работы.

Защита курсовой работы на комиссии в составе руководителя курсовой работы и одного или двух преподавателей кафедры может быть организована разными методами: индивидуально или группой, с привлечением оппонентов из числа студентов.

По-возможности, защита курсовой работы должна проводиться публично в присутствии группы.

Руководитель работы определяет требования к содержанию и продолжительности доклада при защите, устанавливает регламент для оппонентов.

Защита курсовой работы, как правило, состоит в коротком докладе (5 - 7 мин) студента и ответах на вопросы по существу проекта. За такое время можно представить примерно 2 - 3 страницы стандартного машинописного текста.

Доклад представляет конспект выступления студента. В докладе студент должен раскрыть цель и задачи курсового проекта, его актуальность, кратко изложить содержание, делая акцент на выводах. Основное требование к докладу - обеспечение логической последовательности между разделами, подчиненной четкому и полному изложению цели.

Желательно к защите подготовить презентационный материал, поскольку он помогает студенту успешно защищать свою работу, свои идеи и представлять их в самом выгодном свете. Представленная презентация должна быть последовательно и жестко связана с содержанием доклада.

Все слайды, используемые на защите, должны быть представлены в пояснительной записке как последнее приложение.

Презентацию следует оформлять в Microsoft PowerPoint.

Для того чтобы такая презентация действительно делала выступление более эффективным, необходимо соблюдение определенных требований. Эти требования кажутся очевидными. Тем не менее, практика показывает, что они часто нарушаются не только начинающими, но и квалифицированными пользователями и специалистами в области информационных технологий.

Во-первых, информация на слайдах должна хорошо читаться. Это обеспечивается выбором оптимального цветового решения, размера и типа шрифта, объема и структуры информации на слайде.

Выбор цветового решения - соотношения цветов фона слайда и текста - диктуется условиями показа. Для демонстрации презентации на экране монитора или с помощью проектора в хорошо затемненном помещении вполне оправдан выбор ярких цветов, темного фона слайдов и светлого цвета текста. Типичной же ситуацией является отсутствие затемнения, поэтому оптимальным для электронной презентации является светлый фон слайдов и темный цвет текста.

Размер шрифта, позволяющий сделать текст приемлемым для чтения на экране, предусмотрен в шаблонах презентаций, поэтому имеет смысл размещать на слайде такой объем текста, который бы не приводил к автоматическому уменьшению размера шрифта.

Кроме того, нужно помнить, что на экране лучше воспринимаются шрифты без засечек (такие как, например, *Tahoma*, *Verdana*, *Arial*), поэтому использовать привычный для печатных текстов шрифт *Times New Roman* в презентациях не рекомендуется, также как не рекомендуется использовать курсивное начертание.

Текст презентации не должен служить конспектом для докладчика. В презентациях, иллюстрирующих публичное выступление, каковым является защита курсового проекта, текст должен быть свернут до ключевых слов и фраз. Полные развернутые предложения на слайдах таких презентаций используются только при необходимости цитирования. Текст каждого слайда должен быть кратким, но содержательным и хорошо структурированным.

Списки на слайдах не должны включать более 5-7 элементов. Если элементов списка все-таки больше, их лучше расположить в две колонки.

В таблицах не должно быть более 4 строк и 4 столбцов - в противном случае данные в таблице будет просто невозможно увидеть. Ячейки с названиями строк и столбцов и наиболее значимые данные рекомендуется выделять цветом.

Гистограммы не должны включать более 4 категорий, а организационные диаграммы - более 5 элементов.

Если требуются более объемные таблицы и диаграммы, лучше подготовить их для раздаточных материалов с помощью других программ.

Во-вторых, выбранные средства визуализации должны быть адекватными содержанию.

Информация, которая плохо воспринимается на слух - даты, имена, новые термины, названия - должна быть обязательно представлена на слайдах.

При графическом представлении информации должны использоваться адекватные средства визуализации, т.е. подбираться соответствующие содержанию типы графиков и диаграмм, иллюстрации, таблицы.

