МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение высшего образования
«ИВАНОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
Шуйский филиал ИвГУ

Кафедра математики, информатики и методики обучения

ОДОБРЕНО:

Руководитель ОП

_____ А.С. Зинченко
(подпись)

«25» февраля 2025 г.

Рабочая программа дисциплины

БАЗЫ ДАННЫХ

Уровень высшего образования:	бакалавриат
Квалификация выпускника:	бакалавр
Направление подготовки:	44.03.05 Педагогическое образование (с двумя профилями подготовки)
Направленность (профиль) образовательной программы:	Математика; Информатика и VR-технологии

Шуя

1. Цели освоения дисциплины

Дисциплина «Базы данных» имеет своей целью формирование системы знаний, умений и навыков по использованию информационных технологий для решения профессиональных задач, связанных с обработкой большого количества данных.

2. Место дисциплины в структуре ОП

Дисциплина «Базы данных» является обязательной для изучения; относится к обязательной части блока дисциплин Б1, входит в предметно-методический модуль.

Успешное освоение данной дисциплины будет способствовать готовности обучающихся к освоению дисциплин: «Основы искусственного интеллекта»; «Электронная школа XXI века»; «Логическое программирование»; «Компьютерные сети и Web-технологии»; «Информационные технологии управления образовательным учреждением»; успешному прохождению следующих практик: «Производственная практика, научно-исследовательская работа, по предметно-методическому модулю», «Производственная практика, педагогическая (комплексная) практика, по предметно-методическому модулю», подготовке к сдаче и сдаче государственного экзамена, выполнению и защите выпускной квалификационной работы.

Студент, приступающий к изучению данной дисциплины, должен:

Знать:

- особенности системного и критического мышления;
- основы современных технологий сбора, обработки и представления информации;
- основные понятия и термины, связанные с алгоритмизацией и программированием;

Уметь:

- анализировать источники информации с точки зрения временных и пространственных условий их возникновения;
- получать, обрабатывать и анализировать информацию из различных источников;
- проводить все этапы статистической обработки информации, обрабатывать числовую информацию при помощи электронных таблиц;
- учитывать и создавать условия успешного самообразования в процессе учебно-профессиональной деятельности;
- использовать программную поддержку курса и оценивать ее методическую целесообразность;
- использовать различного рода прикладные программы и языки программирования для решения учебных задач.

Владеть:

- навыками сопоставления разных источников информации с целью выявления их противоречий и поиска достоверных суждений;
- технологиями анализа и синтеза информации на основе системного подхода;
- способами совершенствования профессиональных знаний и умений путем использования информационной среды;
- технологиями алгоритмизации и программирования.

3. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

3.1. Компетенции, формированию которых способствует данная дисциплина

При реализации дисциплины формируются следующие компетенции в соответствии с ФГОС ВО по данному направлению подготовки:

а) универсальные компетенции (УК) и индикаторы их достижения:

Наименование категории (группы) универсальных компетенций	Код и наименование универсальной компетенции выпускника	Код и наименование индикатора достижения универсальной компетенции
Системное и критическое мышление	УК-1. Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1.1. Демонстрирует знание особенностей системного и критического мышления и готовность к нему. УК-1.2. Применяет логические формы и процедуры, способен к рефлексии по поводу собственной и чужой мыслительной деятельности. УК-1.3. Анализирует источник информации с точки зрения временных и пространственных условий его возникновения. УК-1.4. Анализирует ранее сложившиеся в науке оценки информации. УК-1.5. Сопоставляет разные источники информации с целью выявления их противоречий и поиска достоверных суждений. УК-1.6. Аргументировано формирует собственное суждение и оценку информации, принимает обоснованное решение. УК-1.7. Определяет практические последствия предложенного решения задачи.

б) общепрофессиональные компетенции (ОПК) и индикаторы их достижения:

Наименование категории (группы) общепрофессиональных компетенций	Код и наименование общепрофессиональной компетенции выпускника	Код и наименование индикатора достижения общепрофессиональной компетенции
Научные основы педагогической деятельности	ОПК-8. Способен осуществлять педагогическую деятельность на основе специальных научных знаний	ОПК-8.1. Применяет методы анализа педагогической ситуации, профессиональной рефлексии на основе специальных научных знаний. ОПК-8.2. Проектирует и осуществляет учебно-воспитательный процесс с опорой на знания основных закономерностей возрастного развития когнитивной и личностной сфер обучающихся, научно-обоснованных закономерностей организации образовательного процесса, научных основ преподаваемых предметов, их истории и места в мировой культуре и науке. ОПК-8.3. Осуществляет мониторинг и отбор программ профессионального и личностного развития.

в) профессиональные компетенции (ПК) и индикаторы их достижения:

Код и наименование профессиональной компетенции выпускника	Код и наименование индикатора достижения профессиональной компетенции	Основание (ПС, анализ опыта)
Тип задач профессиональной деятельности: педагогический		
ПК-4. Способен выделять структурные элементы, входящие в систему познания предметной области (в соответствии с профилем и уровнем обучения), анализировать их в единстве содержания, формы и выполняемых функций	ПК-4.1. Выделяет и анализирует единицы различных уровней знаковой математической структуры в единстве их содержания, формы и функций. ПК-4.2. Выделяет и анализирует закономерности и явления информатики как науки в их структурном и функциональном единстве с информационно-коммуникационными технологиями. ПК-4.3. Знает и умеет анализировать аксиомы, определения и математические доказательства в единстве с прикладными аспектами использования.	Профессиональный стандарт 01.001 «Педагог (педагогическая деятельность в сфере дошкольного, начального общего, основного общего, среднего общего образования) (воспитатель, учитель)»

3.2. Планируемые результаты обучения по дисциплине, соотнесенные с индикаторами достижения формируемых компетенций

В результате освоения дисциплины обучающийся должен:

Знать:

– современные пакеты прикладных программ статистической обработки данных (УК-1 (индикаторы – 1, 2, 3, 4, 5, 6, 7);

– научные основы преподаваемого предмета, его историю и место в мировой культуре и науке (конкретизируются применительно к изучаемой учебной дисциплине) (ОПК-8 (индикаторы – 1, 2, 3);

– основные понятия информатики, их системно-структурные связи в контексте развития информационно-коммуникационных технологий (ПК-4 (индикаторы – 1, 2, 3).

Уметь:

– использовать современные информационно-коммуникационные технологии (включая пакеты прикладных программ, локальные и глобальные компьютерные сети) для сбора, обработки и анализа информации (УК-1 (индикаторы – 1, 2, 3, 4, 5, 6, 7);

– осуществлять корректный подбор методов анализа, проводить обработку данных исследования и правильную интерпретацию результатов (УК-1 (индикаторы – 1, 2, 3, 4, 5, 6, 7);

– совершенствовать свои профессиональные знания и умения на основе постоянного самообразования (ОПК-8 (индикаторы – 1, 2, 3);

– соотносить основные этапы развития предметной области с ее актуальными задачами, методами и концептуальными подходами, тенденциями и перспективами ее современого развития (ОПК-8 (индикаторы – 1, 2, 3);

– устанавливать содержательные, методологические и мировоззренческие связи предметной области со смежными научными областями (ОПК-8 (индикаторы – 1, 2, 3);

– устанавливать взаимосвязь между теоретическими аспектами информатики и реализующими их технологиями (ПК-4 (индикаторы – 1, 2, 3).

Владеть:

– основными методами математической обработки информации (УК-1 (индикаторы – 1, 2, 3, 4, 5, 6, 7);

– средствами математического моделирования и анализа информации на компьютере с помощью электронных таблиц (УК-1 (индикаторы – 1, 2, 3, 4, 5, 6, 7);

– способами совершенствования профессиональных знаний и умений путем использования информационной среды (ОПК-8 (индикаторы – 1, 2, 3);

– научными основами преподаваемого предмета (конкретизируются применительно к изучаемой учебной дисциплине) (ОПК-8 (индикаторы – 1, 2, 3);

– теоретическими и практическими знаниями для постановки и решения исследовательских задач в предметной области и в области образования (ОПК-8 (индикаторы – 1, 2, 3);

– навыками работы в текстовых, графических и видео редакторах, с электронными таблицами и базами данных (ПК-4 (индикаторы – 1, 2, 3).

4. Объём и содержание дисциплины

Очная форма обучения

Объем дисциплины составляет 4 зачетные единицы (108 час.)

Контактная работа 48,25 час.

Самостоятельная работа 59,75 час.

Виды учебных занятий и формы промежуточной аттестации	5 сем.		6 сем.	
	Контактная работа с преподавателем	Самостоятельная работа обучающегося	Контактная работа с преподавателем	Самостоятельная работа обучающегося
Лекции	12	28,5	10,5	31,25
Практические занятия	13,5		12	
Зачет			0,25	
Всего:	25,5	28,5	22,75	31,25

4.1. Содержание дисциплины по разделам (темам), соотнесенное с видами и объемом учебных занятий

Очная форма обучения

№ п/п	Раздел (тема) дисциплины	Семестр	Виды учебных занятий, включая самостоятельную работу обучающихся и объем (в часах)				Формы текущего контроля успеваемости Формы промежуточной аттестации
			Занятия лекционного типа	Занятия семинарского типа		Самостоятельная работа обучающегося	
				Лаб.	Пр.		
1	Сферы применения баз данных	5	1,5		1,5	6	Практические задания на компьютере
2	Принцип независимости данных	5	3		3	4,5	Практические задания на компьютере
3	Этапы проектирования баз данных	5	3		4,5	9	Практические задания на компьютере

4	Логическое проектирование базы данных	5	4,5		4,5	9	Тест; практические задания на компьютере
Всего за 5 семестр			12		13,5	28,5	
5	Реляционная модель данных	6	3		4,5	9	Тест; практические задания на компьютере
6	Реляционная БД	6	4,5		4,5	10,5	Тест; практические задания на компьютере
7	Физическое проектирование базы данных	6	3		3	11,75	Тест; практические задания на компьютере; проект
Всего за 6 семестр			10,5		12	31,25	
ИТОГО:			22,5		25,5	59,75	Зачет

4.2. Развернутое описание содержания учебного материала по разделам (темам)

Тема 1. Сферы применения баз данных

История развития автоматизированных систем обработки данных. Суть концепции баз данных. Развитие систем обработки данных. Современные тенденции в обработке данных. Типовая организация современной СУБД. Функции СУБД.

Тема 2. Принцип независимости данных

Режимы хранения базы данных: централизованное и распределенное хранение. Принципы распределения обработки данных в архитектуре «клиент-сервер», классификация моделей «клиент-сервер» в архитектуре баз данных: файловый сервер, модель удаленного доступа, пассивный и активный сервера баз данных, трехзвенная архитектура и сервер приложений. Принцип репликации, реализация репликации в MS Access. Основные термины и определения: база данных, система управления базой данных, пользователи, приложения. Трехуровневая архитектура баз данных: внешняя, внутренняя и концептуальная модели, понятие логической и физической независимости.

Тема 3. Этапы проектирования баз данных

Проектирование реляционных баз данных. Теория функциональных зависимостей и нормальных форм баз данных. Алгоритмы декомпозиции и синтеза при проектировании нормальных форм для реляционных баз данных. Современная трактовка реляционной базы данных: структурная целостность, уровень языков манипулирования данными в реляционных БД, ссылочная целостность. Проектирование реляционных баз данных. CASE-системы - инструментальные средства разработки систем. Диаграммы UML.

Тема 4. Логическое проектирование базы данных

Структура модели данных. Иерархическая и сетевая модели данных. Иерархическая модель данных: Основные понятия, определения: сегмент, тип сегмента, экземпляр сегмента, логически исходный, логически-подчиненный сегмент. Связи между сегментами на

физическом уровне, экземпляры сегментов-близнецов, понятие физической записи в иерархической модели. Сетевая модель данных: Понятие набора данных, владелец набора, член набора. Языки описания и манипулирования данными в сетевой модели.

Тема 5. Реляционная модель данных

Структура данных. Модели данных. Основные определения: домены, атрибуты, кортежи, отношения. Понятие отношения, домена, атрибута, кортежа, ранга (степени) отношения, схемы отношения. Понятие возможного ключа, первичного ключа. Определение тэта-сравнимых атрибутов. Инфологическое проектирование БД ER – диаграммы. Правила преобразования ER-диаграмм в реляционные таблицы. Схема отношения, его степень и мощность. Разработка баз данных при помощи Microsoft Access. Правила преобразования модели «сущность-связь» в реляционные отношения.

Тема 6. Реляционная БД

Свойства отношений реляционной БД. Нормализация таблиц реляционной базы данных. Первая, вторая и третья нормальные формы. Декомпозиционный метод проектирования. Универсальное отношение и проблемы его использования. Функциональные зависимости. Разработка СУБД с использованием Oracle SQL Developer. Операторы языка: CREATE, INSERT, UPDATE, DELETE, ALTER, SELECT. Использование SQL для выборки данных из таблицы: операторы в условиях AND, OR, IN, BETWEEN, LIKE, IS NULL. Работа с онлайн-тренажером sql-academy.org.

Тема 7. Физическое проектирование базы данных

Глобальная, базовая и конкретные ИТ Файлы и файловые структуры. Индексные файлы: плотный, неплотный индекс, B-деревья, инвертированные списки. Организация взаимосвязанных файлов: однонаправленные и двунаправленные связи. Алгоритмы модификации взаимосвязанных файлов. Физические модели баз данных, основанные на нефайловых структурах: понятие страниц, экстентов.

5. Образовательные технологии, в том числе перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине

В соответствии с требованиями ФГОС ВО в программе данного курса рекомендуется использовать следующие образовательные технологии: лекции, практические занятия, самостоятельная работа студентов.

В преподавании дисциплины существуют темы, изучение которых должно обязательно начинаться с теоретической подготовки. Чтобы повысить результативность изучения теоретического материала, применяются следующие приемы:

- лекции сопровождаются компьютерными презентациями;
- применяются методики «мозгового штурма», разбора практических ситуаций;
- используются краткие сообщения студентов по новейшим разработкам (по материалам Интернет) с использованием демонстрационных материалов.

Основной и самой результативной формой обучения являются лабораторно-практические занятия. Для преподавания дисциплины создан учебно-методический комплекс, обеспечивающий предоставление информации (электронные учебники и пособия) и управление деятельностью обучаемого.

Разработанные практические задания включают в себя обучающие тексты, набор пошаговых инструкций, учебных задач и заданий, демонстрационный материал и тестовые задания.

Такой подход к организации проведения практических занятий стимулирует и организует самостоятельную работу студента. Достигаются не только учебные, но и

развивающие цели, т.к. формируется умения по самоорганизации учебной деятельности.

Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине: мультимедиа технологии; технологии смешанного обучения.

6. Учебно-методическое обеспечение самостоятельной работы обучающихся

Самостоятельная работа обучающихся по дисциплине осуществляется в следующих формах:

- решение задач, связанных с установкой и настройкой программного обеспечения;
- создание и проектирование баз данных;
- заполнение данными и построение запросов к ним;
- изучение и анализ предложенных электронных образовательных ресурсов, содержащих информацию по базам данных;
- подготовка к тестированию на занятиях по каждой теме курса;
- повторение материала с использованием конспектов.

Полностью весь методический материал по обеспечению самостоятельной работы студентов приводится в приложении 1 к настоящей программе. Доступ к методическим материалам осуществляется через сайт Шуйского филиала ИвГУ, в том числе через ЭБС Шуйского филиала ИвГУ на платформе «Руконт».

7. Характеристика оценочных средств для текущего контроля успеваемости и промежуточной аттестации по дисциплине

Текущая аттестация осуществляется с использованием накопительного рейтингового оценивания. Критерии оценки деятельности обучающихся разрабатываются в зависимости от уровня их информационно технологической и дидактической подготовки обучающихся.

Для текущего контроля успеваемости применяются:

- Практические работы на компьютере: 10 заданий по 3 балла (30 баллов);
- Практические задания для самостоятельного выполнения: 20 заданий по 0,5 балла (10 балла);
- Проектное задание (40 баллов);

Помимо этого учитывается:

- степень активности и самостоятельности обучающегося на занятиях – 10 баллов,
- посещаемость (отсутствие пропусков занятий) – 10 баллов.

Промежуточная аттестация по дисциплине (зачёт) проводится с использованием рейтинговой шкалы оценки. На зачёте по дисциплине «Базы данных» контролируемые знания, умения и владения, характеризующие этапы формирования компетенций в процессе освоения образовательной программы. Зачёт проводится в форме интегрального тестового задания, включающего вопросы по всем темам курса (максимально 30 баллов).

Общая накопительная оценка складывается из баллов текущей успеваемости в семестре по дисциплине и баллов, полученных студентами на зачёте, таким образом, чтобы результирующий балл не превышал значения 100.

Шкала оценки результатов освоения дисциплины «Базы данных» при рейтинговой системе оценивания

Повышенный уровень	Базовый уровень	Пороговый уровень	Ниже порогового уровня
зачтено	зачтено	зачтено	не зачтено
85-100 баллов	71-84 балла	61-70 баллов	60 и менее баллов

8. Учебно-методическое и информационное обеспечение дисциплины

Учебная и учебно-методическая литература:

– основная литература:

1. Гущин, А. Н. Базы данных : учебник / А. Н. Гущин. – Москва : Директ-Медиа, 2014. – 266 с. : ил.,табл., схем. – Режим доступа: по подписке. – URL: http://biblioclub.ru/index.php?page=book&id=222149
2. Карпова, Т. С. Базы данных: модели, разработка, реализация : учебное пособие / Т. С. Карпова. - 2-е изд., исправ. - Москва : Национальный Открытый Университет «ИНТУИТ», 2016. - 241 с. : ил. ; То же [Электронный ресурс]. - URL: http://biblioclub.ru/index.php?page=book&id=429003
3. Щелоков, С. А. Базы данных : учебное пособие / С. А. Щелоков ; Министерство образования и науки Российской Федерации, Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «Оренбургский государственный университет», Кафедра программного обеспечения вычислительной техники и автоматизированных систем. – Оренбург : Оренбургский государственный университет, 2014. – 298 с. : ил. – Режим доступа: по подписке. – URL: http://biblioclub.ru/index.php?page=book&id=260752

– дополнительная литература:

1. Базы данных в высокопроизводительных информационных системах : учебное пособие / авт.-сост. Е. И. Николаев ; Министерство образования и науки РФ, Федеральное государственное автономное образовательное учреждение высшего образования «Северо-Кавказский федеральный университет». – Ставрополь : СКФУ, 2016. – 163 с. : ил. – Режим доступа: по подписке. – URL: http://biblioclub.ru/index.php?page=book&id=466799
2. Дейт, К. Дж. Введение в системы баз данных : пер. с англ. / Дейт, К. Дж. - 8-е изд. - Санкт-Петербург : Вильямс, 2008. - 1328 с.
3. Кузин, А. В. Базы данных : учебное пособие для студентов вузов / А. В. Кузин, С. В. Левонисова. - 4-е изд. ; стер. - Москва : Академия, 2010. - 320 с.
4. Кузьменко, И. П. Базы данных и SQL : учебник : [16+] / И. П. Кузьменко ; Ставропольский государственный аграрный университет. – Ставрополь : АГРУС, 2024. – 128 с. : ил. – Режим доступа: по подписке. – URL: https://biblioclub.ru/index.php?page=book&id=721011. – Текст : электронный.
5. Мартишин, С. А. Проектирование и реализация баз данных в СУБД MySQL с использованием MySQL Workbench : учебное пособие / С. А. Мартишин, В. Л. Симонов, М. В. Храпченко. - Москва : Форум : Инфра-М, 2012. - 160 с. : ил. - (Высшее образование). - ISBN 978-5-8199-0517-3 (Форум). - ISBN 978-5-16-005716-3 (Инфра-М) : 247-50.
6. Сирант, О. В. Работа с базами данных / О. В. Сирант, Т. А. Коваленко. – 2-е изд., испр. – Москва : Национальный Открытый Университет «ИНТУИТ», 2016. – 150 с. : схем., ил. – Режим доступа: по подписке. – URL: http://biblioclub.ru/index.php?page=book&id=428978
7. СУБД: язык SQL в примерах и задачах : учебное пособие для студентов высших учебных заведений / И. Ф. Астахова [и др.]. - Москва : Физматлит, 2009. - 168 с. - (Информационные и компьютерные технологии). - ISBN 978-5-9221-0816-4 : 240-00.
8. Чурбанова, О. В. Базы данных и знаний. Проектирование баз данных в Microsoft Access : учебно-методическое пособие / О.В. Чурбанова, А.Л. Чурбанов ; Министерство образования и науки Российской Федерации, Северный (Арктический) федеральный университет имени М.В. Ломоносова. – Архангельск : САФУ, 2015. – 152 с. : ил., схем.,

табл. – Режим доступа: по подписке. – URL: http://biblioclub.ru/index.php?page=book&id=436230
9. Шилин, А. С. Перспективные методы проектирования реляционных баз данных : учебное пособие : [12+] / А. С. Шилин. – Москва ; Берлин : Директ-Медиа, 2021. – 136 с. : ил., схем., табл. – Режим доступа: по подписке. – URL: https://biblioclub.ru/index.php?page=book&id=602240. – Библиогр. в кн. – ISBN 978-5-4499-1890-1. – Текст : электронный.

