

МИНОБРНАУКИ РОССИИ  
ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ЭКОНОМИКИ И СЕРВИСА

**РАБОЧАЯ ПРОГРАММА  
УЧЕБНОЙ ДИСЦИПЛИНЫ**

*ОП.08 Основы проектирования баз данных*

программы подготовки специалистов среднего звена

*09.02.07 Информационные системы и программирование*

Форма обучения: *очная*

Владивосток 2021

Рабочая программа учебной дисциплины *ОП.08 Основы проектирования баз данных* разработана в соответствии с требованиями Федерального государственного образовательного стандарта среднего профессионального образования по специальности *09.02.07 Информационные системы и программирование*, утвержденного приказом Минобрнауки России от 09.12.2016 г., №1547, примерной образовательной программой.

Разработчик(и):

*Федоренко Евгения Владимировна, преподаватель высшей категории Академического колледжа ВГУЭС*

Рассмотрено и одобрено на заседании цикловой методической комиссии

Протокол № 10 от «12» мая 2021 г.

Председатель ЦМК \_\_\_\_\_ *А.Д. Гусакова*  
*подпись*

## СОДЕРЖАНИЕ

1. Общие сведения .....	4
2. Структура и содержание учебной дисциплины .....	5
3. Условия реализации программы дисциплины .....	1
4. Контроль результатов освоения учебной дисциплины .....	3

## 1. Общие сведения

### 1.1. Общая характеристика программы учебной дисциплины

По государственному образовательному стандарту среднего профессионального образования дисциплина «Основы проектирования баз данных» включена в профессиональный учебный цикл общепрофессиональных дисциплин (ОП.08)

### 1.2. Цель и планируемые результаты освоения дисциплины:

Код	Умения	Знания
ОК-1 ОК 2 ОК 4 ОК 5 ОК 9 ОК 10	<ul style="list-style-type: none"><li>– проектировать реляционную базу данных;</li><li>– использовать язык запросов для программного извлечения сведений из баз данных</li></ul>	<ul style="list-style-type: none"><li>– основы теории баз данных;</li><li>– модели данных;</li><li>– особенности реляционной модели и проектирование баз данных;</li><li>– изобразительные средства, используемые в ER- моделировании;</li><li>– основы реляционной алгебры;</li><li>– принципы проектирования баз данных;</li><li>– обеспечение непротиворечивости и целостности данных;</li><li>– средства проектирования структур баз данных;</li><li>– язык запросов SQL</li></ul>

## 2. Структура и содержание учебной дисциплины

### 2.1. Объем учебной дисциплины и виды учебной работы

Вид учебной работы	Объем часов
<b>Объем учебной дисциплины</b>	76
<b>Работа обучающихся во взаимодействии с преподавателем</b>	64
в том числе:	
теоретическое обучение	32
практические занятия	32
<b>Консультации</b>	2
<b>Самостоятельная работа:</b>	
– инфологическое проектирование по выбранной предметной области;	4
– даталогическая модель данных по выбранной предметной области;	
– подготовка к тестам, контрольным работам.	
<b>Промежуточная аттестация</b>	6
<b>Итоговая аттестация в форме:</b> экзамена – 2 семестр.	

## 2.2. Тематический план и содержание учебной дисциплины «ОП.08 Основы проектирования баз данных»

Наименование разделов и тем	Содержание учебного материала и формы организации деятельности обучающихся	Объем часов	Коды компетенций, формированию которых способствует элемент программы
1	2	3	5
<b>Тема 1. Теоретические основы баз данных</b>	<b>Содержание учебного материала</b>	<b>14</b>	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10, ПК 11.1-11.6
	<b>Содержание учебного материала:</b> основные понятия, историческая справка. Состав и архитектура банков данных. Модели данных. Предметная область. Проектирование информационно-логической модели базы данных. Даталогическое проектирование. Типы моделей данных: иерархическая, сетевая, реляционная. Централизованные и распределенные базы данных.	6	
	<b>Практическое занятие № 1</b> - описать предметную область. Разработать информационно-логическую (инфологическую) модель по предложенной предметной области.	2	
	<b>Практическое занятие №2</b> - рассмотреть и проанализировать примеры даталогических моделей.	2	
	<b>Самостоятельная работа обучающихся:</b> проработать и проанализировать конспекты занятий, учебную и специальную литературу.	1	
<b>Тема 2. Проектирование реляционной базы данных</b>	<b>Содержание учебного материала</b>	<b>23</b>	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10, ПК 11.1-11.6
	<b>Содержание учебного материала:</b> понятие процесса проектирования. Основные понятия реляционной базы данных. Этапы проектирования реляционной базы данных. Нормализация отношений. Нормальные формы. Примеры.	9	
	<b>Практическое занятие №3</b> – рассмотреть и проанализировать трехуровневую структуру разработки. Рассмотреть примеры инфологических моделей: диаграммы Бахмана, модель сущность-связь.	2	
<b>Практическое занятие №4</b> – тестирование (вопросы промежуточного тестирования). Разработать даталогическую модель, по представленной инфологической модели (практическая работа №1).	3		

Наименование разделов и тем	Содержание учебного материала и формы организации деятельности обучающихся	Объем часов	Коды компетенций, формированию которых способствует элемент программы
1	2	3	5
	<b>Практическое занятие №5</b> - разработать инструкции пользователя по работе с базой данных, описать сценарии диалога с пользователем. Проанализировать примеры.	2	
	<b>Практическое занятие №6</b> - разработать нормализацию отношений в предложенной базе данных. Проанализировать типы связей, нормальные формы.	2	
	<b>Самостоятельная работа обучающихся:</b> проработать и проанализировать конспекты учебной и специальной литературы, разобрать примеры.	1	
<b>Тема 3. Системы управления базами данных</b>	<b>Содержание учебного материала</b>	<b>31</b>	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10, ПК 11.1-11.6
	<b>Содержание учебного материала:</b> понятие системы управления базами данных (СУБД). Языковые средства СУБД. Классификация СУБД. Исторический обзор СУБД для персональных компьютеров. Режимы работы пользователя с СУБД. Обзор современных СУБД. Тенденции развития СУБД. Инструментальные средства для разработки СУБД. Примеры разработки СУБД с помощью MS Access, PostgreSQL, FoxPro.	9	
	<b>Практическое занятие №7</b> - создать таблицы реляционной базы данных в режиме Конструктор (практическая работа №2, раздел 1,3,4,5).	2	
	<b>Практическое занятие №8</b> – спроектировать формы для ввода и редактирования данных, подчиненные формы в режиме Мастера и Конструктора (практическая работа №2, разделы 2,4,5,7,8).	2	
	<b>Практическое занятие №9</b> - провести нормализацию отношений (практическая работа №2, раздел 6).	2	
	<b>Практическое занятие №10</b> – создать запросы (на выборку, параметрические, с условием, с использованием логических операторов, с повторяющимися записями, перекрестные, вычисляемые). Использовать макросы и модули для выбора данных (практическая работа №2, раздел 9)	3	

Наименование разделов и тем	Содержание учебного материала и формы организации деятельности обучающихся	Объем часов	Коды компетенций, формированию которых способствует элемент программы
1	2	3	5
	<b>Практическое занятие №11</b> – построить отчеты по выбранным данным (простые, с формулами и диаграммами, сводные) (практическая работа №2, раздел 10)	2	
	<b>Практическое занятие №12</b> - разработать пользовательскую форму с кнопками управления (использовать элементы управления). Использовать кнопки управления для перехода из одного уровня Меню в другой, оформить Меню. Разработать пользовательскую форму с помощью Диспетчера форм (практическая работа №2, разделы 11,12,13)	2	
	<b>Самостоятельная работа обучающихся:</b> самостоятельно разработать СУБД, с помощью Access, по выбранной предметной области (таблицы, нормализация, запросы, отчеты, пользовательские формы, макросы, модули) – ИДЗ задания 3 и 4.	1	
<b>Тема 4. Организация запросов SQL</b>	<b>Содержание учебного материала</b>	<b>18</b>	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ОК 10, ПК 11.1-11.6
	<b>Содержание учебного материала:</b> основные понятия языка SQL. Синтаксис операторов, типы данных. Создание, модификация и удаление таблиц. Операторы манипулирования данными. Организация запросов на выборку данных.	8	
	<b>Практическое занятие №13</b> – семинар по теоретическим вопросам. Демонстрация, разработанной самостоятельно, СУБД. Разработать запросы, с использованием языка SQL.	2	
	<b>Практическое занятие №14</b> - работа с переменными. Разработать программный файл для работы с табличными файлами. Заполнить массив из табличного файла. Заполнить табличный файл из массива.	2	
	<b>Практическое занятие №15</b> - работа с командами ввода-вывода. Использовать функций для работы с массивами. Итоговое тестирование.	2	
	<b>Самостоятельная работа обучающихся:</b> в созданной, самостоятельно, СУБД добавить запросы манипулирования данными на языке SQL, кнопки управления в пользовательскую форму (работа с массивами и переменными). Написать отчет по созданной СУБД.	1	



Наименование разделов и тем	Содержание учебного материала и формы организации деятельности обучающихся	Объем часов	Коды компетенций, формированию которых способствует элемент программы
1	2	3	5
<b>ИТОГО часов</b>		<b>76</b>	
<b>Теоретические занятия</b>		<b>32</b>	
<b>Практические занятия</b>		<b>32</b>	
<b>Самостоятельная работа</b>		<b>4</b>	
<b>Консультации</b>		<b>2</b>	
<b>Промежуточная аттестация</b>		<b>6</b>	

### 3. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

**3.1. Для реализации программы учебной дисциплины «ОП.08 Основы проектирования баз данных» образовательной организации, предусмотрено наличие следующих специальных помещений:**

Учебная аудитория для проведения учебных занятий (уроки, лекции, практические занятия, лабораторные занятия, семинарские занятия, курсовое проектирование), групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации.