Средства динамического представления информации (перемещение или одновременное появление фрагментов текста и графических объектов, другие анимационные эффекты и эффекты смены слайдов) должны служить для дозирования информации, привлечения внимания слушателей к той ее части, о которой идет речь в определенный момент выступления, и показа явлений в динамике.

В презентации должно быть не менее 5 слайдов. Количество слайдов определяется исходя из потребностей доклада. Количество и содержание

слайдов должны быть адекватны содержанию и продолжительности выступления.

Избыточное количество слайдов приводит не только к нарушению регламента, но и к утомлению слушателей и рассеиванию их внимания.

Все слайды (кроме первого и последнего, которые не включаются и в приложения пояснительной записки) должны быть пронумерованы в порядке следования. После номера ставится точка ".", пробел " " и с большой буквы приводится заголовок слайда без точки на конце, например: "1. Наименование слайда". Если логически однородный материал разбивается на несколько слайдов, то в их нумерации используют русские буквы, например: "2.А. Наименование слайда", "2.Б. Наименование слайда". Все заголовки слайда должны выделяться большим размером шрифта, жирностью и цветом. Слайды могут содержать подзаголовки.

Первый слайд презентации является слайдом - приветствием, на котором необходимо разместить:

- наименование ВУЗа, в котором обучается студент;
- вид работы (Курсовой проект);
- тема курсового проекта;
- наименование выпускающей кафедры;
- фамилия, имя, отчество студента;
- шифр группы, в которой обучается студент;
- фамилия, имя и отчество руководителя курсового проектирования с указанием его научной степени и звания;
- место защиты и год защиты.

Последний слайд сообщает о завершении доклада. На нем помещаются фразы «Спасибо за внимание. Доклад окончен».

Необходимо помнить, что чрезмерное увлечение дизайном, звуковым сопровождением, анимацией может отвлечь слушателей от сути выступления.

Курсовая работа оценивается по 100-балльной шкале с выставлением оценки по пятибалльной системе. Оценка зависит от качества выполнения и защиты курсового проекта. Критерии оценки качества исполнения курсовой работы приведены в таблице 3.

Таблица 3 – Критерии оценки качества исполнения курсового проекта
(*Оработка*)

№ п/п	Критерии оценки	Оценка работы (по 100-балльной шкале)	
		min количест	max количеств

		во баллов	о баллов
1.	Соответствие содержания курсовой работы утвержденной теме	0-10	10
2.	Выполнение поставленных целей и задач	0-5	5
3.	Оценка работы в разрезе структурных элементов	0-28	28
3.1	Введение	0-5	5
3.2	Основная часть	0-32	32
3.5	Заключение	0-6	6
4.	Общая характеристика работы (сбалансированность по объему теоретической, аналитической и проектной частей, правильность интерпретации результатов)	0-4	4
5.	Оформление работы	0-10	10
ИТОГОВАЯ ОЦЕНКА по курсовой работе		0-100	100

Защита курсовой работы – обязательная процедура, которая оказывает существенное влияние на выставление итоговой оценки проведенного исследования. Качество исполнения курсовой работы оценивается руководителем (Оработка), а результаты защиты либо только руководителем, либо членами комиссии по защите курсовых работ (Озащита) (если это решение было принято на заседании кафедры).

Оценка руководителя ставится на основании отзыва на курсовую работу. Качество исполнения и защиты курсовой работы оценивается по 100-балльной (рейтинговой) системе в соответствии со следующими критериями оценки (таблица 4).

Таблица 4 – Критерии оценки защиты курсового проекта (Озащита)

№ п/п	Критерии оценки	Оценка защиты курсовой работы (по 100-балльной шкале)
1.	Владение содержанием курсового проекта	30
2.	Логическая последовательность изложения материала	5
3.	Краткость изложения работы	5
4.	Умение вычленить главную мысль работы	10
3.	Умение обосновать собственный вклад в работу	20
4	Полнота и грамотность ответов на вопросы при защите	20
5	Наличие подготовленного иллюстрационного материала	10
ИТОГОВАЯ ОЦЕНКА по защите курсовой работы		100

Итоговая оценка за курсовую работу рассчитывается по формуле:

$$O_{итог} = 0,4 \cdot O_{работа} + 0,6 \cdot O_{защита}$$

Полученное количество баллов трансформируется в оценку и проставляется в зачетную книжку обучающегося и зачетную ведомость для курсовых работ.

