

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ
федеральное государственное автономное образовательное учреждение высшего
образования
"САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ"

Кафедра № 43

УТВЕРЖДАЮ

Руководитель направления

доц., к.т.н., доц.

(должность, уч. степень, звание)

А.А. Ключарев

(инициалы, фамилия)



(подпись)

«04» июня 2020 г

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

«Проектирование баз данных»
(Наименование дисциплины)

Код направления подготовки/ специальности	09.03.04
Наименование направления подготовки/ специальности	Программная инженерия
Наименование направленности	Проектирование программных систем
Форма обучения	заочная

Санкт-Петербург– 2020

Лист согласования рабочей программы дисциплины

Программу составил (а)

старший преподаватель
(должность, уч. степень, звание)



03.06.2020

(подпись, дата)

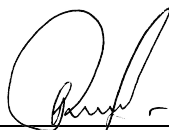
Н.В. Путилова
(инициалы, фамилия)

Программа одобрена на заседании кафедры № 43

«04» июня 2020 г, протокол № 08-2019/20

Заведующий кафедрой № 43

д.т.н., проф.
(уч. степень, звание)



04.06.2020

(подпись, дата)

М.Ю. Охтилев
(инициалы, фамилия)

Ответственный за ОП ВО 09.03.04(02)

доц., к.т.н., доц.
(должность, уч. степень, звание)



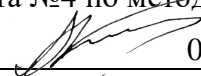
04.06.2020

(подпись, дата)

А.А. Ключарев
(инициалы, фамилия)

Заместитель директора института №4 по методической работе

доц., к.т.н., доц.
(должность, уч. степень, звание)



04.06.2020

(подпись, дата)

А.А. Ключарев
(инициалы, фамилия)

Аннотация

Дисциплина «Проектирование баз данных» входит в образовательную программу высшего образования по направлению подготовки/ специальности 09.03.04 «Программная инженерия» направленности «Проектирование программных систем». Дисциплина реализуется кафедрой «№43».

Дисциплина нацелена на формирование у выпускника следующих компетенций:

ОПК-6 «Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов»

Содержание дисциплины охватывает круг вопросов, связанных с реляционными , объектными, объектно-реляционными и NoSQL СУБД, реляционной алгеброй, языком SQL, проектированием структуры баз данных, разработкой запросов к базам данных, разработкой серверной части базы данных (в том числе триггеров и хранимых процедур), использованием CASE-средств и систем управления базами данных для разработки баз данных, использованием современных моделей данных для разработки систем различных предметных областей.

Преподавание дисциплины предусматривает следующие формы организации учебного процесса: лекции, лабораторные работы, практические занятия, самостоятельная работа обучающегося.

Программой дисциплины предусмотрены следующие виды контроля: текущий контроль успеваемости, промежуточная аттестация в форме экзамена.

Общая трудоемкость освоения дисциплины составляет 6 зачетных единиц, 216 часов.

Язык обучения по дисциплине «русский»

1. Перечень планируемых результатов обучения по дисциплине

1.1. Цели преподавания дисциплины

Целью преподавания дисциплины является получение обучающимися необходимых знаний и навыков в области организации хранения информации в базах данных в среде систем управления базами данных, обеспечения целостности данных, выполнения запросов к информации, хранимой в базах данных, получения знаний о роли баз данных в современных программных системах и современных типах систем управления базами данных.

1.2. Дисциплина входит в состав обязательной части образовательной программы высшего образования (далее – ОП ВО).

1.3. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения ОП ВО.

В результате изучения дисциплины обучающийся должен обладать следующими компетенциями или их частями. Компетенции и индикаторы их достижения приведены в таблице 1.

Таблица 1 – Перечень компетенций и индикаторов их достижения

Категория (группа) компетенции	Код и наименование компетенции	Код и наименование индикатора достижения компетенции
Общепрофессиональные компетенции	ОПК-6 Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов	ОПК-6.3.1 знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий ОПК-6.У.2 умеет применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ

2. Место дисциплины в структуре ОП

Дисциплина может базироваться на знаниях, ранее приобретенных обучающимися при изучении следующих дисциплин:

- «Объектно-ориентированное программирование»;
- «Проектирование программных систем»;
- «Структуры и алгоритмы обработки данных».

Знания, полученные при изучении материала данной дисциплины, имеют как самостоятельное значение, так и могут использоваться при изучении других дисциплин:

- «Разработка и анализ требований»
- «Экспертные системы»

3. Объем и трудоемкость дисциплины

Данные об общем объеме дисциплины, трудоемкости отдельных видов учебной работы по дисциплине (и распределение этой трудоемкости по семестрам) представлены в таблице 2.