Лаборатория организации и принципов построения информационных систем (ауд 1408)

Автоматизированные рабочие места обучающихся (сетевые терминалы) - 14 шт., автоматизированное рабочее место преподавателя с облачным монитором - 1 шт., сервер, шкаф 1 шт., мультимедийный проектор, экран, маркерная доска 1шт.

Помещение для самостоятельной работы обучающихся (ауд 1406)

Рабочие места на базе вычислительной техники с установленным офисным пакетом с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду организации. а также комплектом оборудования для печати: персональные компьютеры; посадочных мест – 30 шт. Стол преподавателя - 1 шт; Стул преподавателя - 1 шт; Доска маркерная - 1 шт; Мультимедийный проектор с экраном

Программное обеспечение общего и профессионального назначения, в том числе включающее в себя следующее ПО: EclipseIDEforJavaEEDevelopers, .NETFrameworkJDK 8, MicrosoftSQLServerExpressEdition, MicrosoftVisioProfessional, MicrosoftVisualStudio, MySQLInstallerforWindows, NetBeans, SQLServerManagementStudio, MicrosoftSQLServer-JavaConnector, AndroidStudio, IntelliJIDEA.

– .

### 3.2. Информационное обеспечение реализации программы

*Для реализации программы учебной дисциплины библиотечный фонд образовательной организации укомплектован печатными и электронными изданиями.*

*Основная литература:*

1. Дадян, Э. Г. Современные базы данных. Основы. Часть 1: Учебное пособие / Дадян Э.Г. - Москва: НИЦ ИНФРА-М, 2017. - 88 с. ISBN 978-5-16-106526-6 (online). - Текст: электронный. - URL: <https://new.znaniium.com/catalog/product/959289>
2. Кумскова, И.А. Базы данных: учебник / Кумскова И.А. — Москва: КноРус, 2016. — 399 с. — (СПО). — ISBN 978-5-406-04521-3. — URL: <https://book.ru/book/919609>
3. Советов, Б. Я. Базы данных: учебник для среднего профессионального образования / Б. Я. Советов, В. В. Цехановский, В. Д. Чертовской. — 2-е изд. — Москва: Издательство Юрайт, 2017. — 463 с. — (Профессиональное образование). — ISBN 978-5-534-01653-6. — Текст: электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/400171>

*Дополнительная литература:*

1. Администрирование баз данных ORACLE в операционной системе UNIX. - М.: СПб: ЦКТиП Газпром, 2015. - 300 с.
2. Базы данных / С.В. Глушаков, Д.В. Ломотько. - М.: Харьков: Фолио, 2017. - 504 с.
3. Базы данных. Проектирование, реализация и сопровождение. Теория и практика / Т. Коннолли. - М.: Вильямс И.Д., 2017. - 1440 с.
4. Основы систем баз данных / Дж. Ульман. - М.: Финансы и статистика, 2017. - 292 с.

*Интернет-ресурсы:*

1. <http://ibase.ru> - независимое экспертное бюро для оказания консультаций разработчикам СУБД.
2. <http://www.sql.ru> – сайт для разработчиков БД, СУБД.
3. <https://support.microsoft.com/ru-ru>- ресурсы для изучения MS Access (служба поддержки Майкрософт)
4. <https://aws.amazon.com> - специальная подборка ресурсов, которые помогут научиться создавать решения для работы с базами данных на AWS.
5. <https://accesshelp.ru> – пошаговое создание СУБД в MS Access.
6. <http://www.oszone.net> - компьютерный информационный портал:

#### 4. Контроль результатов освоения учебной дисциплины «ОП.08 Основы проектирования баз данных»

Контроль и оценка результатов освоения дисциплины осуществляется преподавателем в процессе проведения практических занятий, тестирования, а также выполнения обучающимися типовых индивидуальных заданий.

<i>Результаты обучения</i>	<i>Критерии оценки</i>	<i>Формы и методы оценки</i>
<p><i>Перечень умений, осваиваемых в рамках дисциплины:</i></p> <ul style="list-style-type: none"> <li>- проектировать реляционную базу данных;</li> <li>- использовать язык запросов для программного извлечения сведений из баз данных</li> </ul>	<p>«Отлично» - теоретическое содержание курса освоено полностью, без пробелов, умения сформированы, все предусмотренные программой учебные задания выполнены, качество их выполнения оценено высоко.</p>	<p>Формы и методы контроля и оценки</p> <p>Тестирование на знание терминологии по теме;</p> <p>Самостоятельная работа по разработке СУБД;</p>
<p><i>Перечень знаний, осваиваемых в рамках дисциплины:</i></p> <ul style="list-style-type: none"> <li>- основы теории баз данных;</li> <li>- модели данных;</li> <li>- особенности реляционной модели и проектирование баз данных;</li> <li>- изобразительные средства, используемые в ER- моделировании;</li> <li>- основы реляционной алгебры;</li> <li>- принципы проектирования баз данных;</li> <li>- обеспечение непротиворечивости и целостности данных;</li> <li>- средства проектирования структур баз данных;</li> <li>- язык запросов SQL</li> </ul>	<p>«Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все предусмотренные программой учебные задания выполнены, некоторые виды заданий выполнены с ошибками.</p> <p>«Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки.</p> <p>«Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, выполненные учебные задания содержат грубые ошибки.</p>	<p>Семинар</p> <p>Выполнение проекта;</p> <p>Защита проекта;</p> <p>Наблюдение за выполнением практического задания. (деятельностью студента)</p> <p>Оценка выполнения практического задания(работы)</p>

МИНОБРНАУКИ РОССИИ  
ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ЭКОНОМИКИ И СЕРВИСА

**КОНТРОЛЬНО-ОЦЕНОЧНЫЕ СРЕДСТВА**  
для проведения текущего контроля и промежуточной аттестации  
по учебной дисциплине

*ОП.08 Основы проектирования баз данных*

программы подготовки специалистов среднего звена

*09.02.07 Информационные системы и программирование*

Форма обучения: *очная*

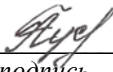
Контрольно-оценочные средства для проведения текущего контроля и промежуточной аттестации по учебной дисциплине *ОП.08 Основы проектирования баз данных* разработаны в соответствии с требованиями ФГОС СПО по специальности *09.02.07 Информационные системы и программирование*, утвержденного приказом Минобрнауки РФ от *09.12.2016 г., №1547*, примерной образовательной программой, рабочей программой учебной дисциплины.

Разработчик(и):

*Федоренко Евгения Владимировна, преподаватель высшей категории Академического колледжа ВГУЭС*

Рассмотрено и одобрено на заседании цикловой методической комиссии

Протокол № 10 от «12» мая 2021 г.

Председатель ЦМК  *А.Д. Гусакова*  
подпись

## 1 Общие сведения

Контрольно-оценочные средства (далее – КОС) предназначены для контроля и оценки образовательных достижений обучающихся, освоивших программу учебной дисциплины ОП.08 Основы проектирования баз данных.

КОС разработаны на основании:

- основной образовательной программы СПО по специальности 09.02.07 Информационные системы и программирование;
- рабочей программы учебной дисциплины ОП.08 Основы проектирования баз данных.

Формой итоговой аттестации является экзамен.

Код ОК, ПК	Код результата обучения	Наименование
ОК 1 ОК 2 ОК 4 ОК 5 ОК 9 ОК 10	У1	Умение проектировать реляционную базу данных.
	У2	Умение использовать язык запросов для программного извлечения сведений из баз данных.
	31	Знание основ теории баз данных.
	32	Знание моделей данных.
	33	Знание особенностей реляционной модели и проектирования баз данных.
	34	Знание изобразительных средств, используемых в ER-моделировании.
	35	Знание основ реляционной алгебры.
	36	Знание принципов проектирования баз данных.
	37	Знание как обеспечить непротиворечивость и целостность данных.
	38	Знание средств проектирования структур баз данных.
	39	Знание языка запросов SQL.

## 2 Распределение типов контрольных заданий по элементам знаний и умений, контролируемых в процессе изучения

Код результата обучения	Содержание учебного материала (темы)	Вид оценочного средства	
		Текущий контроль	Промежуточная аттестация
У1	Разработка СУБД с помощью MS Access: создание таблиц, использование форм для ввода и редактирования данных, проектирование многотабличных баз данных, использование подчиненных форм, создание пользовательских форм, выбор данных с помощью запросов, построение отчетов, создание кнопок управления, создание кнопочных форм. Использование кнопок управления для перехода из одного уровня Меню в другой, оформление Меню.	Защита практической работы, ИДЗ	

Код результата обучения	Содержание учебного материала (темы)	Вид оценочного средства	
		Текущий контроль	Промежуточная аттестация
У2	Работа с переменными. Написание программного файла и работа с табличными файлами. Заполнение массива из табличного файла. Заполнение табличного файла из массива. Работа с командами ввода-вывода. Использование функций для работы с массивами.	Защита практической работы	
31	Основные понятия, историческая справка. Состав и архитектура банков данных. Предметная область. Типы моделей данных: иерархическая, сетевая, реляционная. Централизованные и распределенные базы данных.		
32	Классификация моделей данных.		
33, 36, 37	Понятие процесса проектирования. Этапы проектирования реляционной базы данных. Нормализация отношений.	Промежуточное тестирование	
34, 38	Знание средств проектирования структур баз данных.	Семинар	
35, 39	Основные понятия языка SQL. Синтаксис операторов, типы данных. Создание, модификация и удаление таблиц. Операторы манипулирования данными. Организация запросов на выборку данных.		Итоговое тестирование

### 3 Структура банка контрольных заданий для текущего контроля и промежуточной аттестации

Тип контрольного задания	Количество контрольных заданий (вариантов)	Общее время выполнения обучающимся контрольных заданий
Текущий контроль		
Практическая работа №1 Технология работы с базами данных.	24 (1)	90 мин
Практическая работа №2 Проектирование многотабличных баз данных.	13 разделов (1)	360 мин
Тестирование	14 (1)	30 мин
Проектирование и реализация СУБД по выбранной предметной области (индивидуальная домашняя работа)	12 тем	
Семинар	вопросов 31	60 мин
Промежуточная аттестация		
Итоговое тестирование	22 (1)	60 мин



## Практическая работа №1

### Технология работы с базами данных.

1. Создайте файл базы данных (БД) с именем **Сведения о сотрудниках**.
2. В данной базе данных в *Режиме таблицы* создайте структуру таблицы «Штат» (табл. 1), последовательно заменяя условное имя поля *Код* на *Код должности*, *Должность* и *Оклад* и определяя тип полей.
3. В *режиме Конструктора* создайте структуру таблиц «Сотрудники» (табл. 2) и «Список» (табл. 3). Для этого задайте имена, типы и свойства полей, а также определите ключевое поле.