Результаты выполнения и защиты курсовых работ определяются оценками:

90-100 баллов - «отлично»;

70-89 баллов - «хорошо»;

50-69 баллов - «удовлетворительно»;

0-49 баллов - «неудовлетворительно».

Результат защиты курсовой работы студента оценивается в форме зачета с оценкой (дифференцированного зачета) по бально-рейтинговой и пятибалльным системам. Кафедра разрабатывает критерии оценки, в соответствии с которыми устанавливается качество сформированности у студента компетенций, которые он должен приобрести при подготовке курсового проекта и продемонстрировать в ходе ее защиты, а также уровень знаний, владений (навыков), которые студент должен продемонстрировать для подтверждения освоенных компетенций.

Оценка зачтено (с оценкой «отлично»), зачтено (с оценкой «хорошо»), зачтено (с оценкой «удовлетворительно») проставляется в зачетную книжку студента и зачетную ведомость для защиты курсовых работ. Отрицательная оценка в зачетную книжку не вносится. Полное наименование курсовых работ вносится в зачетную книжку и в приложение к Курсовому проекту.

Студент, не представивший курсовой проект или получивший неудовлетворительную оценку, считается студентом, имеющим академическую задолженность по учебной дисциплине.

По решению кафедры для защиты курсовых работ может быть утверждена комиссия. Число членов комиссии для защиты курсового проекта должно составлять не более трех человек. Состав комиссии определяется заведующим кафедрой.

Повторная защита курсовых работ для обучающихся, которые по уважительной причине не вышли на защиту курсового проекта, назначается либо в период проведения зачетной недели, либо в дополнительную сессию по решению кафедры.

Хранение курсовых работы осуществляется согласно номенклатуре дел кафедры.

3.4 Порядок размещения в ЭБС и автоматизированной (компьютерной)

проверке на объем и характер заимствования курсовой работы

Проверка курсовых работ на объем и характер заимствования курсовых и выпускных квалификационных работ по направлениям подготовки/специальностям высшего образования является составной частью реализуемого в академии процесса контроля соблюдения академических норм при выполнении и защите курсовых работ.

Проверка работ на наличие неправомерных заимствований осуществляется с помощью программных продуктов электронных систем проверки заимствований.

При наличии в курсовой работе менее 45% оригинального текста, она отправляется на доработку при сохранении ранее установленной темы и после этого подвергается повторной проверке.

При повторной проверке курсовой работы, имеющая менее 45% оригинального текста, в течение 3-х дней должна быть доработана при сохранении ранее установленной темы и после этого подвергается окончательной проверке. Если после проведения научным руководителем окончательной проверки уровень оригинальности не достигает установленного минимального рубежа в 45%, курсовой проект не допускается к защите.

Итоговая проверка курсовой работы осуществляется с помощью программных продуктов электронных систем проверки заимствований должна быть выполнена за месяц до начала экзаменационной сессии и сдачи экзамена по дисциплине, по которой учебным планом предусмотрено выполнение курсового проекта.

Обучающийся, не допущенный к защите курсового проекта, считается имеющим академическую задолженность по дисциплине, по которой предусмотрено выполнение курсового проекта.

Все курсовые работы обучающихся (полный текст) подлежат загрузке в электронную среду академии.

Доступ лиц к текстам курсовых работ должен быть обеспечен в соответствии с законодательством Российской Федерации, с учетом изъятия производственных, технических, экономических, организационных и других сведений, в том числе о результатах интеллектуальной деятельности в научно-технической сфере, о способах осуществления профессиональной деятельности, которые имеют действительную или потенциальную коммерческую ценность в силу неизвестности их третьим лицам, в соответствии с решением правообладателя.