Таблица 2 – Объем и трудоемкость дисциплины

Вид учебной работы	Всего	Трудоемкость по семестрам	
		№6	№7
1	2	3	4
Общая трудоемкость дисциплины, ЗЕ/ (час)	6/ 216	3/ 108	3/ 108
Аудиторные занятия, всего час.	40	24	16
в том числе:			
лекции (Л), (час)	14	8	6
практические/семинарские занятия (ПЗ), (час)	4	4	
лабораторные работы (ЛР), (час)	22	12	10
курсовой проект (работа) (КП, КР), (час)			
экзамен, (час)	9	9	
Самостоятельная работа, всего (час)	167	75	92
Вид промежуточной аттестации: зачет, дифф. зачет, экзамен (Зачет, Дифф. зач, Экз.**)	Экз., Зачет	Экз.	Зачет

Примечание: ** кандидатский экзамен

4. Содержание дисциплины

4.1. Распределение трудоемкости дисциплины по разделам и видам занятий.

Разделы, темы дисциплины и их трудоемкость приведены в таблице 3.

Таблица 3 – Разделы, темы дисциплины, их трудоемкость

Разделы, темы дисциплины	Лекции (час)	ПЗ (СЗ) (час)	ЛР (час)	КП (час)	СРС (час)
Семестр 6					
Раздел 1. Введение в проектирование баз данных					
Тема 1.1. История развития и причины появления СУБД					
Тема 1.2. Модели данных.	2	1	2		16
Тема 1.3. Нормализация и денормализация баз данных					
Тема 1.4. Роль баз данных в программных системах					
Раздел 2. Введение в базы данных					
Тема 2.1. Отношения и их свойства, ключи отношений, абстрактные операции манипулирования данными					
Тема 2.2. Реализация отношений в базах данных, типы данных в языке SQL, операторы языка SQL для создания, удаления, модификации таблиц базы данных, манипулирования данными	3	1	4		26
Тема 2.3. Реляционная алгебра					

Раздел 3. Введение в язык SQL					
Тема 3.1. Оператор выборки в языке SQL, агрегатные функции					
Тема 3.2. Запросы с подзапросами, экзистенциальные запросы, объединение, пересечение, разность запросов	3	2	6		33
Тема 3.3. Использование представлений, управляющих конструкций в языке SQL					
Тема 3.4. Хранимые процедуры.					
Итого в семестре:	8	4	12		75
Семестр 7					
Раздел 3. Введение в язык SQL					
Тема 3.5. Триггеры, обеспечение активной целостности	1				18
Раздел 4. . Объектные и объектно-реляционные базы данных					
Тема 4.1. Выбор типа СУБД применительно к особенностям предметной области	2		5		35
Тема 4.2. Объектно-реляционные базы данных					
Тема 4.3. Объектные базы данных					
Раздел 5. NoSQL базы данных					
Тема 5.1. Базы данных «Ключ-значение»	3		5		39
Тема 5.2. Документные базы данных					
Тема 5.3. Графовые базы данных					
Тема 5.4. Базы данных «Семейство столбцов»					
Итого в семестре:	6		10		92
Итого	14	4	22	0	167

4.2. Содержание разделов и тем лекционных занятий.

Содержание разделов и тем лекционных занятий приведено в таблице 4.

Таблица 4 – Содержание разделов и тем лекционного цикла

Номер раздела	Название и содержание разделов и тем лекционных занятий
1	Раздел 1. Введение в проектирование баз данных Тема 1.1. История развития и причины появления СУБД Тема 1.2. Модели данных. Тема 1.3. Нормализация и денормализация баз данных Тема 1.4. Роль баз данных в программных системах
2	Раздел 2. Введение в базы данных Тема 2.1. Отношения и их свойства, ключи отношений, абстрактные операции манипулирования данными Тема 2.2. Реализация отношений в базах данных, типы данных в языке SQL, операторы языка SQL для создания, удаления, модификации таблиц базы данных, манипулирования данными Тема 2.3. Реляционная алгебра
3	Раздел 3. Введение в язык SQL Тема 3.1. Оператор выборки в языке SQL, агрегатные функции

	Тема 3.2. Запросы с подзапросами, экзистенциальные запросы, объединение, пересечение, разность запросов Тема 3.3.Использование представлений, управляющих конструкций в языке SQL Тема 3.4. Хранимые процедуры. Тема 3.5. Триггеры, обеспечение активной целостности
4	Раздел 4. . Объектные и объектно-реляционные базы данных Тема 4.1. Выбор типа СУБД применительно к особенностям предметной области Тема 4.2. Объектно-реляционные базы данных Тема 4.3. Объектные базы данных
5	Раздел 5. NoSQL базы данных Тема 5.1. Базы данных «Ключ-значение» Тема 5.2. Документные базы данных Тема 5.3. Графовые базы данных Тема 5.4. Базы данных «Семейство столбцов»

4.3. Практические (семинарские) занятия

Темы практических занятий и их трудоемкость приведены в таблице 5.

