

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Саратовский государственный технический университет
имени Гагарина Ю.А.»

Филиал федерального государственного бюджетного образовательного
учреждения высшего образования
«Саратовский государственный технический университет
имени Гагарина Ю.А.» в г. Петровске

УТВЕРЖДАЮ
Директор филиала СГТУ
имени Гагарина Ю.А. в г. Петровске
Е.А. Бесшапошникова
«06» июня 2024 г.



МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ВЫПОЛНЕНИЮ ПРАКТИЧЕСКИХ РАБОТ

по дисциплине

МДК.11.01 «Технология разработки и защиты баз данных»

направление подготовки

09.02.07 «Информационные системы и программирование»

Методические указания рассмотрены
на заседании предметной (цикловой)
комиссии общепрофессиональных
дисциплин, профессиональных модулей
специальностей технического профиля
«14» июня 2024 года, протокол №12

Председатель ПЦК Табарова /Ю.А.Табарова/

Петровск 2024

Пояснительная записка

Методические указания по выполнению практических работ подготовлены на основе рабочей программы учебной дисциплины МДК.11.01 «Технология разработки и защиты баз данных», разработанной на основе ФГОС СПО по специальности 09.02.07 «Информационные системы и программирование» и соответствующих общих (ОК) и профессиональных (ПК) компетенций:

ОК 01. Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам.

ОК 02. Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности

ОК 03. Планировать и реализовывать собственное профессиональное и личностное развитие, предпринимательскую деятельность в профессиональной сфере, использовать знания по финансовой грамотности в различных жизненных ситуациях.

ОК 04. Эффективно взаимодействовать и работать в коллективе и команде.

ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста

ОК 06. Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей, в том числе с учетом гармонизации межнациональных и межрелигиозных отношений, применять стандарты антикоррупционного поведения

ОК 07. Содействовать сохранению окружающей среды, ресурсосбережению, применять знания об изменении климата, принципы бережливого производства, эффективно действовать в чрезвычайных ситуациях

ОК 08. Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности

ОК 09. Пользоваться профессиональной документацией на государственном и иностранном языках.

ОК 10. Использовать знания по финансовой грамотности, планировать предпринимательскую деятельность в профессиональной сфере

ПК 11.1. Осуществлять сбор, обработку и анализ информации для проектирования баз данных

ПК 11.2. Проектировать базу данных на основе анализа предметной области

ПК 11.3. Разрабатывать объекты базы данных в соответствии с результатами анализа предметной области

ПК 11.4. Реализовывать базу данных в конкретной системе управления базами данных

ПК 11.5. Администрировать базы данных

ПК 11.6. Защищать информацию в базе данных с использованием технологии защиты информации

При выполнении практических работ студент должен *знать*:

- основные положения теории баз данных, хранилищ данных, баз знаний;
- основные принципы структуризации и нормализации базы данных;
- основные принципы построения концептуальной, логической и физической модели данных; методы описания схем баз данных в современных системах управления базами данных;
- структуры данных систем управления базами данных, общий подход к организации представлений, таблиц, индексов и кластеров;
- методы организации целостности данных;
- способы контроля доступа к данным и управления привилегиями;
- основные методы и средства защиты данных в базах данных

При выполнении практических работ студент должен *уметь*:

- работать с современными CASE-средствами проектирования БД;
- проектировать логическую и физическую схемы базы данных;
- создавать хранимые процедуры и триггеры на базах данных;
- применять стандартные методы для защиты объектов базы данных;
- выполнять стандартные процедуры резервного копирования и мониторинга выполнения этой процедуры;
- выполнять процедуру восстановления базы данных и вести мониторинг выполнения этой процедуры;
- обеспечивать информационную безопасность на уровне базы данных

Содержание практических занятий определено рабочей программой и тематическим планированием, соответствует теоретическому материалу изучаемых разделов учебной дисциплины. Объем практических занятий по дисциплине определяется учебным планом по данной специальности. Продолжительность практического занятия – 2 академических часа. Перед проведением практического занятия преподавателем организуется инструктаж, а по ее окончании – обсуждение итогов.

Комплект методических указаний по выполнению практических работ по дисциплине МДК.11.01 «Технология разработки и защиты баз данных» содержит 21 практических занятий.

Перечень практических работ

по дисциплине МДК.11.01 «Технология разработки и защиты баз данных»

ПРАКТИЧЕСКАЯ РАБОТА № 1.

Тема: Сбор и анализ данных

ПРАКТИЧЕСКАЯ РАБОТА № 2.

Тема: Сбор и анализ данных

ПРАКТИЧЕСКАЯ РАБОТА № 3.

Тема: Проектирование реляционной схемы базы данных в среде СУБД.

ПРАКТИЧЕСКАЯ РАБОТА № 4.

Тема: Проектирование реляционной схемы базы данных в среде СУБД.

ПРАКТИЧЕСКАЯ РАБОТА № 5.

Тема: Проектирование реляционной схемы базы данных в среде СУБД.

ПРАКТИЧЕСКАЯ РАБОТА № 6.

Тема: Проектирование реляционной схемы базы данных в среде СУБД.

ПРАКТИЧЕСКАЯ РАБОТА № 7.

Тема: Проектирование реляционной схемы базы данных в среде СУБД.

ПРАКТИЧЕСКАЯ РАБОТА № 8.

Тема: Создание БД в среде разработки

ПРАКТИЧЕСКАЯ РАБОТА № 9.

Тема: Создание БД в среде разработки

ПРАКТИЧЕСКАЯ РАБОТА № 10.

Тема: Создание БД в среде разработки

ПРАКТИЧЕСКАЯ РАБОТА № 11.

Тема: Создание БД в среде разработки

ПРАКТИЧЕСКАЯ РАБОТА № 12.

Тема: Создание БД в среде разработки

ПРАКТИЧЕСКАЯ РАБОТА № 13.

Тема: Экспорт данных базы в документы пользователя

ПРАКТИЧЕСКАЯ РАБОТА № 14.

Тема: Экспорт данных базы в документы пользователя

ПРАКТИЧЕСКАЯ РАБОТА № 15.

Тема: Импорт данных пользователя в базу данных

ПРАКТИЧЕСКАЯ РАБОТА № 16.

Тема: Импорт данных пользователя в базу данных

ПРАКТИЧЕСКАЯ РАБОТА № 17.

Тема: Выполнение настроек для автоматизации обслуживания БД

ПРАКТИЧЕСКАЯ РАБОТА № 18

Тема: Выполнение резервного копирования

ПРАКТИЧЕСКАЯ РАБОТА № 19

Тема: Восстановление БД из резервной копии

ПРАКТИЧЕСКАЯ РАБОТА № 20.

Тема: Реализация доступа пользователей к БД

ПРАКТИЧЕСКАЯ РАБОТА № 21.

Тема: Мониторинг безопасности работы с базами данных

ИНСТРУКЦИИ ДЛЯ ОБУЧАЮЩИХСЯ ПО ВЫПОЛНЕНИЮ ПРАКТИЧЕСКИХ РАБОТ

Прежде чем приступить к выполнению заданий, внимательно прочитайте данные рекомендации. Практические работы включают в себя задания следующих видов.

1. Работа за компьютером

В ходе выполнения практических работ студент должен:

- выполнять требования по охране труда
- соблюдать инструкцию по правилам и мерам безопасности в кабинете информационных технологий
- строго выполнять весь объем работы, указанный в задании
- соблюдать требования эксплуатации компьютерной техники (правила включения и выключения)
- предоставить отчет о проделанной работе по окончании выполненной работы, который должен содержать:

1. Название работы.
2. Цель работы.
3. Задание и его решение.
4. Вывод о проделанной работе.

Текст отчета по практической работе должен быть набран на компьютере шрифтом Times New Roman размером 14 пт. (при оформлении текста используется текстовый редактор MS Word). Шрифт, используемый в иллюстративном материале (таблицы и рисунки), рекомендуется уменьшить до 12 пт. Межстрочный интервал в основном тексте - полуторный. В иллюстративном материале межстрочный интервал рекомендуется сделать одинарным. Поля страницы должны быть: левое поле - 30 мм; правое поле – 15 мм; верхнее и нижнее поле - 20 мм.

Каждый абзац должен начинаться с красной строки. Отступ абзаца – 1,25 см от левой границы текста.

Студент должен выполнить практическую работу самостоятельно (или в группе, если это предусмотрено заданием). Практическая работа выполняется согласно заданию и методическим рекомендациям. После выполнения практической работы обучающийся самостоятельно себя контролирует путем ответов на вопросы. Результат работы представляется преподавателю в виде файла (файлов) в личном каталоге, защищается обучающимися.

По ходу выполнения работы при возникновении вопросов обучающийся может получить консультацию у преподавателя или самостоятельно воспользоваться лекционным материалом, рекомендуемой литературой.

2. Ответ на поставленные вопросы (с аргументацией)

Прочитайте вопрос и вникните в него.

Для удобства подчеркните ту, фразу, которая, по вашему мнению, является главной. Это поможет вам быстрее сориентироваться при ответе на вопрос.

Если вы считаете, что можете ответить на вопрос без помощи лекции и дополнительной литературы – приступайте. Если же вопрос заставляет вас

сомневаться, откройте лекционную тетрадь (учебник или дополнительную литературу), прочитайте необходимый пункт, вникните в содержание и после этого приступайте за работу.

ГЛАВНОЕ! Не переписывайте отрывки лекции в рабочую тетрадь! Четко отвечайте на ПОСТАВЛЕННЫЙ вопрос!

Не забудьте привести аргументацию (обоснование) вашей позиции, если вопрос предполагает личностное отношение к проблеме.

3. Заполнение таблиц и схем

Прочитайте название таблицы или схемы.

Исходя из названия, вы поймете цель предстоящей работы.

Воспользуйтесь материалами лекций или другими источниками, чтобы заполнить таблицу (схему).

Используйте цветные графические материалы для выделения строк, столбцов или элементов схем.

Особое внимание обращайтесь на четкость при отборе материала: делайте записи кратко и четко!

4. Поиск информации в сети — использование web-браузеров, баз данных, пользование информационно-поисковыми и информационно-справочными системами, автоматизированными библиотечными системами, электронными журналами. Поиск и обработка информации включает подготовку фрагмента практического занятия.

ПРАКТИЧЕСКАЯ РАБОТА № 1

Тема: Сбор и анализ данных

Цель работы: ознакомиться с процессом описания БД и получить навыки по использованию основных методов анализа данных.

Оборудование: ПК, интернет, программное обеспечение – MS Visio Professional, MS Word, инструкции по выполнению работы.

Содержание работы:

Задание 1.

1. Ознакомиться с предложенным вариантом описания предметной области (согласно заданию индивидуального задания).
2. Проанализировать предметную область, уточнив и дополнив ее, руководствуясь собственным опытом, консультациями и любыми источниками (книгами, учебниками или Интернет-источниками).
3. Выполнить структурное разбиение предметной области на отдельные подразделения (подсистемы) согласно выполняемым ими функциям.
4. Определить задачи и функции системы в целом и функции каждого подразделения (подсистемы).
5. Продумать подробное описание работы каждого подразделения (подсистемы), алгоритмов и сценариев выполнения ими отдельных работ. Продумать виды входной и выходной информации для каждого подразделения (подсистемы).
6. Описать схему работы будущей БД, учитывая выделенные и описанные ранее подсистемы.
7. Определить группу пользователей, для которой данная система будет более востребована. Описать перечень функций системы, которые будут доступны данной группе пользователей.
8. Создать физическую диаграмму в соответствии с описанием деятельности предметной области.

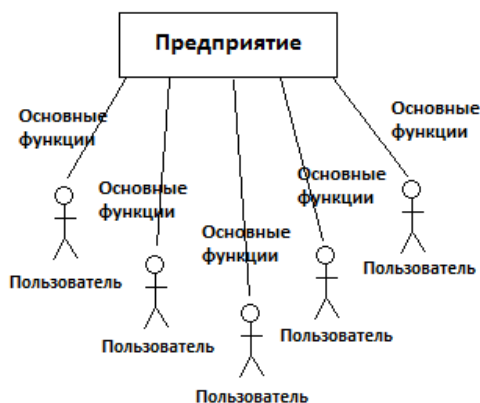


Рисунок 1 – Схематическое отображение физической диаграммы предметной области

9. Осуществить идентификацию опорных точек зрения, построив диаграмму идентификации точек зрения.

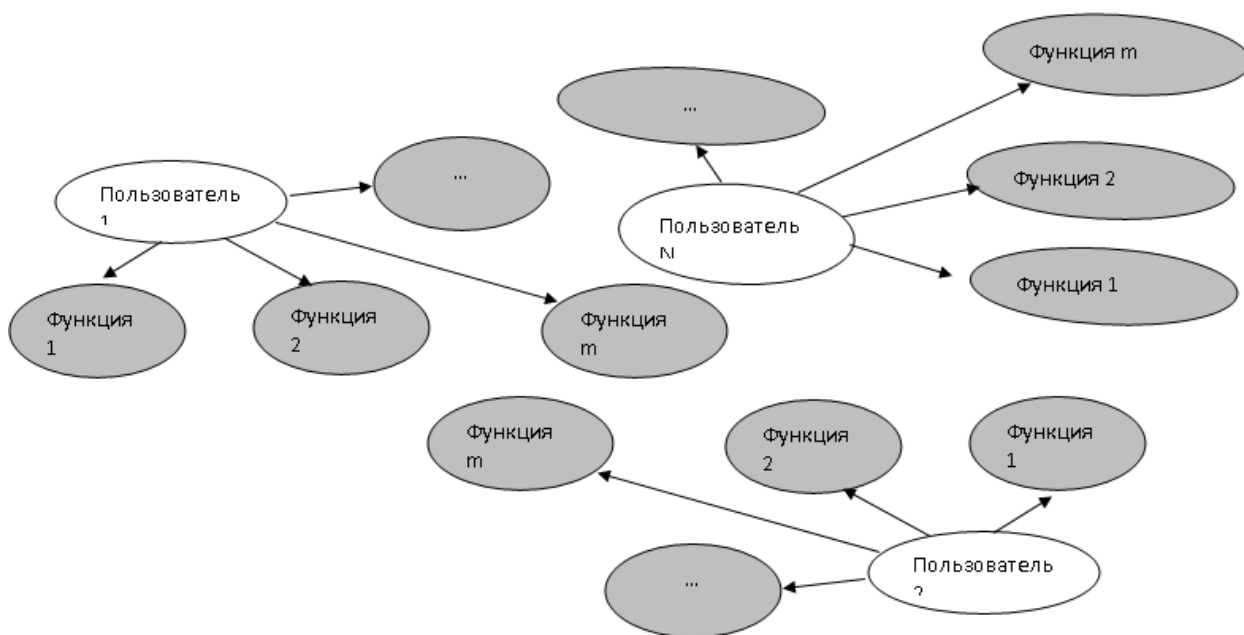


Рисунок 2 – Пример диаграммы идентификации точек зрения

10. Построить таблицу «Соотнесение точек зрения с выполняемыми функциями».

Точка зрения 1	Точка зрения 2	...	Точка зрения N
функция 1_1	функция 2_1		функция N_1
функция 1_2	функция 2_2		функция N_2
...
функция 1_m	функция 2_m		функция N_m

11. Расписать основные функциональные возможности администратора системы, как одного из пользователей системы.

12. Подготовить отчет по работе

Индивидуальные задания:

Вариант 1. Страховая медицинская компания.

Сущность задачи: Страховая медицинская компания (СМК) заключает договора добровольного медицинского страхования с населением и договора с лечебными учреждениями на лечение застрахованных клиентов. При возникновении страхового случая клиент подает заявку на оказание медицинских услуг по условиям договора инспектору, который работает с данным клиентом. Инспектор направляет данного клиента в лечебное учреждение. Отчеты о своей деятельности инспектор предоставляет в бухгалтерию. Бухгалтерия проверяет оплату договоров, перечисляет денежные средства за оказанные услуги лечебным учреждениям, производит отчисления в налоговые органы и предоставляет отчетность в органы государственной статистики. СМК не только оплачивает лечение застрахованного лица при возникновении с ним страхового случая, но и, при возникновении каких-либо осложнений после лечения, оплачивает лечение этих осложнений.

Вариант 2. Агентство недвижимости

Сущность задачи: Агентство недвижимости занимается покупкой, продажей, сдачей в аренду объектов недвижимости по договорам с их собственниками. Агентство управляет объектами недвижимости как

физических, так и юридических лиц. Собственник может иметь несколько объектов. В случае покупки или аренды клиент может произвести осмотр объекта. В качестве одной из услуг, предлагаемых агентством, является проведение инспектирования текущего состояния объекта для адекватного определения его рыночной цены. По результатам своей деятельности агентство производит отчисления в налоговые органы и предоставляет отчетность в органы государственной статистики.

Вариант 3. Компания по разработке программных продуктов

Сущность задачи: Компания заключает договор с клиентом на разработку программного продукта согласно техническому заданию. После утверждения технического задания определяется состав и объем работ, составляется предварительная смета. На каждый проект назначается ответственный за его выполнение – куратор проекта, который распределяет нагрузку между программистами и следит за выполнением технического задания. Когда программный продукт готов, то его внедряют, производят обучение клиента и осуществляют дальнейшее сопровождение. По результатам своей деятельности компания производит отчисления в налоговые органы и предоставляет отчетность в органы государственной статистики.

Вариант 4: Кадровое агентство

Сущность задачи: Кадровое агентство способствует трудоустройству безработных граждан. Агентство ведет учет и классификацию данных о безработных на основании резюме от них. От предприятий города поступают данные о свободных вакансиях, на основании которых агентство предлагает различные варианты трудоустройства соискателям. В случае положительного исхода поиска вакансия считается заполненной, а безработный становится трудоустроенным. По результатам своей деятельности кадровое агентство производит отчисления в налоговые органы и предоставляет отчетность в органы государственной статистики.

ПРАКТИЧЕСКАЯ РАБОТА № 2

Тема: Сбор и анализ данных

Цель работы: ознакомиться с процессом описания БД и получить навыки по использованию основных методов анализа данных.

Оборудование: ПК, интернет, программное обеспечение – MS Visio Professional, MS Word, инструкции по выполнению работы.

Содержание работы:

Задание 1.

1. Ознакомиться с предложенным вариантом описания предметной области (согласно заданию индивидуального задания).
2. Проанализировать предметную область, уточнив и дополнив ее, руководствуясь собственным опытом, консультациями и любыми источниками (книгами, учебниками или Интернет-источниками).
3. Выполнить структурное разбиение предметной области на отдельные подразделения (подсистемы) согласно выполняемым ими функциям.
4. Определить задачи и функции системы в целом и функции каждого подразделения (подсистемы).
5. Продумать подробное описание работы каждого подразделения (подсистемы), алгоритмов и сценариев выполнения ими отдельных работ. Продумать виды входной и выходной информации для каждого подразделения (подсистемы).
6. Описать схему работы будущей БД, учитывая выделенные и описанные ранее подсистемы.
7. Определить группу пользователей, для которой данная система будет более востребована. Описать перечень функций системы, которые будут доступны данной группе пользователей.
8. Создать физическую диаграмму в соответствии с описанием деятельности предметной области.

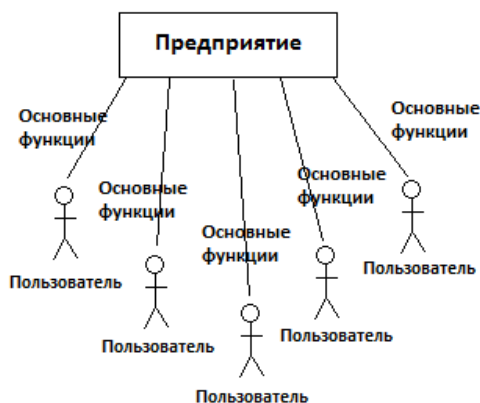


Рисунок 1 – Схематическое отображение физической диаграммы предметной области

9. Осуществить идентификацию опорных точек зрения, построив диаграмму идентификации точек зрения.