Таблица 1

Структура записи таблицы «Штат»

Код должности (текстовый)	Должность (текстовый)	Оклад (числовой)

Таблица 2

Структура записи таблицы «Сотрудники»

Ключ	Имя поля	Тип данных	Свойства		
			Размер	Маска ввода	Обязательное поле
+	Код сотрудника	Текстовый	5		да
	ФИО	Текстовый	30		
	Пол	Текстовый	5		
	Образование	Текстовый	30		
	Дата рождения	Дата/время			да
	Адрес	Текстовый	50		
	Телефон	Текстовый	15	999\ -99\ -99	
	Иждивенцы	Текстовый	5		
	Льготы	Логический			

Таблица 3

Структура записи таблицы «Список»

Ключ	Имя поля	Тип данных	Свойства
			Размер
+	Код сотрудника	Текстовый	5
	Код должности	Текстовый	3
	Дата принятия на работу	Дата/время	

4. Откройте таблицу «Штат» в *режиме Конструктора* и:
  - задайте размер полей для таблицы (Код должности – 3; Должность – 40);
  - определите ключевое поле (Код должности).
5. Заполните таблицы «Список», «Сотрудники» и «Штат» данными в соответствии с табл. 4-6.

Таблица 4

Данные для таблицы «Список»

<b>Код сотрудника</b>	<b>Код должности</b>	<b>Дата принятия на работу</b>
001	01	10.01.2005
002	02	15.01.2005
003	09	20.02.2005
004	09	01.12.2007
005	09	15.03.2007
006	03	12.04.2005
007	05	01.05.2006
008	06	01.08.2006
009	10	20.02.2005
010	10	01.03.2005
011	10	10.12.2005
012	07	15.01.2005
013	03	01.03.2005

Таблица 5

## Данные для таблицы «Сотрудники»

Код сотрудника	ФИО	Пол	Образование	Дата рождения	Адрес	Телефон	Иждивенцы	Льготы
001	Орлов И.Б.	муж	высшее	01.03.1970	Владивосток, К.Маркса, 26/1, 30	346-58-10	2	Да
002	Козлов А.А.	муж	высшее	15.02.1965	Владивосток, Красный пр, 75,10	223-15-20	1	Нет
003	Петрова Н.В.	жен	высшее	03.10.1980	НСО, Бердск, Ленина, 5, 20	46-15-20		Нет
004	Иванова Е.С.	жен	высшее	10.11.1978	Владивосток, Д.Ковальчук, 125, 200	226-65-33	2	Да
005	Иванов С.Б.	муж	высшее	20.02.1975	Владивосток, Д.Ковальчук, 125, 200	226-65-33	2	Да
006	Цой И.Б.	муж	высшее	13.11.1971	Владивосток, Титова, 20, 2	314-50-20	1	Нет
007	Борисов В.П.	муж	высшее	11.09.1969	НСО, Искитим, Народная, 15, 10			Нет
008	Антонова А.А.	жен	спец. среднее	05.04.1980	Владивосток, Новогодняя, 9, 35			Нет
009	Ким М.Б.	жен	спец. среднее	08.03.1990	НСО, Бердск, Маркса, 10, 3			Да
010	Аксенова Т.В.	жен	спец. среднее	22.07.1989	Владивосток, Ленина, 15, 46	221-98-16		Нет
011	Егорова К.В.	жен	спец. среднее	12.02.1975	Владивосток, Космическая, 10, 28	346-53-27	1	Нет
012	Потапов В.В.	муж	высшее	23.12.1970	Владивосток, Плахотного, 26/1, 17	341-99-48	3	Да
013	Дроздова Е.С.	муж	высшее	05.10.1965	Владивосток, Зорге, 106, 45	342-15-15		Нет

Таблица 5

## Данные для таблицы «Штат»

Код должности	Должность	Оклад
01	генеральный. директор	50000
02	исполнительный директор	40000
03	юрист	35000
04	инженер	30000
05	спец. по рекламе	25000
06	спец. по связи с общественностью	25000
07	бухгалтер	30000
08	главный бухгалтер	35000
09	программист	35000
10	оператор	15000
11	уборщица	10000
12	грузчик	10000
13	сетевой администратор	45000

6. Установите связь между таблицами с соблюдением условия обеспечения целостности и каскадных изменений. В результате должна получиться схема данных. Обратите внимание на то, что между таблицами «Сотрудники» и «Список» устанавливается отношение «один-к-одному», а между таблицами «Штат» и «Список» – отношение «один-ко-многим».
7. Для полей *Код должности* и *Код сотрудника* из таблицы «Список» создайте столбцы подстановок на основе данных из таблиц «Штат» и «Сотрудники», соответственно. В результате при раскрытии поля *Код сотрудника* в таблице «Список» можно будет просмотреть соответствие кодов сотрудников и ФИО, а при раскрытии поля *Код должности* соответствие между кодами должностей и их названиями.

**Работа с Формами**

8. Создайте на основе таблицы «Сотрудники» форму с помощью инструмента *Форма* и сохраните её под именем «Сотрудники\_форма».
9. С помощью созданной формы введите свои личные данные, а также еще 2-3 записи (можно о Ваших друзьях или родственниках).
10. На основе таблицы «Список» постройте *Разделенную форму* и сохраните ее под именем «Список\_разделенная».
11. С помощью Мастера форм постройте форму на основе всех таблиц, включите в нее следующие поля: *ФИО; Образование; Телефон; Дата принятия на работу; Должность и Оклад*.
12. Разместите эти данные в наиболее удобном для просмотра и работы виде.
13. Создайте Форму с вкладками *Основная* и *Дополнительная* и разместите на вкладке *Основная* следующие поля – *ФИО, Образование, Должность и Оклад*, а на вкладке *Дополнительная* – поля *ФИО, Дата принятия на работу и Телефон*.

**Работа с запросами**

14. Выведите список сотрудников, имеющих льготы, и сохраните его под именем *Запрос\_льготы*.

15. Сформируйте запрос, содержащий информацию о сотрудниках с высшим образованием, занимающим должность программиста (*Запрос\_программисты*).
16. Сформируйте запрос, содержащий информацию об адресе, телефоне и ФИО генерального директора фирмы (*Запрос\_директор*).
17. Сформируйте запрос, содержащий информацию о сотрудниках, принятых на работу в 2006-2007 гг., имеющих иждивенцев (*Запрос\_иждивенцы*).
18. Сформируйте запрос, содержащий информацию о сотрудниках, не проживающих в городе Владивостоке (*Запрос\_иностранцы*).
19. Определите, есть ли в фирме вакансии в штатном расписании (используйте пункт *Записи без подчиненных* в *Мастере запросов*). Запрос сохраните под именем *Запрос\_вакансии*.
20. Используя итоговый запрос, выведите средний оклад сотрудников с высшим образованием (*Запрос\_группировка*).
21. Используя запрос на создание таблицы, сформируйте новую таблицу, включив в нее поля *ФИО, Образование, Дата принятия на работу, Телефон, Должность* и *Оклад* (*Новая\_объединенная*).
22. Сформируйте запрос, содержащий информацию об окладах сотрудников, и всем сотрудникам, имеющим оклад меньше 20000, начислите премию в размере 30% от оклада. Для этого следует в запрос включить вычисляемое поле *Премия* и использовать *Построитель выражений* (*Запрос\_премия*).

#### **Работа с отчетами**

23. На основе таблицы «Сотрудники» постройте отчет с помощью инструмента *Отчет*, а затем доработайте его в *режиме Конструктора* таким образом, чтобы все поля одновременно отображались на экране.
24. Используя *Мастер отчетов*, постройте отчет. Для построения используйте группировку по полю *Образование* и итоги (среднее значение) по полю *Оклад*. Для вывода величины среднего значения по группе задайте в окне Свойств (F4) *Формат поля* – *Фиксированный* и *Число десятичных знаков* – 2. Сохраните отчет под именем *Сотрудники\_группировка*.

## **Практическая работа № 2**

### **Проектирование многотабличных баз данных.**

#### Раздел 1.

#### **Задание 1:**

Создать в своей папке пустую базу данных. Для этого выполнить следующие действия:

1. Запустить программу MS Access
2. Создать новую базу данных (Файл/Создать базу данных)
3. Из раскрывающегося списка ПАПКА выбрать свою папку.
4. В поле Имя файла ввести имя базы данных “(фамилия)\_(группа)\_БАЗА ДАННЫХ”, нажать кнопку Создать
5. Выбрать вкладку Таблица и нажать кнопку Создать.
6. В окне Новая таблица выбрать пункт Конструктор.