Таблица 5 – Практические занятия и их трудоемкость

№ п/п	Темы практических занятий	Формы практических занятий	Трудоемкость, (час)	№ раздела дисциплины
Семестр 6				
1	Проектирование базы данных: Критерии корректности определения связей и между сущностями нормализация реляционной базы данных	Кейс, занятие по моделированию реальных условий	1	1
2	Определение ключей отношений. Запросы на определение данных, манипулирование данными	Решение задач., кейс,	1	2
3	Построение сложных запросов на SQL: запросов с агрегатными функциями, с внешними соединениями, запросов с подзапросами	Решение задач.	1	3
4	Использование управляющих конструкций в языке SQL, хранимые процедуры	Кейс, занятие по моделированию реальных условий	1	3
Всего			4	

4.4. Лабораторные занятия

Темы лабораторных занятий и их трудоемкость приведены в таблице 6.

Таблица 6 – Лабораторные занятия и их трудоемкость

№ п/п	Наименование лабораторных работ	Трудоемкость, (час)	№ раздела дисциплины
-------	---------------------------------	---------------------	----------------------

Семестр 6			
1.	Разработка физической модели базы данных с учетом декларативной ссылочной целостности	2	1
2.	Создание и модификация базы данных и таблиц базы данных	2	2
3.	Заполнение таблиц и модификация данных	2	2
4.	Разработка SQL запросов: виды соединений и шаблоны	2	3
5.	Разработка SQL запросов: запросы с подзапросами	2	3
6.	Хранимые процедуры	2	3
Семестр 7			
7.	Объектно-реляционные базы данных. Проектирование и создание	3	4
8.	Объектно-реляционные базы данных. Манипуляция данными и пользовательские операторы	2	4
9.	Разработка документной базы данных	3	5
10.	Манипулирование данными в документной базе данных	2	5
Всего		22	

4.5. Курсовое проектирование/ выполнение курсовой работы
Учебным планом не предусмотрено

4.6. Самостоятельная работа обучающихся
Виды самостоятельной работы и ее трудоемкость приведены в таблице 7.

Таблица 7 – Виды самостоятельной работы и ее трудоемкость

Вид самостоятельной работы	Всего, час	Семестр 6, час	Семестр 7, час
1	2	3	4
Изучение теоретического материала дисциплины (ТО)	129	55	74
Контрольные работы заочников (КРЗ)	14	6	8
Подготовка к промежуточной аттестации (ПА)	24	14	10
Всего:	167	75	92

5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю)
Учебно-методические материалы для самостоятельной работы обучающихся указаны в п.п. 7-11.

6. Перечень печатных и электронных учебных изданий
Перечень печатных и электронных учебных изданий приведен в таблице 8.
Таблица 8– Перечень печатных и электронных учебных изданий

Шифр/ URL адрес	Библиографическая ссылка	Количество экземпляров
--------------------	--------------------------	---------------------------

		В В библиотеке (кроме электронных экземпляров)
https://e.lanbook.com/book/126933 Режим доступа: для авториз. пользователей	Волк, В. К. Базы данных. Проектирование, программирование, управление и администрирование : учебник / В. К. Волк. — Санкт-Петербург : Лань, 2020. — 244 с. — ISBN 978-5-8114-4189-1. — Текст : электронный // Лань : электронно-библиотечная система..	
https://znanium.com/catalog/product/1032103 Режим доступа: по подписке	Кукарцев, В.В. Теория баз данных : учебник / В.В. Кукарцев, Р.Ю. Царев, О.А. Антамошкин. - Красноярск : Сиб. федер. ун-т, 2017. - 180 с. - ISBN 978-5-7638-3621-9. - Текст : электронный. - URL:	
https://znanium.com/catalog/product/858603 Режим доступа: по подписке	Тарасов, С. В. СУБД для программиста. Базы данных изнутри: Практическое пособие / Тарасов С.В. - Москва : СОЛОН-Пр., 2015. - ISBN . - Текст : электронный. -	
https://znanium.com/catalog/product/1053934 Режим доступа: по подписке	Голицына, О. Л. Базы данных : учеб. пособие / О.Л. Голицына, Н.В. Максимов, И.И. Попов. — 4-е изд., перераб. и доп. — Москва : ФОРУМ : ИНФРА-М, 2020. — 400 с. — (Высшее образование: бакалавриат). - ISBN 978-5-16-107544-9. - Текст : электронный. -	
https://znanium.com/catalog/product/1007949 Режим доступа: по подписке	Мартишин, С. А. Проектирование и реализация баз данных в СУБД MySQL с использованием MySQL Workbench : учеб. пособие / С.А. Мартишин, В.Л. Симонов, М.В. Храпченко. - М. : ИД «ФОРУМ» : ИНФРА-М, 2019. - 160 с. - (Высшее образование). - ISBN . - Текст : электронный. - URL:	
https://znanium.com/catalog/product/1093646 Режим доступа: по подписке	Мартишин, С. А. Базы данных: Работа с распределенными базами данных и файловыми системами на примере	