Интернет-ресурсы:
1. Документация по PostgreSQL и Postgres Pro https://postgrespro.ru/docs/
2. Официальный сайт Open Server Panel. https://ospanel.io/
3. Образовательные сайты и порталы. http://school2100.com/
4. Сайты учителей. http://rus.1september.ru/
5. Фоксфорд.Учебник https://foxford.ru/wiki
6. Профессиональный сайт учителей «Методисты». http://nsportal.ru/
7. Социальная сеть работников образования. http://nsportal.ru/shkola

Современные профессиональные базы данных и информационно-справочные системы:
ЭБС «Университетская библиотека онлайн»;
Коллекция Шуйского филиала ИвГУ на национальном цифровом ресурсе «Руконт».
КонсультантПлюс http://www.consultant.ru/

Программное обеспечение:

В процессе изучения дисциплины используются офисные пакеты Microsoft Office (в том числе пакет для организации баз данных Microsoft Access 2007 или выше), OpenOffice (в том числе пакет для организации баз данных OpenOffice Base), инструментальное средство Oracle SQL Developer Data Modeler, программная среда для веб-разработчиков Open Server Panel, программа для просмотра и чтения файлов PDF Adobe Acrobat Reader DC, программа для воспроизведения флэш-анимации в браузерах Adobe Flash Player, веб-обозреватели Яндекс.Браузер, Mozilla Firefox, Антивирус Касперского, 1С университет.

9. Материально-техническое обеспечение дисциплины

1. Учебная аудитория для проведения занятий лекционного типа, укомплектованная специализированной мебелью и техническими средствами обучения, служащими для предоставления учебной информации большой аудитории.
2. Учебная аудитория для проведения занятий семинарского типа, укомплектованная специализированной мебелью и техническими средствами обучения.
3. Учебная аудитория для проведения текущего контроля и промежуточной аттестации, укомплектованная специализированной мебелью и техническими средствами обучения.
4. Помещение для самостоятельной работы, оснащенное специализированной мебелью, компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду организации.
5. Учебно-наглядные пособия для занятий лекционного типа, обеспечивающие тематические иллюстрации: презентации по информационным системам и базам данных.

Автор рабочей программы дисциплины: В.А. Смирнов, старший преподаватель кафедры математики, информатики и методики обучения.

Программа рассмотрена на заседании кафедры математики, информатики и методики обучения «19» февраля 2025 г., протокол № 8

Программа обновлена
протокол заседания кафедры № _____ от «_____» _____20 __ г.
Согласовано:
Руководитель ОП _____ А.С. Зинченко
(подпись)

Программа обновлена
протокол заседания кафедры № _____ от «_____» _____20 __ г.
Согласовано:
Руководитель ОП _____ А.С. Зинченко
(подпись)

Программа обновлена
протокол заседания кафедры № _____ от «_____» _____20 __ г.
Согласовано:
Руководитель ОП _____ А.С. Зинченко
(подпись)

Приложение 1

Методические указания для обучающихся по освоению дисциплины

На самостоятельную работу студентов по курсу «Базы данных» отводится более 60% учебного времени от общей трудоемкости курса. Сопровождение самостоятельной работы студентов по данному курсу может быть организовано в следующих формах:

– согласование индивидуальных заданий (виды и темы заданий, сроки представления результатов) самостоятельной работы студента в пределах часов, отведенных на самостоятельную работу;

– консультации (индивидуальные и групповые), в том числе с применением дистанционных технологий взаимодействия;

– промежуточный контроль хода выполнения заданий, строящихся на основе различных способов самостоятельной информационной деятельности в открытой информационной среде.

Преподаватель определяет содержание самостоятельной работы, график её выполнения; создает сетевую информационную и коммуникационную среду для организации самостоятельной работы.

В процессе организации самостоятельной работы особое внимание уделяется не только развитию информационной культуры, но и формированию готовности к кооперации, к работе в коллективе в сетевом пространстве, для чего целесообразно стимулировать студентов к совместной деятельности в малых группах.

Одной из важнейших особенностей заданий для самостоятельной работы по курсу «Базы данных» является их ориентация на формирование способности оценивать приоритеты и ограничения при выборе средств информационных технологий для решения управленческих задач, а также на мотивирование самообразовательной деятельности с использованием ИКТ.

К типовым заданиям для самостоятельной работы студентов относятся:

– Конспектирование и реферирование литературных источников по тематике учебного курса.

– Решение задач, связанных с установкой программного обеспечения, создания и управления базами данных, внесением данных и построением запросов для их обработки;

– Изучение и анализ предложенных электронных образовательных ресурсов, содержащих информацию по базам данных;

– Подготовка к тестированию на занятиях по каждой теме курса;

– Повторение материала с использованием конспектов.

Перечень вопросов для самостоятельного изучения

1. Дайте определение реляционной базы данных? Роль системы управления базами данными (СУБД) в организации.
2. Архитектура современных СУБД. В чем суть логического и инфологического моделирования?
3. Этапы проектирования СУБД. Каковы принципы построения ER-модели?
4. Иерархическая модель данных. Основные понятия, графическое изображение, примеры.
5. Сетевая модель данных. Основные понятия, графическое изображение, примеры.
6. Реляционная модель данных. Основные понятия, графическое изображение, примеры.
7. Нормализация таблиц реляционной базы данных. Определение, виды, понятия.

8. Первая нормальная форма реляционной модели данных. Определение, требования, примеры.

9. Вторая нормальная форма реляционной модели данных. Определение, требования, примеры.
10. Третья нормальная форма реляционной модели данных. Определение, требования, примеры.
11. Какие задачи по проектированию БД могут решаться с помощью инструментальных средств ErWin/BpWin?
12. Каково назначение языков DDL и DML. Как создать таблицу БД используя DDL.
13. Какие существуют операторы языка SQL для чтения и записи данных?
14. Какие существуют операторы языка SQL для обновления (редактирования) и удаления данных?
15. Какие существуют операторы языка SQL предоставления прав доступа к отдельным объектам БД различным пользователям?
16. В чем отличия архитектуры клиент-сервер от архитектуры файл-сервер? Каковы особенности работы БД в многопользовательском режиме?
17. Как осуществить доступ к БД из приложений под Windows, используя VBA и ADO.
18. Опишите основные этапы проектирования приложений БД под WEB. Как осуществить передачу данных с WEB-страницы серверу БД?
19. Перечислите функции администратора БД. Как осуществить физический пере-нос объектов БД с данными между различными серверам (например, в MS SQL Server).
20. Перечислите основные инструментальные средства СУБД MS SQL Server?
21. Перечислите основные инструментальные средства СУБД Oracle?
22. Какие функции управления данными могут быть реализованы с помощью Oracle enterprise manager?
23. Из каких ключевых блоков состоит программа PL/SQL?
24. Как осуществить запрос нескольких строк из базы данных, используя курсоры PL/SQL?
25. Физические модели данных. Определения, назначение информационной модели, цели.
26. Правила составления условий отбора данных. Операторы сравнения, использование функций.

Задания для организации практических занятий:

Практическая работа №1 «Выделение таблиц предметной области»

Одним из основных моментов при создании реляционной базы данных является процесс выделения таблиц, которые должны использоваться для хранения данных конкретной предметной области.

Например, в информационной системе «Расписание занятий» должны присутствовать следующие таблицы:
- Таблица аудиторий (код аудитории, название, расположение в пределах здания);
- Таблица преподавателей (табельный номер, фамилия, имя, отчество, должность);
- Таблица дисциплин (код дисциплины, название дисциплины);

- Таблица расписания звонков (номер пары, время начала занятия, время конца занятия);
- Таблица типов занятий (номер типа, название (лекция, лабораторная и т.д.));
- Таблица расписания (номер пары, код аудитории, табельный номер преподавателя, код дисциплины, номер пары, номер типа занятия, дата проведения).

Упражнение 1. Укажите данные, которые необходимо хранить при организации:

а) *системы регистрации учащихся*.

Подробное описание: Школьному округу ABC требуется информационная онлайн-система для регистрации учащихся и сбора информации об учащихся. Эта система должна быть реализована как интерактивный процесс, чтобы все новые учащиеся могли регистрироваться онлайн. Кроме того, у существующих учащихся должна быть возможность обновлять и просматривать всю информацию. Создайте список важных данных, которые необходимо собрать и сохранить в базе данных регистрации учащихся.

б) *системы управления библиотекой*.

Подробное описание: Сообщество XYZ планирует создать систему управления библиотекой. База данных должна обрабатывать все операции библиотеки. В базе данных должны храниться все данные, относящиеся к управлению библиотечным фондом, обслуживанию посетителей и повседневной работе библиотеки. Создайте список важных данных, которые необходимо собрать и сохранить в базе данных управления библиотекой.

Упражнение 2. Определите возможные таблицы и поля в них, опираясь на предложенные сценарии.

a) Book.com — интернет-магазин, где покупатели могут посмотреть каталог и выбрать интересующие их товары.

a. У каждой книги есть название, номер ISBN, год издания и цена. В базе также хранятся сведения об авторах и издателях книг.

b. Сведения об авторах включают в себя имя, адрес и ссылку на главную страницу сайта.

c. Сведения об издателях включают в себя имя, адрес, номер телефона и ссылку на главную страницу сайта.

d. В базу внесены данные нескольких складов, включая код, адрес и номер телефона.

e. На складе хранится некоторое количество книг. Одна и та же книга может храниться на нескольких складах.

f. В базе данных регистрируется число экземпляров книги, хранящихся на различных складах.

g. Книжный магазин хранит данные покупателей: имя, адрес, адрес электронной почты и номер телефона.

h. У покупателя имеется несколько корзин. Корзине присваивается идентификатор Shopping_Cart_ID, и она содержит несколько книг.

i. Некоторые корзины могут содержать несколько экземпляров одной и той же книги. В базе данных регистрируется число экземпляров каждой книги, находящихся в любой корзине.

j. На данном этапе для выполнения транзакции требуется дополнительная информация. Как правило, покупателю нужно указать или выбрать платежный адрес, адрес доставки, способ доставки и платежные данные, например номер кредитной карты. После размещения заказа покупатель получает уведомление по электронной почте.

б) Компания ABC Ltd планирует компьютеризировать свою систему продаж и управления запасами. Проведенный анализ целесообразности показал, что этой компании настоятельно рекомендуется внедрить реляционную базу данных. Система продаж и управления запасами ABC функционирует следующим образом:

a. Покупатели отправляют заказы на приобретение товаров. Каждый заказ может содержать несколько запросов на различное количество одного или нескольких товаров из ассортимента ABC. ABC ведет складской учет. Складская ведомость содержит информацию о каждом товаре: описание, сведения о предпочтительном поставщике, количество единиц товара в наличии, возможность повторного заказа и прочие данные.

b. Получив заказ от покупателя, ABC доставляет имеющиеся в наличии товары и создает счет на отправленные покупателю товары. Товары, которых нет в наличии, помещаются в список отложенных заказов и, как правило, повторно заказываются у предпочтительного поставщика. В некоторых случаях товары заказываются у других поставщиков.

c. Получив счета от компании ABC, покупатели осуществляют оплату. При этом покупатель может оплатить один счет, некоторые позиции счета или несколько счетов (полностью или частично).

Упражнение 3. Вам предстоит попытаться представить выбранную предметную область в виде набора таблиц. В зависимости от собственных интересов выберите один из следующих вариантов:

- Система хранения тем выпускных квалификационных работ (по окончании обучения студент выполняет дипломную работу, руководителем которой является единственный преподаватель, но при этом преподаватель может руководить несколькими студентами);
- Система хранения данных о студентах (студенты распределены по группам и направлениям подготовки, каждая из групп прикреплена к конкретному факультету);
- Система хранения данных о пациентах в медицинской организации (пациент может посещать различных врачей, каждый из которых может иметь различную специализацию. Система хранит дату посещения специалиста, а также сканированную версию медицинского заключения, но не предполагает поиска и систематизации заболеваний);
- Гостиничный комплекс (жильцы заселяются только в одноместные номера, имеющие разную стоимость. Система хранит дату заселения и дату выселения. Кроме того, в каждом номере есть определенное оборудование: телевизор, холодильник и т.д. Хранить детальную информацию (марку, производителя) об оборудовании необходимости нет, однако важно учитывать, что в номере может быть несколько единиц каждого вида оборудования).

Практическая работа №2 «Создание реляционной модели базы данных»

Описание ситуации: имеем базу данных, состоящую из следующих таблиц:

Сотрудники				
Табельный номер	**ФИО**	**З/П**	**Номер должности**	**Номер отдела**
1	Федоров Петр Иванович	15000	2	2

2	Сидоров Кирилл Игоревич	20000	3	2
3	Петров Владимир Альбертович	18000	2	2
4	Колотилов Иван Владимирович	23000	1	1
...	...	...	...	...

Должности	
Номер должности	**Название должности**
1	Бухгалтер
2	Программист
3	Администратор баз данных
...	...

Отделы	
Номер отдела	**Название отдела**
1	Бухгалтерия
2	IT-отдел
...	...

В этих таблицах присутствуют следующие поля:

Должности		
Номер должности	Целое число	Первичный ключ
Название должности	Строка	

Отделы		
Номер отдела	Целое число	Первичный ключ
Название отдела	Строка	

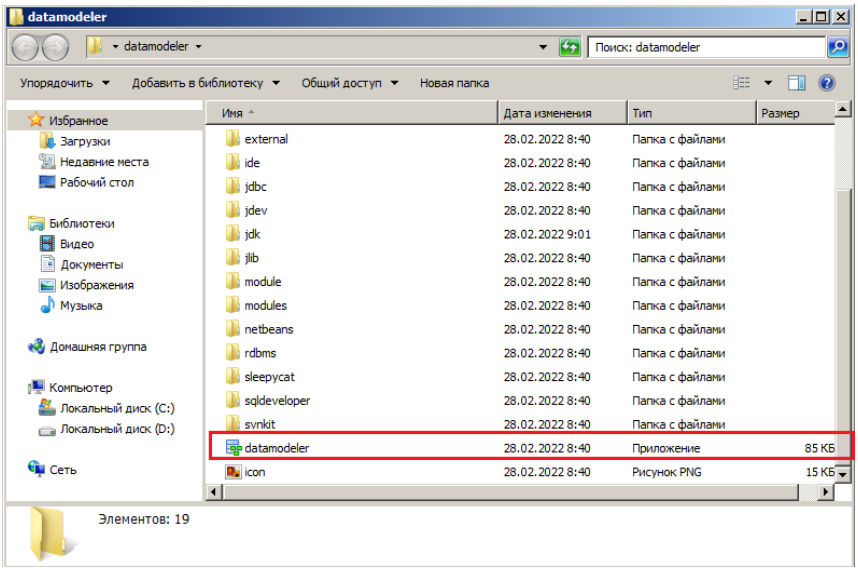
Сотрудники		
Табельный номер	Целое число	Первичный ключ
ФИО	Строка	
З/П	Целое число	
Номер должности	Целое число	Внешний ключ (таблица «Должности»)
Номер отдела	Целое число	Внешний ключ (таблица «Отделы»)

Информация о данных таблицах может быть визуализирована с помощью диаграммы «сущность-связь» (ERD).

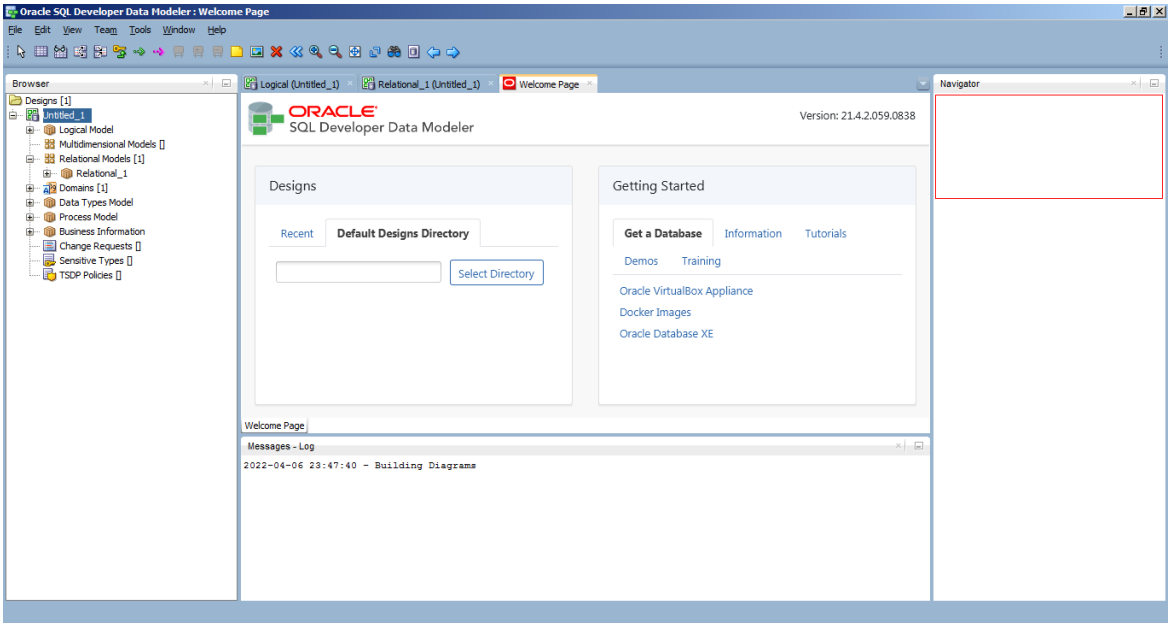
Цель занятия – построение реляционной модели базы данных в Oracle SQL Developer Data Modeler.

Данная программа является бесплатной. Ранее она могла быть загружена с официального сайта Oracle. В данный момент рекомендую воспользоваться ссылкой для скачивания: https://www.softpedia.com/get/Internet/Servers/Database-Utils/Oracle-SQL-Developer-Data-Modeler.shtml

Скачанный файл представляет собой zip-архив, который необходимо распаковать. Из архива будет извлечена папка со следующим содержимым:

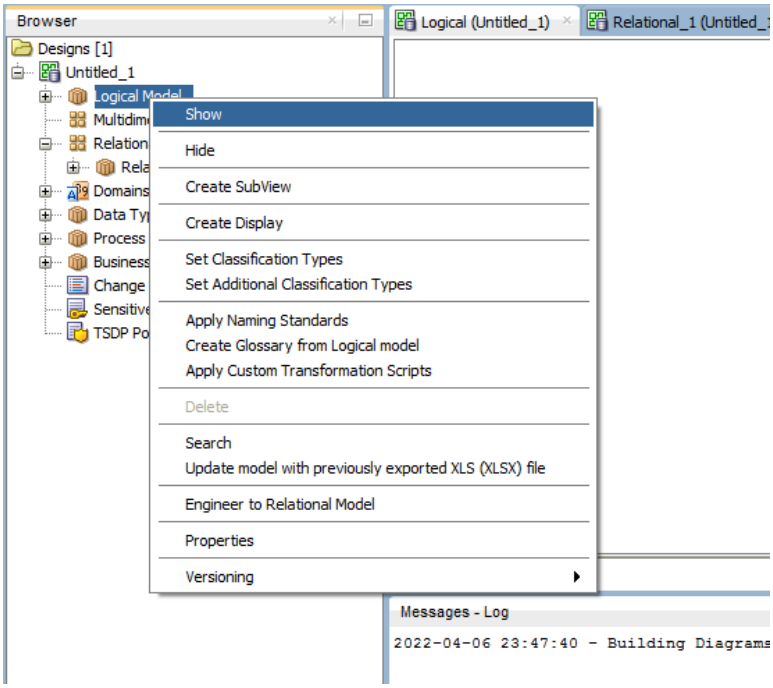


Файл, который запускает среду разработки, выделен красным прямоугольником.
Главное окно программы при запуске:



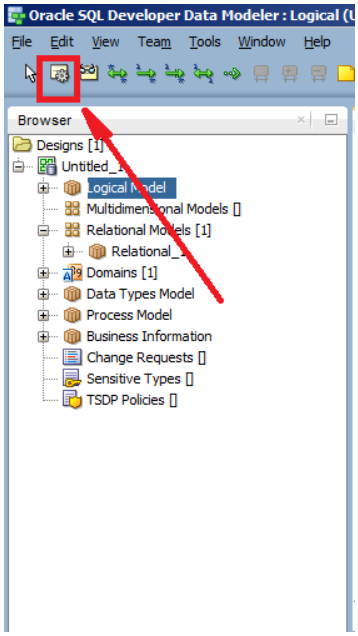
Отсюда переходим во вкладку Logical (логическая модель базы данных).