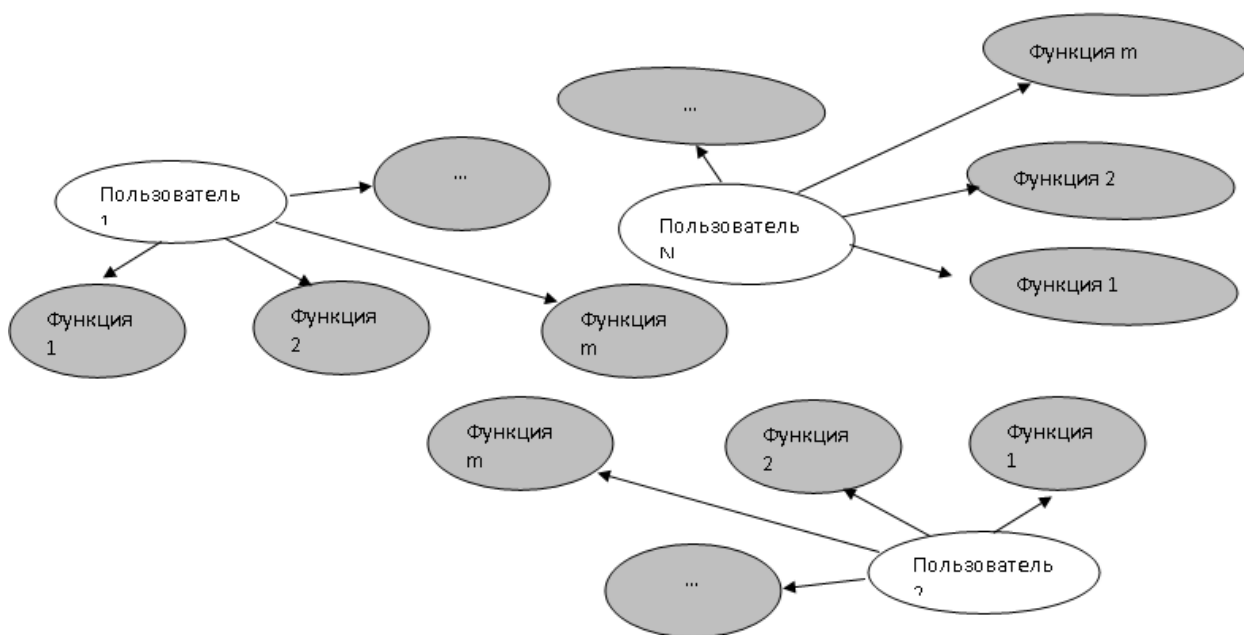


Рисунок 2 – Пример диаграммы идентификации точек зрения

10. Построить таблицу «Соотнесение точек зрения с выполняемыми функциями».

Точка зрения 1	Точка зрения 2	...	Точка зрения N
функция 1_1	функция 2_1		функция N_1
функция 1_2	функция 2_2		функция N_2
...
функция 1_m	функция 2_m		функция N_m

11. Расписать основные функциональные возможности администратора системы, как одного из пользователей системы.

12. Подготовить отчет по работе

Индивидуальные задания:

Вариант 1. Строительная организация

Сущность задачи: Строительная организация занимается строительством объектов по заказам клиентов. Сначала заказ проходит предварительную стадию: сбор различных разрешений на строительство, составление эскиза объекта, расчет объема и закупка строительных материалов. Сами строительные материалы доставляются на объект партиями. По мере поступления очередной партии стройматериалов закладывается фундамент объекта, строится каркас здания. По результатам данной работы происходит согласование с заказчиком, после чего утепляется контур, вставляются окна, устанавливается крыша. Далее идет обсуждение с клиентом внутренней отделки здания, закупаются отделочные материалы. После того, как объект проходит технический контроль, он передается заказчику. В дополнительные услуги строительной организации входят: услуги дизайнера по интерьеру, закупка и доставка мебели, сотрудничество с охранным предприятием по установке сигнализации. По результатам своей деятельности строительная организация производит отчисления в налоговые органы и предоставляет отчетность в органы государственной статистики.

Вариант 2. Компьютерная компания

Сущность задачи: Компьютерная компания занимается продажей, ремонтом, сборкой, тестированием компьютерной техники. Также, специалисты компании предоставляют услуги по разработке и монтажу локальных вычислительных сетей. Вся техника и комплектующие закупаются оптом у дилеров и хранятся на складе. Клиент, который хочет приобрести товар, оформляет заказ в торговом зале, а забирает технику со склада или оставляет заявку на ее доставку. Клиент, который хочет отремонтировать технику, приносит ее в сервисный отдел, откуда, по прошествии некоторого времени, забирает как от-ремонтированную или как технику, не подлежащую ремонту. По желанию клиента, специалисты компании могут выехать к клиенту для общей диагностики возникшей проблемы с техникой. По результатам своей деятельности компьютерная компания производит отчисления в налоговые органы и предоставляет отчетность в органы государственной статистики.

Вариант 3. Компания по предоставлению телекоммуникационных услуг

Сущность задачи: Компания занимается оказанием телекоммуникационных услуг абонентам. Клиент делает заявку на подключение к телекоммуникационным услугам и ему, по необходимости, устанавливают соответствующее оборудование. Оплата за услуги вносится путем авансовых платежей. Каждый факт предоставления услуги фиксируется соответствующим оборудованием и является основанием для списания соответствующей суммы с личного счета абонента. Клиент в любое время суток может получить отчет об оказанных ему услугах, их стоимости и остатку на личном счете абонента. По результатам своей деятельности компания производит отчисления в налоговые органы и предоставляет отчетность в органы государственной статистики.

Вариант 4. Спортивный комплекс

Сущность задачи: Спортивный комплекс предоставляет услуги по проведению спортивных тренировок. Тренировки, относящиеся к одному виду спорта, объединяются в спортивные секции. Клиент обращается в спортивный комплекс, где получает абонемент на посещение спортивной секции. На основе купленных абонементов составляется расписание тренировок на следующий месяц. Также, в зависимости от загруженности спортивного комплекса, распределяются тренеры спортивных секций. По результатам своей деятельности спортивный комплекс производит отчисления в налоговые органы и предоставляет отчетность в органы государственной статистики.

ПРАКТИЧЕСКАЯ РАБОТА № 3

Тема: Проектирование реляционной схемы базы данных в среде СУБД

Цель работы: научить проектировать реляционные схемы БД

Оборудование: ПК, интернет, программное обеспечение – MS Visio Professional, MS Word, инструкции по выполнению работы.

Справочный материал:

Процесс проектирования включает в себя следующие шаги:

1. Определение задач, стоящих перед базой данных.
2. Сбор и анализ документов, относящихся к исследуемой предметной области.
3. Описание особенностей Проектной организации, которые позволяют установить зависимости и связи между объектами (субъектами) предметной области.
4. Создание модели предметной области.
5. Определение групп пользователей и перечня задач, стоящих перед каждой группой.
6. Выбор аппаратной и программной платформы для реализации БД.
7. Выбор СУБД (системы управления базой данных).
8. Создание логической схемы БД.
9. Создание схем отношений, определение типов данных атрибутов и ограничений целостности.
10. Нормализация отношений (до третьей или четвёртой нормальной формы).
11. Определение прав доступа пользователей к объектам БД.
12. Написание текста создания основных объектов базы данных на языке SQL в синтаксисе выбранной СУБД (пользователи, таблицы и др.).
13. Написание текста создания вспомогательных объектов базы данных (представления, индексы, триггеры, роли и т.д.).

Эти шаги можно объединить в 5 этапов:

1. Инфологическое проектирование (1-5).
2. Определение требований к операционной обстановке, в которой будет функционировать информационная система (6).
3. Выбор системы управления базой данных (СУБД) и других инструментальных программных средств (7).
4. Логическое проектирование БД (8-11).
5. Физическое проектирование БД (12-13).

Содержание работы:

Задание 1. Спроектировать БД проектной организации. Основной вид деятельности такой организации – выполнение проектов по договорам с заказчиками.

1. Анализ предметной области. БД создаётся для информационного обслуживания руководства организации, руководителей проектов и участников проектов. БД должна содержать данные об отделах организации, сотрудниках и проектах. В соответствии с предметной областью система строится с учётом следующих особенностей:

– Каждый сотрудник работает в определённом отделе, в каждом отделе могут

работать несколько сотрудников.

- Каждый проект относится к определённому отделу, каждый отдел может отвечать за выполнение нескольких проектов.

- Каждый сотрудник может принимать участие в выполнении нескольких проектов, над каждым проектом может трудиться несколько сотрудников.

- Для каждого проекта назначается руководитель из числа сотрудников того отдела, к которому относится проект.

- Каждый проект должен быть выполнен в заданные сроки, каждый проект может состоять из нескольких этапов. Если проект состоит из одного этапа, то сроки его выполнения должны совпадать со сроками выполнения проекта в целом.

- Оклад сотрудника зависит от занимаемой должности, за участие в проектах сотрудник получает дополнительное вознаграждение.

- Виды участия сотрудников в проектах: руководитель, консультант, исполнитель.

- Каждый отдел занимает одно или несколько помещений (комнат), в каждом помещении может быть один или несколько стационарных телефонов.

Для создания ER-модели необходимо выделить сущности предметной области:

1) **Отделы.** Атрибуты: название, аббревиатура, комнаты, телефоны.

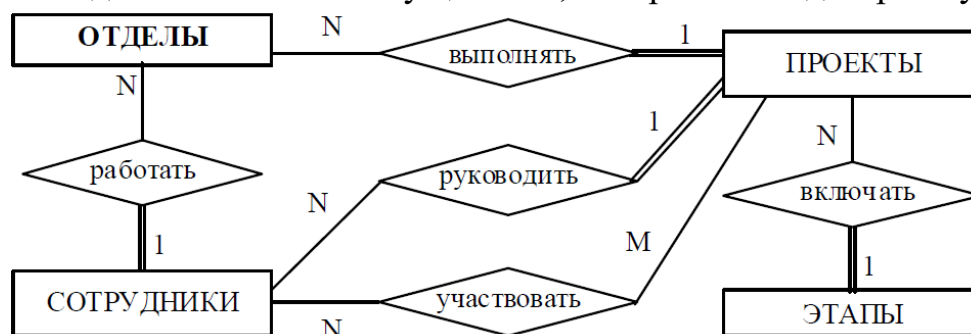
2) **Сотрудники.** Атрибуты: ФИО, паспортные данные, дата рождения, пол, ИНН (индивидуальный номер налогоплательщика), номер пенсионного страхового свидетельства, адреса, телефоны (рабочий, домашний, мобильный), данные об образовании (вид образования (высшее, средне-специальное и т.д.), специальность, номер диплома, дата окончания учебного заведения), должность, оклад, логин (имя пользователя).

Примечание: Логин потребуется нам для назначения дифференцированных прав доступа.

3) **Проекты.** Атрибуты: номер договора; полное название проекта; сокращённое название проекта; дата подписания договора; заказчик; контактные данные заказчика; дата начала проекта; дата завершения проекта; сумма по проекту; дата реальной сдачи проекта; сумма, полученная по проекту на текущую дату.

4) **Этапы проекта.** Атрибуты: номер по порядку, название, дата начала этапа, дата завершения этапа, форма отчетности, сумма по этапу, дата реальной сдачи этапа; сумма, полученная по этапу на текущую дату.

Исходя из выявленных сущностей, построим ER-диаграмму:



2. Анализ информационных задач и круга пользователей системы.

Определим группы пользователей, их основные задачи и запросы к БД:

1. Руководители организации:

- заключение новых договоров;
- назначение руководителей проектов;
- получение списка всех участников проектов;
- изменение должностных окладов и штатного расписания;
- получение полной информации о проектах;
- внесение изменений в данные о проектах;
- архивирование данных по завершённым проектам.

2. Руководитель проекта:

- назначение участников проекта;
- получение списка сотрудников, работающих над конкретным проектом;
- получение полной информации о проекте, руководителем которого он является;
- получение сведений о сотрудниках, которые могут стать участниками проекта;
- определение размера дополнительного вознаграждения сотрудников по конкретному проекту;
- внесение изменений в данные об этапах проекта.

3. Сотрудники отдела кадров:

- приём/увольнение сотрудников;
- внесение изменений в данные о сотрудниках.

4. Бухгалтеры:

- получение ведомости на выплату зарплаты.

5. Сотрудники – участники проектов:

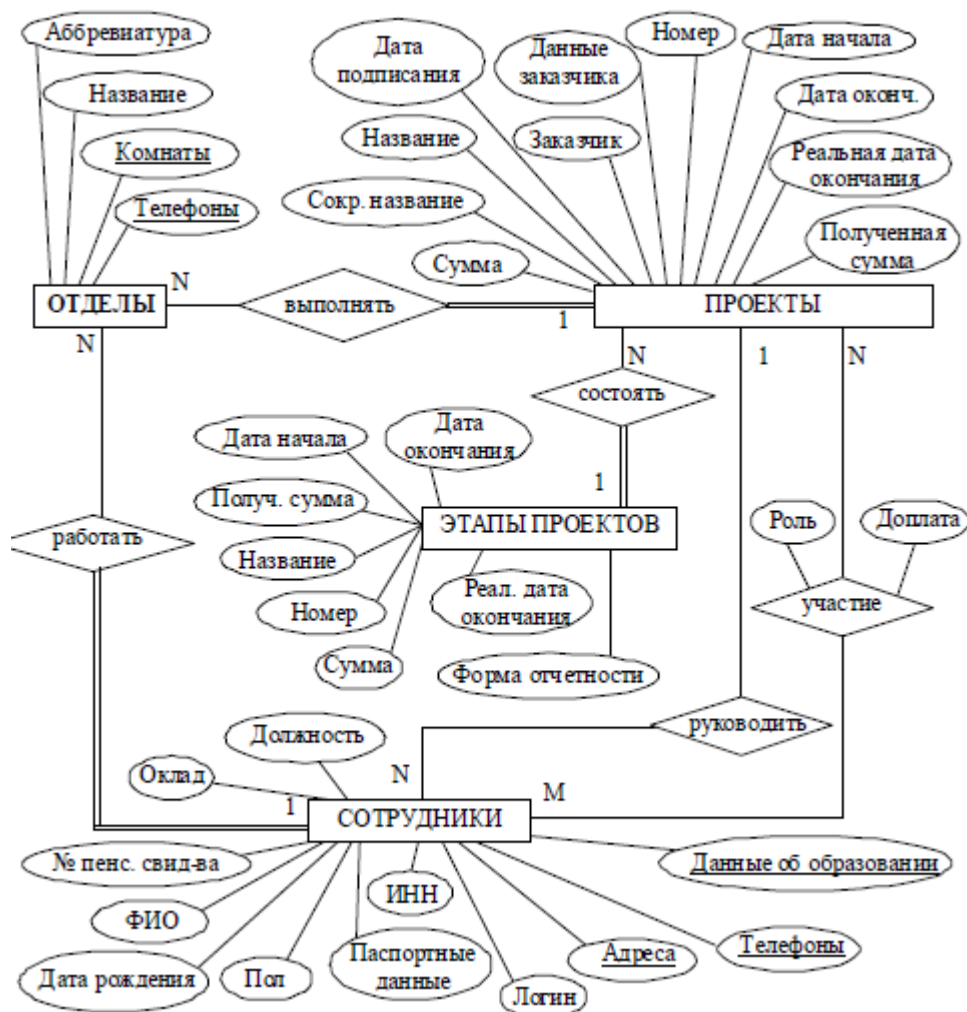
- просмотр данных о других участниках проекта;
- просмотр данных о сроках сдачи проекта и форме отчётности.

3. Логическое проектирование реляционной БД

Преобразование ER–диаграммы в схему базы данных

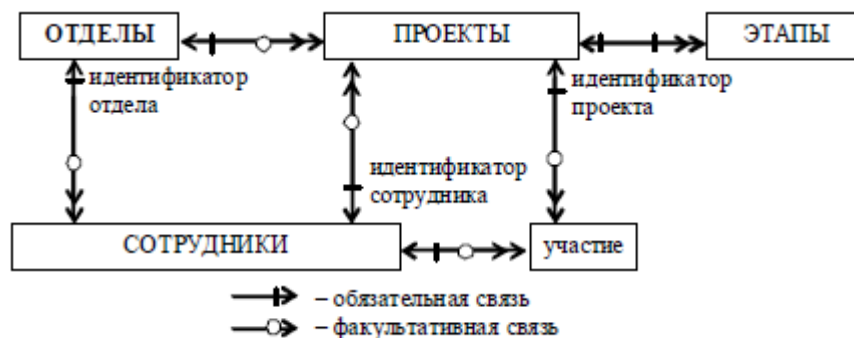
База данных создаётся на основании схемы базы данных. Для преобразования ER–диаграммы в схему БД приведём уточнённую ER–диаграмму, содержащую атрибуты сущностей.

Преобразование ER–диаграммы в схему БД выполняется путем сопоставления каждой сущности и каждой связи, имеющей атрибуты, отношения (таблицы) БД. Связь типа 1:n (один-ко-многим) между отношениями реализуется через внешний ключ. Ключ вводится для того отношения, к которому осуществляется множественная связь. Внешнему ключу должен соответствовать первичный или уникальный ключ основного (родительского) отношения.



Связь *участивать* между *ПРОЕКТАМИ* и *СОТРУДНИКАМИ* принадлежит к типу n:m (многие-ко-многим). Этот тип связи реализуется через вспомогательное отношение *Участие*, которое содержит комбинации первичных ключей соответствующих исходных отношений.

Полученная схема реляционной базы данных:



Бинарная связь между отношениями не может быть обязательной для обоих отношений. Такой тип связи означает, что, например, прежде чем добавить новый проект в отношение *ПРОЕКТЫ*, нужно добавить новую строку в отношение *ЭТАПЫ*, и наоборот. Поэтому для такой связи необходимо снять с одной стороны условие обязательности. Так как все эти связи будут реализованы с помощью внешнего ключа, снимем условие обязательности связей для отношений, содержащих первичные ключи.

Схема содержит три цикла: "сотрудники–проекты–участие–сотрудники", "отделы–сотрудники–проекты–отделы" и "отделы–сотрудники–участие–проекты–отделы". Цикл допустим только в том случае, если связи, входящие в него, независимы друг от друга. Например, для нашей ПрО справедливо такое правило: сотрудник любого отдела может быть участником (исполнителем или консультантом) проекта любого отдела. Эти связи независимы, поэтому цикл "отделы–сотрудники–участие–проекты–отделы" не будет приводить к нарушению логической целостности данных.

С другой стороны, только сотрудник отдела, отвечающего за выполнение проекта, может быть руководителем проекта. Но система не помешает нам назначить руководителем проекта сотрудника любого отдела. При добавлении проекта с внешним ключом *Руководитель* система проверит только, что такой человек есть в таблице *СОТРУДНИКИ*. А значение внешних ключей *Отдел* в таблицах *СОТРУДНИКИ* и *ПРОЕКТЫ* сравнивать не будет.

Таким образом, остальные циклы могут приводить к возможности нарушения логической целостности данных. Существует несколько подходов для разрешения ситуаций, в которых связи, входящие в цикл, зависят друг от друга.

Рассмотрим эту ситуацию в общем случае. Сначала слегка упростим схему: реализуем связь "руководить" через таблицу *УЧАСТИЕ* – это позволит не отвлекаться на малозначительные детали.

Будем считать, что в выполнении проекта могут участвовать только сотрудники, работающие в том же отделе, к которому относится проект (рис. а). При циклической схеме СУБД не сможет гарантировать логическую целостность данных без использования дополнительных средств.

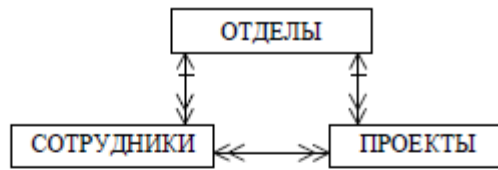
Один из способов разрешения таких ситуаций – разорвать цикл, исключив одну из связей (рис. б) или введя промежуточное отношение (рис. в). В нашем случае можно было бы разорвать связь "сотрудники–проекты", если бы каждый сотрудник участвовал во всех проектах своего отдела. Промежуточное отношение можно было бы использовать, если бы существовала общая связь между сущностями, входящими в цикл. Например, если бы каждый сотрудник заключал договор с отделом на выполнение работ в рамках проекта, то отношение *ДОГОВОРЫ* отражало бы связь между отделом, сотрудником и проектом.