	<p>MongoDB и HDFS с использованием Node.js, Express.js, Apache Spark и Scala : учебное пособие / С.А. Мартишин, В.Л. Симонов, М.В. Храпченко. — Москва : ИНФРА-М, 2020. — 235 с. + Доп. материалы [Электронный ресурс]. — (Высшее образование: Бакалавриат). — DOI 10.12737/textbook_5cc063e18baca3.52928692. - ISBN 978-5-16-107636-1. - Текст : электронный.</p>	
<p>https://znanium.com/catalog/product/1066784 Режим доступа: по подписке</p>	<p>Мартишин, С. А. Базы данных. Практическое применение СУБД SQL и NoSQL-типа для проектирования информационных систем : учебное пособие / С.А. Мартишин, В.Л. Симонов, М.В. Храпченко. — Москва : ИД «ФОРУМ» : ИНФРА-М, 2020. — 368 с. — (Высшее образование: Бакалавриат). - ISBN 978-5-16-104936-5. - Текст : электронный.</p>	

7. Перечень электронных образовательных ресурсов информационно-телекоммуникационной сети «Интернет»

Перечень электронных образовательных ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины приведен в таблице 9.

Таблица 9 – Перечень электронных образовательных ресурсов информационно-телекоммуникационной сети «Интернет»

URL адрес	Наименование
http://citforum.ru/database/	Базы данных
https://dev.mysql.com/doc/	Электронная документация по MySQL
http://www.mysql.ru/docs/man/Reference.html	Электронная документация по MySQL на русском языке

8. Перечень информационных технологий

8.1. Перечень программного обеспечения, используемого при осуществлении образовательного процесса по дисциплине.

Перечень используемого программного обеспечения представлен в таблице 10.

Таблица 10– Перечень программного обеспечения

№ п/п	Наименование
1	Open ModelSphere
2	MySQL Community Server
3	Microsoft Word или OpenOffice Writer
4	Microsoft Visual Studio
5	MongoDB Community Server
6	PostgreSQL

8.2. Перечень информационно-справочных систем, используемых при осуществлении образовательного процесса по дисциплине

Перечень используемых информационно-справочных систем представлен в таблице 11.

Таблица 11– Перечень информационно-справочных систем

№ п/п	Наименование
	Не предусмотрено

9. Материально-техническая база

Состав материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине, представлен в таблице 12.

Таблица 12 – Состав материально-технической базы

№ п/п	Наименование составной части материально-технической базы	Номер аудитории (при необходимости)
1	Мультимедийная лекционная аудитория	—
2	Вычислительная лаборатория кафедры 43	Б.М. 23-08, 23-09, 23-10

10. Оценочные средства для проведения промежуточной аттестации

10.1. Состав оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине приведен в таблице 13.

Таблица 13 – Состав оценочных средств для проведения промежуточной аттестации

Вид промежуточной аттестации	Перечень оценочных средств
Экзамен	Список вопросов к экзамену; Задачи;
Зачет	Список вопросов;

10.2. В качестве критериев оценки уровня сформированности (освоения) компетенций обучающимися применяется 5-балльная шкала оценки сформированности компетенций, которая приведена в таблице 14. В течение семестра может использоваться 100-балльная шкала модульно-рейтинговой системы Университета, правила использования которой, установлены соответствующим локальным нормативным актом ГУАП.