После проведения проверок курсовой работы научным руководителем формируется справка-заключение о проверке на наличие незаконных заимствований и прикладывается к курсовой работе (Приложение Е).

4 Условия обучения лиц с ограниченными возможностями здоровья

Для студентов из числа лиц с ограниченными возможностями здоровья обучение проводится Академией с учетом особенностей их психофизического развития, их индивидуальных возможностей и состояния здоровья.

При проведении обучения по дисциплине обеспечивается соблюдение следующих общих требований:

– проведение обучения для лиц с ограниченными возможностями здоровья в одной аудитории совместно со студентами, не имеющими ограниченных возможностей здоровья, если это не создает трудностей для них в процессе обучения;

– присутствие в аудитории ассистента, оказывающего обучающимся необходимую техническую помощь с учетом их индивидуальных особенностей (занять рабочее место, передвигаться, прочесть и оформить задание, общаться с преподавателем);

– пользование необходимыми обучающимся техническими средствами при выполнении практических и других работ в соответствии с учебным планом с учетом их индивидуальных особенностей;

В зависимости от индивидуальных особенностей обучающихся с ограниченными возможностями здоровья образовательная среда Академии обеспечивает выполнение следующих требований при обучении и проведении промежуточной и итоговой аттестации:

а) для слепых:

– задания и иные материалы для аттестации зачитываются ассистентом;

– письменные задания надиктовываются обучающимся ассистенту;

б) для слабовидящих:

увеличенным шрифтом;

– обеспечивается индивидуальное равномерное освещение не менее 300

– люкс;

– при необходимости обучающимся предоставляется увеличивающее устройство, допускается использование увеличивающих устройств, имеющихся у обучающихся;

в) для глухих и слабослышащих, с тяжелыми нарушениями речи:

– обеспечивается наличие звукоусиливающей аппаратуры коллективного пользования, при необходимости обучающимся предоставляется звукоусиливающая аппаратура индивидуального пользования;

– по их желанию аттестационные испытания проводятся в письменной форме;

г) для лиц с нарушениями опорно-двигательного аппарата (тяжелыми нарушениями двигательных функций верхних конечностей или отсутствием верхних конечностей):

- письменные задания надиктовываются ассистенту;
- по их желанию все аттестационные испытания проводятся в устной форме.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

Основная литература

1. Технологии обеспечения безопасности информационных систем : учебное пособие / А. Л. Марухленко, Л. О. Марухленко, М. А. Ефремов [и др.]. – Москва ; Берлин : Директ-Медиа, 2021. – 210 с. – URL: <https://biblioclub.ru/index.php?page=book&id=598988>. – Режим доступа: по подписке. – Текст : электронный.

2. Марухленко, А. Л. Разработка защищённых интерфейсов Web-приложений : учебное пособие / А. Л. Марухленко, Л. О. Марухленко, М. А. Ефремов. – Москва ; Берлин : Директ-Медиа, 2021. – 175 с. – URL: <https://biblioclub.ru/index.php?page=book&id=599050>. – Режим доступа: по подписке. – Текст : электронный.

3. Кобылянский, В. Г. Операционные системы, среды и оболочки : учебное пособие / В. Г. Кобылянский. – Новосибирск : Новосибирский государственный технический университет, 2018. – 80 с. – URL: <https://biblioclub.ru/index.php?page=book&id=576354>. – Режим доступа: по подписке. – Текст : электронный.

4. Основы администрирования информационных систем : учебное пособие / Д. О. Бобынцев, А. Л. Марухленко, Л. О. Марухленко [и др.]. – Москва ; Берлин : Директ-Медиа, 2021. – 202 с. – URL: <https://biblioclub.ru/index.php?page=book&id=598955>. – Режим доступа: по подписке. – Текст : электронный.