Если вкладки нет, то её можно открыть с помощью контекстного меню пункта «Logical Model»:

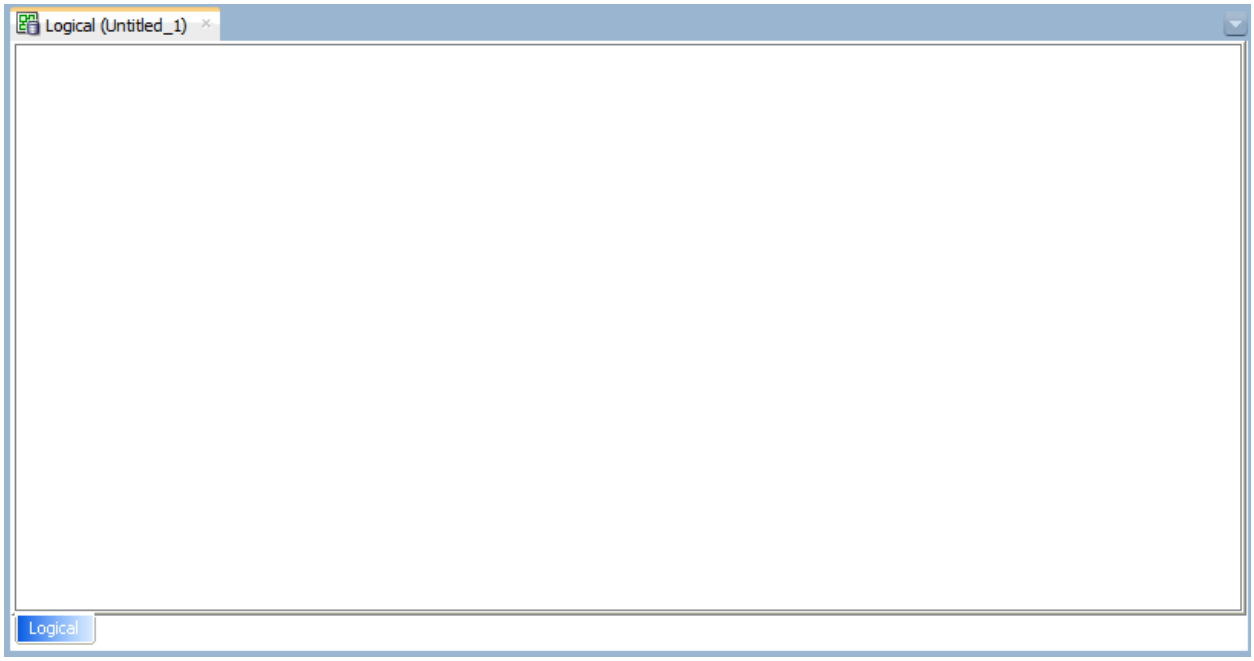


Таблицам в реляционной модели соответствуют сущности в логической модели. Столбцам таблицы – атрибуты сущности.

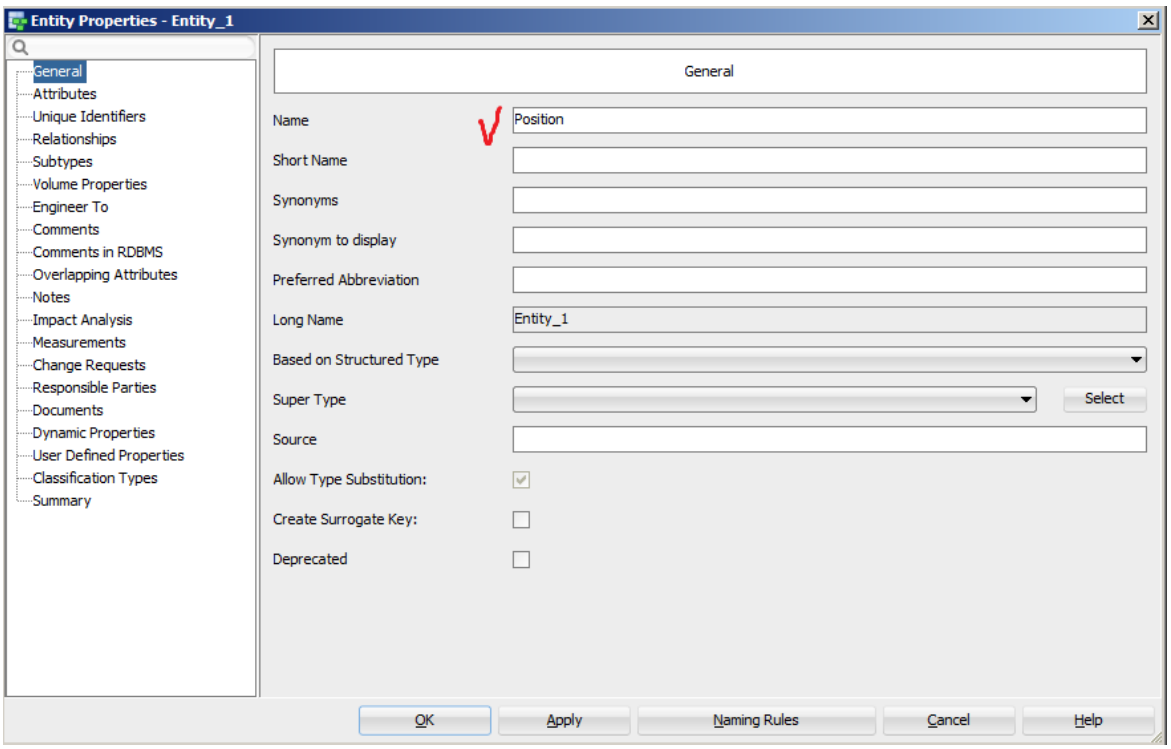
Добавление «сущности» в модель производится при помощи кнопки «New Entity»:



После выбора инструмента курсор в центральной панели (на рис. ниже) сменит изображение на значок «выстрела». На этой панели можно нажать в любое свободное место.

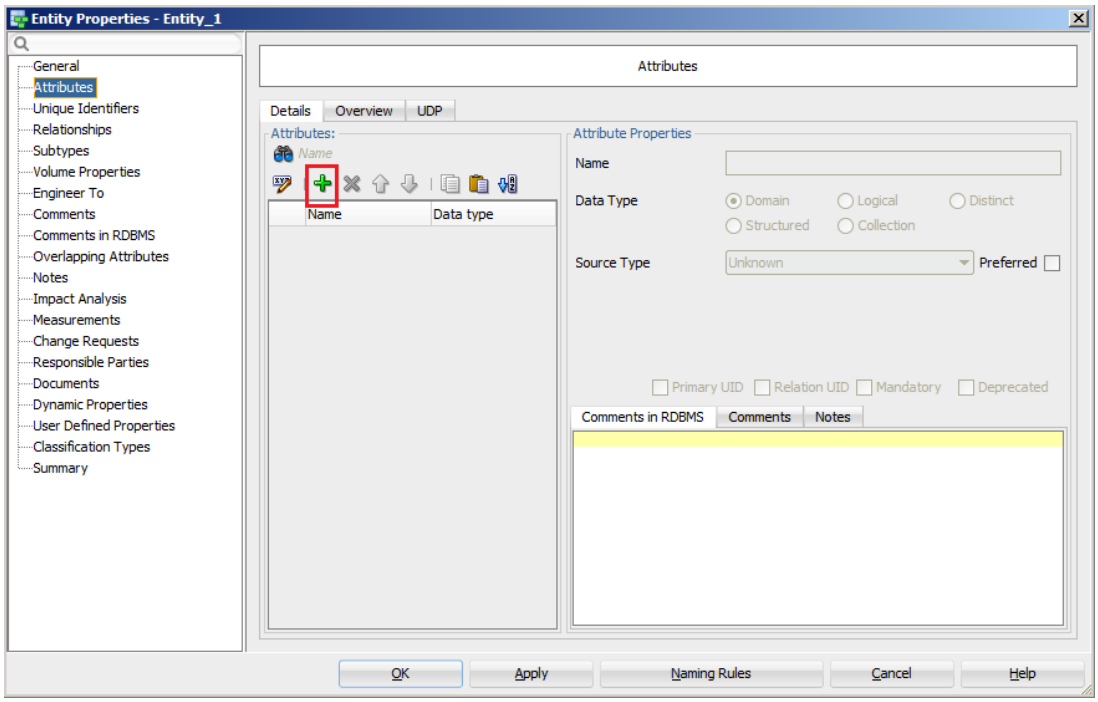


В появившемся окне указываем имя сущности:

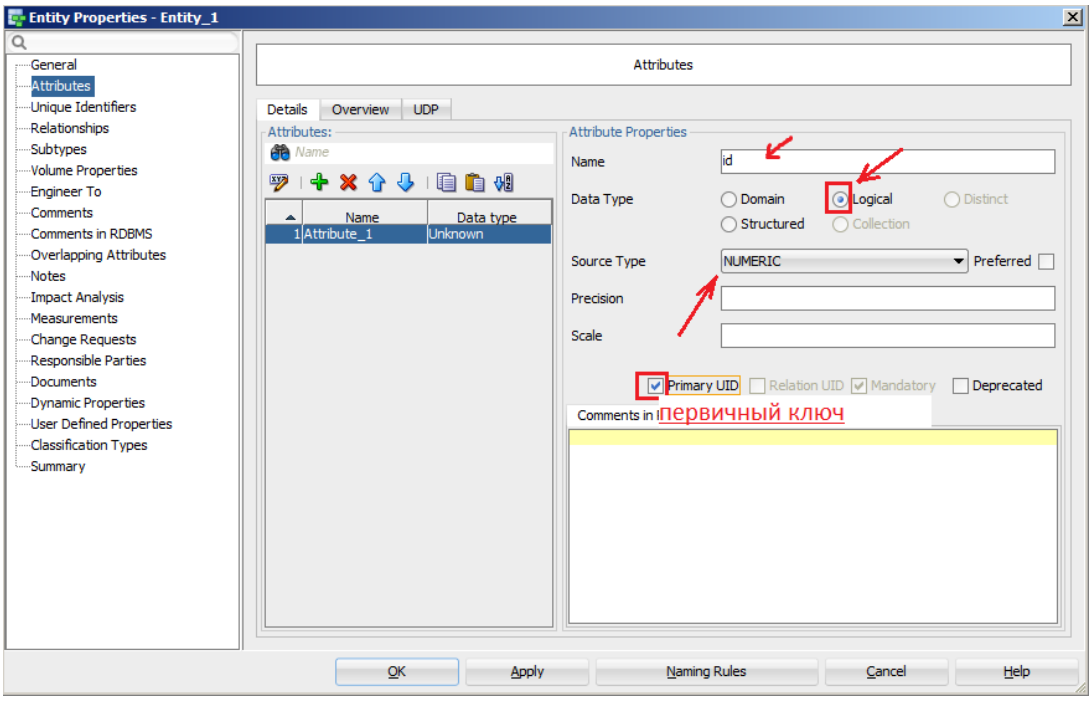


После чего переходим на вкладку «Attributes»:

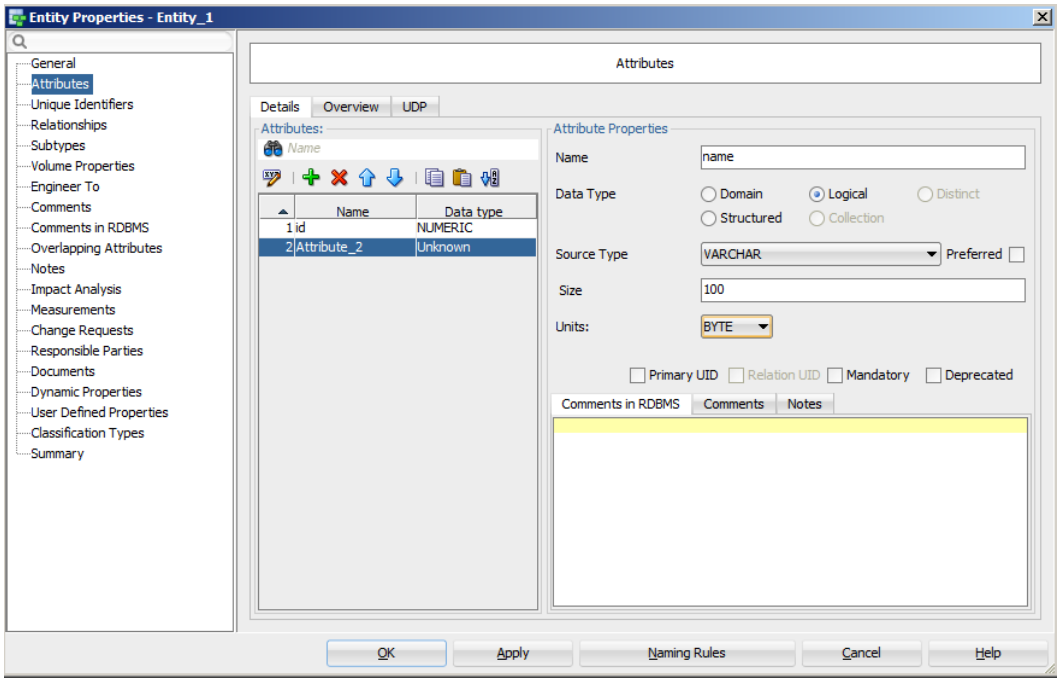
Кнопкой со знаком «+» добавляем новый атрибут:



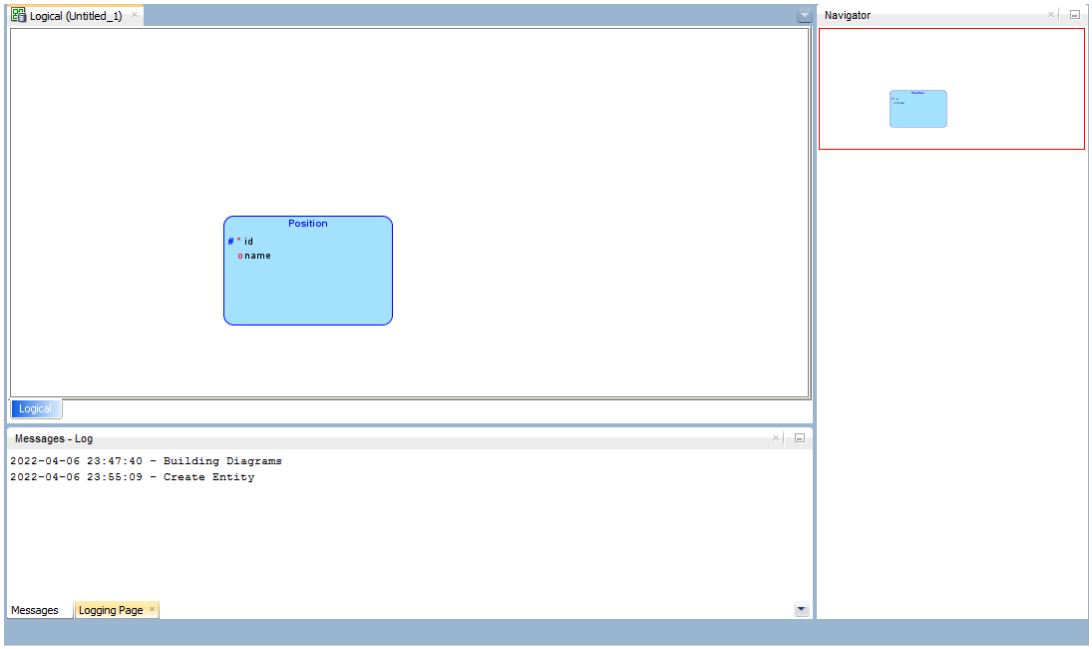
Указываем информацию о добавляемом атрибуте (номер сотрудника – числовой атрибут, будет являться первичным ключом в таблице):



Добавляем кнопкой со знаком «+» еще один атрибут (название должности, строка, состоящая из не более чем 100 символов):

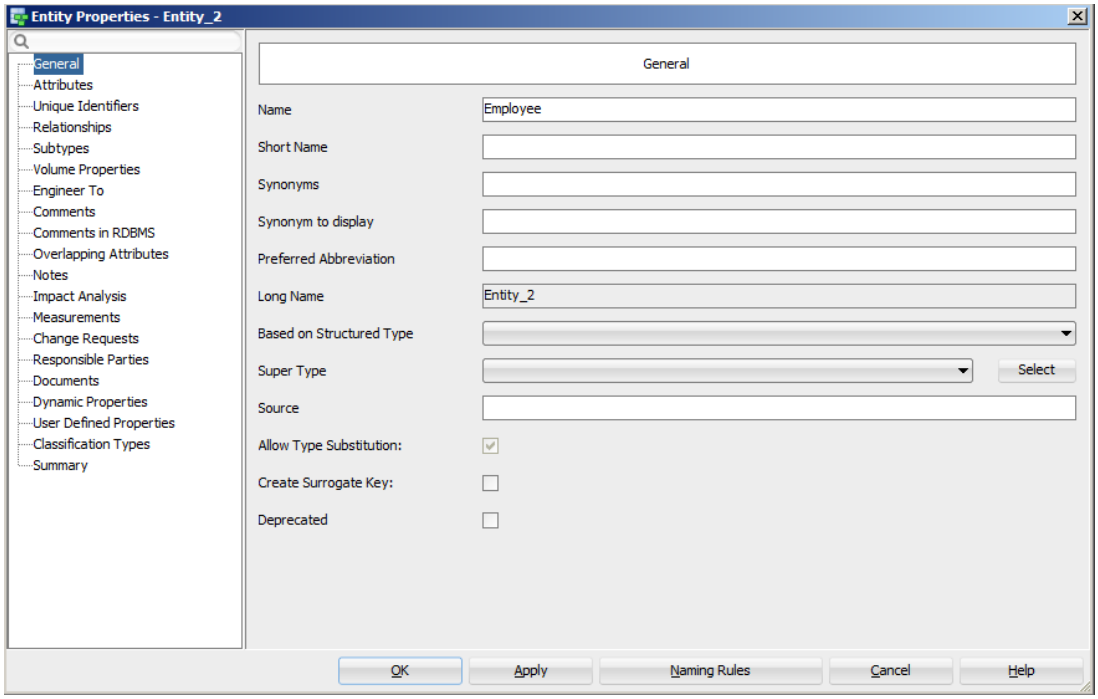


После нажатия «ОК»:

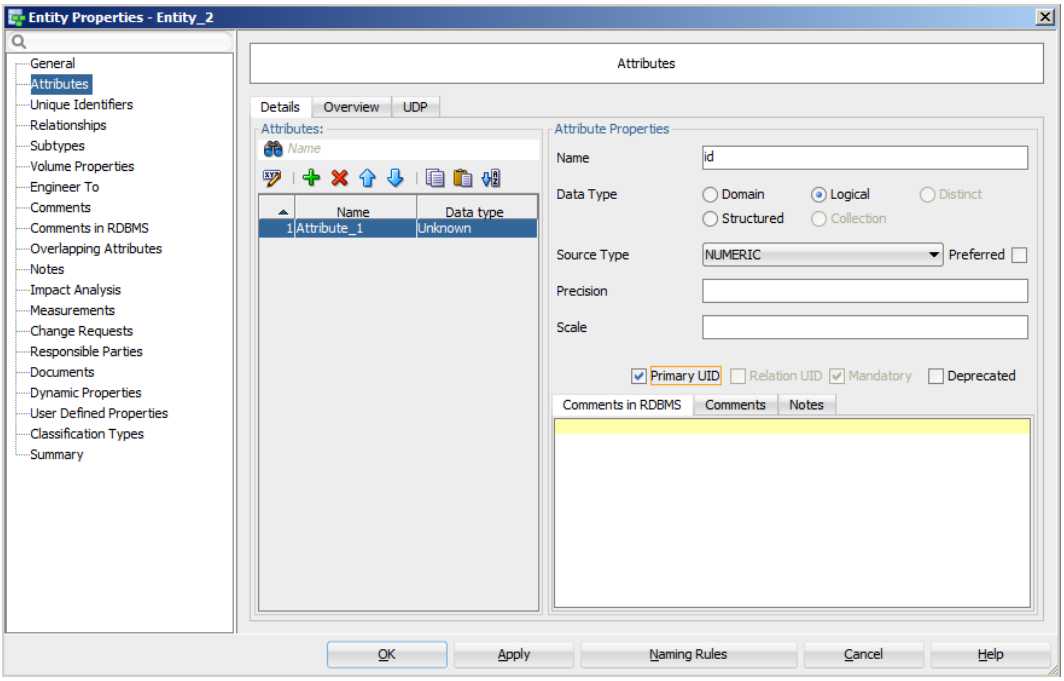


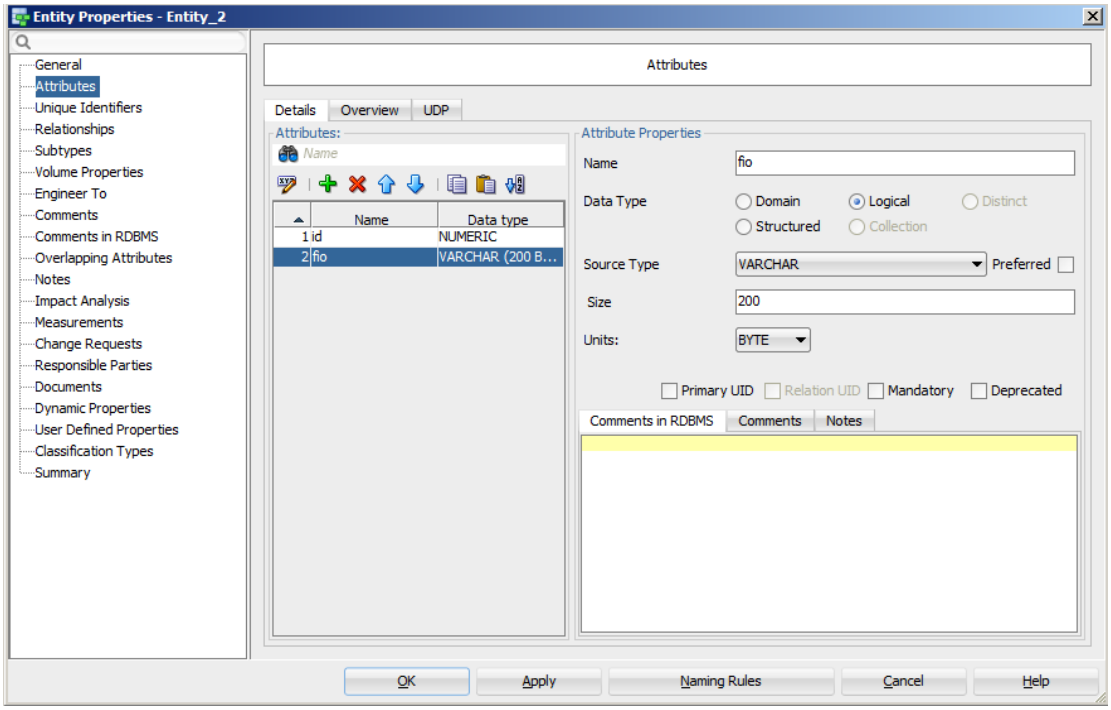
Сущность «Отдел» добавляется аналогично.