7. Создать структуру таблицы, для этого надо заполнить столбцы «имя поля» и «тип данных» следующим образом:

Имя поля	Тип данных
Фамилия	Текстовый
Имя	Текстовый
Отчество	Текстовый
Дата рождения	Дата/время
Адрес	Текстовый
Телефон	Текстовый
Группа	Числовой

8. Завершив ввод структуры таблицы, сохраните ее с именем **Анкета**, выполнив команду **Файл/Сохранить**. В ответ на вопрос **Создать ключевое поле сейчас** нажмите кнопку **Да**.

В вашей таблице появится поле с именем “код”. Это поле называется **первичным ключом** и однозначно идентифицирует каждую запись в таблице. **Тип данных** у этого поля обозначен как **Счетчик**. Это означает, что каждый раз при создании новой записи значение счетчика будет увеличиваться на **1**. В результате у каждой записи формируется свой номер, который и является первичным ключом.

Поля обладают **свойствами**. От свойств полей зависит, какие типы данных можно вносить в поле, а какие нет, а также то, что можно делать с данными, содержащимися в поле. Например, **маска ввода** для поля типа “дата” облегчает ввод дат, позволяет Вам не беспокоиться о разделителях дня, месяца, года.

### Задание 2:

1. Определить свойства полей на вкладке **Общие** следующим образом:

Поле	Свойства полей						
	Размер	Формат поля	Маска ввода	Подпись	Значение по умолчанию	Обязательное поле	Индексированное поле
Фамилия	30			Фамилия		да	Да (совпадения допускаются)
Имя	12			Имя		да	нет
Отчество	30			Отчество		нет	нет
Дата рождения		Краткий формат даты	Краткий формат	Дата рождения		да	нет
Адрес	50			Адрес		да	Нет
Телефон	6					нет	нет
Группа	Целое	основной		Группа	Номер вашей группы	да	Да (совпадения допускаются)

**Задание 3:**

1. Открыть таблицу **Вид/Режим таблицы**.
2. Заполнить строки.

Будьте внимательны при заполнении столбца “**дата рождения**” - *разделители* проставляются автоматически.

**Задание 4:**

1. На основе созданной таблицы создайте список фамилий студентов Вашей группы, для этого в режиме конструктора для поля Фамилия откройте вкладку Подстановка и установите следующие свойства для поля Фамилия.

<b>Тип элемента управления</b>	Поле со списком
<b>Тип источника строк</b>	Таблица/Запрос
<b>Источник строк</b>	<ol style="list-style-type: none"> <li>1. Щелкните мышкой,</li> <li>2. Нажмите кнопку <input type="button" value="..."/>.</li> <li>3. В появившемся окне <b>Построителя запросов</b>: <ul style="list-style-type: none"> <li>• добавить таблицу <b>Анкета</b>;</li> <li>• закрыть окно <b>Добавление таблицы</b>;</li> <li>• перетащить с помощью мыши поле <b>Фамилия</b> в поле бланка запроса;</li> <li>• в поле <b>Вывод на экран</b> должна стоять галочка.</li> </ul> </li> </ol>
<b>Ограничиться списком</b>	Нет

2. В результате выполнения **Задания 4** в таблице **Анкета** в поле **Фамилия** при щелчке мышью должна появиться кнопка, обеспечивающая выбор из списка. Проверьте это.

**3. ФАЙЛ/СОХРАНИТЬ!****Раздел 2.****Использование форм для ввода и редактирования данных****Задание 1:**

1. Создать форму для заполнения таблицы **Анкета**.
  - Откройте окно базы данных.
  - Перейдите на вкладку **Формы**.
  - Нажмите кнопку **Создать**.
  - Выбрать **Автоформа: ленточная**, в качестве **источника данных** выберите таблицу **Анкета**.

**Задание 2:**

Разберитесь с кнопками перемещения по записям

<input type="button" value="↑"/>	Переход на одну запись вперед	<input type="button" value="→"/>	Переход на последнюю запись
<input type="button" value="↓"/>	Переход на одну запись	<input type="button" value="←"/>	Переход на чистую страницу, где можно вве-

	назад		сти новую запись
??	Переход на первую запись		

**Задание 3:**

1. Создайте таблицу для ввода групп, имеющихся на вашем курсе:

- выбрать вкладку **Таблицы**;
- нажать кнопку **Создать**;
- в окне **Новая таблица** выбрать пункт **Конструктор**;
- создать структуру таблицы, для этого надо заполнить столбцы «**имя поля**» и «**тип данных**» следующим образом:

Имя поля	Тип данных
Группа	Текстовый
Фамилия	Текстовый
Имя	Текстовый
Отчество	Текстовый

- в ответ на вопрос **Создать ключевое поле сейчас** нажмите кнопку **Нет**;
- дайте имя таблицы **Группы**.

**Задание 4:**

1. Создать **форму** для заполнения таблицы **Группы**, заполнить эту форму, выполнив следующие действия:

- выбрать вкладку **Формы**;
- нажать кнопку **Создать**;
- выбрать **Автоформу: в столбец**, в качестве **источника данных** выбрать таблицу **Группы**;
- введите данные в форму **Группы**.

**Задание 5:**

Обратите внимание на то, что если вы в поле **фамилии** введете новую фамилию, то она не появится в раскрывающемся списке фамилий. Убедитесь в этом!

**Задание 6:**

Создать самозаполняющийся список фамилий, чтобы при вводе новой фамилии эта фамилия автоматически попадала в список и впоследствии эту фамилию можно было бы выбрать из списка, а не набирать заново. Для этого выполните следующие действия:

1. Создайте **форму**, основанную на таблице **Анкета** с именем

**Форма\_для\_добавления\_записей**. Эта форма будет использоваться для добавления в список новых пунктов.

2. Создайте группу макросов. Будьте очень внимательны при выполнении этого пункта!!!

- выбрать вкладку **Макросы**, нажать кнопку **Создать**;
- ввести макрокоманду **Открыть форму**;
  - ✓ *Имя формы: Форма\_для\_добавления\_записей;*
  - ✓ *Режим: Форма;*



- ✓ *Режим данных: Добавление;*
  - ✓ *Режим окна: Обычное.*
  - закрыть окно и сохранить макрос с именем **Добавление**;
  - создать следующий макрос: ввести макрокоманду **«Выполнить команду»**
    - ✓ *Команда «Сохранить запись (Save Record)»*
  - ввести макрокоманду **Выделить объект**:
    - ✓ *Тип объекта: Форма;*
    - ✓ *Имя объекта: Анкета;*
    - ✓ *В окне базы данных: Нет.*
  - ввести макрокоманду **«Обновление»**
    - ✓ *Имя элемента: Фамилия*
  - закрыть окно и сохранить макрос с именем **Закрытие**.
3. Переключитесь в форму **Форма\_для\_добавления\_записей** в режиме конструктора.
  4. Выбрать режим **Вид/Свойства**.
  5. Открыть вкладку **События**.
  6. Добавить макрос **Закрытие** в свойство **Закрытие формы**.
  7. Закрыть форму.
  8. Откройте в режиме конструктора форму **Анкета**.
  9. Щелкните *правой мышкой* по полю **фамилия** выберете пункт **свойства**:
    - открыть вкладку **События**;
    - указать имя макроса **Добавление** в поле ввода свойства **Двойное нажатие кнопки**.

#### **Задание 7:**

1. Проверьте работу созданных Вами макросов по созданию самозаполняющегося списка:
  - переключитесь в режим формы **Анкета**;
  - дважды щелкните мышкой по полю **фамилия**, введите фамилию **Клинтон**;
  - заполните остальные **обязательные** поля для записи с фамилией **Клинтон**;
  - закройте форму. Вы опять в главной форме **Анкета**;
  - попробуйте раскрыть список фамилий.

Если Вы правильно выполнили **Задание 6**, то в списке Вы обнаружите фамилию Клинтон.

2. Закройте форму, сохранив внесенные изменения. **ФАЙЛ/СОХРАНИТЬ!**

### **Раздел 3.**

#### **Проектирование многотабличных баз данных**

**Задание 1:****1. Создать таблицу Преподаватели. Используйте Мастер таблиц.**

- Выбрать для создания таблицы образец **Сотрудники** и выбрать из предложенного перечня полей следующие:
  - ✓ Фамилия
  - ✓ Имя
  - ✓ Отчество
  - ✓ Должность
  - ✓ Название отдела (переименовать, как “кафедра”)
- Назвать созданную таблицу **Преподаватели**. Связи **не определять**, т.е. нажать кнопку Далее.
- Выбрать режим **Непосредственный ввод данных в таблицу**.
- В режиме конструктора определить тип данных для каждого поля:

Имя поля	Тип данных
Фамилия	Текстовый
Имя	Текстовый
Отчество	Текстовый
Должность	Текстовый
Кафедра	Текстовый

- Определить свойства полей на вкладке **Общие** в режиме конструктора следующим образом:

Поле	Свойства полей						
	Размер	Формат поля	Маска ввода	Подпись	Значение по умолчанию	Обязательное поле	Индексированное поле
Фамилия	30			Фамилия		да	Да (совпадения допускаются)
Имя	12			Имя		да	нет
Отчество	30			Отчество		Да	нет
Должность	20			Должность		нет	нет
Кафедра	50			Кафедра		Да	нет

- Для поля **кафедра** изменить элемент управления следующим образом:
  - ✓ Откройте вкладку **Подстановка** и установите следующие свойства для поля **кафедра**:

<b>Тип элемента управления</b>	Поле со списком
<b>Тип источника строк</b>	Список значений.
<b>Источник строк</b>	Введите список кафедр, используя в качестве разделителя точку с запятой “;”

Ограничиться списком

Да

**Задание 2:**1. Создайте форму для заполнения таблицы **Преподаватели**:

- выбрать вкладку **Формы**, и создать форму, используя **Мастер форм**, определив в качестве источника таблицу **Преподаватели**;
- отметить мышкой все поля для создания формы;
- выбрать вид формы **ленточная**;
- стиль выбрать на свое усмотрение;
- имя формы **Преподаватели**;
- дальнейшие действия **Открыть форму для просмотра**.