Другой способ разрешения цикла заключается в том, что в промежуточное отношение *СОТРУДНИКИ – ПРОЕКТЫ*, которое реализует связь многие-ко-многим, добавляются (мигрируют) внешние ключи *Код отдела* (*D_id*) из отношений *СОТРУДНИКИ* и *ПРОЕКТЫ* (рис.г). Эти ключи проверяются на равенство друг другу с помощью соответствующего ограничения целостности (*check*). Использование этого способа возможно в том случае, когда соответствующие связи (*отдел–проект* и *отдел–сотрудник*) имеют тип один-ко-многим и являются обязательными.

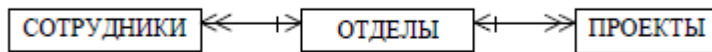
В тех ситуациях, когда все эти способы непригодны, логическая целостность контролируется программой или вручную. Если принято решение переложить обязанности по контролю за логической целостностью данных на

пользователя, то эти обязанности должны быть отражены в документации (в руководстве пользователя).

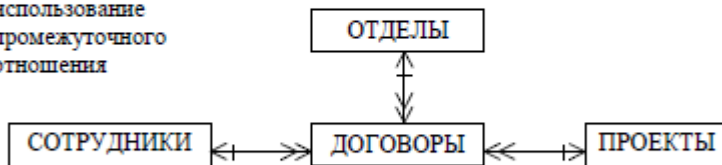
а) пример цикла



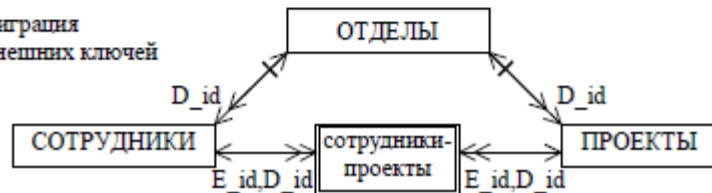
б) разрыв связи



в) использование промежуточного отношения



г) миграция внешних ключей



Примем для нашей ПрО, что руководитель проекта может одновременно выполнять и другие обязанности в этом проекте, чтобы цикл "сотрудники – проекты – участие – сотрудники" не приводил к возможности нарушения логической целостности данных. Зато цикл "отделы–сотрудники (руководители) – проекты – отделы" включает зависимые связи: руководитель проекта назначается из того отдела, который отвечает за выполнение проекта в целом. Здесь можно было бы применить разрыв связи "отделы–проекты" и определять, к какому отделу относится проект через руководителя (по отделу руководителя проекта). Но такой подход в данном случае имеет существенный недостаток.

Заменив руководителя проекта сотрудником другого отдела, можно одновременно изменить отдел, отвечающий за выполнение проекта, т.е. объединить в одно действие два независимых изменения, а это недопустимо.

Исходя из вышесказанного мы не будем разрывать связь, а примем решение реализовать эту проверку программно. Приложение должно будет при назначении руководителя проекта выдавать список сотрудников того отдела, который отвечает за выполнение данного проекта. Руководителя можно будет выбрать только из этого списка, а не вводить вручную.

4. Составление реляционных отношений

Каждое реляционное отношение соответствует одной сущности (объекту ПрО) и в него вносятся все атрибуты этой сущности. Для каждого отношения определяются первичный ключ и внешние ключи (в соответствии со схемой БД). В том случае, если базовое отношение не имеет потенциальных ключей, вводится *суррогатный первичный ключ*, который не несёт смысловой нагрузки и служит только для идентификации записей.

Отношения приведены в табл. 1-5. Для каждого отношения указаны атрибуты с их внутренним названием, типом и длиной. Типы данных обозначаются так: N – числовой, С – символьный тип фиксированной длины, V – символьный тип переменной длины, D – дата (этот тип имеет стандартную длину, зависящую от СУБД, поэтому она не указывается).

Потенциальными ключами отношения *ОТДЕЛЫ* являются атрибуты *Аббревиатура* и *Название отдела*. Первый занимает меньше места, поэтому мы выбираем его в качестве первичного ключа.

Таблица 1. Схема отношения *ОТДЕЛЫ* (Departs)

Содержание поля	Имя поля	Тип, длина	Примечания
Аббревиатура отдела	D_ID	C(10)	первичный ключ
Название отдела	D_NAME	V(100)	обязательное поле
Комнаты	D_ROOMS	V(20)	обязательное многозначное поле
Телефоны	D_PHONE	V(40)	обязательное многозначное поле

Потенциальными ключами отношения *СОТРУДНИКИ* являются поля *Паспортные данные*, *ИНН* и *Номер страхового пенсионного свидетельства*. Все они занимают достаточно много места, а паспортные данные кроме того могут меняться. Введём суррогатный первичный ключ *Номер сотрудника*.

Таблица 2. Схема отношения *СОТРУДНИКИ* (Employees)

Содержание поля	Имя поля	Тип, длина	Примечания
Номер	E_ID	N(4)	суррогатный первичный ключ
Фамилия, имя, отчество	E_NAME	V(50)	обязательное поле
Дата рождения	E_BORN	D	обязательное поле
Пол	E_SEX	C(1)	обязательное поле, 'м' или 'ж'
Паспортные данные	E_PASP	V(50)	обязательное поле
ИНН	E_INN	C(12)	обязательное уникальное поле
Номер пенсионного страхового свидетельства	E_PENS	C(14)	обязательное уникальное поле
Отдел	E_DEPART	C(10)	внешний ключ (к Departs)
Должность	E_POST	V(30)	обязательное поле
Оклад	E_SAL	N(8,2)	обязательное поле, > 4500 руб.
Данные об образовании	E_EDU	V(200)	обязательное многозначное поле
Адреса	E_ADDR	V(100)	многозначное поле
Телефоны	E_PHONE	V(30)	многозначное поле
Логин	E_LOGIN	V(30)	

Примечание. Суррогатный первичный ключ также может вводиться в тех случаях, когда потенциальный ключ имеет большой размер (например, длинная символьная строка) или является составным (не менее трёх атрибутов).

В отношении *ПРОЕКТЫ* три потенциальных ключа: *Номер проекта*, *Название проекта* и *Сокращённое название*. Меньше места занимает первый из них, но он малоинформативен. Зато сокращённое название, используемое в качестве внешнего ключа в других таблицах, позволит специалисту идентифицировать проект без необходимости соединения с отношением *ПРОЕКТЫ*.

Таблица 3. Схема отношения ПРОЕКТЫ (Projects)

Содержание поля	Имя поля	Тип, длина	Примечания
Номер проекта	P_ID	N(6)	обязательное уникальное поле
Название проекта	P_TITLE	V(100)	обязательное поле
Сокращённое название	P_ABBR	C(10)	первичный ключ
Отдел	P_DEPART	C(10)	внешний ключ (к Departs)
Заказчик	P_COMPANY	V(40)	обязательное поле
Данные заказчика	P_LINKS	V(200)	обязательное поле
Руководитель	P_CHIEF	N(4)	внешний ключ (к Employees)
Дата начала проекта	P_BEGIN	D	обязательное поле
Дата окончания проекта	P_END	D	обязательное поле, больше даты начала проекта
Реальная дата окончания	P_FINISH	D	
Стоимость проекта	P_COST	N(10)	обязательное поле
Полученная сумма	P_SUM	N(10)	обязательное поле, значение по умолчанию – 0

Потенциальным ключом отношения ЭТАПЫ является комбинация внешнего ключа и номера этапа, а потенциальным ключом вспомогательного отношения УЧАСТИЕ является комбинация первых трёх полей этого отношения. Можно вообще не вводить первичный ключ для данных отношений, т.к. на них никто не ссылается. Но уникальность этих комбинации является в данном случае ограничением целостности данных, поэтому мы возьмём эти комбинации в качестве первичных ключей соответствующих отношений.

Таблица 4. Схема отношения ЭТАПЫ ПРОЕКТА (Stages)

Содержание поля	Имя поля	Тип, длина	Примечания	
Проект	S_PRO	C(10)	внешний ключ (к Projects)	составной первичный ключ
Номер этапа	S_NUM	N(2)		
Название этапа	S_TITLE	V(200)	обязательное поле	
Дата начала этапа	S_BEGIN	D	обязательное поле	
Дата окончания этапа	S_END	D	обязательное поле, > даты начала	
Реальная дата окончания	S_FINISH	D	больше даты начала этапа	
Стоимость этапа	S_COST	N(10)	обязательное поле	
Полученная сумма по этапу	S_SUM	N(10)	обязательное поле, значение по умолчанию – 0	
Форма отчётности	S_FORM	V(100)	обязательное поле	

Таблица 5. Схема отношения УЧАСТИЕ (Job)

Содержание поля	Имя поля	Тип, длина	Примечания *
Проект	J_PRO	C(10)	внешний ключ (к Projects)
Сотрудник	J_EMP	N(4)	внешний ключ (к Employees)
Роль	J_ROLE	V(20)	обязательное поле
Доплата	J_BONUS	N(2)	

* – в отношении УЧАСТИЕ первичный ключ состоит из первых 3-х полей этого отношения.

5. Нормализация полученных отношений (до 4НФ)

Механизм нормализации подразумевает определённую последовательность преобразования отношений к третьей нормальной форме. Мы не будем чётко придерживаться этой последовательности, т.к. она избыточна, и многозначные атрибуты сразу вынесем в отдельные отношения на первом же этапе.

1НФ. Для приведения таблиц к 1НФ требуется составить прямоугольные таблицы (одно значение атрибута – одна ячейка таблицы) и разбить сложные

атрибуты на простые.

Примечание. В реальных БД сложные атрибуты разбиваются на простые, если:

- а) этого требует внешнее представление данных;
- б) в запросах поиск может осуществляться по отдельной части атрибута.

Разделим атрибут *Фамилия, имя, отчество* на два атрибута *Фамилия* и *Имя, отчество*, *Паспортные данные* на *Номер паспорта* (уникальный), *Дата выдачи* и *Кем выдан*, а *Данные об образовании* – на *Вид образования*, *Специальность*, *Номер диплома* и *Год окончания учебного заведения*.

Многозначные атрибуты *Комнаты* и *Телефоны* из отношения *ОТДЕЛЫ* вынесем в отдельное отношение *КОМНАТЫ*, а домашние и мобильные телефоны и адреса сотрудников – в отношение *АДРЕСА-ТЕЛЕФОНЫ*. Так как в комнате может не быть телефона, первичный ключ отношения *КОМНАТЫ* не определен (ПК не может содержать null-значения), но на этих атрибутах можно определить составной уникальный ключ. В отношении *АДРЕСА-ТЕЛЕФОНЫ* также нет потенциальных ключей: оставим это отношение без первичного ключа, т.к. на это отношение никто не ссылается. Данные об образовании сотрудников также вынесем в отдельное отношение.

Что касается рабочих телефонов сотрудников, то один из этих номеров – основной – определяется рабочим местом сотрудника (рассматриваются только стационарные телефоны). Будем хранить этот номер в атрибуте *Рабочий телефон*. Наличие других номеров зависит от того, есть ли в том же помещении (комнате) другие сотрудники, имеющие стационарные телефоны. Добавим в отношение *СОТРУДНИКИ* атрибут *Номер комнаты*, чтобы дополнительные номера телефонов сотрудника можно было вычислить из других кортежей с таким же номером комнаты.

Связь между отношениями *СОТРУДНИКИ* и *КОМНАТЫ* реализуем через составной внешний ключ (*Номер комнаты, Рабочий телефон*).

Мы также удалим вычисляемый атрибут *Полученная сумма* из отношения *ПРОЕКТЫ*, т.к. он является суммой значений аналогичного атрибута из отношения *ЭТАПЫ ПРОЕКТОВ*. Но атрибут *Стоимость проекта* оставим, т.к. она фигурирует в документации по проекту. А для обеспечения логической целостности данных предусмотрим в приложении проверку того, что сумма по всем этапам совпадает со стоимостью проекта.

2НФ. В нашем случае составные первичные ключи имеют отношения *ЭТАПЫ ПРОЕКТА* и *УЧАСТИЕ*. Неключевые атрибуты этих отношений функционально полно зависят от составных первичных ключей.

3НФ. В отношении *ПРОЕКТЫ* атрибут *Данные заказчика* зависит от атрибута *Заказчик*, а не от первичного ключа, поэтому его следует вынести в отдельное отношение *ЗАКАЗЧИКИ*. Но при этом первичным ключом нового отношения станет атрибут *Заказчик*, т.е. длинная символьная строка. Целесообразнее перенести в новое отношение атрибуты *Заказчик* и *Данные заказчика* и ввести для него суррогатный ПК. Так как с каждым заказчиком может быть связано несколько проектов, связь между отношениями *ПРОЕКТЫ* и *ЗАКАЗЧИКИ*

будет 1:n и суррогатный ПК станет внешним ключом для отношения *ПРОЕКТЫ*.

В отношении *СОТРУДНИКИ* атрибут *Оклад* зависит от атрибута *Должность*. Поступим с этой транзитивной зависимостью так же, как в предыдущем случае: создадим отношение *ДОЛЖНОСТИ*, перенесём в него атрибуты *Должность* и *Оклад*, а первичным ключом сделаем название должности. В отношениях *СОТРУДНИКИ* и *ОБРАЗОВАНИЕ* атрибуты (*Дата выдачи* и *Кем выдан*) и (*Номер диплома* и *Год окончания учебного заведения*) зависят не от первичного ключа, а от атрибутов соответственно *Номер паспорта* и *Специальность*. Но если мы выделим их в отдельное отношение, то получим связи типа 1:1. Следовательно, здесь декомпозиция нецелесообразна.

4НФ. Отношение *АДРЕСА-ТЕЛЕФОНЫ* нарушают 4НФ, т.к. не всякий телефон привязан к конкретному адресу (т.е. мы имеем две многозначных зависимости в одном отношении). Но выделять *Телефоны* в отдельное отношение не стоит, т.к. эти сведения носят справочный характер и не требуется их автоматическая обработка.

Отношения, полученные после нормализации, приведены в табл. 6-15.

Таблица 6. Схема отношения *ОТДЕЛЫ* (Departs)

Содержание поля	Имя поля	Тип, длина	Примечания
Аббревиатура отдела	D_ID	V(12)	первичный ключ
Название отдела	D_NAME	V(100)	обязательное поле

Таблица 7. Схема отношения *КОМНАТЫ* (Rooms)

Содержание поля	Имя поля	Тип, длина	Примечания
Отдел	R_DEPART	V(12)	внешний ключ (к Departs)
Номер комнаты	R_ROOM	N(4)	составной уникальный ключ
Телефон	R_PHONE	V(20)	

Таблица 8. Схема отношения *ДОЛЖНОСТИ* (Posts)

Содержание поля	Имя поля	Тип, длина	Примечания
Название должности	P_POST	V(30)	первичный ключ
Оклад	P_SAL	N(8,2)	обязательное поле, > 4500 руб.

Таблица 9. Схема отношения *СОТРУДНИКИ* (Employees)

Содержание поля	Имя поля	Тип, длина	Примечания
Идентификатор сотрудника	E_ID	N(4)	суррогатный первичный ключ
Фамилия	E_FNAME	V(25)	обязательное поле
Имя, отчество	E_LNAME	V(30)	обязательное поле
Дата рождения	E_BORN	D	обязательное поле
Пол	E_SEX	C(1)	обязательное поле
Серия и номер паспорта	E_PASP	C(10)	обязательное уникальное поле
Когда выдан паспорт	E_DATE	D	обязательное поле
Кем выдан паспорт	E_GIVEN	V(50)	обязательное поле
ИНН	E_INN	C(12)	обязательное уникальное поле
Номер пенсионного страхового свидетельства	E_PENS	C(14)	обязательное уникальное поле
Отдел	E_DEPART	V(12)	внешний ключ (к Departs)
Должность	E_POST	V(30)	внешний ключ (к Posts)
Номер комнаты	E_ROOM	N(4)	составной внешний ключ (к Rooms)
Рабочий телефон	E_PHONE	V(20)	
Логин	E_LOGIN	V(30)	

Таблица 10. Схема отношения ОБРАЗОВАНИЕ (Edu)

Содержание поля	Имя поля	Тип, длина	Примечания
Идентификатор сотрудника	U_ID	N(4)	внешний ключ (к Employees)
Вид образования	U_TYPE	V(20)	обязательное поле
Специальность	U_SPEC	V(40)	
Номер диплома	U_DIPLOM	V(15)	
Год окончания учебного заведения	U_YEAR	N(4)	обязательное поле

Таблица 11. Схема отношения АДРЕСА-ТЕЛЕФОНЫ (AdrTel)

Содержание поля	Имя поля	Тип, длина	Примечания
Идентификатор сотрудника	A_ID	N(4)	внешний ключ (к Employees)
Адрес	A_ADDR	V(50)	
Телефон	A_PHONE	V(30)	

Таблицы ОБРАЗОВАНИЕ и АДРЕСА-ТЕЛЕФОНЫ не имеют потенциальных ключей, но мы не будем вводить суррогатные первичные ключи, т.к. на эти таблицы никто не ссылается.

Таблица 12. Схема отношения ЗАКАЗЧИКИ (Clients)

Содержание поля	Имя поля	Тип, длина	Примечания
Номер заказчика	C_ID	N(4)	суррогатный первичный ключ
Заказчик	C_COMPANY	V(40)	обязательное поле
Адрес заказчика	C_ADR	V(50)	обязательное поле
Контактное лицо	C_PERSON	V(50)	обязательное поле
Телефон	C_PHONE	V(30)	

Таблица 13. Схема отношения ПРОЕКТЫ (Projects)

Содержание поля	Имя поля	Тип, длина	Примечания
Номер проекта	P_ID	N(6)	обязательное уникальное поле
Название проекта	P_TITLE	V(100)	обязательное поле
Сокращённое название	P_ABBR	C(10)	первичный ключ
Отдел	P_DEPART	V(12)	внешний ключ (к Departs)
Заказчик	P_COMPANY	N(4)	внешний ключ (к Clients)
Руководитель	P_CHIEF	N(4)	внешний ключ (к Employees)
Дата начала проекта	P_BEGIN	D	обязательное поле
Дата окончания проекта	P_END	D	обязательное поле, больше даты начала проекта
Реальная дата окончания	P_FINISH	D	больше даты начала проекта
Стоимость проекта	P_COST	N(10)	обязательное поле, > 0

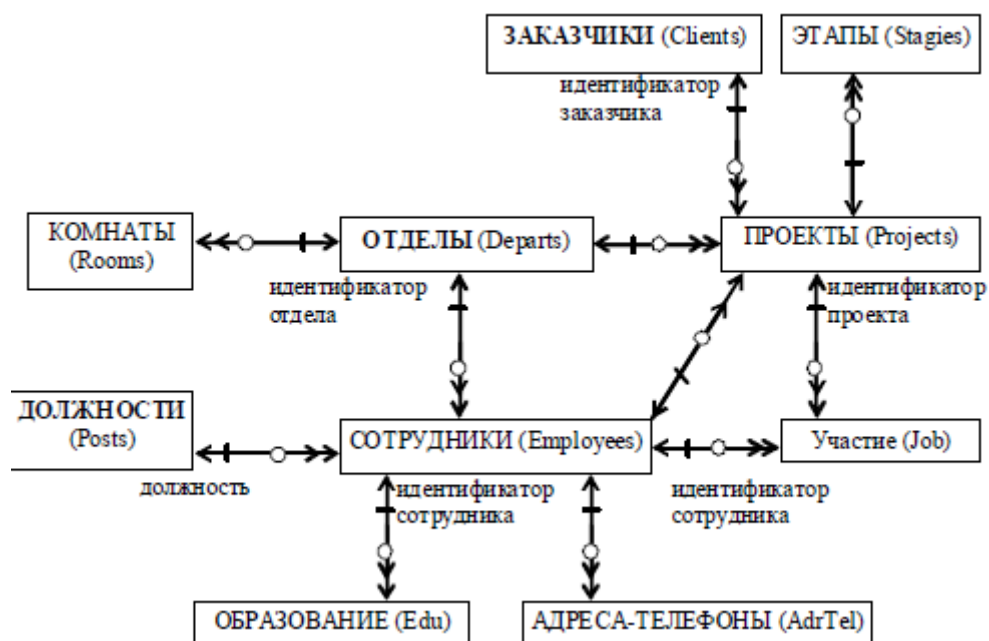
Таблица 14. Схема отношения ЭТАПЫ ПРОЕКТА (Stages)

Содержание поля	Имя поля	Тип, длина	Примечания
Проект	S_PRO	C(10)	внешний ключ (к Projects)
Номер этапа	S_NUM	N(2)	составной первичный ключ
Название этапа	S_TITLE	V(200)	обязательное поле
Дата начала этапа	S_BEGIN	D	обязательное поле
Дата окончания этапа	S_END	D	обязательное поле, больше даты начала этапа
Реальная дата окончания	S_FINISH	D	больше даты начала этапа
Стоимость этапа	S_COST	N(10)	обязательное поле
Полученная сумма по этапу	S_SUM	N(10)	обязательное поле, значение по умолчанию – 0
Форма отчётности	S_FORM	V(100)	обязательное поле

Таблица 15. Схема отношения УЧАСТИЕ (Job)

Содержание поля	Имя поля	Тип, длина	Примечания
Проект	J_PRO	C(10)	внешний ключ (к Projects)
Сотрудник	J_EMP	N(4)	внешний ключ (к Employees)
Роль	J_ROLE	V(20)	обязательное поле
Доплата	J_BONUS	N(2)	

Схема базы данных после нормализации:



5. Определение дополнительных ограничений целостности

Перечислим ограничения целостности, которые не указаны в табл. 6–15.