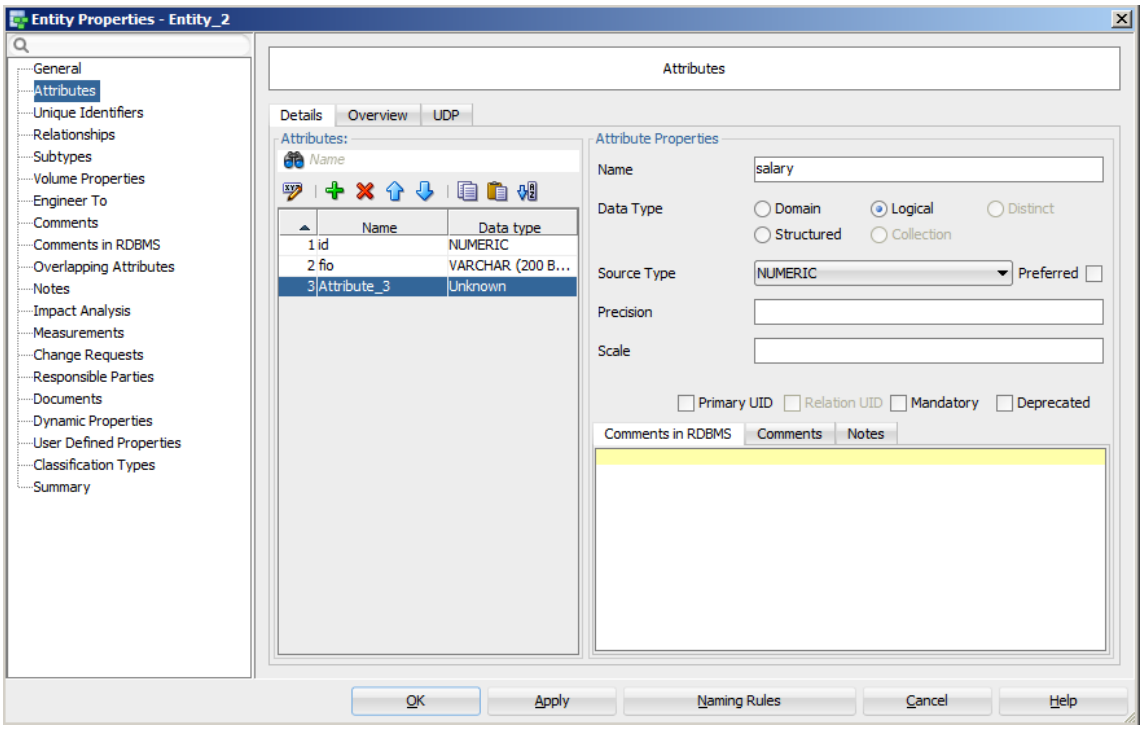
Добавляем сущность «Сотрудник»:



Атрибуты:

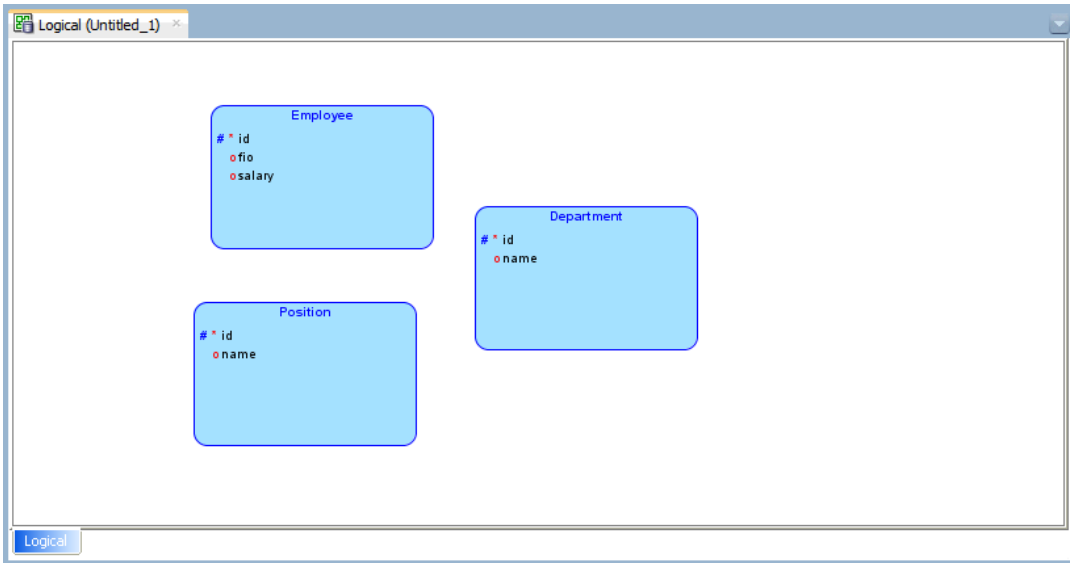






Внешние ключи не добавляем. Они будут добавлены автоматически после установки связи.

Результат:

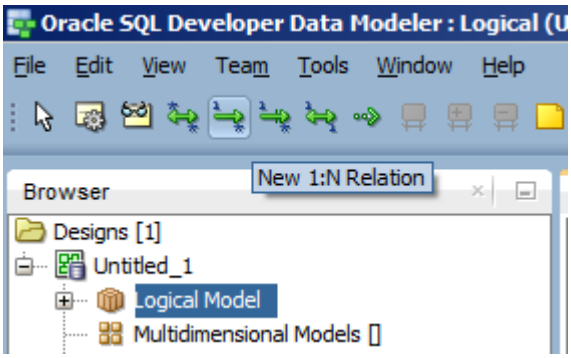


Расположение сущностей на схеме относительно друг друга выбирается произвольно.

Еще одним этапом является создание связей.

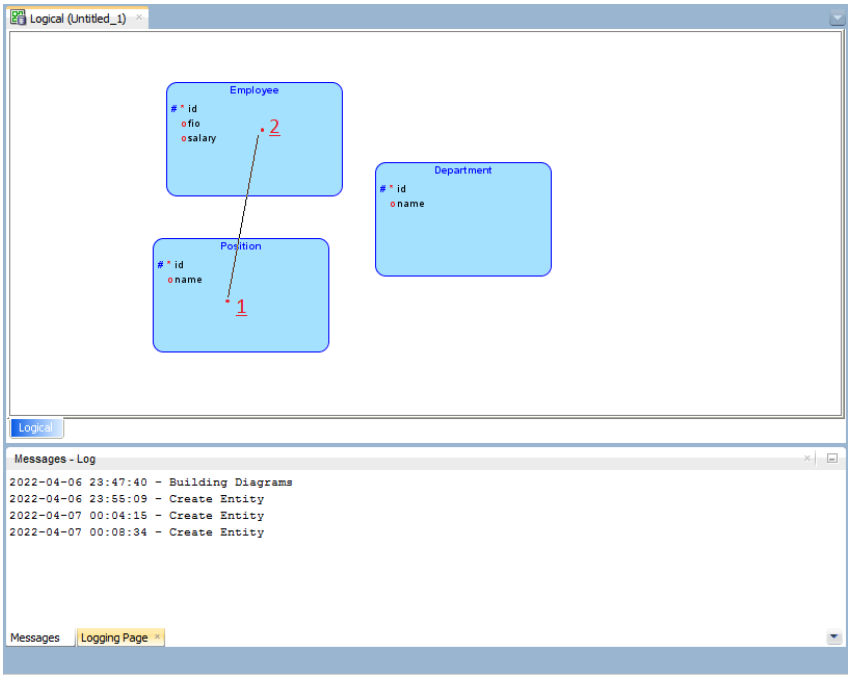
Будем полагать, что любой сотрудник имеет какую-то одну должность и работает в каком-либо одном отделе. При этом в отделе может быть любое количество сотрудников. Соответственно, речь идет о связи «один-ко-многим».

Выбираем этот тип связи на панели инструментов:

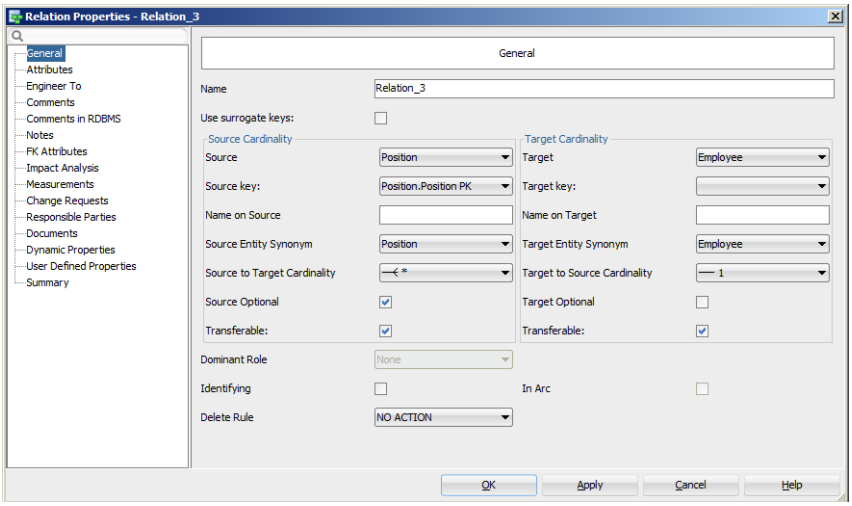


После этого нажимаем сначала на ту сущность, поле из которой будет добавлено (например, «Position»).

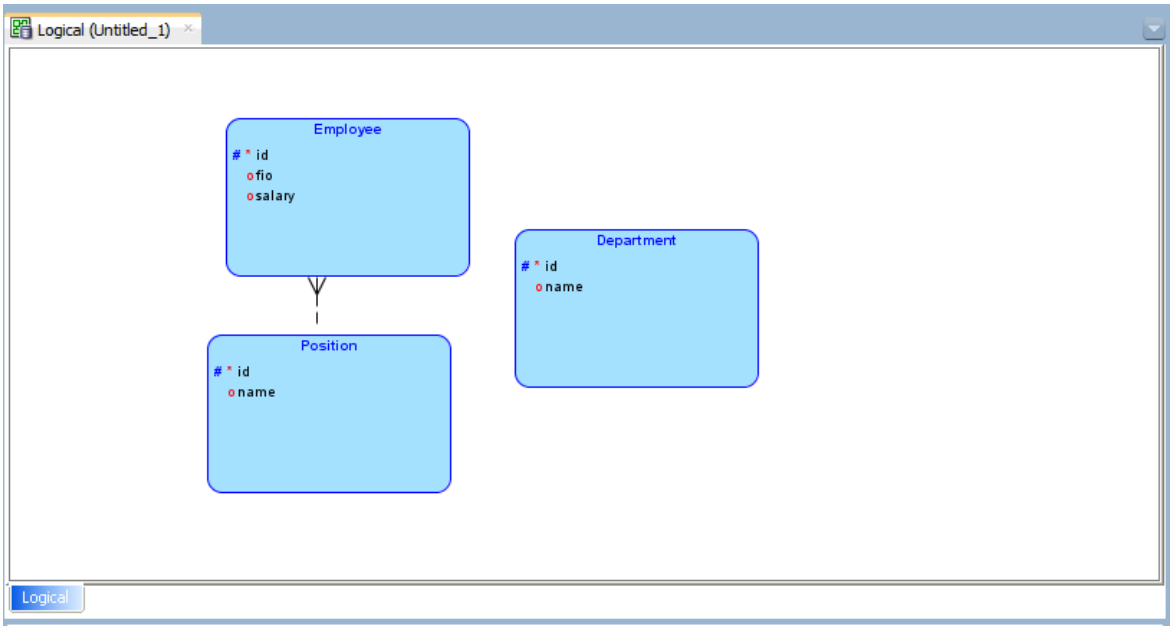
После этого нажимаем на сущность, в которую будет добавлено поле в качестве внешнего ключа («Employee»).



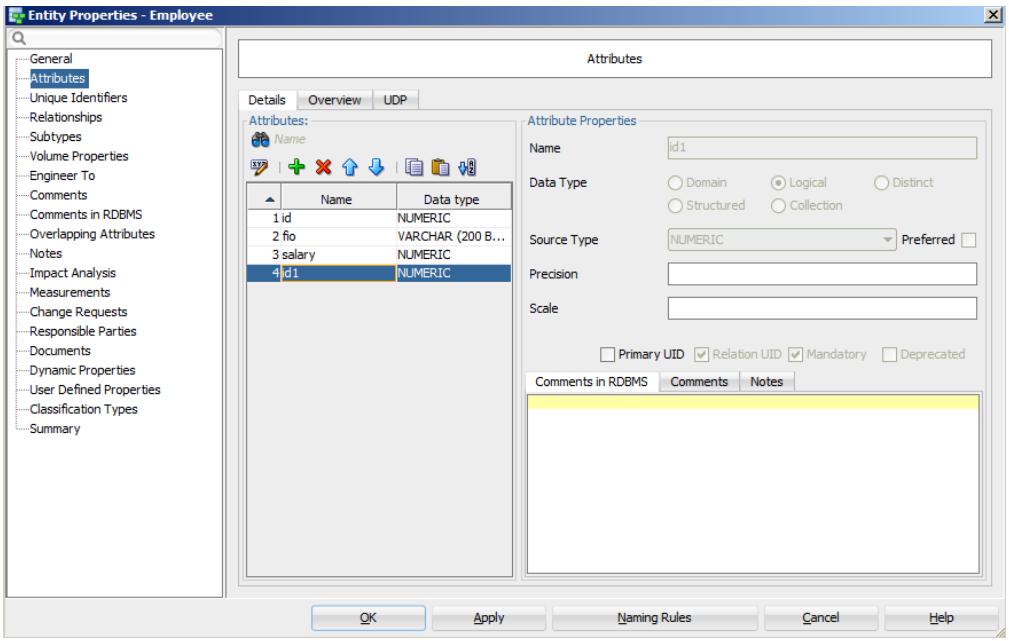
В появившемся окне можно ничего не настраивать.



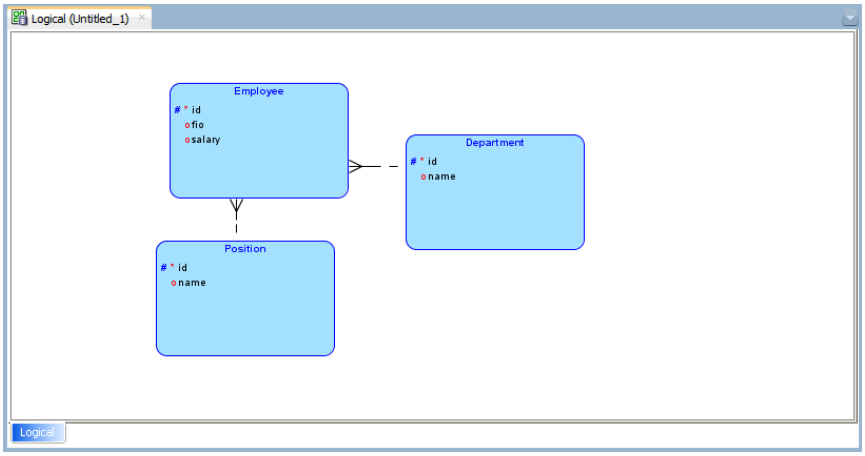
Результат:



Если открыть окно свойств сущности «Employee», то можно заметить появление нового атрибута:

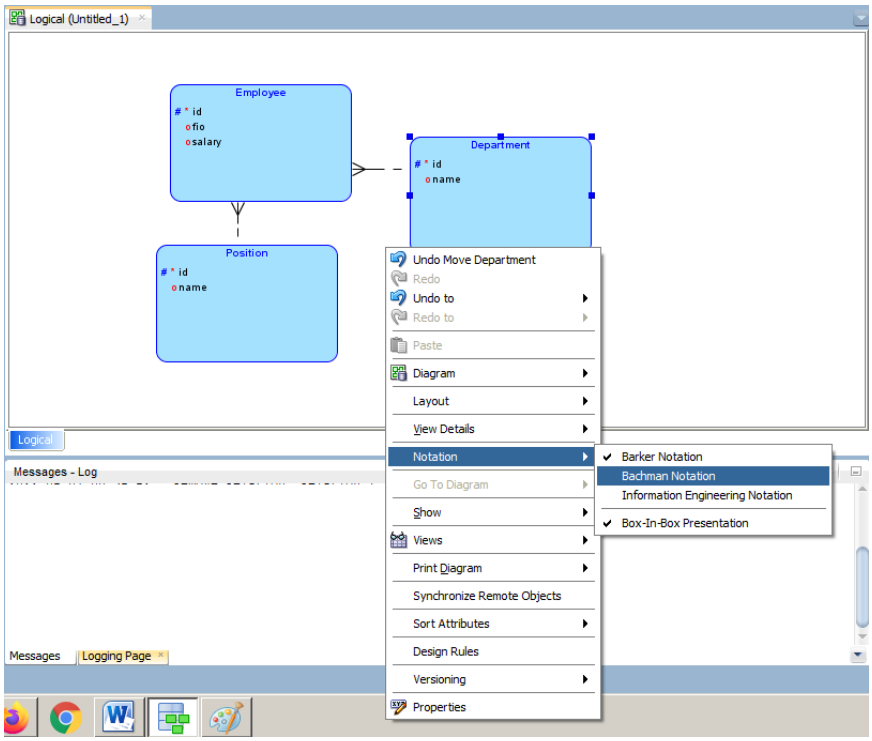


Аналогично добавляем вторую связь:

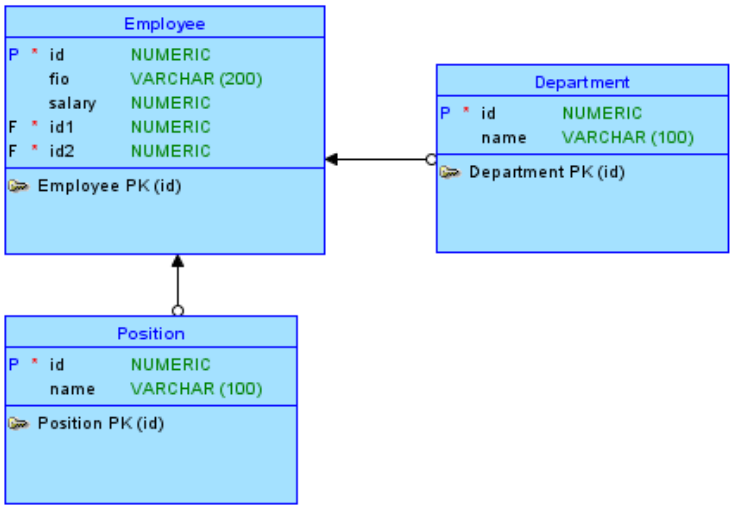


В итоге была создана логическая модель базы данных.

В некоторых случаях удобнее сменить ее внешний вид на изображение в нотацию Баркера (это можно сделать через контекстное меню).