2. Заполните форму. Обратите внимание на то, столбец **кафедра** заполняется путем подстановки из сформированного Вами списка. **ФАЙЛ/СОХРАНИТЬ!****Раздел 4.****Проектирование многотабличных баз данных****Задание 1:**1. Создать таблицу **Дисциплины** в режиме конструктора с параметрами:

Имя поля	Тип данных
Дисциплина	Текстовый

2. Определить свойства полей таблицы **Дисциплины** на вкладке **Общие** следующим образом:

Поле	Свойства полей			
	Размер	Подпись	Обязательное поле	Индексированное поле
Дисциплина	20	Дисциплина	Да	Да (совпадения не допускаются)

3. Создайте **форму** для заполнения таблицы **Дисциплины**. Для этого:

- перейдите на вкладку **формы**;
- нажмите кнопку **Создать**;
- выбрать **Автоформа: ленточная**, в качестве источника данных выберете таблицу **Дисциплины**.

4. Введите названия дисциплин, которые читаются на третьем курсе в **форму**.**Задание 2:**1. Создайте таблицу **Экзамен** в режиме конструктора таблиц с параметрами:

Имя поля	Тип данных
Дисциплина	Текстовый
Группа	Числовой
Студент	Текстовый
Результат контроля	Текстовый

2. Определить **свойства** полей на вкладке **Общие** следующим образом:

Поле	Свойства полей				
	Размер	Формат поля	Подпись	Обязательное поле	Индексированное поле
Дисциплина	15		Дисциплина	Да	Да (совпадения допускаются)
Группа	3		Группа	Да	Да (совпадения допускаются)
Студент	30		Студент	Да	Да (совпадения допускаются)
Результат контроля	10		Результат контроля	Да	Да (совпадения допускаются)

Так как у Вас в базе уже **имеется таблица с перечнем групп**, таблица с **перечнем дисциплин**, а также таблица **Анкета с перечнем студентов** этим надо воспользоваться для автоматического заполнения полей **группа**, **дисциплина** и **Студент** в создаваемой Вами таблице **Экзамен**. Это можно сделать с помощью элемента **управления полем**.

### Задание 3:

1. Для поля **группа** изменить элемент управления:

- Откройте вкладку **Подстановка** и установите следующие свойства для поля **группа**:

<b>Тип элемента управления</b>	Поле со списком
<b>Тип источника строк</b>	Таблица/Запрос.
<b>Источник строк</b>	<ol style="list-style-type: none"> <li>Щелкните мышкой,</li> <li>Нажмите кнопку <input type="button" value="..."/>.</li> <li>В появившемся окне “построителя запросов”: <ul style="list-style-type: none"> <li>добавить таблицу <b>Группы</b>;</li> <li>закрыть окно добавление таблицы;</li> <li>перетащить с помощью мыши поле <b>Группа</b> в поле бланка запроса.</li> </ul> <p><i>в поле “вывод на экран” должна стоять галочка</i></p> </li> </ol>
<b>Ограничиться списком</b>	Да

2. Аналогично изменить элемент управления для полей **дисциплина** и **студент**, используя в качестве строк соответственно таблицы **Дисциплины** и **Анкета**.

3. Для поля **результат контроля** изменить элемент управления следующим образом:

- откройте вкладку **Подстановка** и установите следующие свойства для поля **результат контроля**:

<b>Тип элемента управления</b>	Поле со списком
<b>Тип источника строк</b>	Список значений.
<b>Источник строк</b>	Введите список оценок, используя в качестве разделителя <b>точку с запятой “ ; ”</b> (отлично; хорошо; удовлетвори-

	тельно; неудовл)
<b>Ограничиться списком</b>	Да

**Задание 4:**

1. Создайте **форму** для заполнения таблицы **Экзамен**:

- перейдите на вкладку **формы**;
- нажмите кнопку **Создать**;
- выбрать **Автоформа: ленточная**, в качестве **источника данных** выберете таблицу **Экзамен**.

**Задание 5:**

Пользуясь созданной в **Задании 4** формой **Экзамен** введите данные по двум дисциплинам на **Ваш выбор** по всем студентам своей группы в таблицу **Экзамен**. Оценки проставлять произвольно (*желательно разнообразные*).

**Раздел 5.****Проектирование многотабличных баз данных****Задание 1:**

1. Создайте таблицу **Зачет** в режиме конструктора таблиц с параметрами:

Имя поля	Тип данных
Дисциплина	<b>Текстовый</b>
Группа	<b>Числовой</b>
Студент	<b>Текстовый</b>
Результат	<b>Логический</b>

2. Определить свойства полей на вкладке **Общие** следующим образом:

Поле	Свойства полей				
	Размер	Формат поля	Подпись	Обязательное поле	Индексированное поле
Дисциплина	15		Дисциплина	Да	Да (совпадения допускаются)
Группа	3		Группа	Да	Да (совпадения допускаются)
Студент	30		Студент	Да	Да (совпадения допускаются)
Результат		Да/Нет	Результат	Да	Да (совпадения допускаются)

Так как у Вас в базе уже имеется таблица с **перечнем групп**, таблица с **перечнем дисциплин**, а также таблица **Анкета** с **перечнем студентов** этим надо воспользоваться для автоматического заполнения полей **группа**, **дисциплина** и **Студент** в создаваемой Вами таблице **Зачет**.

Это можно сделать с помощью элемента **управления полем**

Для поля **группа** изменить элемент управления:

- Откройте вкладку **Подстановка** и установите следующие свойства для поля **группа**:

<b>Тип элемента</b>	Поле со списком
---------------------	-----------------

<b>управления</b>	
<b>Тип источника строк</b>	Таблица/запрос.
<b>Источник строк</b>	<ol style="list-style-type: none"> <li>1. щелкните мышкой;</li> <li>2. нажмите кнопку <input type="button" value="..."/></li> <li>3. в появившемся окне <b>построителя запросов</b>: <ul style="list-style-type: none"> <li>• добавить таблицу <b>Группы</b>;</li> <li>• закрыть окно добавление таблицы;</li> <li>• перетащить с помощью мыши поле <b>группа</b> в поле бланка запроса;</li> <li>• в поле <b>вывод на экран</b> должна стоять галочка.</li> </ul> </li> </ol>
<b>Ограничиться списком</b>	Да

Аналогично изменить элемент управления для полей **дисциплина** и **студент**, используя в качестве строк соответственно таблицы **Дисциплины** и **Анкета**.

### Задание 2:

1. Создайте форму для заполнения таблицы **Зачет**:

- перейдите на вкладку **формы**;
- нажмите кнопку **Создать**;
- выбрать **Автоформа: ленточная**, в качестве **источника данных** выберете таблицу **Зачет**.

2. Пользуясь созданной формой **Зачет** введите данные по всем студентам своей группы по двум дисциплинам на свой выбор.

Зачет отмечайте флажком.

У такой формы контроля, как **зачет** может быть два состояния: «**зачет**» или «**незачет**». Для индикации состояния, которое может иметь одно из двух допустимых значений, используется элемент управления типа **флажок**, в чем Вы и убедились.

Теперь сделаем так, чтобы **рядом с флажком появлялась надпись «зачет»**, а если флажка нет (то есть Вы не отметили его галочкой), то пусть появляется надпись «**незачет**»

### Задание 3:

Для того, чтобы посредством флажка ввести поле одно из двух значений надо воспользоваться функцией управления **If**, а для этого следует выполнить следующие действия.

1. Откройте форму **Зачет** в режиме **конструктора**;

2. Справа от поля типа флажок добавить элемент типа **поле**:

- щелкните мышкой по значку  на панели инструментов и **справа** от флажка еще раз щелкните мышкой, появится поле с произвольной надписью.

Нажмите **правую мышь** на этом поле, раскройте пункт **свойства**. Введите в свойство **данные** следующее выражение:

**=If([Результат];"Зачет";"Незачет")**

3. Установить для свойства **Вывод на экран** для **надписи** (надпись находится слева от поля) поля значение **нет**.

**Задание 4:**

1. Просмотрите форму Зачет в режиме формы. Теперь установленный флажок будет соответствовать значению зачет, а снятый флажок – значению незачет.
2. Пользуясь новым элементом управления флажок проставьте результаты по зачету заново. СОХРАНИТЬ!

**Раздел 6.**

Таблица состоит из строк и столбцов и имеет уникальное имя в базе данных. База данных содержит множество таблиц, связь между которыми устанавливается с помощью совпадающих полей. MS Access поддерживает четыре типа отношений между таблицами:

**Один-к-одному** (каждая запись в одной таблице соответствует только одной записи в другой таблице)

**Один-ко-многим** (каждая запись в одной таблице соответствует многим записям в другой таблице)

**Много-к-одному** (аналогична записи «один-ко-многим»)

**Много-ко-многим** (одна запись из первой таблицы может быть связана более чем с одной записью из второй таблицы либо одна запись из второй таблицы может быть связана более чем с одной записью из первой таблицы)

**Тип отношения «один-ко-многим» является наиболее общим**

Рассмотрим на конкретном примере, для чего может понадобиться связь между таблицами.