5. Ищейнов, В. Я. Информационная безопасность и защита информации : теория и практика : учебное пособие / В. Я. Ищейнов. – Москва ; Берлин : Директ-Медиа, 2020. – 271 с. – URL: <https://biblioclub.ru/index.php?page=book&id=571485>. – Режим доступа: по подписке. – Текст : электронный.

6. Криптографическая защита информации : учебное пособие / С.О. Крамаров, О.Ю. Митясова, С.В. Соколов [и др.] ; под ред. С.О. Крамарова. — Москва : РИОР : ИНФРА-М, 2023. — 321 с. — (Высшее образование). — DOI: <https://doi.org/10.12737/1716-6>. - ISBN 978-5-369-01716-6. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1899016>. – Режим доступа: по подписке. – Текст : электронный.

7. Петров, А. А. Компьютерная безопасность. Криптографические методы защиты : практическое руководство / А. А. Петров. - 2-е изд. - Москва : ДМК Пресс, 2023. - 451 с. - ISBN 978-5-89818-453-7. - Текст : электронный. - URL: <https://znanium.com/catalog/product/2106222>. - Режим доступа: по подписке. - Текст : электронный.

8. Вострецова, Е. В. Основы информационной безопасности : учебное пособие / Е. В. Вострецова. - Екатеринбург : Изд-во Уральского ун-та, 2019. - 204 с. - ISBN 978-5-7996-2677-8. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1936350>. - Режим доступа: по подписке. - Текст : электронный.

9. Подсобляева, О. В. Безопасность информационных систем и баз данных : практикум / О. В. Подсобляева. - 2-е изд., стер. - Москва : Флинта, 2022. - 99 с. - ISBN 978-5-9765-5148-0. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1906158>. - Режим доступа: по подписке. - Текст : электронный.

10. Методы и средства комплексной защиты информации в технических системах : учебное пособие / Э. В. Запонов, А. П. Мартынов, И. Г. Машин [и др.]. - Саров : РФЯЦ-ВНИИЭФ, 2019. - 224 с. - ISBN 978-5-9515-0429-6. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1230827>. - Режим доступа: по подписке. - Текст : электронный.

11. Сычев, Ю. Н. Защита информации и информационная безопасность : учебное пособие / Ю.Н. Сычев. — Москва : ИНФРА-М, 2023. — 201 с. — (Высшее образование: Бакалавриат). — DOI 10.12737/1013711. - ISBN 978-5-16-014976-9. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1912987>. - Режим доступа: по подписке. - Текст : электронный.

Дополнительная литература

1. Беловицкий, К. Б. Основы экономической безопасности : учебное пособие / К. Б. Беловицкий, Н. Ф. Кузовлева. - 2-е изд., доп. - Москва : Дашков и К, 2022. - 218 с. - ISBN 978-5-394-04816-6. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1926418>. - Режим доступа: по подписке. - Текст : электронный.

2. Карпухин, Е. О. Технологии и методы защиты инфокоммуникационных систем и сетей : учебное пособие для вузов / Е. О. Карпухин. - Москва : Горячая линия-Телеком, 2021. - 120 с. - ISBN 978-5-9912-0896-3. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1911612>. - Режим доступа: по подписке. - Текст : электронный.

3. Ларина, Т. Б. Операционные системы : учебно-методическое пособие / Т. Б. Ларина. - Москва : РУТ (МИИТ), 2018. - 58 с. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1895290>. - Режим доступа: по подписке. - Текст : электронный.

4. Чекулаева, Е. Н. Управление информационной безопасностью : учебное пособие / Е. Н. Чекулаева, Е. С. Кубашева. - Йошкар-Ола : Поволжский государственный технологический университет, 2020. - 154 с. - ISBN 978-5-8158-2165-1. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1894130>. - Режим доступа: по подписке. - Текст : электронный.

5. Гришина, Н. В. Основы моделирования процессов и систем защиты информации : учебное пособие / Н.В. Гришина. — Москва : ИНФРА-М, 2022. — 107 с. — (Высшее образование). - ISBN 978-5-16-110840-6. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1891122>. - Режим доступа: по подписке. - Текст : электронный.