1. Атрибут *Вид образования* может принимать одно из следующих значений: 'начальное', 'среднее', 'средне-специальное', 'высшее'.

2. Атрибут *Роль* может принимать одно из двух значений: 'исполнитель' или 'консультант'.

3. В поле *Доплата* хранится величина доплаты сотруднику за участие в проекте (в процентах к его окладу). Значение поля больше либо равно 0.

4. Нумерация в поле *Номер этапа* начинается с 1 и является непрерывной для каждого проекта.

5. Дата начала первого этапа проекта должна соответствовать началу проекта в целом, дата завершения последнего этапа должна соответствовать завершению проекта в целом. Этапы не должны пересекаться по времени и между ними не должно быть разрывов.

6. Стоимость проекта должна быть равна сумме стоимостей всех этапов этого проекта.

6. Описание групп пользователей и прав доступа

Опишем для каждой группы пользователей права доступа к каждой таблице. Права доступа должны быть распределены так, чтобы для каждого объекта БД был хотя бы один пользователь, который имеет право добавлять и удалять данные из объекта. Используются следующие сокращения: s – чтение данных (select); i – добавление данных (insert); u – модификация данных (update); d – удаление данных (delete).

Права на изменение данных в таблице *УЧАСТИЕ* будут назначены через представление, т.к. изменять данные этой таблицы может только руководитель проекта. Описание представлений приведено в п.2.5.2. "Создание представлений (готовых запросов)".

Права назначает администратор БД (или администратор безопасности, если система сложная и администраторов несколько).

Таблица 16. Права доступа к таблицам для групп пользователей

Таблицы	Группы пользователей (роли)				
	Руководители организации	Сотрудники отд. кадров	Руководители проектов	Бухгалтеры	Участники проектов
Отделы	S	SIUD	S	S	
Комнаты	S	SUID	S	S	S
Должности	SIUD			S	
Сотрудники	S	SUID	S	S	
Адреса- телефоны	S	SUID	S	S	
Образование	S	SUID	S	S	
Заказчики	SIUD		S		
Проекты	SIUD		S		
Этапы про- ектов	SIUD		SUI		
Участие	S		S	S	

ПРАКТИЧЕСКАЯ РАБОТА № 4

Тема: Проектирование реляционной схемы базы данных в среде СУБД

Цель работы: научить проектировать реляционные схемы БД

Оборудование: ПК, интернет, программное обеспечение – MS Visio Professional, MS Word, инструкции по выполнению работы.

Содержание работы:

Задание 1. Спроектировать БД «Спортивный комплекс»

Сущность задачи: Спортивный комплекс предоставляет услуги по проведению спортивных тренировок. Тренировки, относящиеся к одному виду спорта, объединяются в спортивные секции. Клиент обращается в спортивный комплекс, где получает абонемент на посещение спортивной секции. На основе купленных абонементов составляется расписание тренировок на следующий месяц. Также, в зависимости от загруженности спортивного комплекса, распределяются тренеры спортивных секций. По результатам своей деятельности спортивный комплекс производит отчисления в налоговые органы и предоставляет отчетность в органы государственной статистики.

Задание 2. Спроектировать БД «Страховая медицинская компания».

Сущность задачи: Страховая медицинская компания (СМК) заключает договора добровольного медицинского страхования с населением и договора с лечебными учреждениями на лечение застрахованных клиентов. При возникновении страхового случая клиент подает заявку на оказание медицинских услуг по условиям договора инспектору, который работает с данным клиентом. Инспектор направляет данного клиента в лечебное учреждение. Отчеты о своей деятельности инспектор предоставляет в бухгалтерию. Бухгалтерия проверяет оплату договоров, перечисляет денежные средства за оказанные услуги лечебным учреждениям, производит отчисления в налоговые органы и предоставляет отчетность в органы государственной статистики. СМК не только оплачивает лечение застрахованного лица при возникновении с ним страхового случая, но и, при возникновении каких-либо осложнений после лечения, оплачивает лечение этих осложнений.

ПРАКТИЧЕСКАЯ РАБОТА № 5

Тема: Проектирование реляционной схемы базы данных в среде СУБД

Цель работы: научить проектировать реляционные схемы БД

Оборудование: ПК, интернет, программное обеспечение – MS Visio Professional, MS Word, инструкции по выполнению работы.

Содержание работы:

Задание 1. Спроектировать БД «Компания по разработке программных продуктов»

Сущность задачи: Компания заключает договор с клиентом на разработку программного продукта согласно техническому заданию. После утверждения технического задания определяется состав и объем работ, составляется предварительная смета. На каждый проект назначается ответственный за его выполнение – куратор проекта, который распределяет нагрузку между программистами и следит за выполнением технического задания. Когда программный продукт готов, то его внедряют, производят обучение клиента и осуществляют дальнейшее сопровождение. По результатам своей деятельности компания производит отчисления в налоговые органы и предоставляет отчетность в органы государственной статистики.

Задание 2. Спроектировать БД «Агентство недвижимости»

Сущность задачи: Агентство недвижимости занимается покупкой, продажей, сдачей в аренду объектов недвижимости по договорам с их собственниками. Агентство управляет объектами недвижимости как физических, так и юридических лиц. Собственник может иметь несколько объектов. В случае покупки или аренды клиент может произвести осмотр объекта. В качестве одной из услуг, предлагаемых агентством, является проведение инспектирования текущего состояния объекта для адекватного определения его рыночной цены. По результатам своей деятельности агентство производит отчисления в налоговые органы и предоставляет отчетность в органы государственной статистики.

ПРАКТИЧЕСКАЯ РАБОТА № 6

Тема: Проектирование реляционной схемы базы данных в среде СУБД

Цель работы: научить проектировать реляционные схемы БД

Оборудование: ПК, интернет, программное обеспечение – MS Visio Professional, MS Word, инструкции по выполнению работы.

Содержание работы:

Задание 1. Спроектировать БД «Компьютерная компания»

Сущность задачи: Компьютерная компания занимается продажей, ремонтом, сборкой, тестированием компьютерной техники. Также, специалисты компании предоставляют услуги по разработке и монтажу локальных вычислительных сетей. Вся техника и комплектующие закупаются оптом у дилеров и хранятся на складе. Клиент, который хочет приобрести товар, оформляет заказ в торговом зале, а забирает технику со склада или оставляет заявку на ее доставку. Клиент, который хочет отремонтировать технику, приносит ее в сервисный отдел, откуда, по прошествии некоторого времени, забирает как от-ремонтированную или как технику, не подлежащую ремонту. По желанию клиента, специалисты компании могут выехать к клиенту для общей диагностики возникшей проблемы с техникой. По результатам своей деятельности компьютерная компания производит отчисления в налоговые органы и предоставляет отчетность в органы государственной статистики.

Задание 2. Спроектировать БД «Строительная организация»

Сущность задачи: Строительная организация занимается строительством объектов по заказам клиентов. Сначала заказ проходит предварительную стадию: сбор различных разрешений на строительство, составление эскиза объекта, расчет объема и закупка строительных материалов. Сами строительные материалы доставляются на объект партиями. По мере поступления очередной партии стройматериалов закладывается фундамент объекта, строится каркас здания. По результатам данной работы происходит согласование с заказчиком, после чего утепляется контур, вставляются окна, устанавливается крыша. Далее идет обсуждение с клиентом внутренней отделки здания, закупаются отделочные материалы. После того, как объект проходит технический контроль, он передается заказчику. В дополнительные услуги строительной организации входят: услуги дизайнера по интерьеру, закупка и доставка мебели, сотрудничество с охранным предприятием по установке сигнализации. По результатам своей деятельности строительная организация производит отчисления в налоговые органы и предоставляет отчетность в органы государственной статистики.

ПРАКТИЧЕСКАЯ РАБОТА № 7

Тема: Проектирование реляционной схемы базы данных в среде СУБД

Цель работы: научить проектировать реляционные схемы БД

Оборудование: ПК, интернет, программное обеспечение – MS Visio Professional, MS Word, инструкции по выполнению работы.

Содержание работы:

Задание 1. Спроектировать БД «Компания по предоставлению телекоммуникационных услуг»

Сущность задачи: Компания занимается оказанием телекоммуникационных услуг абонентам. Клиент делает заявку на подключение к телекоммуникационным услугам и ему, по необходимости, устанавливают соответствующее оборудование. Оплата за услуги вносится путем авансовых платежей. Каждый факт предоставления услуги фиксируется соответствующим оборудованием и является основанием для списания соответствующей суммы с личного счета абонента. Клиент в любое время суток может получить отчет об оказанных ему услугах, их стоимости и остатку на личном счете абонента. По результатам своей деятельности компания производит отчисления в налоговые органы и предоставляет отчетность в органы государственной статистики.

Задание 2. Спроектировать БД «Кадровое агентство»

Сущность задачи: Кадровое агентство способствует трудоустройству безработных граждан. Агентство ведет учет и классификацию данных о безработных на основании резюме от них. От предприятий города поступают данные о свободных вакансиях, на основании которых агентство предлагает различные варианты трудоустройства соискателям. В случае положительного исхода поиска вакансии считается заполненной, а безработный становится трудоустроенным. По результатам своей деятельности кадровое агентство производит отчисления в налоговые органы и предоставляет отчетность в органы государственной статистики.

ПРАКТИЧЕСКАЯ РАБОТА № 8

Тема: Создание БД в среде разработки

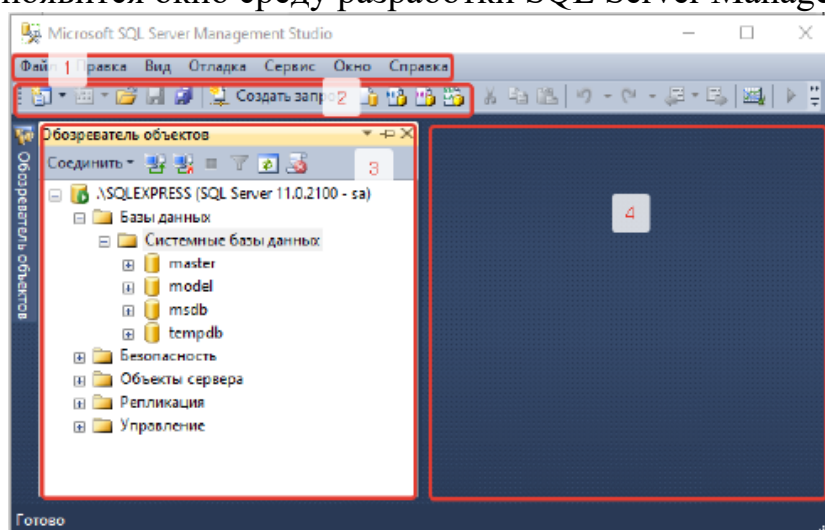
Цели работы: научить создавать БД в среде разработки MS SQL Server Management Studio.

Оборудование: ПК, интернет, программное обеспечение – MS SQL Server Management Studio, инструкции по выполнению работы.

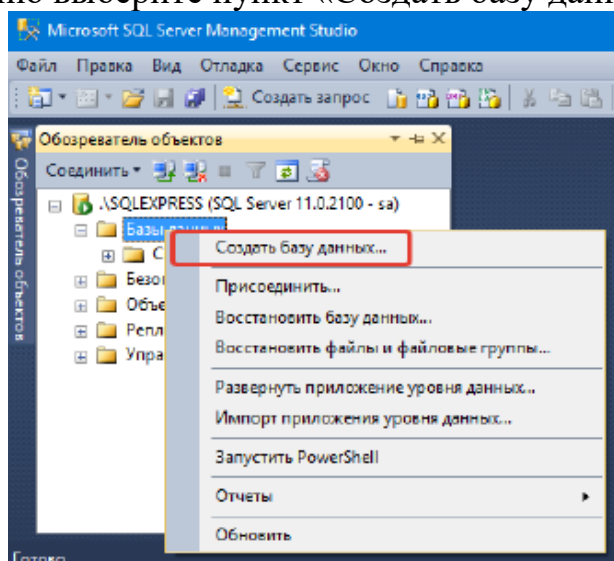
Содержание работы:

Задание 1. Используя инструмент Microsoft SQL Management Studio создать базу данных.



1. Запустить MS SQL Server Management Studio. После запуска среды разработки появится окно подключения к серверу Соединение с сервером. В этом окне необходимо нажать кнопку Соединить. После нажатия кнопки «Соединить» появится окно среду разработки SQL Server Management Studio



2. Теперь перейдём непосредственно к созданию файла данных. Для этого в обозревателе объектов щёлкните правой кнопкой мыши на папке Базы данных и в появившемся меню выберите пункт «Создать базу данных...».

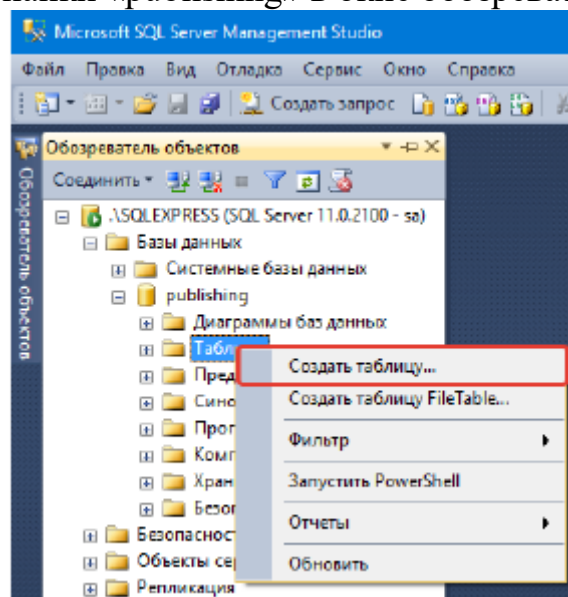


3. Появится окно настроек параметров файла данных новой базы данных Создание базы данных. В левой части окна настроек имеется список «Выбор страницы». Этот список позволяет переключаться между группами настроек.


4. Новую БД можно создать, используя стандартные команды языка T-SQL. Все команды языка T-SQL набираются на вкладке нового запроса (SQLQuery). Для того чтобы создать новый запрос на панели инструментов необходимо нажать кнопку  Создать запрос. Для выполнения команд языка T-SQL на панели инструментов необходимо нажать кнопку  Выполнить или на вкладке нового запроса набрать команду GO. Для создания нового файла данных используется команда CREATE DATABASE, которая имеет следующий синтаксис:

```
CREATE DATABASE [Имя БД] ON PRIMARY
(
    NAME = <Логическое имя>,
    FILENAME = <Имя файла>,
    SIZE = <Нач.размер>,
    MAXSIZE = <Макс.размер>,
    FILEGROWTH = <Шаг> )
LOG ON
(
    NAME = <Логическое имя>,
    FILENAME = <Имя файла>,
    SIZE = <Нач.размер>,
    MAXSIZE = <Макс.размер>,
    FILEGROWTH = <Шаг> )
```

5. Перейдём теперь к созданию таблиц. Все таблицы нашей БД находятся в подпапке «Таблицы» папки «publishing» в окне обозревателя объектов.



6. Создадим таблицу «Сотрудники». Для этого необходимо кликнуть правой кнопкой мыши по папке «Таблицы» и в появившемся меню выбрать пункт «Создать таблицу». Появится окно создания новой таблицы

Имя столбца	Тип данных	Разрешит...
▶		

В правой части окна расположена таблица определения полей новой таблицы. Данная таблица имеет следующие столбцы:

- Имя столбца должно всегда начинаться с буквы и не должно содержать различных специальных символов и знаков препинания. Если имя поля содержит пробелы, то оно автоматически заключается в квадратные скобки.
- Тип данных столбца.

- Разрешить значения Null. Если эта опция поля включена, то в случае не заполнения поля в него будет автоматически подставлено значение Null. То есть, поле необязательно для заполнения.

Под таблицей определения полей располагается таблица свойств выделенного поля «Свойства столбца». В данной таблице настраиваются свойства выделенного поля.

7.Перейдём к созданию полей и настройке их свойств. В таблице определения полей задайте значения столбцов «Имя столбца», «Тип данных» и «Разрешить значения Null»


	Имя столбца	Тип данных	Разрешить значения NULL
	E_ID	int	<input type="checkbox"/>
	E_SURNAME	nvarchar(20)	<input type="checkbox"/>
	E_NAME	nvarchar(20)	<input type="checkbox"/>
	E_FNAME	nvarchar(30)	<input checked="" type="checkbox"/>
	E_BIRTHDAY	date	<input checked="" type="checkbox"/>
	E_SEX	bit	<input checked="" type="checkbox"/>
	E_POST	int	<input type="checkbox"/>
	E_ROOM	int	<input checked="" type="checkbox"/>
	E_PHONE	nvarchar(10)	<input checked="" type="checkbox"/>
	E_INN	nvarchar(12)	<input type="checkbox"/>
	E_PASSP_NUMBER	nvarchar(12)	<input type="checkbox"/>
	E_PASSP_ORG	nvarchar(50)	<input type="checkbox"/>
	E_PASSP_DATE	date	<input type="checkbox"/>
	E_ADDRESS	nvarchar(80)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Таблица Сотрудники (Employees) имеет следующий набор столбцов:



- E_ID –идентификатор сотрудника (первичный ключ, уникальный);
- E_SURNAME – фамилия сотрудника (обязательное, строка 20 символов);
- E_NAME – имя сотрудника (обязательное, строка 20 символов);
- E_FNAME – отчество (не обязательно, строка 30 символов);
- E_BIRTHDAY – дата рождения (дата);
- E_SEX – пол (логический тип данных);
- E_POST – идентификатор должности, служит для связи с таблицей «должности» (posts), в которой хранятся все должности;
- E_ROOM – номер комнаты (составной внешний ключ, служит для связи с таблицей «Комнаты»;
- E_TEL – номер телефона (составной внешний ключ, служит для связи с таблицей «Комнаты»;
- E_INN – идентификационный номер налогоплательщика;
- E_PASSP_NUMBER – номер паспорта (строка 12 символов, обязательное);
- E_PASSP_ORG – кем выдан паспорт(строка, 50 символов, обязательной);
- E_PASSP_DATE – дата выдачи паспорта (дата, обязательное);

- E_ADDRESS – адрес проживания (строка 80 символов).


Так как, поле E_ID будет являться первичным полем связи, то мы должны сделать его числовым счётчиком. То есть данное поле должно автоматически заполняться числовыми значениями. Более того, оно должно быть ключевым. Для этого выделите поле, просто щёлкнув по нему мышкой в таблице определения полей. В таблице свойств столбца отобразятся свойства поля E_ID. Разверните группу свойств Спецификация идентификатора Свойство (Идентификатор) установите в значение Да. Задайте свойства Начальное значение идентификатора и Шаг приращения идентификатора равными 1


Сжатый тип данных	bigint
<input checked="" type="checkbox"/> Спецификация вычисляемого столбца	
<input checked="" type="checkbox"/> Спецификация идентификатора	Да
(Идентификатор)	Да
Начальное значение идентификатора	1
Шаг приращения идентификатора	1
<input checked="" type="checkbox"/> Спецификация полнотекстового столбца	Нет

Эти настройки показывают, что значение поля E_ID у первой записи в таблице будет равным 1, у второй – 2, у третьей 3 и т.д.

8.Теперь сделаем поле E_ID ключевым полем. Выделите поле, а затем на панели инструментов нажмите кнопку с изображением ключа . В таблице определения полей, рядом с полем E_ID появится изображение ключа  SID, говорящее о том, что поле ключевое.