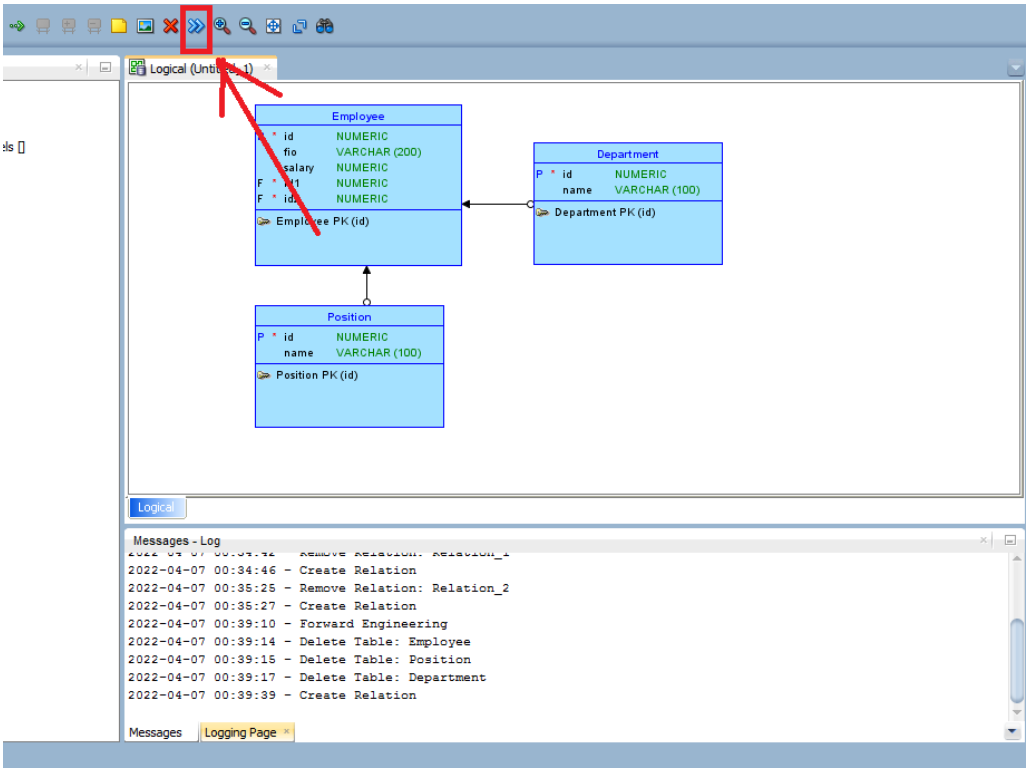


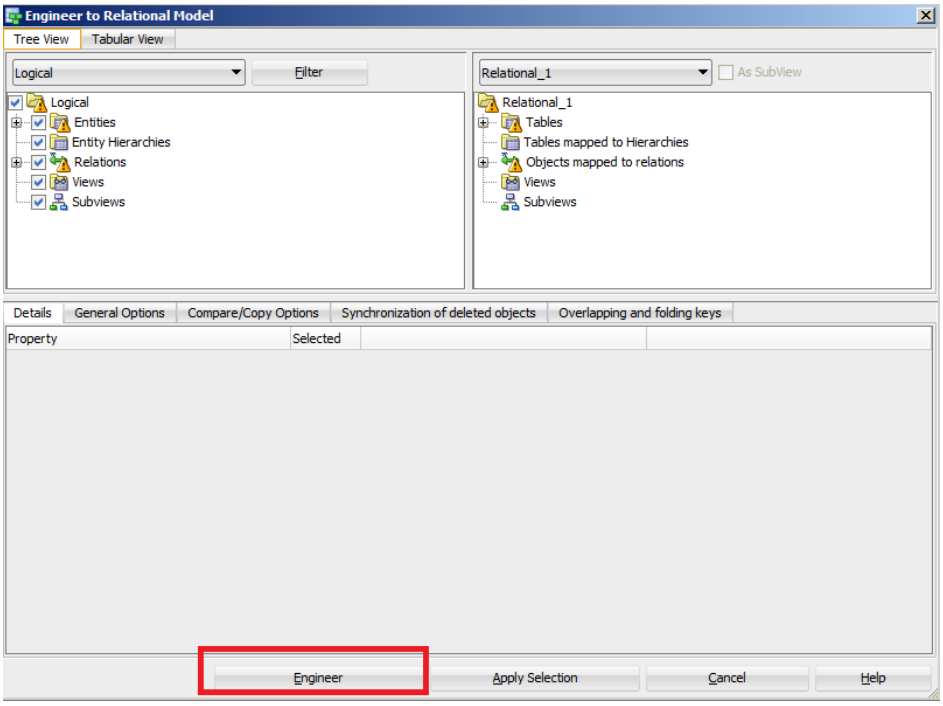
Результат:



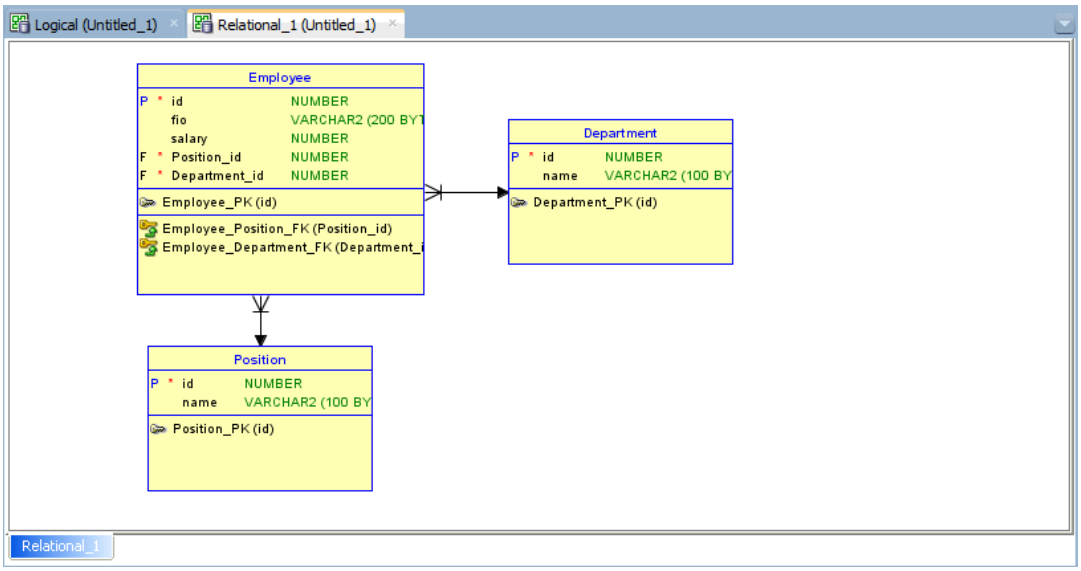
В случае дальнейшего использования данной диаграммы в технической документации поля id1 и id2 нужно будет переименовать.

Для получения реляционной модели базы данных воспользуемся встроенным инструментом:





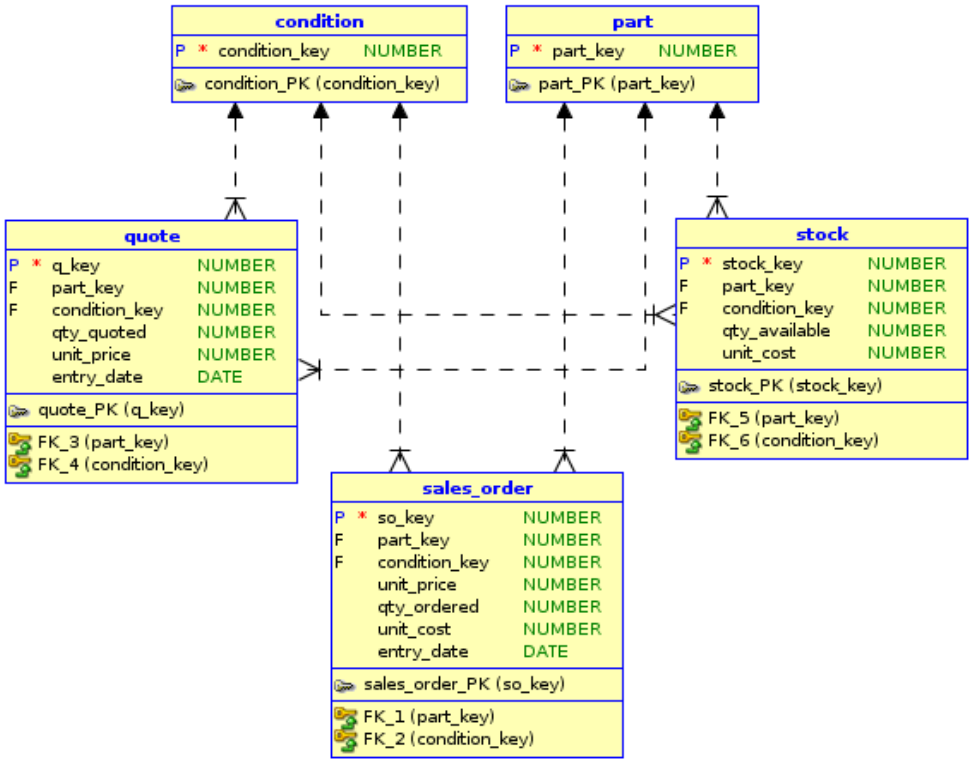
Итог:



Задание:

а) проделать все описанные выше этапы. Результат сохранить в стандартном формате Oracle SQL Developer Data Modeler и прислать в виде архива zip.

б) построить модель, представленную на рисунке. Результат сохранить в стандартном формате Oracle SQL Developer Data Modeler и прислать в виде архива zip.



Практическая работа №3 «Объекты, атрибуты и уникальные идентификаторы»

Задание 1. Определить объекты и их атрибуты для базы данных системы управления образовательным учреждением.

Принципы работы системы:

a. Школа или университет имеет множество подразделений, где учащимся преподаются курсы в течение определенного учебного периода.

b. Каждый из этих курсов ведут преподаватели.

c. Учащиеся регистрируются на различные курсы в течение учебного периода.

d. Помимо регистрационных данных, школа или университет хранит информацию о родителях учащихся.

e. Каждое подразделение ведет учет посещаемости, которая является критерием допуска к экзаменам за данный учебный период.

f. В конце каждого учебного периода проводятся экзамены, и результаты экзаменов сообщаются учащимся в установленные сроки.

g. В целях отчетности подразделение ведет журнал входов и выходов из системы, выполняемых преподавателями.

Создайте реляционную модель базы данных для системы управления образовательным учреждением.

Задание 2. Определить уникальные идентификаторы и соответствующие первичные ключи на основе предложенных сценариев.

1. Как найти определенную композицию в коллекции? Каким будет уникальный идентификатор объекта SONG?

2. У каждого учащегося в аудитории есть определенные черты или атрибуты. Какой атрибут (или атрибуты) вы выберете, чтобы выделить определенного учащегося среди всех остальных?

3. Каждого объекта из диаграммы системы управления образовательным учреждением в предыдущем задании.

В качестве ответа на данное задание необходимо представить:

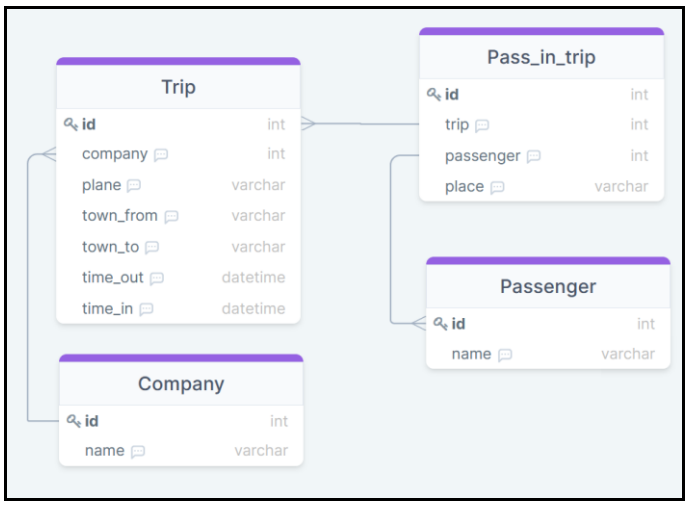
а) файл Microsoft Word с описанием объектов, из которых должна состоять спроектированная база данных (задание 1), и ответы на вопросы (задание 2).

б) архив с файлами диаграммы в Oracle SQL Developer Data Modeler (задание 1).

Практическая работа №4 «Выполнение SQL-запросов»

Ознакомиться с решением задач:

1. Вывести имена всех когда-либо обслуживаемых пассажиров авиакомпаний (https://sql-academy.org/ru/trainer/tasks/1):



Ответ:

```
SELECT name FROM Passenger
```

2. Вывести все рейсы, совершенные из Москвы (https://sql-academy.org/ru/trainer/tasks/3):

```
SELECT * FROM Trip WHERE town_from = 'Moscow'
```

3. Вывести имена людей, которые заканчиваются на "man" (https://sql-academy.org/ru/trainer/tasks/4):

```
SELECT name FROM Passenger WHERE name LIKE '%man'
```

4. Какие компании организуют перелеты с Владивостока (Vladivostok)? (https://sql-academy.org/ru/trainer/tasks/9)

```
SELECT DISTINCT Company.name FROM Company, Trip WHERE (Company.id = Trip.company) AND (town_from='Vladivostok')
```

5. Вывести количество рейсов, совершенных на TU-134 (https://sql-academy.org/ru/trainer/tasks/5)

```
SELECT COUNT(*) AS count FROM Trip WHERE plane='TU-134'
```

6. Удалить все перелеты, совершенные из Москвы (https://sql-academy.org/ru/trainer/tasks/56)

```
DELETE FROM Trip WHERE town_from = 'Moscow'
```

7. Измените имя "Andie Quincey" на новое "Andie Anthony" (https://sql-academy.org/ru/trainer/tasks/53)

```
UPDATE FamilyMembers SET member_name='Andie Anthony' WHERE member_name='Andie Quincey'
```

Самостоятельно решить задачи:

1. https://sql-academy.org/ru/trainer/tasks/2
2. https://sql-academy.org/ru/trainer/tasks/22
3. https://sql-academy.org/ru/trainer/tasks/31
4. https://sql-academy.org/ru/trainer/tasks/6
5. https://sql-academy.org/ru/trainer/tasks/34
6. https://sql-academy.org/ru/trainer/tasks/54
7. https://sql-academy.org/ru/trainer/tasks/59
8. https://sql-academy.org/ru/trainer/tasks/35
9. https://sql-academy.org/ru/trainer/tasks/38

Практическая работа № 5 «Знакомство с Access. Создание таблиц»

База данных (БД) – упорядоченная совокупность данных, предназначенных для хранения, накопления и обработки с помощью ЭВМ. Для создания и ведения баз данных (их обновления, обеспечения доступа по запросам и выдачи данных по ним пользователю) используется набор языковых и программных средств, называемых *системой управления базами данных* (СУБД).

Объекты базы данных Access

К объектам базы данных Access относятся:

1. *Таблицы* – предназначены для упорядоченного хранения данных.
2. *Запросы* – предназначены для поиска, извлечения данных и выполнения вычислений.
3. *Формы* – предназначены для удобного просмотра, изменения и добавления данных в

таблицах.

4. *Отчеты* – используются для анализа и печати данных.

5. *Страницы доступа к данным* – предназначены для просмотра, ввода, обновления и анализа данных через сеть или из любого места компьютера.

6. *Макросы* – используются для выполнения часто встречающегося набора макрокоманд, осуществляющих обработку данных.

7. *Модули* – предназначены для описания инструкций и процедур на языке VBA.

Основным объектом базы данных является таблица, которая состоит из записей (строк) и полей (столбцов). На пересечении записи и поля образуется ячейка, в которой содержатся данные.

Каждому полю таблицы присваивается уникальное имя, которое не может содержать более 64 символов. В каждом поле содержатся данные одного типа.

Типы данных

Тип	Описание
Текстовый	Используется для хранения символьных или числовых данных, не требующих вычислений. В свойстве Размер поля задается максимальное количество символов, которые могут быть введены в данное поле. По умолчанию размер устанавливается в 50 знаков. Максимальное количество символов, которые могут содержаться в текстовом поле, – 255.
Поле МЕМО	Предназначено для ввода текстовой информации, по объему превышающей 255 символов; может содержать до 65 536 символов
Числовой	Предназначен для хранения числовых данных, используемых в математических расчетах. На вкладках Общие и Подстановка можно установить свойства числового поля, среди которых Размер поля, Формат поля, Число десятичных знаков
Дата/Время	Используется для представления даты и времени. Выбор конкретного формата даты или времени устанавливается в свойстве *Формат даты*
Денежный	Предназначен для хранения данных, точность представления которых колеблется от 1 до 4 знаков после запятой. Целая часть может содержать до 15 десятичных знаков
Счетчик	Предназначен для автоматической вставки уникальных последовательных (увеличивающихся на 1) или случайных чисел в качестве номера новой записи. Номер, присвоенный записи, не может быть удален или изменен. Поля с этим типом

	данных используются в качестве ключевых полей таблицы
Логический	Предназначен для хранения одного из двух значений, интерпретируемых как «Да / Нет», «Истина / Ложь», «Вкл. / Выкл.»
Поле объекта OLE	Содержит данные, созданные в других программах, которые используют протокол OLE. Это могут быть, например, документы Word, электронные таблицы Excel, рисунки, звуковые и видеозаписи и др. Объекты OLE связываются с базой данных Access или внедряются в нее. Сортировать, группировать и индексировать поля объектов OLE нельзя
Гиперссылка	Специальный тип, предназначенный для хранения гиперссылок
Мастер подстановок	Предназначен для автоматического определения поля. С его помощью будет создано поле со списком, из которого можно

Создание базы данных

1. Запустите Microsoft Access 2007.

2. Нажмите на кнопку 

.

3. Задайте имя новой базы данных – «Записная книжка.accdb».

4. На вкладке ленты Создание в панели инструментов Таблицы нажмите на кнопку Конструктор таблиц.

5. Введите имена полей и укажите типы данных, к которым они относятся.

Имя поля	Тип данных
№ п/п	Счетчик
Фамилия	Текстовый
Имя	Текстовый
Адрес	Текстовый
Индекс	Числовой
Телефон	Текстовый
Хобби	Текстовый
Эл. почта	Гиперссылка

6. Выйдите из режима Конструктора, предварительно сохранив таблицу под именем

«Друзья»; ключевые поля не задавайте.

7. Откройте таблицу «Друзья» двойным щелчком мыши и заполните в ней 10 строк.

8. Добавьте поля «Отчество» и «Дата рождения», для этого:

1) установите курсор на поле, перед которым нужно вставить новый столбец;

2) выполните команду: вкладка ленты Режим таблицы → панель инструментов Поля и столбцы → Вставить;

3) щелкнув два раза на Поле1, переименуйте его в «Отчество», а Поле2 – «Дата рождения».

9. Перейдите в режим Конструктора командой: вкладка ленты Главная → Режимы → Конструктор.

10. Для поля «Дата рождения» установите тип данных Дата / время; в свойствах поля выберите Краткий формат даты.

11. Отформатируйте таблицу следующим образом: 1) цвет сетки – темно-красный; 2) цвет фона – голубой;

3) цвет текста – темно-красный, размер – 12 пт, начертание – курсив.

12. Переименуйте поле «Хобби» в «Увлечения».

13. Удалите запись под номером 8.

14. Измените размер ячеек так, чтобы были видны все данные. Для этого достаточно два раза щелкнуть левой кнопкой мыши на границе полей.

15. Расположите поля в следующем порядке: «№», «Фамилия», «Имя», «Отчество», «Телефон», «Дата рождения», «Увлечения», «Адрес», «Индекс», «Фото», «Эл_почта».

16. Заполните пустые ячейки таблицы.

17. В режиме Конструктора добавьте поле «Семейное положение», в котором будет содержаться фиксированный набор значений – замужем, не замужем, женат, не женат. Для создания раскрывающегося списка будем использовать Мастер подстановок:
1) установите тип данных Мастер подстановок;
2) в появившемся диалоговом окне выберите строку «Будет введен фиксированный набор значений» и нажмите кнопку Далее;
3) число столбцов – 1;
4) введите данные списка – замужем, не замужем, женат, не женат;
5) нажмите кнопку Готово.

18.С помощью раскрывающегося списка заполните новый столбец. Поскольку таблица

получилась широкая, то при заполнении данного столбца возникают некоторые неудобства: не видно фамилии человека, для которого заполняется поле «Семейное положение». Чтобы фамилия была постоянно видна при заполнении таблицы, необходимо воспользоваться командой Закрепить столбцы из контекстного меню поля «Фамилия».

19.Покажите работу преподавателю.