Допустим, уже по всем группам заполнили таблицы **Экзамен**, и **Зачет** полностью и выяснилось, что одна из дисциплин в таблице **Дисциплины** названа неверно, что в свою очередь отразилось на заполнении таблиц **Зачет**, либо **Экзамен**. Аналогичная ситуация может случиться с фамилией студента и.т.д.

Как сделать, чтобы, исправив название дисциплины в таблице **Дисциплины**, либо, изменив фамилию студента в таблице **Анкета**, таблицы **Зачет** и **Экзамен** были бы исправлены автоматически. Для этого надо установить связи между таблицами.

**Задание 1:**

Установите связь между таблицами **Зачет – [Дисциплины, Анкета]**

Отношение **один-ко-многим** создается в том случае, когда поле, по которому устанавливается связь (со стороны отношения **один**) должно иметь свойство.

**Индексированное поле - Да, совпадения не допускаются**

Это относится к полю **фамилия** в таблице **Анкета** и полю **дисциплина** - в таблице **Дисциплины**. Проверьте наличие этих свойств в ваших таблицах прежде, чем приступить к выполнению задания.

1. Для этого выполните следующие действия:

- перейти в режим **Таблицы**;
- если у вас есть открытые таблицы, или формы - закройте их;
- нажмите кнопку **Схема данных** на панели управления;
- пользуясь правой мышкой добавьте в схему все таблицы **Зачет**, **Анкета**, **Дисциплины**;
- закройте окно **Добавление таблицы**. В схеме данных должны появиться три схемы выше указанных таблиц;

- поместите указатель мыши на поле **дисциплины** таблицы **Зачет** и, нажав левую кнопку и, **не отпуская** ее, перетащите появившийся значок поля на поле **дисциплина** таблицы **Дисциплины**. Появится диалоговое окно **Связи**;
  - поставьте флажок **Обеспечение целостности данных**;
  - поставьте флажок **Каскадное обновление связанных полей**;
  - нажмите кнопку **Объединение**;
  - выберите опцию 3, **ОК**;
  - нажмите кнопку **Создать**, чтобы подтвердить создание связи;
  - перейдите в окно **Схема данных**, Вы должны увидеть появившуюся связь **одно-ко-многим**.
2. Аналогично создайте связь между полем **фамилия** таблицы **Анкета** и полем **студент** таблицы **Зачет**. *(последовательность действий та же, что и при установлении связи между таблицами **Дисциплины** – **Зачет**, только связываются другие поля)*

### Задание 2:

Если Вы правильно выполнили **Задание 1**, то:

После установления связи изменение полей **фамилия** и **дисциплина** соответственно в таблицах **Анкета**, **Дисциплины** автоматически повлечет связанных с этими полями полей **дисциплина** и **студент** в таблице **Зачет**. В свою очередь Вы не сможете ввести в таблицу **Зачет** новые фамилии, или дисциплины, если они отсутствуют в таблицах **Анкета** и **Дисциплины**.

**Проверьте это!!!!!!...**

### Задание 3:

Установите связь между таблицами **Экзамен** - [**Анкета**, **Дисциплины**] Проверьте, как работают связи!!! **СОХРАНИТЬ!**

## Раздел 7.

### Подчиненные формы.

Допустим, Вам требуется посмотреть данные результатов экзамена по каждому студенту в отдельности, либо по каждой дисциплине в отдельности, используя уже готовую таблицу, где у Вас занесены общие результаты по всем студентам и по всем дисциплинам.

Для этого используется механизм создания подчиненных форм.

#### Задание 1:

Создать подчиненную форму по результатам сдачи экзаменов с выборкой по каждому студенту в отдельности.

**Выполнить последовательно действия:**

1. Откройте Вашу базу данных;
2. Выберите вкладку **Формы**;
3. Нажмите кнопку **Создать** и выберите опцию **Мастер форм**, не выбирая источник данных, нажмите кнопку **ОК**.
4. В окне **Создание форм** выберите из списка **Таблицы/запросы** таблицу **Экзамен** (это будет подчиненная форма). Поместите в выбранные поля, поля **дисциплина**, **студент**, **результат контроля**.



5. Не выходя из окна **Создание форм**, выберите из списка **Таблицы/запросы** таблицу **Анкета** (это будет главная форма) и поместите в список **Выбранные поля** поле **фамилия**.
6. Нажмите кнопку **Далее**, в результате откроется следующее окно диалога **Создание форм**. В этом окне Вам предстоит определить главную и подчиненную таблицы.
7. Щелкните мышкой по таблице **Анкета**.
8. Обратите внимание на правую часть окна: в верхней части отображаются поля главной (**Анкета**), а в нижней части – подчиненной (**Экзамен**) таблицы.
9. Выбрать опцию **подчиненные формы**.
10. Определив главную и подчиненную таблицы, нажмите кнопку **Далее**.
11. Выберите **внешний вид – табличный**.
12. Выберите **стиль** на Ваше усмотрение.
13. Форму назвать **Анкета\_для\_экзамена**, название подчиненной формы не изменять.
14. Нажать **Готово**.

Созданная форма вполне приемлема для использования.

Попробуйте **менять фамилию студента** (используйте кнопки перемещения по записям внизу окна), и Вы увидите, как меняется содержимое таблицы **Экзамен**. Причем информация, появляющаяся в таблице, **относится только к выбранному Вами студенту**.

**Убедитесь в этом!!!** В этой форме явно лишняя информация в подчиненной таблице **Экзамен**, ведь **вполне достаточно фамилии** студента, которая появляется сверху над результатами контроля. Это можно исправить в конструкторе форм в следующем упражнении.

## Раздел 8.

### Форматирование форм.

#### Задание 1:

Измените стиль формы **Анкета** по своему усмотрению, для этого:

- откройте форму в режиме **конструктора**;
- нажмите на панели инструментов кнопку **Автоформат**;
- выберите нужный **стиль**.

#### Задание 2:

1. Измените **цвет фона и цвет шрифта** для каждого поля по своему усмотрению.

2. Вставьте на место поля **код рисунок**:

- нажмите кнопку **рисунок** на панели инструментов, щелкните мышкой по тому месту, где должен быть рисунок (поле «код») и выберите любой из файлов в папке **«Programm Files\Microsoft Office\Clipart»**.

#### Задание 3:

1. Попробуйте изменить размер поля для ввода фамилии по высоте, для этого надо щелкнуть мышкой по этому полю и изменить появившиеся границы поля.
2. Измените шрифт (размер, вид, цвет) для поля фамилия.

**Задание 4:**

1. Убрать в подчиненной форме Экзамен поле студент:
  - открыть форму **Анкета\_для\_экзамена** (в подчиненной форме **Экзамен** желательно убрать поле **Студент**, так как фамилия уже присутствует в верхней части формы)
  - уберите поле «студент»:
    - открыть форму в режиме конструктора;
    - выделить поле **студент** и вырезать это поле.
2. Выйдите из режима конструктора и посмотрите, как теперь выглядит форма.

**Раздел 9.****Выбор данных из таблиц с помощью запросов**

Вы уже познакомились с конструктором запросов, когда использовали их для подстановки в качестве источника строк при построении таблиц.

**Задание 1:**

1. Добавьте в таблицу **Экзамен** поле **преподаватель** со следующими характеристиками:
  - тип данных – **текстовый**;
  - подпись – **преподаватель**;
  - подстановка – используйте поле со списком (**источник строк** – таблица **Преподаватели**).

**Задание 2:**

У Вас в таблице **Экзамен** должно быть не менее 20 записей и эти записи должны содержать информацию:

- не менее, чем по **трем** дисциплинам;
- оценки должны быть **разные** (от отлично до неудовлетворительно);
- студенты должны быть из **разных** групп (хотя бы из **двух**).

Убедитесь в том, что записей в таблице достаточное количество, в противном случае добавьте новые записи.

**Задание 3:**

1. Создайте **запрос** по выбору записей, содержащих студентов, сдавших все экзамены на **отлично**. Для этого надо выполнить следующие действия:
  - выбрать в окне базы данных вкладку **Запрос**, нажать кнопку **Создать**;
  - в окне диалога **Новый запрос** выберете опцию **Конструктор**;
  - Access предложит Вам выбрать таблицу, выберете таблицу **Экзамен**, нажмите кнопку **Добавить** и закройте окно диалога;
  - из таблицы последовательно выберете поля **студент**, **дисциплина**, **результат** и перетаскиваете выбранные поля по очереди в бланк запроса (*можно просто дважды щелкнуть мышкой по каждому полю*);

- в строке **условие отбора** введите условие:
  - ✓ щелкните по кнопке **Построить** на панели управления;
  - ✓ в появившемся построителе щелкните по кнопке **Like**;
  - ✓ напечатайте **Like “отлично”**, **ОК**.
- закройте окно конструктора запросов.

2. Сохраните запрос с именем **Запрос\_по\_отличникам**.

3. Откройте созданный запрос и убедитесь в том, что выборка была сделана правильно.

#### Задание 4:

1. Создайте аналогичный запрос по двоечникам и сохраните с именем **Запрос\_по\_двоечникам**.
2. Откройте созданный запрос и убедитесь в том, что выборка была сделана правильно.

#### Задание 5:

1. Создайте запрос на выборку записей по результатам контроля по дисциплине, которую преподает **Быковская** (можно выбрать другого преподавателя). Запрос сохраните с именем **Запрос\_по\_дисциплине**.
2. Откройте созданный запрос и убедитесь в том, что выборка была сделана правильно.