На этом настройку таблицы «Сотрудники» можно считать за данным этапом завершённой.

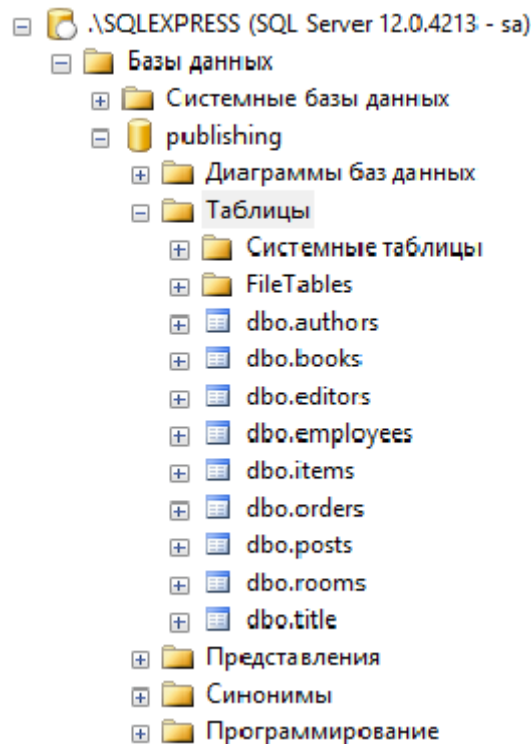
9.Закройте окно создания новой таблицы, нажав кнопку закрытия  в верхнем правом углу окна, над таблицей определения полей.

IDEA-PC\SQLXPRES...ing - dbo.Table_1* 	
Имя столбца	Тип
E_SURNAME	nvarchar(20)
E_NAME	nvarchar(20)

Появится окно с запросом о сохранении таблицы. В этом окне необходимо нажать Да. Появится окно Выбор имени, предназначенное для определения имени новой таблицы. В этом окне задайте имя новой таблицы как «employees» и нажмите кнопку ОК. Таблица «employees» отобразится в обозревателе объектов в папке Таблицы базы данных «publishing».

Выбор имени	?	×
Введите имя таблицы:		
<input type="text" value="employees"/>		
OK		Отмена

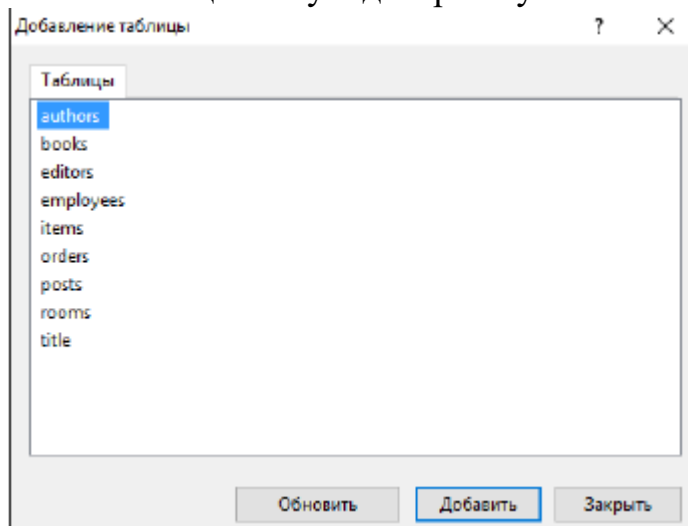
10. По аналогии создадим все таблицы



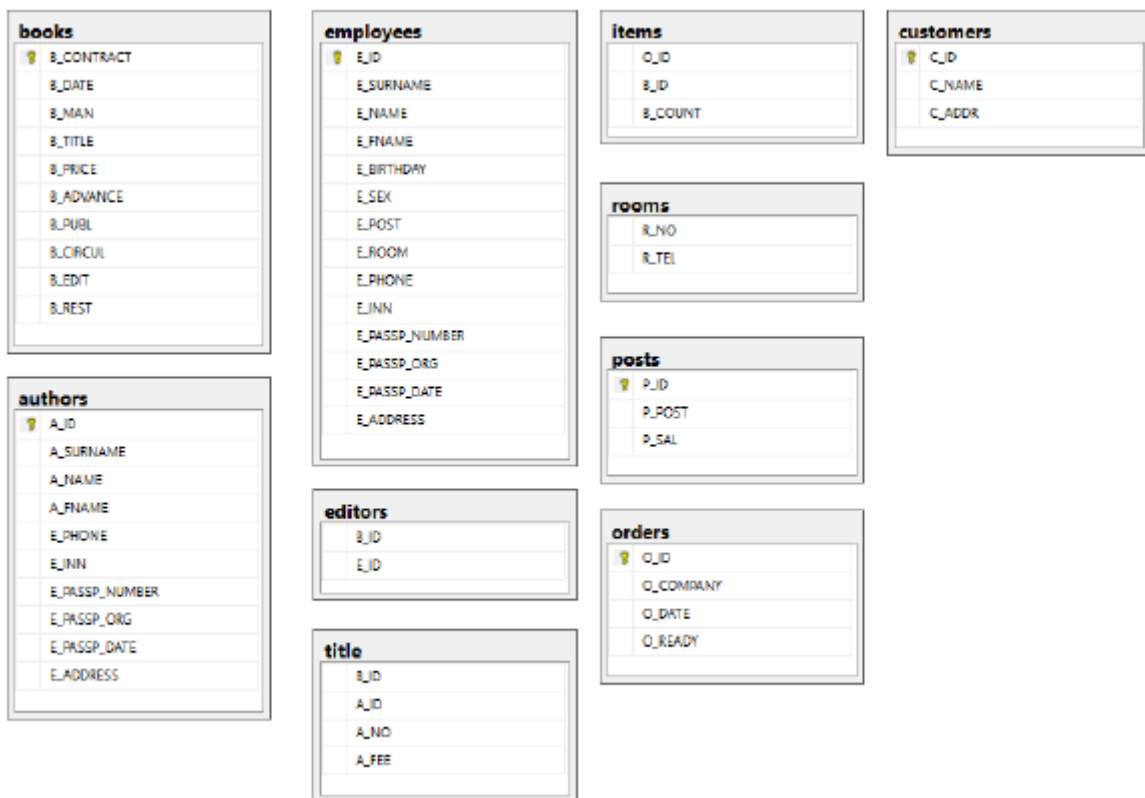
11. Связывание таблиц

Создадим диаграмму, обеспечивающую целостность данных нашей БД «publishing».

Для создания новой диаграммы в БД «publishing» щёлкните правой кнопкой мыши по папке Диаграммы базы данных и в появившемся меню выберем пункт Новая диаграмма базы данных. Сначала появится окно с вопросом о добавлении нового объекта Диаграмма. В этом окне нужно нажать кнопку Да. Затем появится окно «Добавление таблицы» предназначенное для добавления таблиц в новую диаграмму.



В окне добавления таблиц выделите все таблицы нашей БД и нажмите кнопку «Добавить». Закройте окно «Добавление таблицы» нажатием на кнопку «Закрыть». Появится окно диаграммы, где будут отображены отобранные таблицы. Обратите внимание, что на данном этапе таблицы являются не связанными, т.е. отсутствует ограничение целостности по внешнему ключу



12. Теперь необходимо определить связи между таблицами. Давайте организуем связь между сотрудником и должностью. Перед этим уточним, что в случае если мы захотим удалить из БД должность, то необходимо обнулить должность у сотрудника. Для этого необходимо перетащить поле P_ID из таблицы Posts на такое же поле в таблице Employees. Появится окно создания связи между таблицами «Таблицы и столбцы»

Таблицы и столбцы

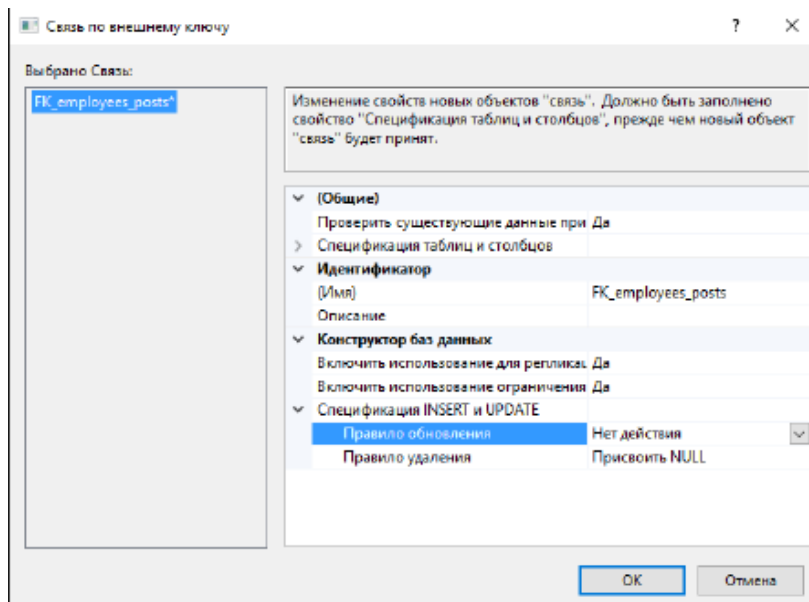
Имя связи:

Таблица первичного ключа: Таблица внешнего ключа:

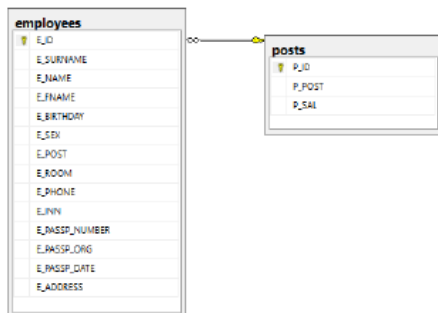
P_ID	E_POST

OK Отмена

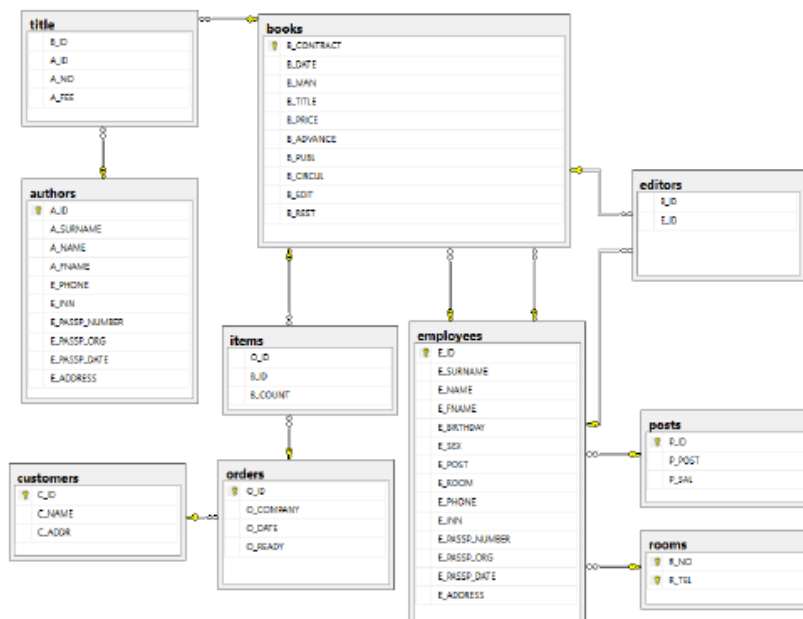
В окне создания связи нажмите кнопку ОК. Появится окно настройки свойств связи Связь по внешнему ключу



В диаграмме между таблицами employees и posts появится связь в виде линии.



13. Аналогично для всех остальных таблиц необходимо проделать те же самые действия. В итоге схема отношений будет выглядеть следующим образом.



14. Добавление данных в таблицу. Заполнение таблиц производится при помощи следующей команды:

```
INSERT <Имя таблицы> (<Список полей>)
VALUES (<Значения полей>)
```

ПРАКТИЧЕСКАЯ РАБОТА № 9

Тема: Создание БД в среде разработки

Цели работы: научить создавать БД в среде разработки MS SQL Server Management Studio.

Оборудование: ПК, интернет, программное обеспечение – MS SQL Server Management Studio, инструкции по выполнению работы.

Содержание работы:

Задание 1: Создать БД проектной организации (спроектированную в предыдущей работе № 1) на языке SQL. Основной вид деятельности такой организации – выполнение проектов по договорам с заказчиками.

Создание таблиц

1. Отношение Departs (отделы):

```
create table departs (  
  d_id varchar(12) primary key,  
  d_name varchar(100) not null);
```

2. Отношение Rooms (комнаты):

```
create table rooms (  
  d_depart varchar(12) references departs(d_id),  
  r_room numeric(4) not null,  
  r_phone varchar(20),  
  unique(r_room, r_phone));
```

3. Отношение Posts (должности):

```
create table posts (  
  p_post varchar(30) primary key,  
  p_salary numeric(8,2) not null check(p_salary>=4500));
```

4. Отношение Employees (сотрудники):

```
create table employees (  
  e_id numeric(4) primary key,  
  e_fname varchar(25) not null,  
  e_lname varchar(30) not null,  
  e_born date not null,  
  e_sex char(1) check(e_sex in ('ж','м')),  
  e_pasp char(10) not null unique,  
  e_date date not null,  
  e_given varchar(50) not null,  
  e_inn char(12) not null unique,  
  e_pens char(14) not null unique,  
  e_depart varchar(12) references departs,  
  e_post varchar(30) references posts,  
  e_room numeric(4) not null,  
  e_phone varchar(20) not null,  
  e_login varchar(30),  
  foreign key(e_room,e_phone)  
  references rooms(r_room,r_phone));
```

(Если внешний ключ ссылается на первичный ключ отношения, его можно не указывать, как в случае ссылок на Departs и Posts).

5. Отношение Edu (образование):

```
create table edu (  
  u_id numeric(4) references employees,  
  u_type varchar(20) not null,  
  u_spec varchar(40),  
  u_diplom varchar(15),  
  u_year number(4) not null,  
  check(u_spec in ('начальное', 'среднее', 'высшее', 'средне-специальное')));
```

6. Отношение AdrTel (адреса-телефоны):

```
create table adrtel (  
  a_id numeric(4) references employees,  
  a_adr varchar(50),  
  a_phone varchar(30));
```

7. Отношение Clients (заказчики):

```
create table clients (  
  c_id numeric(4) primary key,  
  c_company varchar(40) not null,  
  c_adr varchar(50) not null,  
  c_person varchar(50) not null,  
  c_phone varchar(30));
```

8. Отношение Projects (проекты):

```
create table projects (  
  p_id numeric(6) not null unique,  
  p_title varchar(100) not null,  
  p_abbr char(10) primary key,  
  p_depart varchar(12) references departs,  
  p_company numeric(4) references clients,  
  p_chief numeric(4) references employees,  
  p_begin date not null,  
  p_end date not null,  
  p_finish date,  
  p_cost numeric(10) not null check(p_cost>0),  
  check (p_end>p_begin),  
  check (p_finish is null or p_finish>p_begin));
```

9. Отношение Stages (этапы проектов):

```
create table stages (  
  s_pro char(10) references projects,  
  s_num numeric(2) not null,  
  s_title varchar(200) not null,  
  s_begin date not null,  
  s_end date not null,  
  s_finish date,  
  s_cost numeric(10) not null,
```

```
s_sum numeric(10) not null,
s_form varchar(100) not null,
check (s_cost>0),
check (s_end>s_begin),
check (s_finish is null or s_finish>s_begin));
```

10. Отношение Job (участие):

```
create table job (
j_pro char(10) references projects,
j_emp numeric(2) references employees,
j_role varchar(20) not null,
j_bonus numeric(2) not null,
check(j_bonus>0),
check (j_role in ('исполнитель', 'консультант')));
```

Создание представлений (готовых запросов)

Приведём примеры нескольких готовых запросов (представлений):

1. Список всех текущих проектов (sysdate – функция, возвращающая текущую дату, определена в СУБД Oracle; в других системах аналогичная функция может называться по-другому, например, getdate() в Transact-SQL, now() в MS Access, currrdate() в MySQL и т.д.):

```
create view curr_projects as
select *
from projects
where p_begin<=sysdate and sysdate<=p_end;
```

2. Определение суммы по текущим проектам, полученной на текущую дату:

```
create or replace view summ (title, cost, total) as
select p_title, p_cost, sum(s_sum)
from curr_projects, stages
where p_abbr=s_pro
group by p_title, p_cost;
```

3. Список сотрудников, участвующих в текущих проектах:

```
create view participants (project, name, role) as
select p_abbr, e_fname||' '||e_lname, 'руководитель'
from curr_projects, employees
where p_chief=e_id
union all
select p_abbr, e_fname||' '||e_lname, j_role
from curr_projects, employees, job
where p_abbr=j_pro and e_id=j_emp
order by 1, 3 desc;
```

4. Список рабочих телефонов сотрудников:

```
create or replace view worktel (name, room, phone) as
select e_fname||' '||e_lname, e_room, e_phone
from employees
order by 1;
```

5. Форма отчётности и сроки выполнения этапов по текущим проектам:

```
create or replace view reports as
select s_pro, s_num, s_title, s_begin, s_end, s_form
from stages
order by 1, 2;
```

6. Данные о проектах для руководителя проектов:

```
create or replace view my_projects as
select *
from projects p
where exists (select * from employees e
where e.e_id=p.p_chief and e.e_login=user);
```

Функция user возвращает имя пользователя, выполняющего текущий запрос. Таким образом, каждый пользователь получит данные только о тех проектах, руководителем которых является. Используя аналогичный способ, можно ограничить участника проекта данными только о сотрудниках тех проектов, в которых он сам участвует.

7. Данные об этапах проектов для руководителя проектов:

```
create or replace view my_stages as
select s.*
from stages s
where exists (select *
from employees e, projects p
where e.e_id=p.p_chief and e.e_login=user
and s.s_pro=p.p_abbr);
```

8. Данные об участниках проектов для руководителя проектов:

```
create or replace view my_staff as
select j.*
from job j
where exists (select *
from employees e, projects p
where e.e_id=p.p_chief and e.e_login=user
and j.j_pro=p.p_abbr);
```

9. Данные о других участниках проекта:

```
create or replace view my_emps as
select je.j_pro, e.e_fname||' '||e.e_lname e_name,
e_depart, e_post, e_phone, e_room
from employees e, job je
where e.e_id=je.j_emp and exists (select *
from job jm, employees m
where m.e_id=jm.j_emp and
m.e_login=user and je.j_pro=jm.j_pro);
```

Для того чтобы можно было работать с этими представлениями, соответствующим пользователям нужно назначить права доступа к представлениям.

Права доступа к представлениям

Представления	Группы пользователей (роли)		
	Руководители организации	Руководители проектов	Участники проектов
Текущие проекты (curr_projects)	S	S	
Сумма по текущим проектам (summ)	S	S	
Рабочие телефоны (worktel)	S	S	S
Участники проектов (participants)	S	S	S
Отчетность (reports)	S	S	S
Проекты для руководителя (my_projects)		SIUD	
Стадии проектов (my_stages)		SIUD	
Участники проектов для руководителей (my_staff)		SIUD	
Участники проектов (my_emps)			S

ПРАКТИЧЕСКАЯ РАБОТА № 10

Тема: Создание БД в среде разработки

Цели работы: научить создавать БД в среде разработки MS SQL Server Management Studio.

Оборудование: ПК, интернет, программное обеспечение – MS SQL Server Management Studio, инструкции по выполнению работы.

Содержание работы:

Задание 1: Создать таблицы, представляющие собой фрагмент базы данных торговой фирмы.

Таблица «Партнер» с полями: «Код партнера» - числовое поле, размер поля – целое, ключевое поле; «ИНН» - числовое поле, размер поля – целое, обеспечить проверку на уникальность поля; «КПП» - числовое поле, размер поля – целое, обеспечить проверку на уникальность поля; «Юридический статус» - текстовое поле, размер поля – 50; «Наименование» - текстовое поле, размер поля – 150; «Юридический адрес» - текстовое поле, размер поля – 150; «Адрес e-mail» - текстовое поле, размер поля – 20.

Таблица «Телефоны партнеров»: «Код партнера» - числовое поле, размер поля – целое; «Телефон» - текстовое поле, размер поля – 15. Создать составной ключ, включающий оба эти поля.