Критерии оценки:
Всего баллов – 3 (Максимально)
3 балла выставляется все шаги выполнены полностью правильно и самостоятельно, а студент способен уверенно отвечать на вопросы преподавателя о проделанной работе;
2 балла выставляется, если работа выполнена в целом корректна, но имеются неточности в ответах на вопросе преподавателя или некоторые действия выполнены неоптимально;
1 балл выставляется, когда работа не может быть завершена студентом самостоятельно, без помощи преподавателя.
0 баллов выставляется, если работа не была выполнена.

Практическая работа № 6 «Создание связей между таблицами»

1. Запустите Microsoft Access 2007.

2. Создадим базу данных «Фирма». Сотрудники данной организации работают с клиентами и выполняют их заказы.

Если все сведения поместить в одной таблице, то она станет очень неудобной для работы. В ней начнутся повторы данных. Всякий раз, когда сотрудник Иванов будет работать с какой-либо фирмой, придется прописывать данные о сотруднике и клиенте заново, в результате чего можно допустить множество ошибок. Чтобы уменьшить число ошибок, можно исходную таблицу разбить на несколько таблиц и установить связи между ними. Это будет более рационально, чем прежде.

Таким образом, необходимо создать 3 таблицы: *Сотрудники, Клиенты и Заказы*.

Сотрудники

Имя поля	**Тип данных**
Код сотрудника	Счетчик
Фамилия	Текстовый
Имя	Текстовый
Отчество	Текстовый
Должность	Текстовый
Телефон	Текстовый
Адрес	Текстовый
Дата рождения	Дата/Время
Заработная плата	Денежный

Фото	Объект OLE
Эл_почта	Гиперссылка

Клиенты

Имя поля	**Тип данных**
Код клиента	Счетчик
Название компании	Текстовый
Адрес	Текстовый
Номер телефона	Текстовый
Факс	Числовой
Адрес электронной почты	Гиперссылка
Заметки	Поле MEMO

Заказы

Имя поля	**Тип данных**
Код заказа	Счетчик
Код клиента	Числовой
Код сотрудника	Числовой
Дата размещения	Дата/Время
Дата исполнения	Дата/Время
Сумма	Денежный
Отметка о выполнении	Логический

3. Отдельные таблицы, содержащие информацию по определенной теме, необходимо связать в единую структуру базы данных. Для связывания таблиц следует задать ключевые поля. Ключ состоит из одного или нескольких полей, значения которых однозначно определяют каждую запись в таблице. Наиболее подходящим в качестве ключевого поля является «Счетчик», так как значения в данном поле являются уникальными (т. е. исключают повторы).

4. Откройте таблицу Сотрудники в режиме Конструктора.

5. Нажмите правой кнопкой мыши на поле Код сотрудника и в появившемся контекстном меню выберите команду Ключевое поле. Если в таблице необходимо установить несколько ключевых полей, то выделить их можно, удерживая клавишу Ctrl.

6. Для таблицы Клиенты установите ключевое поле Код клиента, а для таблицы Заказы – Код заказа.

7. Таблица Заказы содержит поля Код сотрудника и Код клиента. При их заполнении могут возникнуть некоторые трудности, так как не всегда удается запомнить все предприятия, с которыми работает фирма, и всех сотрудников с номером кода. Для удобства можно создать раскрывающиеся списки с помощью Мастера подстановок.

8. Откройте таблицу Заказы в режиме Конструктора.

9. Для поля Код сотрудника выберите тип данных Мастер подстановок.

10. В появившемся окне выберите команду «Объект «столбец подстановки» будет использовать значения из таблицы или запроса» и щелкните на кнопке Далее.

11. В списке таблиц выберите таблицу Сотрудники и щелкните на кнопке Далее.

12. В списке Доступные поля выберите поле Код сотрудника и щелкните на кнопке со стрелкой, чтобы ввести поле в список Выбранные поля. Таким же образом добавьте поля Фамилия и Имя и щелкните на кнопке Далее.

13. Выберите порядок сортировки списка по полю Фамилия.

14. В следующем диалоговом окне задайте необходимую ширину столбцов раскрывающегося списка.

15. Установите флажок Скрыть ключевой столбе» и нажмите кнопку Далее.

16. На последнем шаге Мастера подстановок замените при необходимости надпись для поля подстановок и щелкните на кнопке Готово.

17. Аналогичным образом создайте раскрывающийся список для поля

Код клиента.

18. После создания ключевых полей можно приступить к созданию связей. Существует несколько типов отношений между таблицами:

1) при отношении «один-к-одному» каждой записи ключевого поля в первой таблице соответствует только одна запись в связанном поле другой таблицы, и наоборот. Отношения такого типа используются не очень часто. Иногда их можно использовать для разделения таблиц, содержащих много полей, для отделения части таблицы по соображениям безопасности;

2) при отношении «один-ко-многим» каждой записи в первой таблице соответствует несколько записей во второй, но запись во второй таблице не может иметь более одной связанной записи в первой таблице;

3) при отношении «многие-ко-многим» одной записи в первой таблице могут соответствовать несколько записей во второй таблице, а одной записи во второй таблице могут соответствовать несколько записей в первой.

19. Закройте все открытые таблицы, так как создавать или изменять связи между открытыми таблицами нельзя.

20.Выполните команду: вкладка ленты Работа с базами данных →

кнопка 

.

21. Если ранее никаких связей между таблицами базы не было, то при открытии окна Схема данных одновременно открывается окно Добавление таблицы, в котором выберите таблицы Сотрудники, Клиенты и Заказы.

22. Если связи между таблицами уже были заданы, то для добавления в схему данных новой таблицы щелкните правой кнопкой мыши на схеме данных и в контекстном меню выберите пункт Добавить таблицу.

23. Установите связь между таблицами Сотрудники и Заказы, для этого выберите поле Код сотрудника в таблице Сотрудники и перенесите его на соответствующее поле в таблице Заказы.

24. После перетаскивания откроется диалоговое окно Изменение связей (рис. 1), в котором включите флажок Обеспечение условия целостности. Это позволит предотвратить случаи удаления записей из одной таблицы, при которых связанные с ними данные других таблиц останутся без связи.

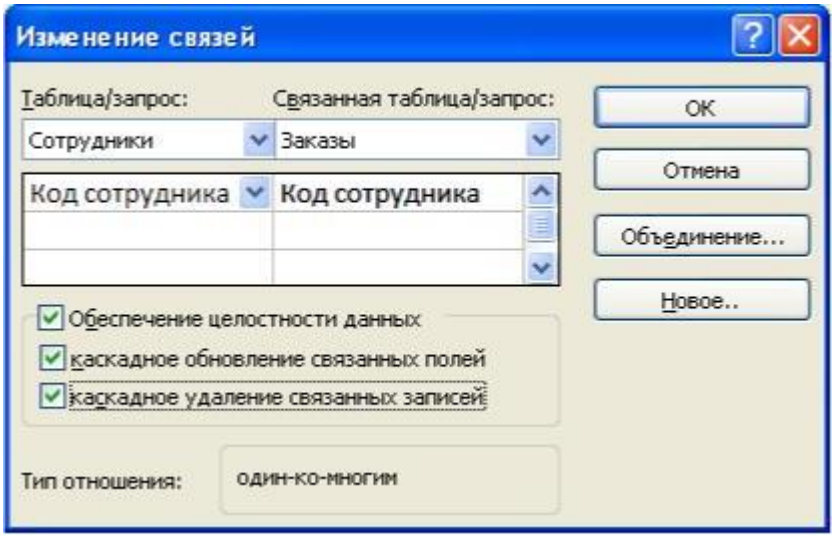


Рис. 1. Создание связи между таблицами

25. Флажки Каскадное обновление связанных полей и Каскадное удаление связанных записей обеспечивают одновременное обновление или удаление данных во всех подчиненных таблицах при их изменении в главной таблице.

26. Параметры связи можно изменить, нажав на кнопку Объединение.

27. После установления всех необходимых параметров нажмите кнопку ОК.

28. Связь между таблицами Клиенты и Заказы установите самостоятельно.

29. В результате должна получиться схема данных, представленная на рис. 2.

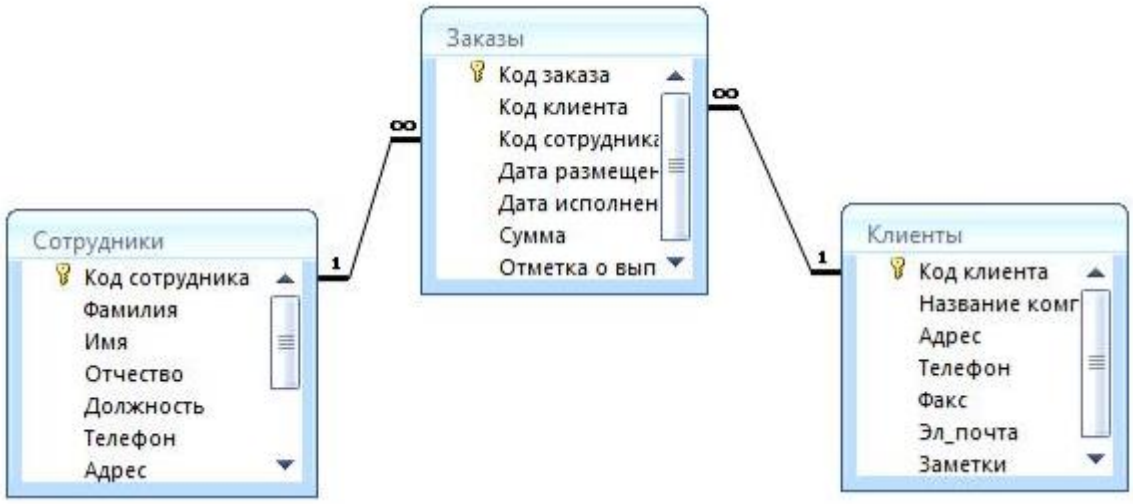


Рис. 2. Схема данных

В приведенном примере используются связи «один-ко-многим». На схеме данных они отображаются в виде соединительных линий со специальными значками около таблиц. Связь «один-ко-многим» помечается «1» вблизи главной таблицы (имеющей первичный ключ) и «∞» вблизи подчиненной таблицы (имеющей внешний ключ). Связь «один-к-одному» помечается двумя «1» (оба поля таблиц имеют первичные ключи). Неопределенная связь не имеет никаких знаков. Если установлено объединение, то его направление отмечается стрелкой на конце соединительной линии (ни одно из объединенных полей не является ключевым и не имеет уникального индекса).

30. В таблицу Сотрудники внесите данные о семи работниках.

31. В таблицу Клиенты внесите данные о десяти предприятиях, с которыми работает данная фирма.

32. В таблице Заказы оформите несколько заявок, поступивших на фирму.

33. Покажите работу преподавателю.

Критерии оценки:
Всего баллов – 3 (Максимально)
3 балла выставляется все шаги выполнены полностью правильно и самостоятельно, а студент способен уверенно отвечать на вопросы преподавателя о проделанной работе;
2 балла выставляется, если работа выполнена в целом корректна, но имеются неточности в ответах на вопросе преподавателя или некоторые действия выполнены неоптимально;
1 балл выставляется, когда работа не может быть завершена студентом самостоятельно, без помощи преподавателя.
0 баллов выставляется, если работа не была выполнена.

Практическая работа № 7 «Отбор данных с помощью запросов»

Запросы являются основным средством просмотра, отбора, изменения и анализа информации, которая содержится в одной или нескольких таблицах базы данных.

Существуют различные виды запросов, но наиболее распространенными являются *запросы на выборку*, с них и начнем наше знакомство.

1. Откройте базу данных «*Фирма*», созданную ранее.

2. Выполните команду: вкладка ленты Создание → Мастер запросов → Простой запрос.

3. В появившемся диалоговом окне (рис. 3) укажите таблицу Сотрудники и выберите поля Фамилия, Имя, Телефон. Нажмите кнопку Далее.

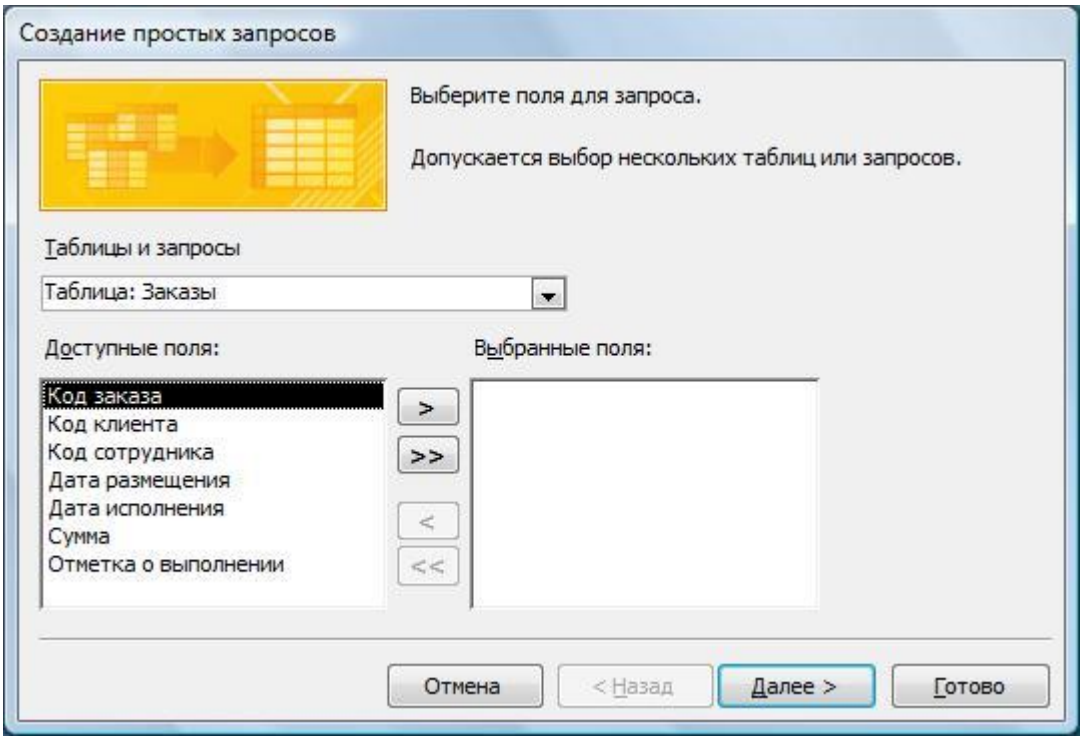


Рис. 3. Создание простого запроса

4. Введите имя запроса – *Телефоны* – и нажмите кнопку *Готово*. Перед вами появится запрос, в котором можно просмотреть телефоны сотрудников.

5. Следующий запрос попробуйте создать с помощью Конструктора, для этого выполните команду: вкладка ленты *Создание → Конструктор запросов.*

6. В диалоговом окне *Добавление таблиц* выберите таблицу *Клиенты* и щелкните на кнопке *Добавить*, а затем – на кнопке *Закрыть*.

7. Чтобы перенести нужные поля в бланк запроса, необходимо по ним дважды щелкнуть левой кнопкой мыши (рис. 4).

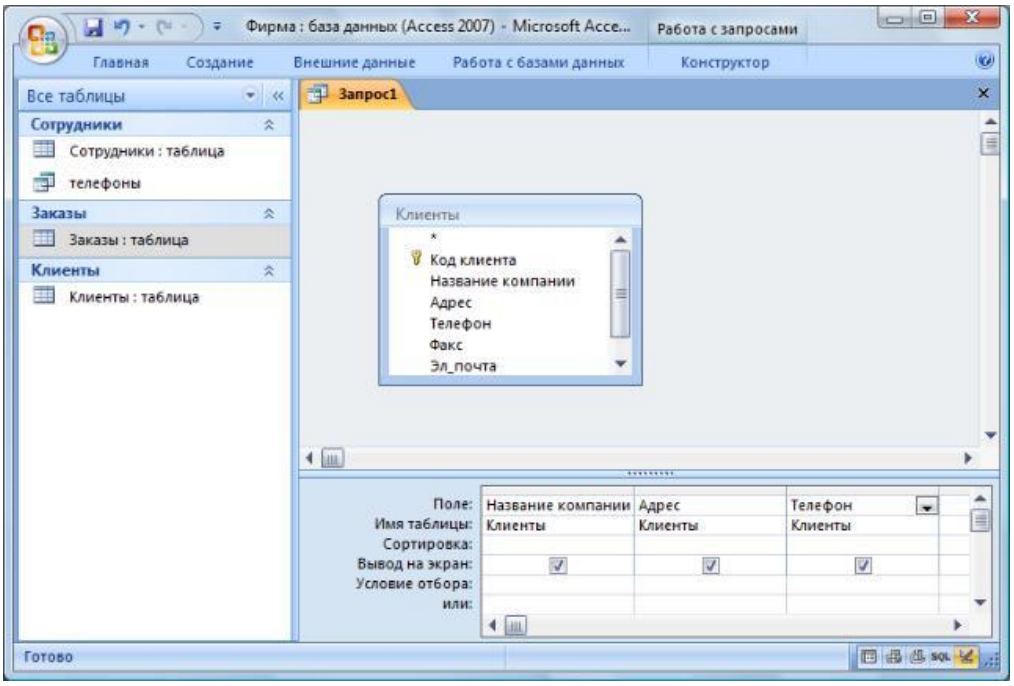


Рис. 4. Создание запроса в режиме Конструктора

8. Чтобы отсортировать записи в поле Название компании в алфавитном порядке, необходимо в раскрывающемся списке строки Сортировка выбрать пункт по возрастанию.

9. Сохраните запрос с именем «Адреса клиентов».

10. Самостоятельно создайте запрос «Дни рождения», в котором можно будет просмотреть дни рождения сотрудников.

11. Допустим, мы хотим узнать, у кого из сотрудников день рождения в текущем месяце, например в апреле. Для этого откройте запрос в режиме Конструктора.

12. В строке Условие отбора для поля «Дата рождения» введите значение *.04.*. В данной записи * означают, что дата и год рождения могут быть любыми, а месяц 4-м (т. е. апрель). После этого окно запроса должно выглядеть так, как оно представлено на рис. 5.

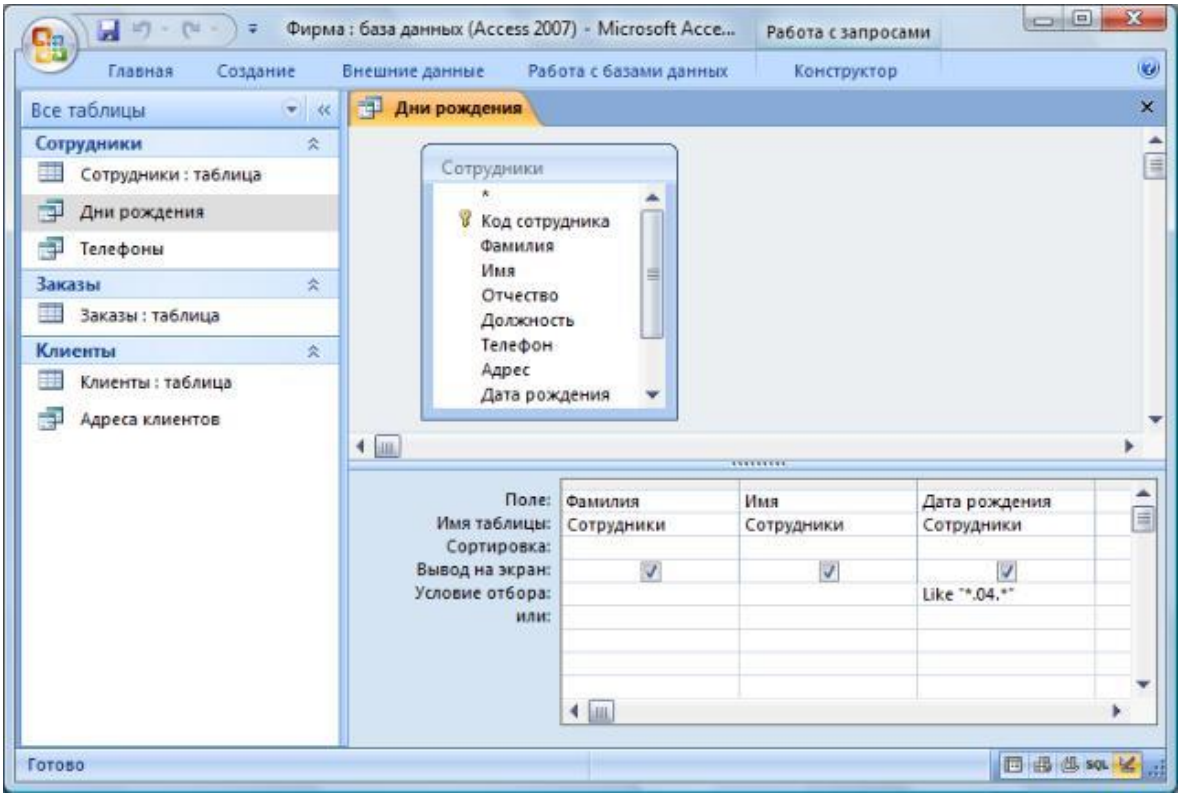


Рис. 5. Создание запроса

13. Закройте Конструктор и просмотрите полученный результат. Если в запросе Дни рождения нет ни одной записи, значит, в таблице *Сотрудники* нет ни одного человека, родившегося в апреле. Добавьте в таблицу *Сотрудники* несколько человек, родившихся в апреле, и посмотрите, как изменится запрос. Запросы автоматически обновляются при каждом открытии.