#### Задание 6:

1. Создайте на базе таблица **Анкета** запрос для формирования списка студентов, которые родились с **1 мая 1982 года по 31 декабря 1982 года**.

Для формирования запроса Вам следует вызвать построитель выражений (кнопка **Построить**) и выбрать в списке операторов - оператор сравнения **Between**, нажать кнопку **Вставить**, **ОК**.

Теперь откорректируйте появившееся выражение следующим образом:

**Between #01.05.82# And #31.12.82#.**

Если вдруг оказалось, что таких студентов в Вашей группе нет, **измените** условия запроса.

Запрос **сохранить** с именем **Запрос\_по\_дате\_рождения**.

2. Откройте созданный запрос и убедитесь в том, что выборка была сделана правильно.

#### Задание 7:

Создайте формы для всех запросов (*автоформа ленточная*)

Желательно, чтобы режим запросов должен быть организован только для просмотра нужной информации, а не для ее корректировки.

#### Задание 8:

Закройте доступ к корректировке базы данных в режиме запросов.

Чтобы в режиме запросов нельзя было бы изменить информацию в базе данных надо выполнить следующие действия:

- Открыть форму для запроса в режиме конструктора.
- Пометить поле и правой кнопкой вызвать свойства помеченного поля, затем на вкладке **Данные** установить свойство **Блокировка – Да**.

Аналогично **установите** Свойство **Блокировка – Да** для всех полей во всех формах для запросов (формы для запросов Вы создали в **Задании 7**).

## Раздел 10.

### Построение отчетов.

Вы уже научились создавать таблицы, вводить и редактировать в них данные, как в самих таблицах, так и при использовании форм. Вы можете также просматривать и анализировать базу, выбирая нужную информацию с помощью запросов. Результаты просмотра и анализа могут быть представлены в виде отчетов.

В этом упражнении Вы научитесь создавать и оформлять отчеты для их последующей распечатки.

#### Задание 1:

1. Создать отчет по результатам сдачи экзаменов, используя таблицу **Экзамен**. Для этого выполните последовательность действий:

- откройте вкладку **Отчеты**;
- нажать кнопку **Создать**, выбрать **Мастер отчетов**, в качестве источника выбрать таблицу **Экзамен**
- перенести в **выбранные поля** следующие поля: дисциплина, группа, студент, результат контроля;
- нажать **Далее**.

Теперь надо определить, требуется ли **сгруппировать** данные по какому-либо из полей. **Поля**, по которым будет осуществляться **группировка**, помещаются в **верхней** части правого списка в **отдельной** рамке и выделяются на экране **синим** цветом:

- выбрать группировку по **дисциплине**, нажать **Далее**;
- задать режим сортировки в возрастающем порядке по полю **Студент**, нажать **Далее**;
- определить вид **макета** - **Ступенчатый** (*или на свое усмотрение, можете поэкспериментировать*);
- нажать **Далее**;
- выбрать **стиль** на свое усмотрение, нажать **Далее**;
- задать имя отчета **Отчет\_по\_экзаменам**, нажать кнопку **Готово**.


Отчет готов, можете просмотреть его в режиме **Файл\Предварительный просмотр**. Для того, чтобы привести отчет в требуемый вид, его можно откорректировать в режиме конструктора.

#### Задание 2:

1. Откройте Ваш отчет в режиме конструктора.
2. Теперь, пользуясь теми же приемами, что и при форматировании форм (*напоминаю - надо выделить изменяемое поле, щелкнуть правой мышкой и и выбрать пункт **свойства***), измените в отчете в **области заголовка** и в **области данных** размер шрифта на **12**, каждое поле (столбец) раскрасьте **разным** цветом и т.д.
3. Обычно в **заголовке** отчет размещают **дату** печати отчета.

Для размещения даты выполните следующие действия:

- выберете на панели инструментов кнопку поле **аб!**;

- установите указатель мыши на место в области **верхнего** колонтитула, где Вы **предполагаете** разместить **поле** даты. В отчете появится связанный объект, состоящий из поля ввода и надписи к нему;
  - выделите надпись связанного поля и удалите ее, нажав клавишу **delete**, Вы подобную операцию выполняли, когда работали с **флажком**;
  - выделите поля ввода, где будет размещаться дата, и откройте для него окно свойств (*правой мышкой*);
  - перейдите на вкладку **данные** и нажмите кнопку построителя . На экране откроется окно диалога **Построитель выражения**;
  - откройте раздел **Встроенные функции** папки **Функции** и, используя функцию **Now** из списка функций, задайте выражение **Now ()**, нажав последовательно кнопки **Вставить**, **ОК**.
  - нажмите кнопку раскрытия списка свойства **Формат** вкладки **Макет** и выберете устраивающий Вас формат отображения даты в отчете.
4. Отчет сохраните с именем **Экзамен**.
  5. Просмотрите отчет.
  6. Аналогично созданию отчета **Экзамен** создайте отчет **Зачет**.

## Раздел 11.

### Создание кнопок управления.

Кнопки используются в формах для выполнения определенного действия или ряда действий.

#### Задание 1:

1. Создать **кнопки выхода** из всех форм, которые Вы создали:
  - Анкета;
  - Преподаватели;
  - Дисциплины;
  - Группы;
  - Экзамен;
  - Зачет.

#### Начнем с формы **Анкета**:

- открыть форму в режиме конструктора;
- установите режим использования мастера на панели элементов (кнопка **Мастера**);
- выберите инструмент **Кнопка** на панели элементов;
- установите указатель мыши на место в форме, в котором Вы предполагаете расположить **кнопку для выхода из формы** и нажмите кнопку мыши. Запускается мастер создания кнопки конструктора форм.

В первом окне диалога расположены два списка: **Категории** и **Действия**

- выберите в списке **Категории** пункт **Работа с формой**, а в списке **Действия** пункт **Закрытие формы**, Далее;
- в окне **Создание кнопок** введите текст **«Выход»**, Далее;
- задать **имя кнопки Выход из анкеты**, **Готово**.

Теперь можно выйти из конструктора и попробовать, как работает кнопка.

#### **Задание 2:**

Создать **кнопки выхода** для **всех форм** (в том числе и запросов), которые Вы создали.

#### **Задание 3:**

Создать кнопки выхода для **всех отчетов**, которые Вы создали.

Оформление кнопок можно изменить в режиме конструктора, используя правую мышку **Цвет текста**.

## **Раздел 12.**

### **Создание кнопочных форм**

Кнопочная форма есть не что иное, как **Меню** для работы в базе данных. В меню может быть несколько вложенных подменю. Вы будете создавать меню следующей структуры:

**Схема меню:** в меню 3 уровня

#### **Задание 1:**

1. Создать меню 1-го уровня:

- **закройте** все таблицы, формы, отчеты, запросы;
- настройте **Диспетчер кнопочных форм**;
- нажать кнопку **Изменить**;
- ввести название кнопочной формы **Студенты**;
- нажать кнопку **Закреть**;
- нажать кнопку **Создать**;
- в окне **Создание** ввести текст **Формы, ОК**;
- нажать кнопку **Создать**;
- в окне **Создание** ввести текст **Запросы, ОК**;
- нажать кнопку **Создать**;
- аналогично создать следующие страницы кнопочной формы **Отчеты, Выход**.

2. Создать меню 2-го уровня.

Теперь последовательно будем раскрывать каждый пункт меню 1-го уровня (для наглядности можно ориентироваться на схему меню):

- отметить курсором главную кнопочную форму **Студент** и нажать кнопку **Изменить**;
- создать элемент кнопочной формы;
- текст **Формы**;
- команда **Переход к кнопочной форме**;

- кнопочная форма - выбрать из списка **Формы, ОК;**
- создать элемент кнопочной формы;
- текст **Запросы;**
- команда **Переход к кнопочной форме;**
- кнопочная форма - выбрать из списка **Запросы, ОК;**
- создать элемент кнопочной формы;
- текст **Отчеты;**
- команда **Переход к кнопочной форме;**
- кнопочная форма - выбрать из списка **Отчеты, ОК;**
- создать элемент кнопочной формы;
- текст **Выход;**
- команда **Выход из приложения;**
- кнопочная форма - выбрать из списка **Выход, ОК;**

Закрывать окно **Изменение страницы кнопочной формы**, Вы вернулись в окно **Диспетчер кнопочных форм**.

3. Создать меню 3-го уровня:

Установить курсор на страницу **Формы** и нажать кнопку **Изменить**.

Вы находитесь в окне **Изменение страниц кнопочной формы**:

- создать элемент кнопочной формы;
- текст **Анкета;**
- команда **Открытие формы в режиме редактирования;**
- кнопочная форма - выбрать из списка **Анкета, ОК;**
- создать элемент кнопочной формы;
- текст **Преподаватели;**
- команда **Открытие формы в режиме редактирования;**
- кнопочная форма - выбрать из списка **Преподаватели, ОК;**
- создать элемент кнопочной формы;
- текст **Дисциплины;**
- команда **Открытие формы в режиме редактирования;**
- кнопочная форма - выбрать из списка **Дисциплины, ОК;**
- создать элемент кнопочной формы;
- текст **Группы;**

- команда **Открытие формы в режиме редактирования;**
- кнопочная форма - выбрать из списка **Группы, ОК.**

Вы создали меню для ввода и корректировки информации в базу данных, теперь создадим меню для вызова запросов.

#### **Задание 2:**

Создать меню 3-го уровня для вызова запросов.