6. Кришталюк, А. Н. Конфиденциальное делопроизводство и защита коммерческой тайны : курс лекций / А. Н. Кришталюк. - Орел : МАБИВ, 2018. - 199 с. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1514363>. - Режим доступа: по подписке. - Текст : электронный.

7. Козьминых, С. И. Организационное и правовое обеспечение информационной безопасности : учебное пособие / С. И. Козьминых. - Тбилиси : Справедливая Грузия, 2020. - 309 с. - ISBN 978-9941-9663-2-3. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1359091>. - Режим доступа: по подписке. - Текст : электронный.

8. Костин, В. Н. Методы и средства защиты компьютерной информации : криптографические методы для защиты информации : учебное пособие / В. Н. Костин. - Москва : Изд. Дом НИТУ «МИСиС», 2018. - 40 с. - ISBN 978-5-90695-334-6. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1232230>. - Режим доступа: по подписке. - Текст : электронный.

ПРИЛОЖЕНИЕ А
(обязательное)
Титульный лист курсового проекта

Негосударственное аккредитованное некоммерческое частное образовательное учреждение высшего образования «Академия маркетинга и социально-информационных технологий – ИМСИТ» (г. Краснодар)

(НАН ЧОУ ВО Академия ИМСИТ)

Институт информационных технологий и инноваций

Кафедра математики и вычислительной техники

КУРСОВОЙ РАБОТА

по дисциплине: «Безопасность систем баз данных»

на тему: РАЗРАБОТКА ЗАЩИЩЕННОЙ БАЗЫ ДАННЫХ
СОТРУДНИКОВ ОРГАНИЗАЦИИ

направление подготовки 10.03.01 Информационная безопасность

Направленность (профиль) образовательной программы

«Безопасность автоматизированных систем (по отрасли или в сфере профессиональной деятельности)»

Работу выполнил студент

4 курса очной формы

обучения, группы 24-ИБ-01 _____ А.В. Ермоленко

Научный руководитель:

канд. техн. наук, доцент _____ С.А. Капустин

Работа защищена с оценкой « _____ »

Краснодар 2023

ПРИЛОЖЕНИЕ Б

(обязательное)

Форма задания на курсовую работу

Негосударственное аккредитованное некоммерческое частное образовательное учреждение высшего образования «Академия маркетинга и социально-информационных технологий – ИМСИТ» (г. Краснодар)

Кафедра математики и вычислительной техники

УТВЕРЖДАЮ

Зав. кафедрой _____
« ____ » _____ 20__ г.

З А Д А Н И Е на курсовую работу

Студенту: _____ группы _____ курса
(Ф.И.О.) (№ группы и курса)

факультета _____

направления _____

(шифр и наименование)

Тема работы: _____

Содержание задания: _____

Объем работы:

а) пояснительная записка к работе _____ с.

б) задачи

Рекомендуемая литература: _____

Срок выполнения работы: с " ____ " _____ по " ____ " _____ 20__ г.

Срок защиты: " ____ " _____ 20__ г.

Дата выдачи задания: " ____ " _____ 20__ г.

Дата сдачи проекта на кафедру: " ____ " _____ 20__ г.

Руководитель работы _____

(подпись, ф.и.о., звание, степень)

Задание принял студент _____

(подпись, дата)

ПРИЛОЖЕНИЕ В (справочное)

Примеры библиографических описаний документов

Статья в периодических изданиях и сборниках статей:

1 Гуреев В.Н., Мазов Н.А. Использование библиометрии для оценки значимости журналов в научных библиотеках (обзор)//Научно-техническая информация. Сер. 1. - 2015. - N 2. - С. 8 - 19.

2 Колкова Н.И., Скипор И.Л. Терминосистема предметной области "электронные информационные ресурсы": взгляд с позиций теории и практики//Научн. и техн. б-ки. - 2016. - N 7. - С. 24 - 41.