14. Если нам нужно узнать, кто из сотрудников родился в мае, то придется создать новый запрос или изменить условие в существующем запросе *Дни рождения*. Данная процедура является неудобной и занимает много времени. Если приходится часто выполнять запрос, но каждый раз с новыми значениями условий используют запрос с параметром. При запуске такого запроса на экран выводится диалоговое окно для ввода значения в качестве условия отбора. Чтобы создать запрос с параметром, пользователю необходимо ввести текст сообщения в строке *Условие отбора* бланка запроса (рис. 6).

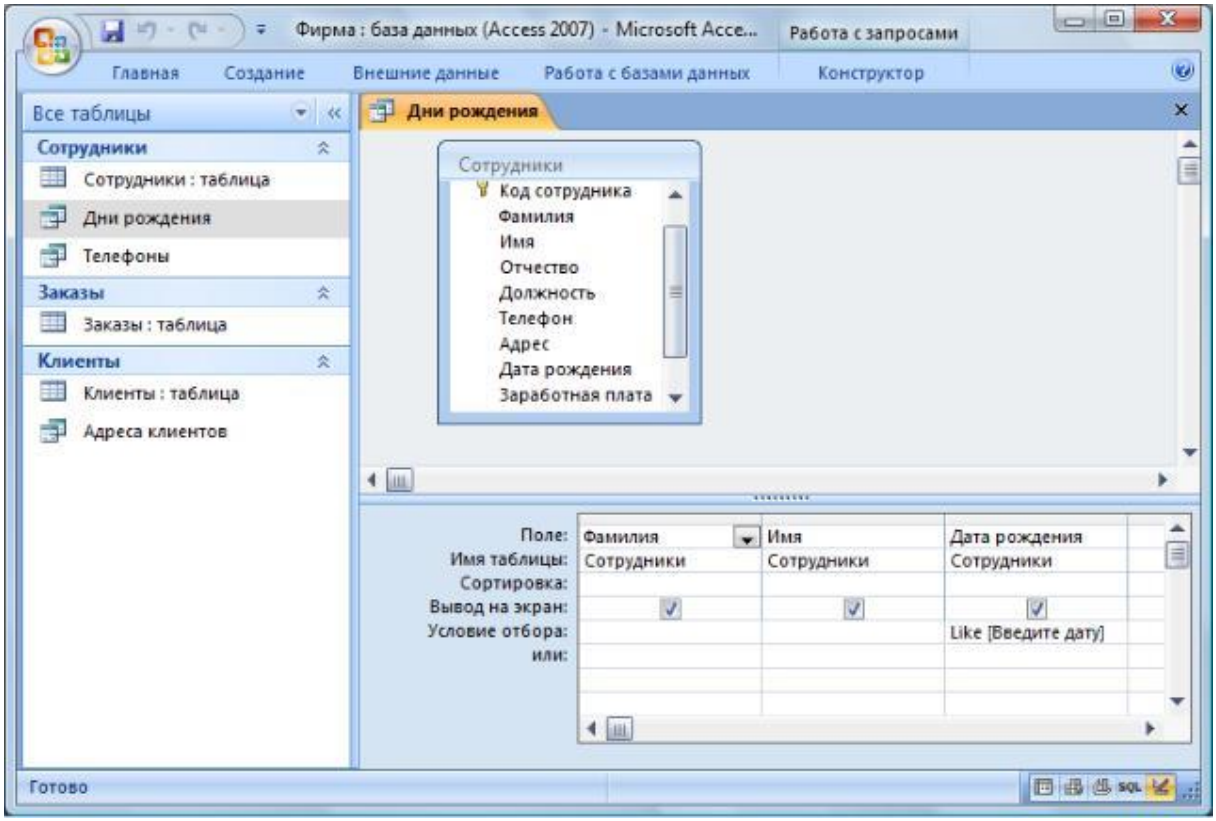


Рис. 6. Создание запроса с параметром

15. Запись Like[*Введите дату*] означает, что при открытии запроса появится диалоговое окно (рис. 7) с текстом «*Введите дату*» и полем для ввода условия отбора. Если ввести условие *.04.*, то в запросе появится список сотрудников, родившихся в апреле. Запустите запрос еще раз и введите значение *.05.*, посмотрите, как изменился запрос.

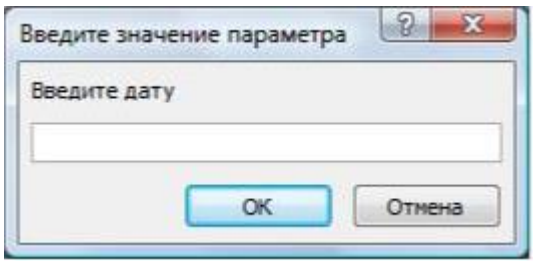


Рис. 7. Окно для ввода условия отбора

16. Измените запрос «*Телефоны*» так, чтобы при его запуске выводилось диалоговое окно с сообщением «*Введите фамилию*». Поскольку в запросе нужно вывести конкретную фамилию, в условии отбора слово *Like* писать не надо.

17. Измените запрос «Телефоны» так, чтобы при его запуске запрашивались не только фамилия, но и имя сотрудника.

18. Самостоятельно создайте запрос «Выполненные заказы», содержащий следующие сведения: фамилия и имя сотрудника, название компании, с которой он работает, отметка о выполнении и сумма заказа. Данные запроса возьмите из нескольких таблиц.

19. В условии отбора для логического поля *Отметка о выполнении* введите *Да*, чтобы в запросе отображались только выполненные заказы.

20. Сделайте так, чтобы столбец *Отметка о выполнении* не выводился на экран.

21. Создайте запрос *Сумма заказа*, в котором будут отображаться заказы на сумму более 50 000 руб.

22. Измените запрос, чтобы сумма заказа была от 20 000 до 50 000 руб. Для данных запросов в условии отбора можно использовать операторы сравнения >, <, =, >=, <=, < > и логические операторы And, Or, Not и др.

23. Иногда в запросах требуется произвести некоторые вычисления, например посчитать подоходный налог 13 % для каждой сделки. Для этого откройте запрос Сумма заказа в режиме Конструктора.

24. В пустом столбце бланка запроса щелкните правой кнопкой мыши на ячейке Поле и в появившемся контекстном меню выберите команду *Построить*. Перед вами появится окно *Построитель выражений* (рис. 8), который состоит из трех областей: поля выражения, кнопок операторов и элементов выражения. Сверху располагается поле выражения, в котором оно и создается. Вводимые в это поле элементы выбираются в двух других областях окна Построителя.

25. В левом списке откройте папку Запросы и выделите запрос Сумма заказа. В среднем списке выделите поле Сумма и нажмите кнопку Вставить. Идентификатор этого поля появится в поле выражения Построителя.

26. Щелкните на кнопке * и введите 0,13 (см. рис. 8). Таким образом, мы посчитаем подоходный налог 13 %.

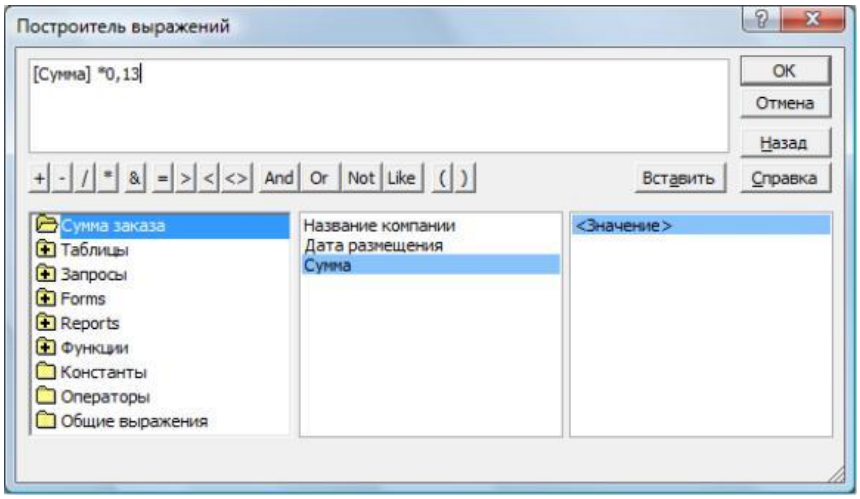


Рис. 8. Построитель выражений

27. Нажмите кнопку ОК, после чего в ячейке свойства Поле появится значение «Выражение1: [Сумма]*0,13».

28. Замените Выражение1 на Налог и закройте Конструктор.

29. Откройте запрос и посмотрите, что у вас получилось.

30. Используя Построитель выражений, добавьте в запрос Сумма заказа поле Прибыль, в котором будет вычисляться доход от заказа (т. е. сумма минус налог).

31. Создайте запрос Менеджеры, с помощью которого в таблице Сотрудники найдите всех менеджеров фирмы.

32. Покажите работу преподавателю.

Критерии оценки:
Всего баллов – 3 (Максимально)
3 балла выставляется все шаги выполнены полностью правильно и самостоятельно, а студент способен уверенно отвечать на вопросы преподавателя о проделанной работе;
2 балла выставляется, если работа выполнена в целом корректна, но имеются неточности в ответах на вопросе преподавателя или некоторые действия выполнены неоптимально;
1 балл выставляется, когда работа не может быть завершена студентом самостоятельно, без помощи преподавателя.
0 баллов выставляется, если работа не была выполнена.

Практическая работа № 8 «Использование форм в базе данных»

Формы – это объекты базы данных, предназначенные для просмотра данных из таблиц и запросов, для ввода данных в базу, корректирования существующих данных и выполнения заданных действий. Форма может содержать графики, рисунки и другие внедренные объекты.

Можно вносить данные в таблицы и без помощи каких-либо форм. Но существует несколько причин, которые делают формы незаменимым средством ввода данных в базу:

1) при работе с формами ограничен доступ к таблицам (самому ценному в базе данных);

2) разные люди могут иметь разные права доступа к информации, хранящейся в базе. Для ввода данных им предоставляются разные формы, хотя данные из форм могут поступать в одну таблицу;

3) вводить данные в форму легче, чем в таблицу, и удобнее, так как в окне формы отображается, как правило, одна запись таблицы;

4) в большинстве случаев информация для баз данных берется из бумажных бланков (анкет,

счетов, накладных, справок и т. д.). Экранные формы можно сделать точной копией бумажных бланков, благодаря этому уменьшается количество ошибок при вводе и снижается утомляемость персонала.

Создавать формы можно на основе нескольких таблиц или запросов с помощью Мастера, используя средство автоформы, «вручную» в режиме Конструктора, сохраняя таблицу или запрос как форму. Созданную любым способом форму можно затем изменять в режиме Конструктора. Рассмотрим некоторые из перечисленных способов.

1. Выполните команду: вкладка ленты Создание → панель инструментов

Формы → Другие формы → Мастер форм.

2. В диалоговом окне Создание форм выберите таблицы (запросы) и поля, которые будут помещены в форму. Щелкните по кнопке Далее.

3. В следующих диалоговых окнах мастера выберите внешний вид формы, стиль, задайте имя формы. Щелкните по кнопке Готово.

4.С помощью Мастера создайте формы Сотрудники, Клиенты, Заказы, Менеджеры.

5. Откройте форму Сотрудники в режиме Конструктора. Этот режим предназначен для создания и редактирования форм.

6. Разместите элементы в удобном для вас порядке, измените размер и цвет текста.

7. В заголовок формы добавьте текст Сотрудники фирмы.

8. В примечание формы добавьте объект Кнопка (вкладка ленты Конструктор → панель инструментов Элементы управления).

9. После того как вы «нарисуете» кнопку указателем, на экране появится диалоговое окно Создание кнопок (рис. 9).

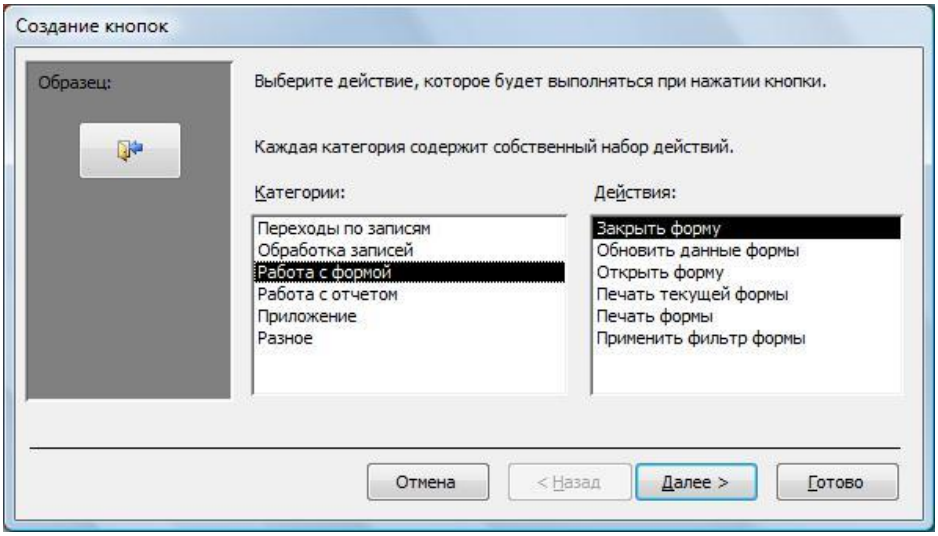


Рис. 9. Создание кнопок на форме

10. В категории Работа с формой выберите действие Закрытие формы и нажмите кнопку Далее.

11. Выберите рисунок или текст, который будет размещаться на кнопке.

12. В последнем диалоговом окне Мастера кнопок задайте имя кнопки и нажмите Готово.

13. Мастер кнопок написал для данной кнопки процедуру на языке Microsoft Visual Basic. Просмотреть процедуру обработки события можно с помощью команды Обработка событий контекстного меню кнопки.

14. Самостоятельно создайте кнопки Выход из приложения, Поиск записи, Удаление записи.

15. Иногда на форме требуется разместить несколько страниц, содержащих данные из различных источников, справочную или вспомогательную информацию. Для этой цели можно использовать набор вкладок.

16. Создайте пустую форму.

17. Для добавления к форме набора вкладок щелкните по кнопке Вкладка на панели инструментов Элементы управления. Сначала добавятся только две вкладки с формальными именами Вкладка 1 и Вкладка 2.

18. Добавьте еще одну вкладку.

19. Переименуйте ярлычки вкладок так, чтобы на них отображались названия данных, которые будут в них располагаться: Сотрудники, Менеджеры, Помощь.

20. Перейдите на вкладку Сотрудники и перетащите на нее мышкой из базы данных форму Сотрудники.

21. Аналогичным образом поместите форму Менеджеры на вкладку Менеджеры.

22. На вкладку Помощь поместите некоторые советы по работе с базой данных.

23. Данную форму сохраните с именем Сотрудники фирмы.

24. В Microsoft Access можно создавать кнопочные формы. Они содержат только кнопки и предназначены для выбора основных действий в базе данных. Для создания кнопочной формы необходимо на вкладке ленты Работа с базами данных выбрать команду Диспетчер кнопочных форм.

25. Если кнопочной формы в базе данных нет, то будет выведен запрос на подтверждение ее создания. Нажмите Да в диалоговом окне подтверждения.

26. Перед вами появится Диспетчер кнопочных форм, в котором щелкните по кнопке Создать.

27. В диалоговом окне Создание (рис. 10) введите имя новой кнопочной формы и нажмите ОК.

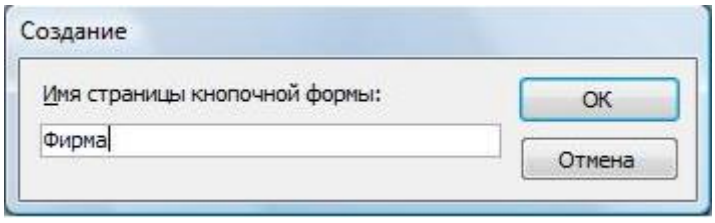


Рис. 10. Задание имени кнопочной формы

28. Имя новой кнопочной формы добавится в список Страницы кнопочной формы окна Диспетчер кнопочных форм (рис. 11). Выделите имя новой кнопочной формы и щелкните по кнопке Изменить.

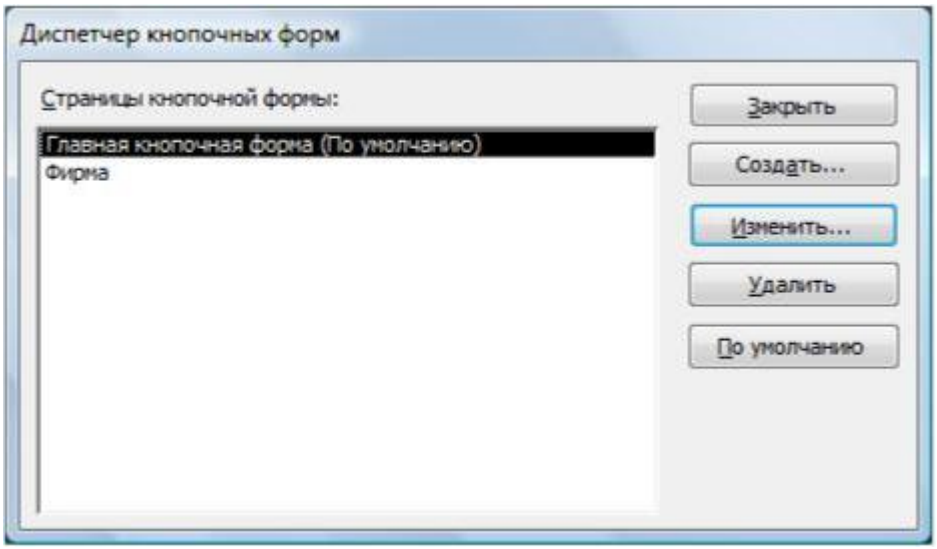


Рис. 11. Диспетчер кнопочных форм

29. В диалоговом окне Изменение страницы кнопочной формы щелкните по кнопке Создать. Появится диалоговое окно Изменение элемента кнопочной формы (рис. 12).

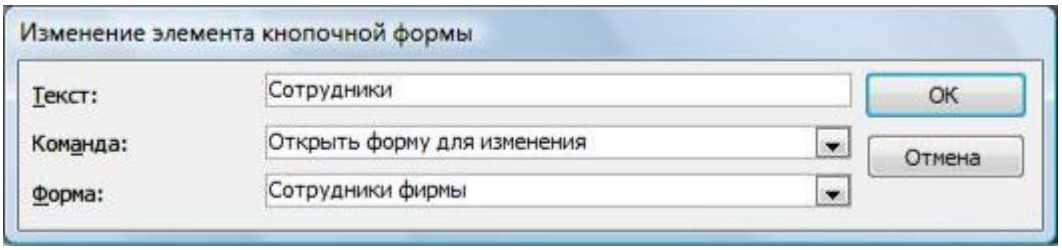


Рис. 12. Создание кнопок на форме

30. В поле Текст введите текст подписи для первой кнопки кнопочной формы, а затем выберите команду из раскрывающегося списка в поле Команда.

В поле Форма выберите форму, для которой будет выполняться данная команда.

31. Аналогичным образом добавьте кнопки Клиенты, Заказы, Выход.

32. В диалоговом окне Диспетчер кнопочных форм выберите имя вашей кнопочной формы и щелкните по кнопке По умолчанию.

Рядом с названием кнопочной формы появится надпись «(по умолчанию)».

33. Чтобы закончить создание кнопочной формы, щелкните по кнопке Закрыть.

34. В результате должна получиться форма, представленная на рис. 13.

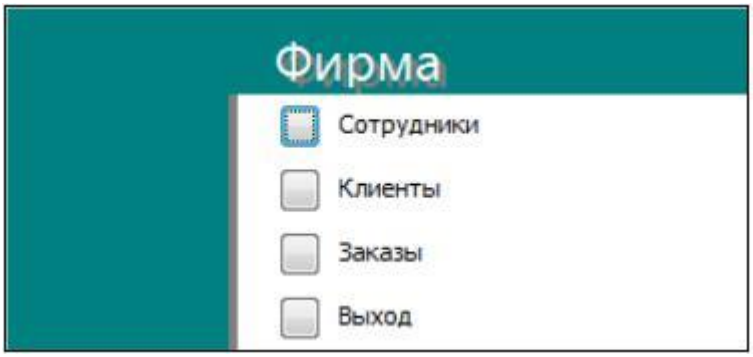


Рис. 13. Главная кнопочная форма

35.Добавьте в форму какой-нибудь рисунок.