Таблица «Товары»: «Код товара» - числовое поле, размер поля – целое, ключевое поле; «Наименование товара» - текстовое поле, размер поля – 100; «Цена» - денежный тип данных; «Наличие на складе» - числовое поле, размер поля – целое.

Таблица «Сотрудник»: «Код сотрудника» - тип данных «Счетчик», поле создать автоматически при завершении описания структуры таблицы, после чего переименовать его; «ФИО» - текстовое поле, размер поля – 30.

Таблица «Счета»: «Номер счета» - тип данных «Счетчик», поле создать автоматически как ключ таблицы; «Код партнера» - числовое поле, размер поля – целое; «Дата» - поле типа «Дата/время»; «Код менеджера» - числовое поле, размер поля – длинное целое.

Таблица «Позиция счета»: «Номер счета» - числовое поле, размер поля – длинное целое; «Код товара» - числовое поле, размер поля – целое; «Количество» - числовое поле, размер поля – целое. Создать составной ключ, включающий поля «Номер счета» и «Код товара».

Создать схему данных для таблиц.

ПРАКТИЧЕСКАЯ РАБОТА № 11

Тема: Создание БД в среде разработки

Цели работы: научить создавать БД в среде разработки MS SQL Server Management Studio.

Оборудование: ПК, интернет, программное обеспечение – MS SQL Server Management Studio, инструкции по выполнению работы.

Содержание работы:

Задание 1. Создать таблицы, представляющие собой фрагмент базы данных учебного центра.

Таблица «Курс» с полями: «Код курса» - поле типа «Счетчик», создать автоматически как ключ при завершении описания таблицы; «Наименование курса» - текстовое поле длиной 120 символов; «Продолжительность» - числовое поле, размер поля – целое; «Стоимость обучения» - поле денежного типа.

Таблица «Преподаватель» с полями: «Код преподавателя» - поле типа «Счетчик», ключевое поле; «ФИО преподавателя» - текстовое поле, 50 символов; «Дата рождения» - поле типа «Дата/время», «Должность» - текстовое поле, 25 символов, «Научно-педагогический стаж» - числовое; «Общий стаж работы» - числовое; «Контактный телефон» - текстовое поле, 15 символов.

Таблица «Владение предметами» с полями: «Код преподавателя», «Код курса» - числовое поле, размер поля – длинное целое. Создать составной ключ, включающий оба эти поля.

Таблица «График учебного процесса» с полями: «Код потока» - поле типа «Счетчик», создать автоматически как ключ при завершении описания таблицы; «Код курса» - числовое поле, размер поля – длинное целое; «Дата начала» - поле типа «Дата/время»; «Дата завершения» - поле типа «Дата/время»; «Время начала» - поле типа «Дата/время»; «Время завершения» - поле типа «Дата/время».

Таблица «Слушатель» с полями: «Код слушателя» - числовое поле, размер поля – длинное целое, поле ключа; «ФИО слушателя» - текстовое поле, 50 символов; «Контактный телефон» - текстовое поле, 10 символов.

Таблица «Запись на курс» с полями: «Код потока» - числовое поле, размер поля – длинное целое; «Код слушателя» - числовое поле, размер поля – длинное целое. Создать составной ключ, включающий оба эти поля.

Создать схему данных для таблиц.

ПРАКТИЧЕСКАЯ РАБОТА № 12

Тема: Создание БД в среде разработки

Цели работы: научить создавать БД в среде разработки MS SQL Server Management Studio.

Оборудование: ПК, интернет, программное обеспечение – MS SQL Server Management Studio, инструкции по выполнению работы.

Содержание работы:

Задание 1: Создать БД на языке SQL, спроектированные в работе 3 – 7 (по вариантам).

ПРАКТИЧЕСКАЯ РАБОТА № 13

Тема: Экспорт данных базы в документы пользователя

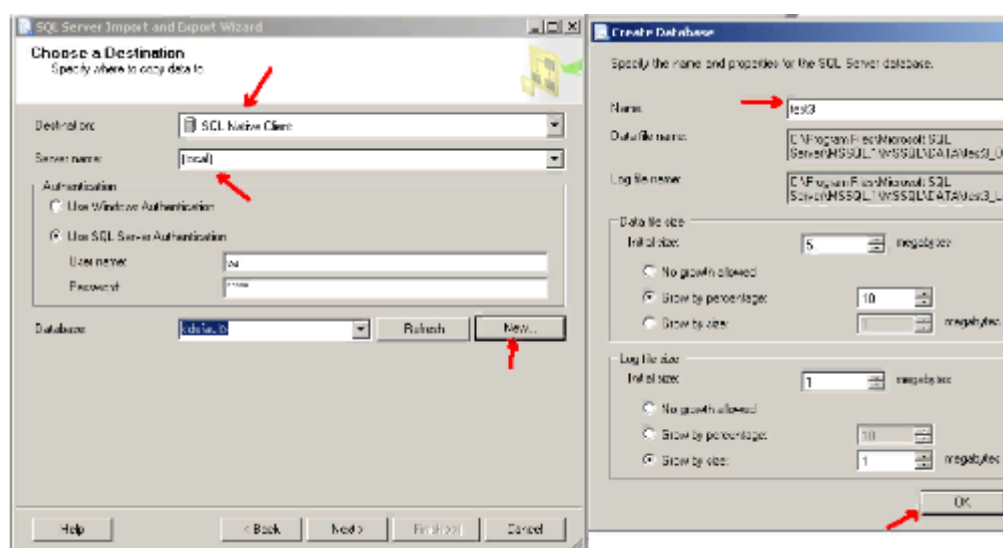
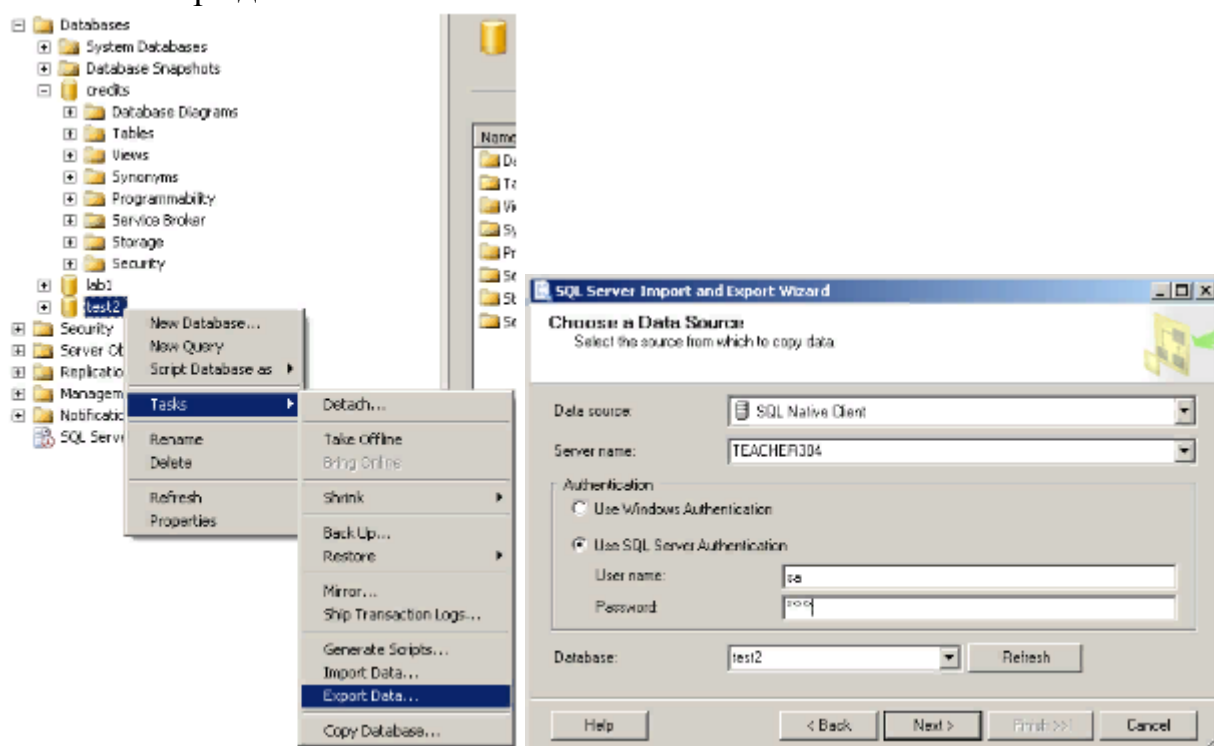
Цель работы: Научить создавать компоненты БД в программе MS SQL Server Management Studio, обменивать информацию между программами MS Office 2010.

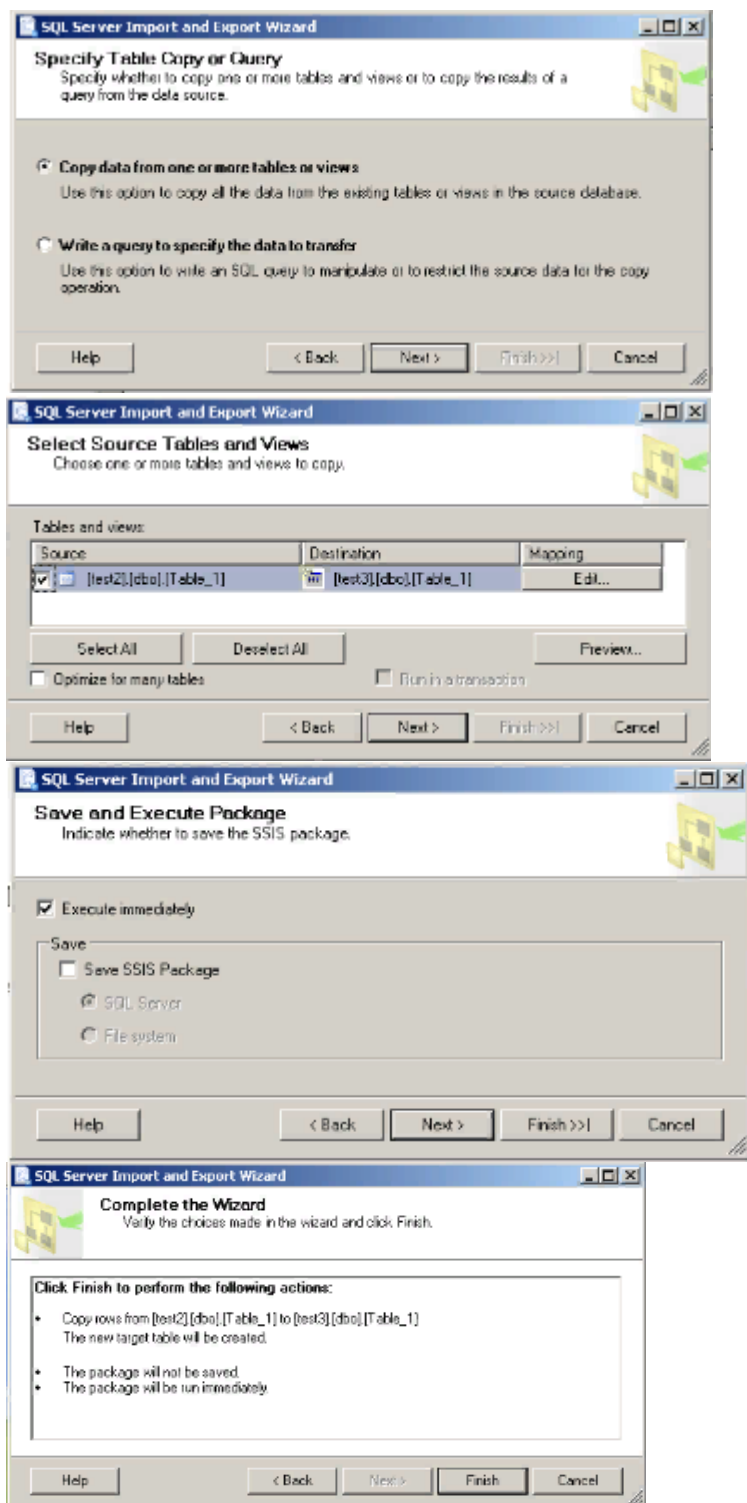
Оборудование: ПК, интернет, программное обеспечение – MS SQL Server Management Studio, MS Office, инструкции по выполнению работы.

Содержание работы:

Задание 1. Экспорт данных, созданную в Практической работе № 8 в другую СУБД

Экспорт данных выполняется по шагам:





Задание 2. Выполним экспорт данных из таблицы Сотрудники (проекты) в текстовый документ в формате RTF. Для этого выполните следующие простые действия:

1. Откройте базу данных, созданную в Практической работе № 8.
2. Перейдите на закладку **Таблицы**, выделите таблицу Сотрудники.
3. Выполните меню **Файл/Экспорт**.
4. Укажите тип файла – формат RTF, имя файла оставьте без изменения, укажите свою папку и экспортируйте туда файл.

Процесс экспорта закончен. Чтобы просмотреть результат откройте свою папку и откройте тестовый файл Сотрудники.rtf.

ПРАКТИЧЕСКАЯ РАБОТА № 14

Тема: Экспорт данных базы в документы пользователя

Цель работы: Научить создавать компоненты БД в программе MS SQL Server Management Studio, обменивать информацию между программами MS Office 2010.

Оборудование: ПК, интернет, программное обеспечение – MS SQL Server Management Studio, MS Office, инструкции по выполнению работы.

Содержание работы:

Задание 1. Выполнить экспорт данных из БД «Учебный центр», созданной в Практической работе № 11, в документы формата .docx.

Задание 2. Выполнить экспорт данных из БД «Торговая фирма», созданной в Практической работе № 10, в документы формата .docx.

ПРАКТИЧЕСКАЯ РАБОТА № 15

Тема: Импорт данных пользователя в базу данных

Цель работы: Научить создавать компоненты БД в программе MS SQL Server Management Studio, обменивать информацию между программами MS Office 2010.

Оборудование: ПК, интернет, программное обеспечение – MS SQL Server Management Studio, MS Office, инструкции по выполнению работы.

Содержание работы:

Задание 1. В Excel хранятся данные о поставщиках (файл ПоставщикиExcel.xls), клиентах (КлиентыExcel.xls), товарах (ТоварыExcel.xls). Откройте эти файлы на своих компьютерах (D:/Studentu/Access/) и просмотрите их.

Чтобы эти данные не вводить в базу данных можно воспользоваться мастером **Импорта данных**. Для этого выполните следующие действия:

1. Откройте базу данных.
2. Перейдите на вкладку **Таблицы**.
3. Откройте меню **Файл/Внешние данные/Импорт**.
4. Укажите путь к файлу **ПоставщикиExcel.xls** и тип файлов – Microsoft Excel (*.xls) и нажмите кнопку **Импорт**.
5. У вас появится диалоговое окно мастера Импорта данных.
6. В первом шаге мастера установите флаг «**Первая строка содержит заголовки столбцов**». Нажмите **Далее**.
7. Во втором шаге мастер будет спрашивать, куда сохранять данные - либо в новую таблицу, либо в уже созданную. Выберите **созданную** таблицу Поставщики. Нажмите **Далее**.
8. В следующем шаге мастер ждет подтверждения импорта данных в выбранную таблицу. Нажмите **Готово**.

Если в вашей таблице Поставщики все поля были созданы верно и имена совпадают с именами полей добавляемой таблицы ПоставщикиExcel.xls и порядок следования имен полей тоже совпадают, то в результате импорта будут добавлены данные о Поставщиках. Откройте таблицу и просмотрите ее.

Самостоятельно аналогично добавьте данные в таблицы **Клиенты** и **Товары**.

Однако добавление данных этих трех таблиц недостаточно, так как мы не ввели главные данные, с которыми потом в дальнейшем будем работать, а именно это данные наших приходных и расходных накладных.

Необходимо добавить данные из таблиц ПриходExcel.xls, Ввод_приходаExcel.xls и РасходExcel.xls, Ввод_расходаExcel.xls.

И тут возникает проблема. Дело в том, чтобы при осуществлении операции импорта данных возможно однозначное добавление данных в простые таблицы. При этом данные добавляются в список, где каждой новой записи соответствует номер из добавленной таблицы.

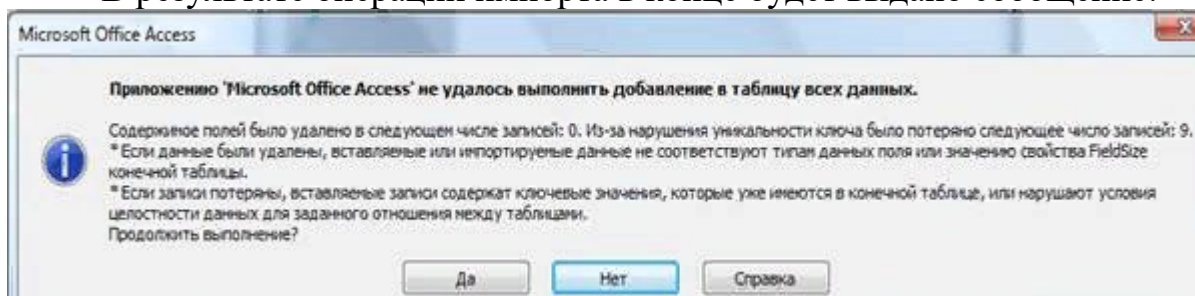
Например, если таблица **Товары** содержала бы следующие данные:

Код_товара	Название_товара	Цена_приходная	Цена_расходная	Примечание	Единица_измерения
	Телевизор Sony	\$2,00	\$4,00		шт
	Телевизор Nokia	\$3,00	\$3,50		шт
	Флеш-носитель 1Гб	\$2,00	\$3,00		шт
	Флеш-носитель 2Гб	\$3,00	\$3,50		шт
	Флеш-носитель 4Гб	\$4,00	\$4,50		шт
	Флеш-носитель 8Гб	\$5,00	\$5,80		шт
	Фото-принтер	\$80,00	\$90,00		шт
	Лазерный принтер	\$51,00	\$15,00		шт
	Струнный принтер	\$120,00	\$170,00		шт

А необходимо добавить данные из таблицы **ТоварыExcel.xls**:

Код_товара	Название_товара	Цена_приходная	Цена_расходная	Примечание	Единица_измерения
	Каша рисовая	2,00р.	4,00р.		литр
	Каша гречневая	3,00р.	3,50р.		ящик
	Каша с курицей	2,00р.	3,00р.		литр
	Каша с грибами	3,00р.	3,50р.		ящик
	Крышка закаточная	4,00р.	4,50р.		ящик
	Паста томатная	5,00р.	5,80р.		шт
	Печенье овсяное	8,00р.	9,00р.		шт
	Печенье ороматное	4,00р.	4,30р.		шт
	Молоко	6,70р.	7,00р.		шт
	Сливки	12,10р.	12,30р.		шт
	Масло сливочное	5,00р.	8,00р.		шт
	Масло оливковое	1,00р.	2,00р.		шт
	Повидло яблочное	2,00р.	5,00р.		ящик
	Повидло сливовое	1,00р.	2,00р.		ящик
	Повидло абрикосовое	4,00р.	6,00р.		шт
	Торт Изобелла	1,00р.	2,00р.		шт
	Торт классический	6,00р.	7,00р.		шт
	Торт пикатный	6,00р.	7,00р.		шт
	Кофе Нескафе	9,80р.	10,00р.		
	Кофе Классик	9,60р.	10,00р.		шт

В результате операции импорта в конце будет выдано сообщение:



Если вы нажмете кнопку **Да**, то после операции импорта данных получится следующее:

Код_товара	Название_товара	Цена_приходная	Цена_расходная	Примечание	Единица_измерения
	Телевизор Sony	\$2,00	\$4,00		шт
	Телевизор Nokia	\$3,00	\$3,50		шт
	Флеш-носитель 1Гб	\$2,00	\$3,00		шт
	Флеш-носитель 2Гб	\$3,00	\$3,50		шт
	Флеш-носитель 4Гб	\$4,00	\$4,50		шт
	Флеш-носитель 8Гб	\$5,00	\$5,80		шт
	Фото-принтер	\$80,00	\$90,00		шт
	Лазерный принтер	\$51,00	\$15,00		шт
	Струнный принтер	\$120,00	\$170,00		шт
	Сливки	\$12,10	\$12,30		шт
	Масло сливочное	\$5,00	\$8,00		шт
	Масло оливковое	\$1,00	\$2,00		шт
	Повидло яблочное	\$2,00	\$5,00		ящик
	Повидло сливочное	\$1,00	\$2,00		ящик
	Повидло абрикосовое	\$4,00	\$6,00		шт
	Торт Изобелла	\$1,00	\$2,00		шт
	Торт классический	\$6,00	\$7,00		шт
	Торт пикатный	\$6,00	\$7,00		шт
	Кофе Нескафе	\$9,80	\$10,00		шт
	Кофе Классик	\$9,60	\$10,00		шт

Если проанализировать, то первые 9 записей остались без изменения, а новые данные добавились в конец списка, где номер кода товара соответствуют кодам товаров из добавляемой таблицы.