Книги, монографии:

1 Земсков А.И., Шрайберг Я.Л. Электронные библиотеки: учебник для вузов. - М: Либерея, 2003. - 351 с.

2 Костюк К.Н. Книга в новой медицинской среде. - М.: Директ-Медиа, 2015. - 430 с.

Тезисы докладов, материалы конференций:

1 Леготин Е.Ю. Организация метаданных в хранилище данных//Научный поиск. Технические науки: Материалы 3-й науч. конф. аспирантов и докторантов/отв. за вып. С.Д. Ваулин; Юж.-Урал. гос. ун-т. Т. 2. - Челябинск: Издательский центр ЮУрГУ, 2011. - С. 128 - 132.

2 Антопольский А.Б. Система метаданных в электронных библиотеках//Библиотеки и ассоциации в меняющемся мире: Новые технологии и новые формы сотрудничества: Тр. 8-й Междунар. конф. "Крым-2001"/г. Судак, (июнь 2001 г.). - Т. 1. - М., 2001, - С. 287 - 298.

3 Парфенова С.Л., Гришакина Е.Г., Золотарев Д.В. 4-я Международная научно-практическая конференция "Научное издание международного уровня - 2015: современные тенденции в мировой практике редактирования, издания и оценки научных публикаций"//Наука. Инновации. Образование. - 2015. - N 17. - С. 241 - 252.

Патентная документация согласно стандарту ВОИС:

1 ВУ (код страны) 18875 (N патентного документа) С1 (код вида документа), 2010 (дата публикации).

Электронные ресурсы:

1 Статистические показатели российского книгоиздания в 2006 г.: цифры и рейтинги [Электронный ресурс]. - 2006. - URL: http://bookhamber.ru/stat_2006.htm (дата обращения 12.03.2009).

2 Прогноз научно-технологического развития Российской Федерации на период до 2030 года. - URL:

<http://government.ru/media/files/41d4b737638891da2184/pdf> (дата обращения 15.11.2016).

3 Web of Science. - URL: <http://apps.webofknowledge.com/>(дата обращения 15.11.2016).

Нормативные документы:

1. ГОСТ 7.0.96-2016 Система стандартов по информации, библиотечному и издательскому делу. Электронные библиотеки. Основные виды. Структура. Технология формирования. - М.: Стандартинформ, 2016. - 16 с.

2 Приказ Минобразования РФ от 19 декабря 2013 г. N 1367 "Об утверждении Порядка организации и осуществления образовательной деятельности по образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры". - URL: http://www.consultant.ru/document/cons_doc_LAW_159671/(дата обращения: 04.08.2016).

3 ISO 25964-1:2011. Information and documentation - Thesauri and interoperability with other vocabularies - Part 1: Thesauri for information retrieval. - URL: http://www.iso.org/iso/catalogue_detail.htm?csnumber53657 (дата обращения: 20.10.2016).

ПРИЛОЖЕНИЕ Г
(обязательное)

Пример оформления реферата

РЕФЕРАТ

Курсовая работа 44 с., 15 рис., 1 табл., 20 источн., 1 прил.

АВТОМАТИЗИРОВАННЫЕ ИНФОРМАЦИОННЫЕ СИСТЕМЫ, БАЗЫ ДАННЫХ, СИСТЕМА ЗАЩИТЫ ИНФОРМАЦИИ, СРЕДСТВА ЗАЩИТЫ, РИСКИ, ТЕХНИЧЕСКОЕ ЗАДАНИЕ, УГРОЗЫ, ВЫБОР СРЕДСТВ ЗАЩИТЫ, МОДЕЛЬ УГРОЗ, ПРОЕКТИРОВАНИЕ

Объектом исследования является база данных организации.

Цель работы: разработка защищенной базы данных организации, на базе комплекса локальных сетей, соединенных каналами передачи данных коллективного пользования.

Методы исследования: экспериментальный, методы системного анализа, методы аналогий и сравнений.