**Закрыть** окно **Изменение страницы кнопочной формы**, если Вы в нем находитесь. Вы опять в **диспетчере кнопочных форм.**

Установить курсор на страницу **Запросы** и нажать кнопку **Изменить**. Вы находитесь в окне **Изменение страниц кнопочной формы:**

- создать элемент кнопочной формы;
- текст **Запрос\_по\_двоечникам;**
- команда **Открытие формы в режиме редактирования;**
- кнопочная форма - выбрать из списка **Запрос\_по\_двоечникам, ОК;**

Аналогично создать элементы кнопочной формы для других запросов.

#### **Задание 3:**

Создать меню 3-го уровня для **просмотра отчетов**, выполнив действия аналогичные действиям при создании меню для форм и запросов.

#### **Задание 4:**

1. Закройте **диспетчер кнопочных форм.**
2. Теперь в Вашей Базе на вкладке **Формы** появилась новая форма с именем **Кнопочная форма.**
3. Откройте эту форму и проверьте как работает меню.

### **Раздел 13.**

#### **Оформление Меню для работы с базой**

Меню работы с базой можно внешне изменить, дополнив его рисунками, значками, изменить размер кнопок, изменить шрифт надписей пунктов меню и т.д. Кроме того, создав Меню работы с базой желательно от посторонних глаз скрыть окно базы данных, чтобы пользователь разработанного Вами программного продукта **не смог увидеть и изменить** таблицы, запросы, макросы.

#### **Задание 1:**

1. Изменить шрифт надписей пунктов меню:
  - открыть **кнопочную форму** в режиме **конструктора;**
  - выделить мышкой то место, где может быть надпись пункта меню **Формы;**
  - выбрать **свойства**, (*либо просто щелкнуть правой мышкой*);
  - откройте вкладку **Макет** и можете на свое усмотрение изменить:
    - ✓ Размер шрифта
    - ✓ Цвет текста
    - ✓ Цвет границы



✓ Насыщенность

2. Аналогично можно изменить оформление для всех пунктов меню.
3. Теперь можно выйти из режима конструктора и посмотреть, как выглядит кнопочная форма (*это и есть главное меню для работы с базой данных*).

#### Задание 2:

1. **Вставьте рисунок** в кнопочную форму на **пустое место слева** от меню:
  - открыть **кнопочную форму** в режиме конструктора;
  - щелкните правой кнопкой **по пустому месту** справа от меню;
  - выбрать пункт **Свойства**;
  - выбрать вкладку **Макет/ Рисунок**, нажать кнопку  ;
  - выбрать рисунок в папке **C:\Windows**.
2. Оформление можно продолжать дальше, меняя цветовые гаммы оставшегося места на панели **Меню**.

#### Промежуточное тестирование

1. Современной формой информационных систем являются банки данных, включающие в свой состав базы данных
  - a) Одну
  - b) Несколько
  - c) Одну или несколько
2. Информация в БД должна быть
  - a) Противоречивой
  - b) Целостной
  - c) Избыточной
3. Одной из основных моделей представления данных в БД является
  - a) Реляционная
  - b) Операционная
  - c) Традиционная
  - d) Объектная
4. Операции над объектами данных имеют характер
  - a) Теоретический
  - b) Множественный
  - c) Теоретико-множественный
5. Одним из видов связей (отношений) в реляционной модели БД является связь

- a) Один-к-двум
  - b) Много-к-одному
  - c) Два-к-одному
  - d) Два-ко-многим
6. К разновидностям графовых моделей данных относится
- a) Реляционная
  - b) Иерархическая
  - c) Объектно-ориентированная
7. Транзакция – это
- a) Операция
  - b) Выполнение последовательности операций
  - c) Последовательность операций
8. При выполнении операций с таблицами используется один из способов доступа к данным
- a) Коммуникационный
  - b) Навигационный
9. Автором реляционной модели представления данных в БД является
- a) Э. Додд
  - b) Э. Тодд
  - c) Э. Кодд
10. Информационная модель системы без привязки к типу ЭВМ и типу системных программных средств создается на этапе
- a) Формирования и анализа требований к системе
  - b) Концептуального проектирования
  - c) Проектирования реализации
  - d) Физической реализации
11. Главным исходным данным для даталогического проектирования БД является
- a) Инфологическая модель
  - b) Множество идентификаторов групп атрибутов
  - c) Описание потоков информации и форм документов
12. В реляционной модели данных порядок строк и столбцов
- a) Существенен

- b) Не существенен
13. Расставьте по порядку этапы проектирования и создания БД
- Создание схемы данных средствами СУБД
  - Построение модели данных предметной области
  - Конструирование таблиц БД средствами СУБД
  - Определение структуры реляционной БД
  - Ввод данных в таблицы
14. Связи "многие ко многим" непосредственно не поддерживаются в модели
- Реляционной
  - Иерархической
  - Сетевой

**Варианты предметных областей для самостоятельной работы (ИДЗ).**

1. Учет движения материалов на складе.
2. Учет студентов в колледже.
3. Учет движения поездов.
4. Учет проживающих в гостинице.
5. Автоматизация работы транспортного предприятия.
6. Учет посещений в поликлинике.
7. Магазин компьютерной техники.
8. Учет больных в стационаре.
9. Учет клиентов в ателье.
10. Учет движения лекарств в аптеке.
11. Автоматизация службы рекламы.
12. Магазин подержанных автомобилей.

**Задания для ИДЗ:**

- составить анализ предметной области;
- разработать инфологическую модель;
- разработать даталогическую модель;
- описать инструкции пользователя с СУБД.

**Вопросы для семинара  
по дисциплине «Основы проектирования баз данных»**

1. Определение банка данных.
2. Определение СУБД.
3. Стадии развития банка данных.
4. Определение базы данных.
5. Свойства базы данных.
6. Требования к базе данных.
7. Категории пользователей.
8. Функции СУБД.
9. Трехуровневая структура разработки СУБД.
10. Классификация моделей данных.
11. Типы моделей данных.
12. Три части реляционной модели.
13. Понятие нормализации.
14. Типы связей.
15. Архитектура СУБД.
16. Классификация СУБД.
17. Основные понятия (сущность, атрибут, связь, ключ, типы ключей).
18. Архитектура баз данных. Три уровня: логический, концептуальный, физический.
19. Данные. Объекты и атрибуты БД.
20. Взаимосвязи между объектами и атрибутами. Особенности реализации взаимосвязей
21. Модели данных.
22. Реляционная модель данных: основные определения, представление данных, аномалии работы с данными, преимущества и недостатки модели.
23. Иерархическая модель данных: основные определения, представление данных, аномалии работы с данными, преимущества и недостатки модели.
24. Сетевая модель данных: основные определения, представление данных, аномалии работы с данными, преимущества и недостатки модели.
25. Проектирование баз данных. Общий подход к проектированию
26. Проектирование информационно-логической модели предметной области. Тестирование пользователей.
27. Выявление взаимосвязей между данными. Реляционные основы проектирования.
28. Нормализация отношений.
29. Методы хранения и доступа к данным. Интерфейс между пользователем и базой данных.
30. Теория реляционных БД. Реляционная алгебра, реляционное исчисление. Теория множеств. Основные операции реляционной алгебры.
31. Модель «Сущность-связь».

#### **Итоговое тестирование.**

1. Главным исходным данным для даталогического проектирования БД является:
  - а) инфологическая модель

- b) множество идентификаторов групп атрибутов
  - c) описание потоков информации и форм документов
2. Описание физической структуры БД называют схемой:
    - a) доступа
    - b) хранения
    - c) обеспечения
  3. Число этапов проектирования информационно-логической модели равно:
    - a) 2
    - b) 3
    - c) 4
  4. Под архитектурой БД понимается совокупность:
    - a) основных, функциональных компонентов
    - b) задач и методов
    - c) знаний
  5. В реляционной модели данных порядок строк и столбцов:
    - a) существенен
    - b) не существенен

Дополните:

6. Функциональный компонент системы базы данных, механизмы которого служат для поддержки уровня абстракции данных, называется \_\_\_\_\_ уровнем системы баз данных.
7. Обобщенное представление пользователей о базе данных – это \_\_\_\_\_ уровень.
8. Для поддержки представления базы данных в среде хранения служат механизмы \_\_\_\_\_ уровня.
9. Для описания частных представлений пользователей о данных служат механизмы \_\_\_\_\_ уровня.
10. Модель данных, которая позволяет представить базу данных в виде совокупности таблиц, называется \_\_\_\_\_.
11. Процесс выявления объектов и их взаимосвязей с помощью концепции реляционной модели и табличной формы представления называется процессом \_\_\_\_\_.
12. Иерархическая и сетевая модели – это разновидности \_\_\_\_\_ моделей данных.
13. Удаление исходных объектов базы данных, которое влечет за собой удаление порожденных, является недостатком \_\_\_\_\_ модели данных.
14. Данные предметной области в виде совокупности информационных объектов и связей между ними отображаются в \_\_\_\_\_ модели предметной области.
15. Проектирование структуры базы данных в среде конкретной СУБД является проектированием \_\_\_\_\_ модели базы данных.
16. Поименованная совокупность связанных данных в сетевой модели данных называется \_\_\_\_\_.
17. Иерархическая древовидная структура строится из \_\_\_\_\_.
18. В иерархической модели узел – это совокупность \_\_\_\_\_.

19. Ключ, состоящий из нескольких атрибутов, называется \_\_\_\_\_.
20. Обобщенная модель архитектуры СУБД содержит \_\_\_\_\_ уровня.
21. Процесс выделения информационных объектов предметной области, отвечающих требованиям нормализации, может производиться на основе подходов:
- a) последовательного анализа
  - b) интуитивного
  - c) двоичного поиска
  - d) формального, предложенного Дж. Мартином
22. Основные операции, производимые над данными в БД:
- a) сортировка
  - b) запоминание
  - c) обмен
  - d) вставка
  - e) деление
  - f) удаление
  - g) модификация