А если мы хотим добавить данные в связанные таблицы, как это в таблицах Приход и Ввод_прихода и Расход и Ввод_Расхода, то простой операцией импорта ничего не получится, так как мы не сможем обеспечить однозначность и целостность данных.

С чем это связано? Дело в том, что при выполнении операции импорта данных будут добавлены записи под старыми кодами. А коды в связанных таблицах должны совпадать, например код_прихода или код_расхода. Код_прихода или код_расхода являются ключевыми полями, они однозначно определяют набор записей в связанных таблицах Приход и Ввод_приход или Расход и Ввод_расхода.

Например, запись в таблице Приход

Код_прихода	Поставщик	Дата_накладной	Идентифик№поставщика	ФИОпоставщика
	МЧП ЛЕЛЕКА	02.10.2002		Моринский

Однозначно соответствуют записи в таблице Ввод_прихода:

Код_ввода_прихода	Код_прихода	Товар	цена	кол-во	сумма
		Торт классический	6,00р.		12,00р.
		Масло оливковое	1,00р.		89,00р.

Проблема заключается в том, что нам нужно будет добавить данные не в одну таблицу, а сразу в две связанные. При этом вначале необходимо добавить

данные в главную таблицу, например Приход. Сразу же возникает особенность данной операции. Если бы мы выполняли добавление через импорт данных, то будут созданы записи под кодами, совпадающими с таблицей ПриходExcel.xls, при этом будут частично потеряны данные. Затем необходимо будет добавить данные в подчиненную ей таблицу Ввод_прихода. А так как уже были потеряны данные, то и к связанной таблице они тоже не добавятся.

Как же решить данную проблему? Решим ее с помощью специального вида запросов – **запроса на добавление**. При этом мы будем добавлять все записи в конец списка и создавать новые уникальные ключи записей.

Но прежде чем переходить к этим видам запросов создадим новые таблицы для хранения внешних данных.

Для этого выполните следующие действия:

1. Перейдите на вкладку **Таблицы**.
2. Откройте меню **Файл/Внешние данные/Импорт.../**
3. Выберите Тип файлов – **Microsoft Excel (*.xls)**
4. Найдите файл под именем **ПриходExcel.xls** и нажмите кнопку **Импорт**.
5. В открывшемся диалоговом окне мастера Импорта электронной таблицы на вопрос «Файлы электронной таблицы содержит несколько листов и диапазонов. Выберите нужный объект» установите опцию **Листы**.
6. Во втором шаге мастера выберите Первая строка содержит заголовки столбцов.
7. На третьем шаге мастера выберите Данные необходимо добавить в **новую таблицу**.
8. На четвертом шаге имеется возможность описать каждое поле импорта. По умолчанию для всех полей автоматически определяются типы данных, кроме индексированного поля. Для этого выберите из списка Имя поля– поле **Код_прихода**, а в списке Индекс выберите – **Да (совпадения не допускаются)**.
9. На пятом шаге мастера рекомендуется создать ключевое поле либо автоматически, либо самостоятельно определить, либо не создавать ключ. Выберите Определить ключ – **Код_прихода**.
10. На шестом шаге мастера введите имя файла – **Приход_импорт**.

Самостоятельно аналогично создайте новые таблицы под именами Ввод_прихода_импорт, Расход_импорт, Ввод_расхода_импорт используя импортированные электронные таблицы.

ПРАКТИЧЕСКАЯ РАБОТА № 16

Тема: Импорт данных пользователя в базу данных

Цель работы: Научить создавать компоненты БД в программе MS SQL Server Management Studio, обменивать информацию между программами MS Office 2010.

Оборудование: ПК, интернет, программное обеспечение – MS SQL Server Management Studio, MS Office, инструкции по выполнению работы.

Содержание работы:

Задание 1. Произведите импорт данных из таблиц Excel в базу данных «Торговая фирма», созданную в Практической работе № 10

Задание 2. Произведите импорт данных из таблиц Excel в базу данных «Учебный центр», созданную в Практической работе № 11

ПРАКТИЧЕСКАЯ РАБОТА № 17

Тема: Выполнение настроек для автоматизации обслуживания БД

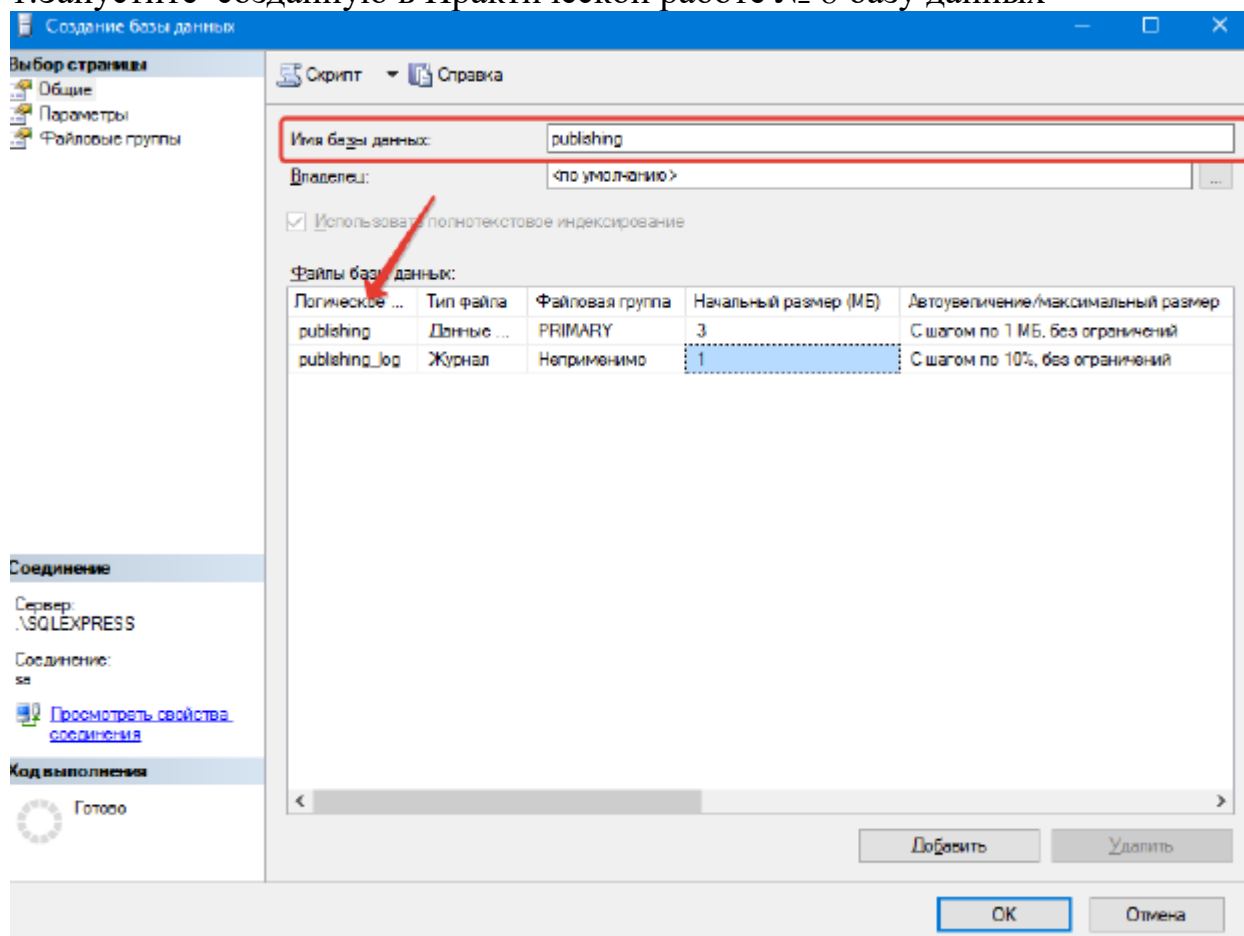
Цель работы: Освоение некоторых возможностей автоматизации управления базой данных. Создание и применение макросов. Создание пользовательского кнопочного и ниспадающего меню.

Оборудование: ПК, интернет, программное обеспечение – MS SQL Server Management Studio, инструкции по выполнению работы.

Содержание работы:

Задание 1. Произвести настройки параметров файла данных созданной в Практической работе № 8 базы данных.

1. Запустите созданную в Практической работе № 8 базу данных



В левой части окна настроек имеется список «Выбор страницы». Этот список позволяет переключаться между группами настроек.

2. Для начала настроим основные настройки. Для выбора основных настроек нужно просто щёлкнуть мышью по пункту «Общие» в списке «Выбор страницы». В правой части окна «Создание базы данных» появятся основные настройки.

В верхней части окна расположено два параметра: «Имя базы данных» и «Владелец». Задайте параметр «Имя базы данных», таким образом, чтобы имя полностью описывало Выбранную Вами предметную область, в первой лабораторной работе был рассмотрен процесс проектирования БД издательства, поэтому название будет соответствующее, но на обязательно на английском языке – «Publishing». Параметр Владелец оставьте без изменений.

Под вышеприведёнными параметрами в виде таблицы располагаются настройки файла данных и журнала транзакций. Таблица имеет следующие столбцы:

- Логическое имя – логическое имя файла данных и журнала транзакций. По этим именам будет происходить обращение к вышеприведённым файлам в БД. Можно заметить, что файл данных имеет то же имя что и БД, а имя файла журнала транзакций составлено из имени БД и суффикса «_log».
- Тип файла – этот параметр показывает, является ли файл файлом данных или журналом транзакций.
- Файловая группа – группа файлов, показывает к какой группе файлов относится файл. Группы файлов настраиваются в группе настроек Файловая группа.
- Начальный размер (МБ) – начальный размер файла данных и журнала транзакций в мегабайтах.
- Авторасширение – как только файл заполняется информацией его размер автоматически увеличивается на величину, указанную в параметре Авторасширение. Увеличение можно задавать как в мегабайтах так и в процентах. Здесь же можно задать максимальный размер файлов. Для изменения этого параметра надо нажать кнопку «...». В нашем случае размер файлов не ограничен. Файл данных увеличивается на 1 мегабайт, а файл журнала транзакций на 10%.
- Путь – путь к папке, где хранятся файлы. Для изменения этого параметра также надо нажать кнопку «...».
- Имена файлов – по умолчанию имена файлов аналогичны логическим именам. Однако файл данных имеет расширение .mdf, а файл журнала транзакций – расширение .ldf.

В рассматриваемом случае все основные настройки без изменений были оставлены без изменений.

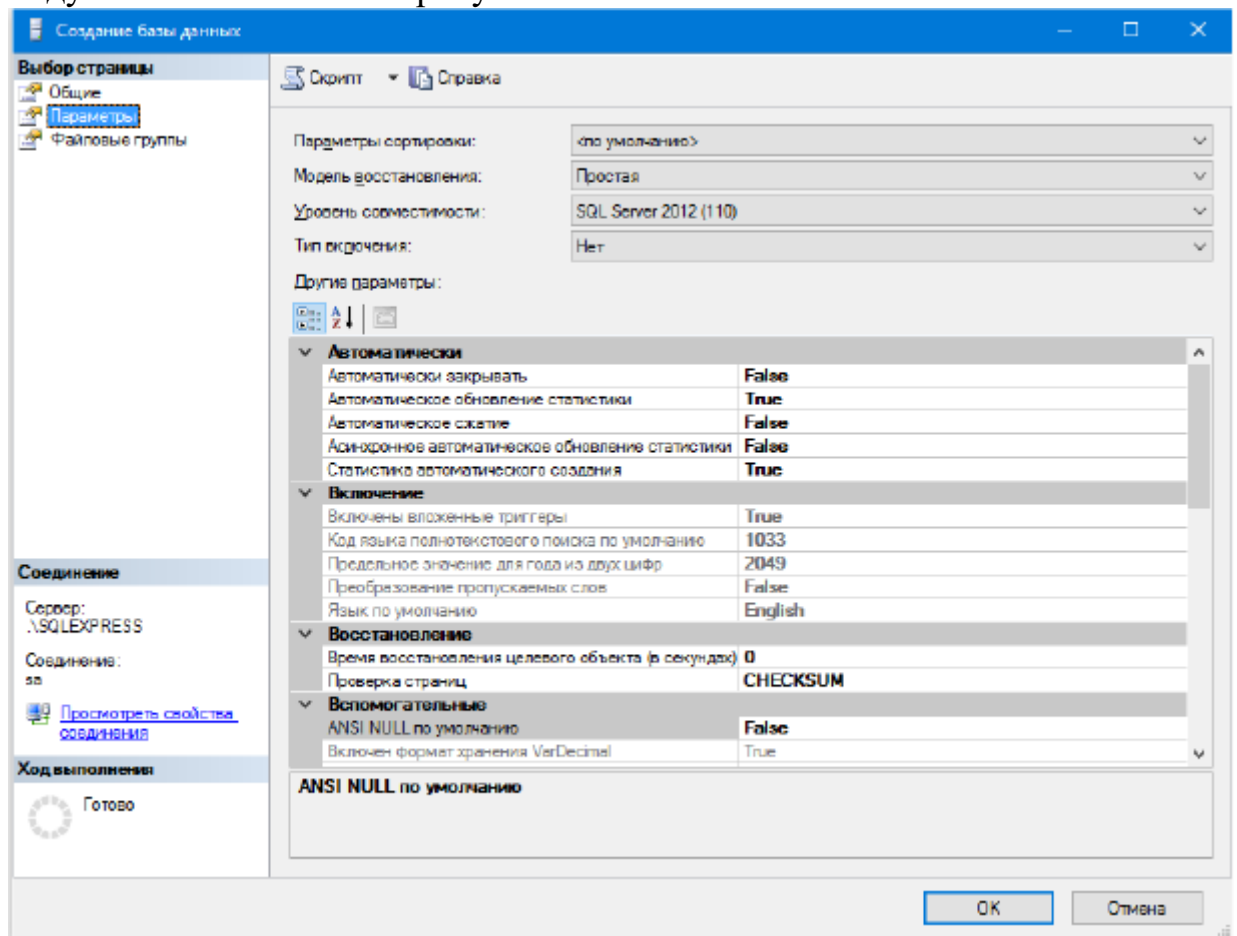
Теперь перейдём к другим второстепенным настройкам файла данных. Для доступа к этим настройкам необходимо щёлкнуть мышью по пункту Параметры в списке Выбор страницы. Появится следующее окно.

В правой части окна мы видим следующие настройки:

- Параметры сортировки – этот параметр отвечает за обработку текстовых строк, их сравнение, текстовый поиск и т.д. Рекомендуется оставить его как <по умолчанию сервера>. При этом данный параметр будет равен значению, заданному на вкладке Параметры сортировки, при установке сервера.
- Модель восстановления – данный параметр отвечает за информацию, предназначенную для восстановления БД, хранящуюся в файле транзакций. Чем полнее модель восстановления, тем больше вероятность восстановления данных при сбое системы или ошибках пользователей, но и больше размер файла журнала транзакций. При наличии места на диске, рекомендуется оставить этот параметр в значении Простая.
- Уровень совместимости – определяет совместимость файла данных с более ранними версиями сервера. Если планируется перенос данных на другую, более раннюю версию сервера, то её необходимо указать в этом параметре.

- Другие параметры – данные параметры являются необязательными для изменения.

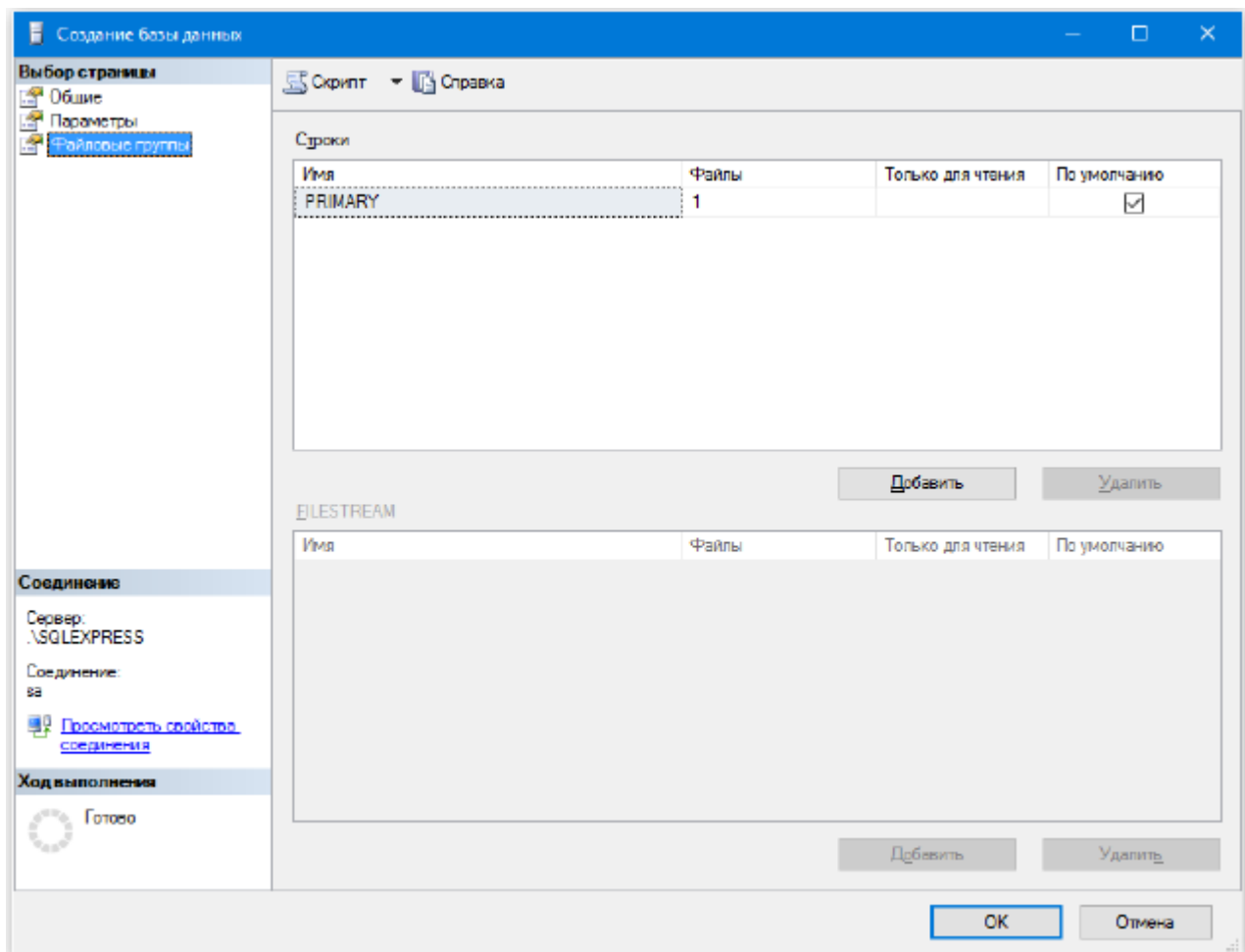
В нашем случае все параметры в разделе Другие параметры, рекомендуется оставить как на рисунке



3. Наконец рассмотрим последнюю группу настроек Файловые группы. Данная группа настроек отвечает за группы файлов. Группы файлов представлены в таблице Строки в правой части окна. Данная таблица имеет следующие столбцы:

- Имя – имя группы файлов.
- Файлы – количество файлов, входящих в группу.
- Только для чтения – файлы в группе будут только для чтения. То есть, их можно только просматривать, но нельзя изменять.
- По умолчанию – группа по умолчанию. Все новые файлы данных будут входить в эту группу.

В рассматриваемой БД нет необходимости добавлять новые группы файлов. Поэтому оставим группу настроек Файловые группы без изменений.