Основные результаты: разработана защищенная база данных организации на базе комплекса локальных сетей, соединенных каналами передачи данных коллективного пользования.

Область применения системы – организации с ограничением пользователей в допуске к различным разделам информационной базы для комплекса локальных сетей, соединенных каналами передачи данных коллективного пользования.

Разработанный проект готов для внедрения при разработке системы защиты автоматизированной информационной системы организации.

Эффективность разработки заключается в разработке оригинальной защиты базы данных для автоматизированной информационной системы организации.

Предметом дальнейшего исследования является разработка системы защиты автоматизированной информационной системы организации.

ПРИЛОЖЕНИЕ Д
(обязательное)

Образец отзыва руководителя на курсовой проект
Негосударственное аккредитованное некоммерческое частное образовательное
учреждение высшего образования "Академия маркетинга и социально-
информационных технологий - ИМСИТ" (г. Краснодар)

Кафедра Математики и вычислительной техники

**РЕЦЕНЗИЯ РУКОВОДИТЕЛЯ
на курсовую работу обучающегося
по дисциплине Безопасность систем баз данных**

Фамилия, имя, отчество обучающегося

Тема курсовой работы: _____

Регистрационный номер _____ **Курс** _____ **Группа** _____

Направление подготовки: _____
код, направление

**Сформированность компетенций у обучающегося по итогам выполнения
заданий на курсовую работу**

Задания*	Уровень сформированности компетенций
1) разработка модели угроз;	
2) выбор средств защиты;	
3)	

*Сформулировать задания в соответствии с содержанием курсового проекта

Соответствие курсового проекта требованиям

Наименование требования	Заключение о соответствии требованиям (отметить «соответствует», «соответствует не в полной мере», или «не соответствует»)
1. Актуальность темы	
2. Соответствие содержания теме	
3. Полнота, глубина и обоснованность решения поставленных задач	
4. Корректность проектных решений	

5. Практическая значимость	
6. Оценка личного вклада автора	
7. Наглядность (информативность) представления результатов исследования	

Достоинства содержательной части курсового проекта:

Ошибки и недостатки содержательной части курсового проекта:

Общее заключение научного руководителя о соответствии курсового проекта требованиям, установленным федеральным государственным образовательным стандартом и основной профессиональной образовательной программой по направлению подготовки 10.03.01 Информационная безопасность

Курсовой проект соответствует требованиям ФГОС и ОПОП по направлению 10.03.01 Информационная безопасность

Обобщенная оценка содержательной части курсового проекта*

Соответствует _____
*соответствует / частично соответствует / не соответствует

Руководитель:
 Полное наименование должности и основного
 места работы, ученая степень, ученое звание

_____ _____
 Подпись расшифровка подписи

« ____ » _____ 20__ г.

ПРИЛОЖЕНИЕ Е
(обязательное)
Образец заключения

Негосударственное аккредитованное некоммерческое частное образовательное учреждение высшего образования "Академия маркетинга и социально-информационных технологий - ИМСИТ" (г. Краснодар)

Кафедра Математики и вычислительной техники

ЗАКЛЮЧЕНИЕ

ФАМИЛИЯ ИМЯ ОТЧЕСТВО ОБУЧАЮЩЕГОСЯ

.....

.....
Форма обучения _____ Курс _____ Группа _____

НАПРАВЛЕНИЕ/СПЕЦИАЛЬНОСТЬ _____

Курсовая работа по дисциплине _____,

выполненная на тему: _____

_____ в соответствии с «Положением о порядке размещения в ЭБС и автоматизированной (компьютерной) проверке на объем и характер заимствования курсовых и выпускных квалификационных работ по направлениям подготовки/специальностям высшего образования» курсовая работа по дисциплине _____ **прошла** автоматизированный анализ в Программной системе для обнаружения заимствований в учебных и научных работах.

Доля авторского текста (оригинальности) в результате автоматизированной проверки составила _____ %.

Зав. кафедрой
математики и вычислительной
техники, к.э.н., доцент

Н.П. Исикова

« _____ » _____