36.Для того чтобы главная кнопочная форма появлялась на экране при запуске приложения, необходимо в главном меню нажать на кнопку Параметры Access (рис. 14). Для текущей

базы данных установите форму просмотра – «кнопочная форма».

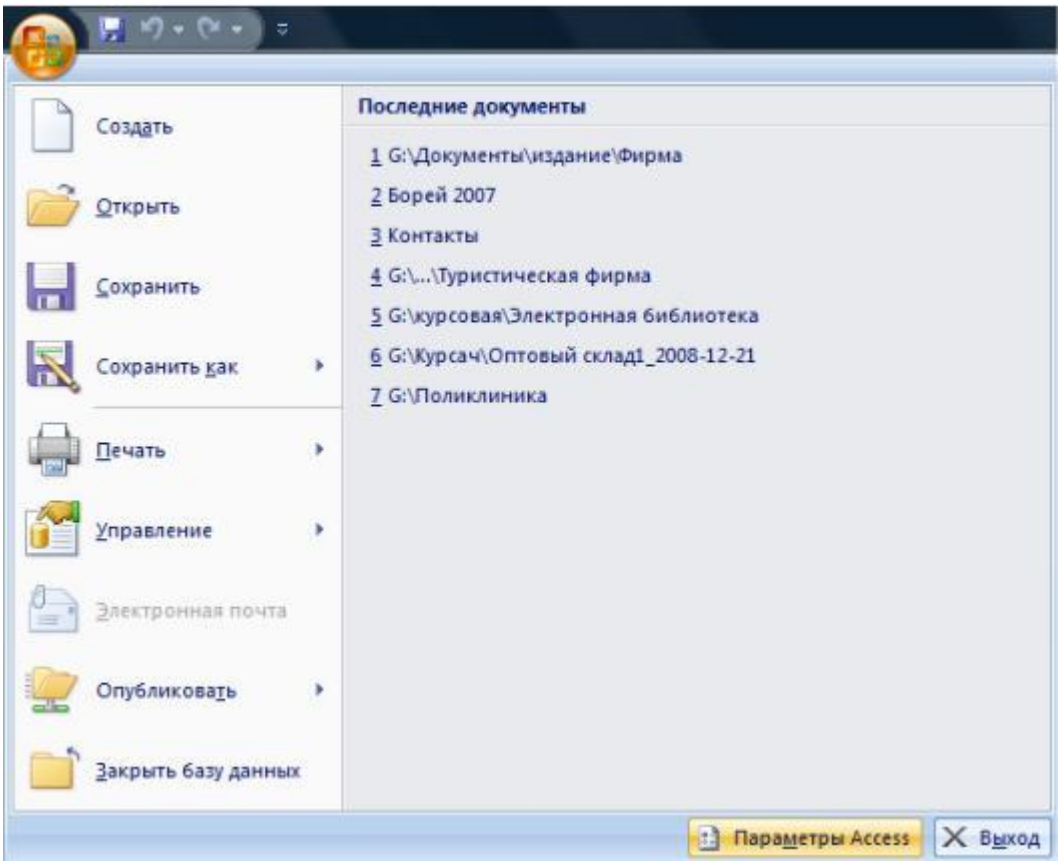


Рис. 14. Задание параметров Access

37. Покажите работу преподавателю.

Критерии оценки:
Всего баллов – 3 (Максимально)
3 балла выставляется все шаги выполнены полностью правильно и самостоятельно, а студент способен уверенно отвечать на вопросы преподавателя о проделанной работе;
2 балла выставляется, если работа выполнена в целом корректна, но имеются неточности в ответах на вопросе преподавателя или некоторые действия выполнены неоптимально;
1 балл выставляется, когда работа не может быть завершена студентом самостоятельно, без помощи преподавателя.
0 баллов выставляется, если работа не была выполнена.

Практическая работа № 9 «Создание отчетов»

Отчеты предназначены для вывода информации на печать. Часто данные в них располагаются в табличной форме. В отличие от распечаток таблиц или запросов отчет дает более широкие возможности сортировки и группировки данных, он предоставляет возможность добавлять итоговые значения, а также поясняющие надписи, колонтитулы, номера страниц, стили и различные графические элементы.

Создавать отчеты в базе данных Access можно несколькими способами: 1) с помощью Мастера отчетов;

2) на основе таблиц или запросов;

3) в режиме Конструктора.

1. В окне базы данных выполните команду: вкладка ленты Создание → панель инструментов Отчеты → Мастер отчетов.

2. Выберите из списка таблицу (или запрос), которая будет использована как источник данных (например, запрос Адреса клиентов).

3. В появившемся диалоговом окне Создание отчетов (рис. 15) переместите все доступные поля в область «выбранные поля».

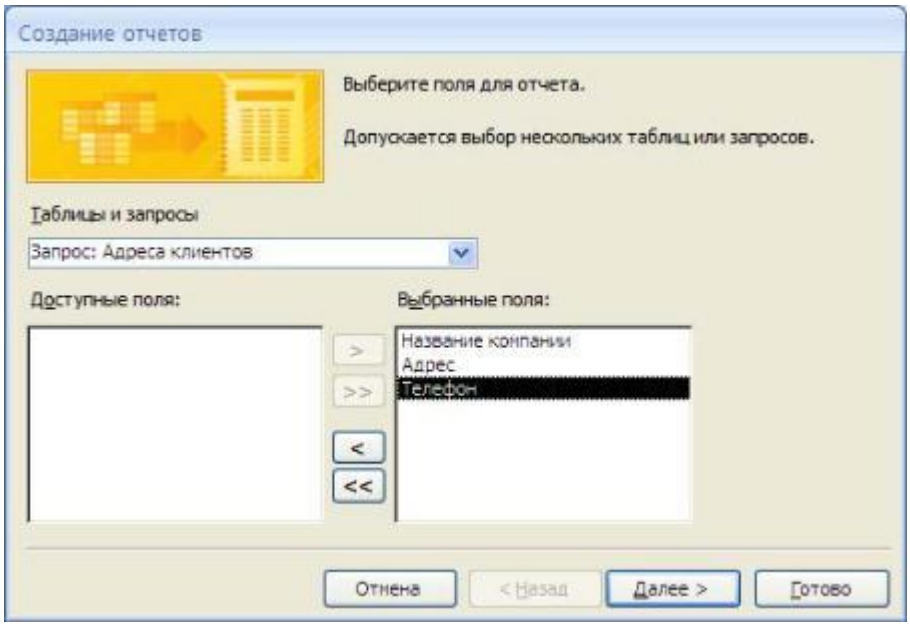


Рис. 15. Мастер отчетов

4. С помощью *Мастера отчетов* создайте отчет *Дни рождения*. В качестве источника данных используйте таблицу *Сотрудники*.

5. Если требуется напечатать почтовые наклейки, Access предоставляет такую возможность. Для этого выделите таблицу *Клиенты* и выполните команду: вкладка ленты *Создание* → панель инструментов *Отчеты → Наклейки*.

6. В появившемся диалоговом окне (рис. 16) укажите размер наклейки, систему единиц, тип наклейки и нажмите кнопку *Далее*.

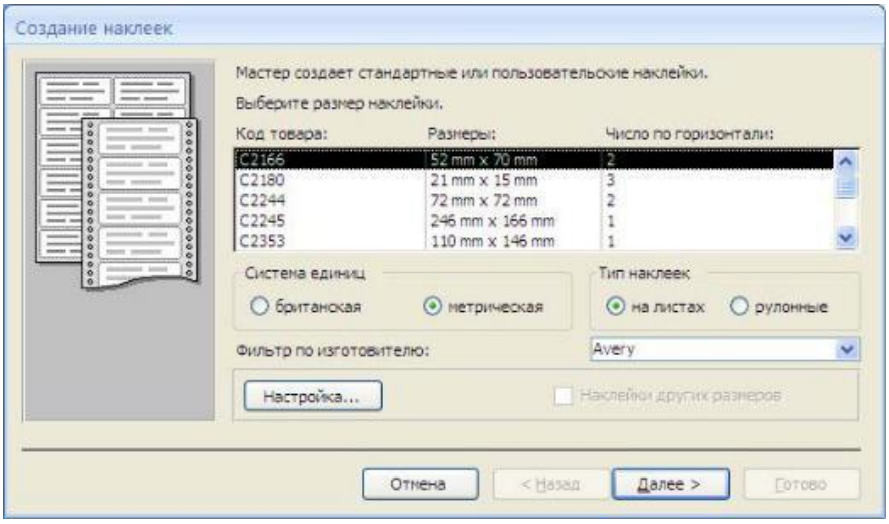


Рис. 16. Диалоговое окно Создание наклеек

7. На следующем шаге создания отчета установите шрифт, размер, цвет текста и начертание. Нажмите кнопку Далее.

8. Выберите поля, которые будут размещаться на наклейке. Например,

Название компании, Адрес, Телефон и Факс. Если на каждой наклейке требуется вывести определенный текст, то введите его в прототип наклейки.

9. При необходимости измените название отчета с наклейками и нажмите кнопку Готово.

10. Иногда в отчетах требуется вычислять итоговые значения, среднее, минимальное или максимальное значения, а также проценты. Для этого запустите Мастер отчетов и в качестве источника данных укажите запрос Сумма заказа.

11. В диалоговом окне Мастера, в котором задается порядок сортировки записей, нажмите кнопку Итоги (рис. 17).

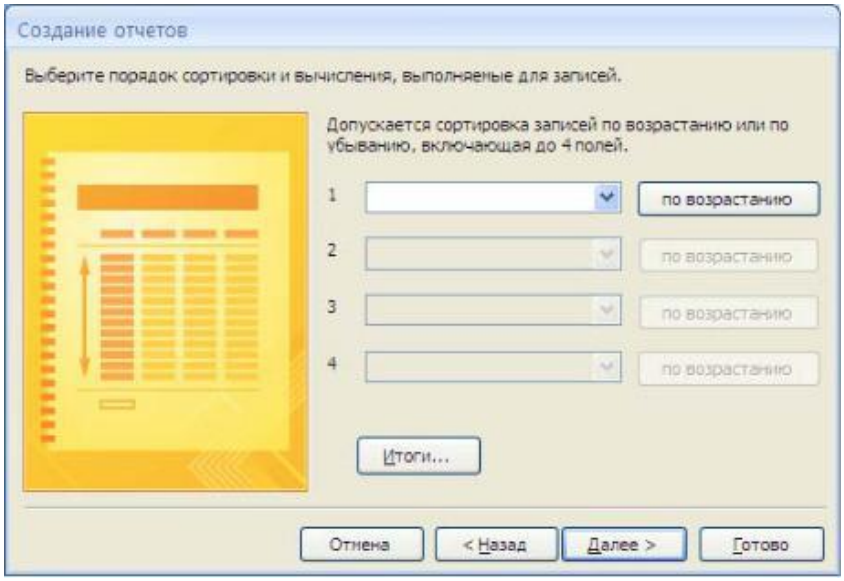


Рис. 17. Вычисление итоговых значений в отчетах

12. В диалоговом окне Итоги (рис. 18) для полей Сумма и Налог установите флажки в столбце sum, чтобы посчитать итоговую сумму.

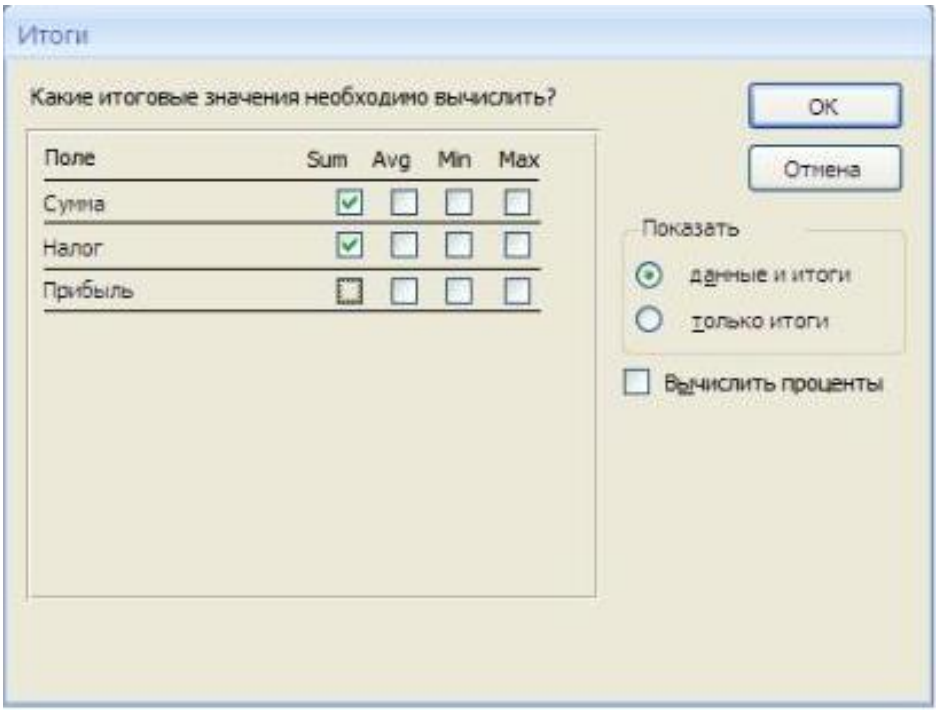


Рис. 18. Вычисление итоговых значений суммы

13. Далее выполните все шаги Мастера и нажмите кнопку *Готово*.

14. Создайте отчет Дни рождения, используя в качестве источника данных запрос *Дни рождения*.

15. Составьте отчет *Выполненные заказы*, в котором будут данные о компании и сумме заказа. Вычислите итоговую сумму, среднее значение (Avg) и максимальную сумму для каждой фирмы.

16. Покажите работу преподавателю.

Критерии оценки:
Всего баллов – 3 (Максимально)
3 балла выставляется все шаги выполнены полностью правильно и самостоятельно, а студент способен уверенно отвечать на вопросы преподавателя о проделанной работе;
2 балла выставляется, если работа выполнена в целом корректна, но имеются неточности в ответах на вопросе преподавателя или некоторые действия выполнены неоптимально;
1 балл выставляется, когда работа не может быть завершена студентом самостоятельно, без помощи преподавателя.
0 баллов выставляется, если работа не была выполнена.

Задания для самостоятельной работы студентов:

1. Создать проект базы данных «Фирма», представленный отношениями:

- Филиалы (Филиал, Страна_Филиала),
- Заказы (Филиал, Заказчик, № Заказа),
- Товары (№ заказа, Товар, Количество, Цена),
- Клиенты (Заказчик, Страна_Заказчика).

Обозначить ключевые поля, поля связи, тип связи.

2. Создать проект базы данных «Банк», представленный отношениями:

- Клиент (ФИО клиента, № филиала, № счета, Остаток, Кредит),
- Филиал (№ Филиала, Район)

Обозначить ключевые поля, поля связи, тип связи.

3. Создать проект базы данных «Программные комплексы», представленный отношениями:

- Файлы (Файл, Сотрудник),
- ПК_Файлы (Код_ПК, Файл),
- Программные комплексы (Код_ПК, Название ПК),
- Сотрудники (Сотрудник, Отдел)

Обозначить ключевые поля, поля связи, тип связи.

4. Открыть БД Банк. Составить запрос: Клиентов, которые имеют счета с остатком, превышающим 10 000 руб.

5. Открыть БД Банк. Составить запрос: Клиентов, которые имеют по одному счету

6. Открыть БД Банк. Составить запрос: Выбрать клиентов, имеющих на счете до 200руб.

7. Открыть БД Банк. Составить запрос: Выбрать клиентов, имеющих на счете 0 руб.

8. Открыть БД Банк. Составить запрос: Выбрать клиентов из Салавата и Ишимбая

9. Открыть БД Банк. Составить запрос: Выбрать клиентов, у которых сумма кредита больше суммы остатка

10. Открыть БД Банк. Составить запрос: Клиентов, у которых сумма кредита не превышает сумму остатка более чем в два раза

11. Открыть БД Банк. Составить запрос: Клиентов, которые имеют на счету остаток ноль.

12. Открыть БД Банк. Составить запрос: Клиентов, которые имеют несколько счетов

13. Открыть БД Банк. Составить запрос: Филиалы с общей суммой выданного кредита

14. Открыть БД Банк. Составить запрос: Клиентов, обслуживающихся в филиале №3

15. Открыть БД «Сотрудники». Составить запрос: «Должности, занимаемые сотрудниками». Поле «ФИО» в запросе отсортировать по возрастанию

16. Открыть БД «Сотрудники». Составить запрос: «Работающие, свыше 7 лет», поле - стаж - не выводить на экран

17. Открыть БД «Сотрудники». Составить запрос: «Сложность работ» (сложность выполняемых работ – по возрастанию)

18. Открыть БД «Сотрудники». Составить запрос: «Сотрудники – мужчины 1978 и 1979 г. рождения». Вывести в запросе поля: ФИО, Должность. Поле «дата рождения» - не выводить.

19. Открыть БД «Сотрудники». Составить запрос: «Средняя заработная плата»

20. Открыть БД «Сотрудники». Составить запрос: «Минимальная заработная плата»

Критерии оценки:
Всего баллов – 20 (Максимально)
По 0,5 балла за каждое правильное самостоятельно решённое задание.

Пример проектного задание для студентов:

Необходимо разработать следующие SQL-скрипты:

1. SQL-скрипт для создания таблиц БД и связей между ними, который проверяет наличие всех таблиц и объектов на их существование, если они есть, удаляет их из системной

базы данных и создает заново.
2. SQL-скрипт, создающий хранимые процедуры для добавления, удаления и редактирования каждой из таблиц. Каждая такая хранимая процедура должна обеспечивать проверку на обеспечение целостности данных.
3. SQL-скрипт для наполнения БД тестовыми значениями. Для вставки значений в таблицу необходимо использовать ранее разработанные хранимые процедуры, поэтому запрещается использовать операторы INSERT.
4. SQL-скрипт, позволяющий создавать индексы для каждой из таблиц.
5. SQL-скрипт, производящий очистку всех таблиц с последующим их удалением.
6. SQL-скрипт, содержащий набор хранимых процедур (этот набор можно поместить в SQL-скрипт №2), обеспечивающих все необходимые функции обработки данных, предусмотренных вариантом задания, а именно:
   - Запрос «Исходные данные» - выдает все данные из всех таблиц, представляя их в удобной для восприятия форме при этом исключить повторение первичных ключевых полей из главных таблиц;
   - Запрос "Проживающие по заданному адресу", который выдает список проживающих на заданной улице;
   - Запрос "Алфавитный список", который выдает список фамилий в указанном буквенном диапазоне. Начальную и конечную буквы диапазона выбрать самостоятельно. Отсортировать список в алфавитном порядке.
   - Запрос "Список с условием" – согласно варианту задания (см. ниже).
   - Запрос "Фамилия, преобразованная к верхнему регистру", который выдает список всех фамилий, преобразованных к верхнему регистру.
   - Запрос "Фамилии и инициалы", который выдает список всех фамилий и инициалов. В запрос включить поля: Фамилия, Имя, Отчество (для контроля выполнения запроса).
   - Запрос с расчетами – согласно варианту задания (см. ниже).
   - Запрос с параметрами – согласно варианту задания (см. ниже).
   - Запрос с групповой операцией – согласно варианту задания (см. ниже).
   - Запрос с групповой операцией SUM – согласно варианту задания (см. ниже).
   - Запрос с групповой операцией COUNT – согласно варианту задания (см. ниже).

Изобразить в любой программе UML-диаграмму классов, отображающую все таблицы и связи между ними.

Пример варианта: (5)

База данных "Библиотека" должна хранить следующую информацию:

1. Фамилия, Имя, Отчество читателя.
2. Домашний адрес читателя.
3. Телефон читателя.
4. Дата рождения читателя.
5. Номер читательского билета.
6. Автор книги, которую взял читатель.
7. Название книги.
8. Дата выдачи книги.

9. Дата возврата книги.
10. Цена книги.

Запрос «Список с условием»: список книг, цена которых не превышает определенного значения. Значением цены книги задаться самостоятельно.

Запрос «Запрос с расчётами»: список, отсортированный по номеру билета, содержащий:

- Фамилия
- Имя
- Отчество
- Номер билета
- Название книги
- Цена книги
- Цена книги, увеличенная на 30%

Запрос «с параметрами»: Список читателей, пользовавшихся указанной книгой, и стоимость этой книги (параметр: название книги). Формат вывода стоимости: пользовательский, 2 знака после запятой.

Запрос «с групповой операцией»: вывести среднюю цену книг

Запрос с групповой операцией SUM: вывести суммарную стоимость книг, выданных каждому читателю.

Запрос с групповой операцией COUNT: вывести количество книг, выданных каждому читателю.

Критерии оценки:
Максимально можно получить 40 баллов.
За каждый скрипт можно получить от 0 до 5 баллов, в зависимости от правильности выполнения. Всего 6 * 5 = 30 баллов.
UML-диаграмма – от 0 до 5 баллов.
От 0 до 5 баллов можно получить за полноту и корректность отчёта и выступления по проделанной работе.