Для принятия всех настроек и создание фала данных и журнала транзакций нашей БД в окне «Создание базы данных» нажмём кнопку ОК. Произойдёт возврат в окно среду разработки SQL Server Management Studio.

ПРАКТИЧЕСКАЯ РАБОТА №18

Тема: Выполнение резервного копирования

Цель работы: Освоить технологию резервного копирования

Оборудование: ПК, интернет, программное обеспечение – MS SQL Server Management Studio, инструкции по выполнению работы.

Справочный материал:

Существует 2 режима создания резервных копий:

1. **Статическое резервное копирование** — режим, при котором в процессе создания копии только одна активная сессия, поддерживаемая системой, является той сессией, которая создает резервную копию. Другие пользовательские процессы во время выполнения копирования недопустимы.
2. **Динамическое резервное копирование** — режим, при котором копирование баз данных может выполняться без остановки сервера баз данных, удаления пользователей или даже закрытия файлов.

MS SQL Server поддерживает оба режима создания резервных копий.

MS SQL Server предоставляет 4 различных метода резервного копирования:

1. **Полное копирование базы данных (Full)** — При полном резервном копировании создается резервная копия всей базы данных целиком, она включает в себя схему всех таблиц, соответствующие файловые структуры, а также содержит все данные из этой базы на момент *завершения* резервного копирования. Это осуществляется благодаря тому, что полное копирование захватывает состояние базы данных на момент начала копирования. А затем, если копирование выполняется динамически, система записывает любые действия, которые имеют место в процессе создания копии.

2. **Дифференцированное (разностное) резервное копирование (Differential)** — В этом случае создается копия только частей баз данных, которые менялись с момента последнего полного копирования баз данных. Эта полная резервная копия называется **основой** для разностной копии. Как и в случае полного резервного копирования, любые действия, имеющие место во время создания копии, также копируются.

3. **Резервное копирование протокола транзакций (Transaction log)** — Данный вид копирования применяется при полной модели восстановления (или при неполном протоколировании) баз данных, и учитывает только изменения, записанные в протокол транзакций. Поэтому такая форма резервного копирования не основывается на физических страницах баз данных, а только на логических операциях.

4. **Резервное копирование файлов или файловых групп** — Данный метод позволяет копировать указанные файлы баз данных вместо копирования всей базы данных. Отдельные файлы могут быть восстановлены из копии, позволяя выполнить восстановление после сбоя, который повлиял лишь на небольшое подмножество файлов баз данных.

Содержание работы:

Задание 1. Выполнить резервное копирование баз данных «Проектная организация», «Учебный центр», «Студенты», «Торговая фирма».

ПРАКТИЧЕСКАЯ РАБОТА № 19

Тема: Восстановление БД из резервной копии

Цель работы: освоить технологию восстановления БД из резервной копии.

Оборудование: ПК, интернет, программное обеспечение – MS SQL Server Management Studio, инструкции по выполнению работы.

Справочный материал:

При восстановлении базы данных из резервной копии существующая база данных будет перезаписана.

Предусмотрены следующие модели восстановления:

- **полное восстановление (Full).** При его применении в журнал транзакций записывается информация обо всех операциях. Происходит восстановление БД из резервной копии с применением к ней всех операций, сохраненных в журнале. При использовании этой модели не происходит потери данных. То есть, когда важна сохранность информации, применяют именно эту модель;

- **восстановление без массовых операций (Bulk-Logged).** Эта модель может рассматриваться как упрощенный вариант полного восстановления. Запись в журнал информации об обычных транзакциях происходит так же, как в модели полного восстановления, а информация об операциях массового копирования (таких, как массовый импорт данных или создание индекса) записывается не в полном объеме. То есть восстановленная из резервной копии БД будет содержать информацию обо всех изменениях, не относящихся к массовым операциям, поэтому необходимо выполнить все массовые информации повторно;

- **простое восстановление (Simple).** В этой модели журнал транзакций применяется только для поддержки транзакций в ходе их осуществления. Регулярно производится очистка журнала, в ходе которой удаляется информация обо всех завершенных транзакциях. Такой журнал гораздо меньше и обеспечивает более высокую производительность, но он практически не пригоден для восстановления БД.

Для восстановления базы данных используется команда:

```
RESTORE { LOG | DATABASE } имя_БД 'файл_или_файловая_группа'  
[ FROM логическое_имя_устройства ]  
[ WITH [ DBO_ONLY ] ]  
[ MOVE 'логическое_имя_файла' TO 'физическое_имя' ] ... ]
```

DBO_ONLY – разрешается доступ к восстановленной базе только владельцам;
MOVE – указывает, какое физическое имя будет соответствовать восстанавливаемому файлу. По умолчанию файл восстанавливается с тем же физическим именем, которое было определено при резервном копировании.

Содержание работы:

Задание 1. Удалите свои базы данных (Студенты, Проектная организация, Учебный центр, Торговая фирма). Восстановите удаленные базы данных с созданного физического устройства. Проверьте результат.

ПРАКТИЧЕСКАЯ РАБОТА № 20

Тема: Реализация доступа пользователей к БД

Цель работы: научить применять способы защиты информации в БД на примере СУБД MS SQL Server Management Studio.

Оборудование: ПК, интернет, программное обеспечение – MS SQL Server Management Studio, инструкции по выполнению работы.

Справочный материал:

Система безопасности БД должна обеспечивать физическую целостность БД и защиту от несанкционированного вторжения с целью чтения содержимого и изменения данных.

Защита БД производится на двух уровнях:

- на уровне пароля;
- на уровне пользователя (защита учетных записей пользователей и идентифицированных объектов).

Компонентами системы безопасности SQL Server на уровне сервера являются: система аутентификации средствами Windows и средствами SQL Server, учетные записи пользователей и встроенные роли сервера. На уровне базы данных компонентами системы безопасности являются: идентификация пользователей баз данных, фиксированные и пользовательские роли баз данных, а также роли приложений.

Фиксированными ролями сервера являются:

Sysadmin – для выполнения любых действий в сервере;

Sereradmin– для конфигурирования и выключения сервера;

Setupadmin– для управления связанными серверами и процедурами, автоматически запускающимися при старте сервера;

Securityadmin– для управления учетными записями и правами на создание базы данных, а также для контроля журнала ошибок;

Processadmin - для управления процессами, запущенными на сервере

Dbcreator – для создания и модификации баз данных;

Diskadmin– для управления файлами сервера;

Bulcadmin– для массивного копирования баз данных

При создании базы данных сервер автоматически создает для нее фиксированные роли, которые, как и фиксированные роли сервера, нельзя удалить или модифицировать:

Db_owner – для выполнения любых действий в базе данных;

Db_accessadmin – для добавления и удаления пользователей;

Db_securityadmin – для управления всеми разрешениями, объектами, ролями и именами ролей;

Db_ddladmin – для выполнения любых команд DDL, кроме GRANT, DENY и REVOKE;

Db_backupoperator– для выполнения команд DBCC, CHECK, POINT и BACKUP;

Db_datareader – для контроля данных во всех таблицах базы данных и чтения;

Db_datawriter – для модификации данных в любых таблицах базы данных;

Db_denydatareader – для запрета просмотра данных в любой таблице базы данных;

Db_denydatawriter – для запрета модификации данных во всех таблицах базы данных.

Кроме этих фиксированных ролей любой базы данных есть еще одна роль public, членами которой автоматически становятся все пользователи, имеющие тот или иной доступ к базе данных. Эта роль имеет специальное назначение и обеспечивает минимальные права доступа к базе данных тем пользователям, для которых их права не определены явно. Эта роль имеется во всех базах данных, включая системные master, tempdb, msdb, model и не может быть удалена.

Если в базе данных разрешен пользователь quest, то установленный для public доступ будут иметь все пользователи, получившие доступ к SQL Server.

В отличие от сервера базы данных могут иметь пользовательские роли и роли приложения, которые создает администратор с помощью Enterprise Manager или Transact_SQL индивидуально для групп пользователей и групп приложений, наделяя их необходимыми правами доступа к конкретной базе данных.

В любую роль базы данных можно включать:

- а) пользователей сервера;
- б) роли сервера;
- в) пользователей Windows;
- г) группы пользователей Windows.

Средствами Enterprise Manager можно включать только пользователей сервера.

Содержание работы:

Задание 1. Создать учетную запись SQL сервера, используя графическую утилиту Enterprise Manager, выполнив следующие действия:

1. Выбрать нужный сервер.
2. Открыть папку Security этого сервера.
3. Выбрать объект Logins, щелкнув по соответствующему значку.
4. В правом окне просмотреть список учетных записей данного сервера:

Name – имя учетной записи сервера;

Type – происхождение учетной записи:

User W– пользователь Windows;

Group W– группа пользователей Windows;

Standard – пользователь SQL сервера;

Server Access – доступ к серверу SQL:

Permit – разрешен;

Deny – запрещен;

Default Database – база данных по умолчанию, к которой подключен пользователь(обязательный параметр)

User – имя пользователя базы данных;

Default Language – язык по умолчанию для данной учетной записи.

5. Для создания новой учетной записи сервера открыть контекстное меню объекта Logins, щелкнув по нему правой клавишей мыши или по значку на панели инструментов левой клавишей мыши.
6. В появившемся диалоговом окне на вкладке General (общие) ввести имя учетной записи в поле Name.
7. Выбрать тип аутентификации: Windows Authentication или SQL Server Authentication.
8. Если выбрана аутентификация Windows, то задать в поле Domain имя домена, в котором учтен пользователь или группа Windows. Имя заданного домена добавиться впереди имени пользователя также и в поле Name (для выбора домена использовать кнопку "...").
9. В группе Security Access (безопасный доступ) установить переключатель Grant Access (доступ разрешен). Установка переключателя Deny Access навсегда запретит регистрацию пользователя или домена Windows.
10. Если выбрана аутентификация SQL Server, то задать только пароль для учетной записи.
11. Задав параметры аутентификации Windows или SQL Server, указать в группе Defaults (умолчания) имя базы данных в поле Database, к которой пользователь будет подключаться автоматически, и язык Language (Russian). Если имя базы данных не задать, то сервер будет автоматически подключать к базе master.
12. Включить создаваемую учетную запись в требуемую встроенную роль сервера:
Sysadmin, Serveradmin, Setupadmin, Securityadmin, Processadmin, Dbcreator, Diskadmin, Bulkadmin, установив флажок против этой роли на вкладке Server Role.
13. На вкладке Database Access выбрать требуемую базу данных, установив флажок слева от ее имени, и задать имя пользователя, в которое будет отображаться рассматриваемая учетная запись, а в нижней части вкладки с помощью флажка включить пользователя в ту или иную роль в зависимости от его работы с базой данных.
14. Щелкнув по кнопке Properties (свойства) и просмотреть список пользователей, включенных в выбранную роль рассматриваемой базы данных.
15. Щелкнув по кнопке Permissions (права) и просмотреть список прав, предоставленных выбранной роли базы данных.
16. Закрыть все окна.
17. Вновь открыть список учетных записей сервера, дважды щелкнуть по вновь созданной записи и проверить правильность введенных параметров.
18. Закрыть все окна.
19. Приступить к работе с базами данных, используя новую учетную запись.

Задание 2. Включить учетную запись пользователя или группы пользователей Windows в фиксированную роль сервера SQL с помощью Enterprise Manager, выполнив следующие действия:

1. Выбрать требуемый сервер в левом окне Tree.
2. Открыть объекты сервера, щелкнув по его кнопке “+”.
3. Открыть объекты Security этого сервера, щелкнув по кнопке “+” для Security.
4. Выбрать объект Logins (записи) и щелкнуть по нему, при этом в правом окне откроется список учетных записей сервера.
5. Дважды щелкнуть по требуемой учетной записи сервера.
6. В открывшемся окне на Server Login Properties открыть вкладку Server role.
7. Установить флажки возле тех ролей сервера, в которые требуется включить конфигурируемую запись.
8. Закрыть все открытые окна, щелкая по кнопкам “ОК” этих окон.
9. Повторить задания, используя следующий набор команд:
 - а) Security/Server Rolees;
 - б) Щелкнуть левой клавишей;
 - в) В правом окне выбрать требуемую фиксированную роль;
 - г) Два раза щелкнуть по выбранной роли;
 - д) В открывшемся окне Server Role Properties щелкнуть по кнопке Add вкладки General;
 - е) Добавить учетные записи в заданную роль;
 - ж) Закрыть окно со списком учетных записей;
 - з) На вкладке Permission окна Server Role Properties просмотреть предоставляемые права для рассматриваемой роли.
10. Закрыть все открытые диалоговые окна, щелкая по кнопкам ОК.
11. Проверить правильность выполненных действий.

Задание 3. Создать нового пользователя базы данных для учетной записи Windows с помощью Enterprise Manager, выполнив следующие действия:

1. Выбрать требуемый сервер и требуемую базу данных в левом окне Tree.
2. Открыть объекты выбранной базы данных, щелкнув по значку “+” этой базы.
3. Выбрать в раскрывшемся списке объектов рассматриваемой базы данных объект Users (пользователи).
4. Щелкнуть правой клавишей мыши и открыть контекстное меню объекта Users (пользователи).
5. В контекстном меню исполнить команду New Database User (новый пользователь базы данных).
6. В открывшемся диалоговом окне ввести:
 - а) в поле Login Name – имя учетной записи пользователя или группы пользователей Windows;
 - б) в поле User Name – имя нового пользователя рассматриваемой базы данных.
7. Включить нового пользователя в необходимые роли базы данных: public, db – owner, db – denydatareader и т.д. Для этого требуемые роли надо отметить флажками в списке фиксированных ролей базы данных, расположенном в правой части окна.
8. Щелкнуть по кнопке Properties и, просмотрев список всех пользователей базы данных, убедиться, что новый пользователь включен этот список.

9. Щелкнуть по кнопке Permission и задать права доступа пользователя к объектам базы данных: SELECT, INSERT, UPDATE, DELET, EXEC, DRI. В окне находится полный список объектов базы данных.
10. Щелкнуть по кнопке Columns (столбцы) для выбранной базы данных и задать права доступа к конкретным столбцам таблицы: SELECT и/или UPDATE.
11. Закрыть все открытые диалоговые окна, щелкая по кнопкам ОК.
12. Проверить работу нового пользователя с рассматриваемой базой данных и его права.

Задание 4. Создать учетную запись SQL сервера, используя мастер Create Login Wizard, выполнив следующие действия:

1. Выполнить команду в Enterprise Manager Run an Wizard/Create Login Wizard.
2. В открывшемся окне мастера ознакомиться с этапами создания учетной записи сервера:
 - а) Select an authentication mode – выбрать режим аутентификации;
 - б) Grant access to security roles – представить доступ секретным ролям;
 - в) Grant access to databases – предоставить доступ к базам данных.
3. Щелкнуть по кнопке Next.
4. Выбрать режим аутентификации: Windows или SQL Server.
5. Щелкнуть по кнопке Next.
6. Если выбран режим аутентификации Windows, то в открывшемся окне в поле Windows account задать имя учетной записи или группы Windows и домена и указать тип доступа: Grant access to the server (предоставить доступ к серверу) или Deny access to the server (запретить доступ к серверу).
7. Если выбран режим аутентификации SQL Server, то помимо имени учетной записи, задаваемой в поле Login I указать пароль в поле Password (пароль) и в поле Confirm Password (подтвердить пароль). Этот пароль пользователь будет использовать при подключении к SQL серверу.
8. Щелкнуть по кнопке Next в том или в другом случае.
9. Включить учетную запись в требуемые фиксированные роли сервера, устанавливая против роли флажок.
10. Щелкнуть по кнопке Next.
11. Разрешить для учетной записи доступ к базам данных, отмечая их флажком.
12. Щелкнуть по кнопке Next.
13. Мастер автоматически создаст имена пользователей баз данных.
14. Проверить сделанные установки.

ПРАКТИЧЕСКАЯ РАБОТА № 21

Тема: Мониторинг безопасности работы с базами данных

Цель работы: изучить технология мониторинга безопасности работы с БД.

Оборудование: ПК, интернет, программное обеспечение – MS SQL Server Management Studio, инструкции по выполнению работы.

Справочный материал:

Мониторинг СУБД и баз данных проводится для поддержания работоспособности и производительности СУБД, а также с целью отслеживания аварийных ситуаций и сбора статистики.

Реализуется мониторинг с помощью отдельных утилит СУБД, представляющих собой программные продукты, входящие в состав СУБД, но загружаемые отдельно от ядра СУБД, либо в виде набора прикладных интерфейсов — API (Application Program Interface). Эту утилиту или группу утилит и в операционной системе, и в СУБД часто называют монитором или системным монитором.

Для осуществления мониторинга ядро СУБД собирает информацию от приложений, работающих с базой данных, и от системных средств самой СУБД. Эта информация может использоваться администратором баз данных для следующих целей:

- обеспечение необходимого объема аппаратных ресурсов (на основе информации об их использовании);
- анализ производительности отдельных приложений или SQL-запросов;
- отслеживание интенсивности использования отношений;
- оценка эффективности используемых методов доступа;
- настройка параметров ядра СУБД в целях повышения производительности;
- оценка последствий вносимых оптимизационных изменений.

Сбор статистики

Администратор системы должен следить за тем, чтобы приложения, работающие с БД, имели средства сбора или предоставления статистики. Например, каждое приложение должно учитывать общее время работы, системное время, процессорное время (total time, system time, process time).

Цель сбора статистики — настроить производительность и параметры, выяснить активность пользователей и затраты по каждому из запросов и операций.

Сбор статистики может начинаться вместе с запуском ядра СУБД или с началом сессии данного приложения. Необходимо с помощью утилит мониторинга собирать статистику по БД в целом, а именно:

- статистику открытий БД (open на базу, как говорят программисты);
- число операций ввода-вывода и время;
- статистику закрытий БД (close на базу);
- число установленных соединений в течение работы сеанса ядра СУБД;
- число взаимоблокировок при блокировании записей БД (deadlock);
- число транзакций в единицу времени;
- статистику по кодам возврата от операций с БД.

Особо отметим, что АБД должен требовать от прикладных программистов обработки кода возврата от любой операции с БД. При некоторых, определенных для каждой СУБД, кодах возврата возникают фатальные события, требующие немедленного реагирования администратора БД.

Необходимо также собирать статистику по отдельным запросам приложений, работающих с СУБД, таким как:

- стоимость процессора (сколько команд процессора, тратится на запрос);
- стоимость ввода-вывода (сколько команд ввода-вывода тратится на запрос);
- число предикатов, используемых в запросе;
- избирательность, т. е. вероятность того, что каждая найденная строка удовлетворяет предикату; обычно избирательность должна составлять около 10%);
- число занятых при запросе страниц в буферном пуле СУБД.

Еще один вид статистики, который надо собирать — это статистика по отдельным отношениям БД и по соответствующим индексным файлам. Например, какой объем памяти занят под индексы, под области переполнения, непосредственно под отношение, под рабочую область СУБД

Содержание работы:

Задание 1: Осуществить мониторинг безопасности созданных БД «Студенты», «Учебный центр», «Проектная организация», «Торговая фирма».

Контрольные вопросы:

1. Какова цель мониторинга безопасности БД?
2. С помощью чего осуществляется мониторинг?
3. Для чего производится сбор статистики?

Информационное обеспечение обучения по дисциплине

Основные учебные издания:

1. Кумскова, И.А. Базы данных: учебник / Кумскова И.А. — Москва: КноРус, 2021. — 400 с. — ISBN 978-5-406-08303-1. — URL: <https://book.ru/book/940108>
2. Молдованова, О. В. Информационные системы и базы данных: учебное пособие для СПО / О. В. Молдованова. — Саратов: Профобразование, 2021. — 177 с. — ISBN 978-5-4488-1177-7. — Текст: электронный // Электронный ресурс цифровой образовательной среды СПО PROФобразование: [сайт]. — URL: <https://profspo.ru/books/106617>

Дополнительные учебные издания:

3. Разработка и защита баз данных в Microsoft SQL Server 2005 [Электронный ресурс]: учебное пособие для СПО/. — Электрон. текстовые данные. — Саратов: Профобразование, 2019. — 148 с. — ISBN 978-5-4488-0366-6. — Режим доступа: <http://www.iprbookshop.ru/86207.html>

Электронно-библиотечная система:

4. ЭБС «IPRbooks», ООО «Ай Пи Ар Медиа»
5. ЭБС «Znanium»
6. ЭБС «PROФобразование»
7. ЭБС «Book.ru»