Таблица 14 –Критерии оценки уровня сформированности компетенций

Оценка компетенции	Характеристика сформированных компетенций
5-балльная шкала	
«отлично» «зачтено»	<ul style="list-style-type: none"> – обучающийся глубоко и всесторонне усвоил программный материал; – уверенно, логично, последовательно и грамотно его излагает; – опираясь на знания основной и дополнительной литературы, тесно привязывает усвоенные научные положения с практической деятельностью направления; – умело обосновывает и аргументирует выдвигаемые им идеи; – делает выводы и обобщения; – свободно владеет системой специализированных понятий.

Оценка компетенции	Характеристика сформированных компетенций
5-балльная шкала	
«хорошо» «зачтено»	<ul style="list-style-type: none"> – обучающийся твердо усвоил программный материал, грамотно и по существу излагает его, опираясь на знания основной литературы; – не допускает существенных неточностей; – увязывает усвоенные знания с практической деятельностью направления; – аргументирует научные положения; – делает выводы и обобщения; – владеет системой специализированных понятий.
«удовлетворительно» «зачтено»	<ul style="list-style-type: none"> – обучающийся усвоил только основной программный материал, по существу излагает его, опираясь на знания только основной литературы; – допускает несущественные ошибки и неточности; – испытывает затруднения в практическом применении знаний направления; – слабо аргументирует научные положения; – затрудняется в формулировании выводов и обобщений; – частично владеет системой специализированных понятий.
«неудовлетворительно» «не зачтено»	<ul style="list-style-type: none"> – обучающийся не усвоил значительной части программного материала; – допускает существенные ошибки и неточности при рассмотрении проблем в конкретном направлении; – испытывает трудности в практическом применении знаний; – не может аргументировать научные положения; – не формулирует выводов и обобщений.

10.3. Типовые контрольные задания или иные материалы.

Вопросы (задачи) для экзамена представлены в таблице 15.

Таблица 15 – Вопросы (задачи) для экзамена

№ п/п	Перечень вопросов (задач) для экзамена
1	История причины появления СУБД. Роль СУБД в программных системах
2	История развития СУБД и модели данных.
3	Объектно-ориентированные и NoSQL модели данных.
4	Нормальные формы и денормализация
5	Отношения и их свойства, ключи отношений
6	Типы данных в языке SQL, оператор языка SQL для создания таблиц
7	Операторы языка SQL для удаления и модификации таблиц
8	Абстрактные операции манипулирования данными, их реализация в языке SQL
9	Булевы операции над отношениями
10	Операция выборки, ее свойства
11	Операция проекции, ее свойства
12	Операция соединения, ее свойства
13	Операция деления
14	Операция переименования атрибутов
15	Операции эквисоединения, θ -выборки и θ -соединения
16	Оператор выборки в языке SQL
17	Агрегатные функции в операторе выборки языка SQL
18	Объединение, пересечение, разность запросов в языке SQL
19	Запросы с подзапросами в языке SQL
20	Экзистенциальные запросы в языке SQL
21	Виды соединений в языке SQL
22	Трехзначная логика и обработка NULL-значений в языке SQL

23	Представления в языке SQL
24	Управляющие конструкции в языке SQL
25	Хранимые процедуры в языке SQL
26	Триггеры в языке SQL
27	Индексация данных
28	Защита информации в базах данных
29	Целостность баз данных и параллельный доступ
	Перечень задач для экзамена (выполняются на языке реляционной алгебры)
1	Создайте схему отношений для хранения следующих сведений: алфавитный каталог книг в библиотеке, читатель, формуляр читателя (выданные и возвращенные книги). Составьте запросы, позволяющие выбрать: читателей, которые брали книги на прошлой неделе; читателей, которые брали книги Ахо и Ульмана; количество книг, находящихся на руках у каждого из читателей; читателей, которые читают только книги жанра «приключения»; читателей, у которых на руках две или более книги одного автора; читателей, прочитавших более ста книг; читателей, взявших книги, которые больше никому не выдавались; читателей, читающих книги всех жанров; читателей, у которых на руках максимальное количество книг.
2	Создайте схему отношений для хранения следующих сведений: кафедра, преподаватель, дисциплина, группа, курс, вид занятия. Составьте запросы, позволяющие выбрать: преподавателей, ведущих Базы данных на различных факультетах; преподавателей, ведущих как Базы данных, так и Логическое программирование; количество дисциплин для каждого преподавателя; группы, у которых в среднем менее 4-х пар в неделю; преподавателей, которые ведут занятия только на старших курсах; преподавателей, которые ведут более двух видов занятий по одной дисциплине; преподавателей, ведущих более трех различных дисциплин; преподавателей, ведущих все виды занятий; преподавателей, ведущих занятия в максимальном количестве групп.
3	Создайте схему отношений для хранения следующих сведений: турфирма, тур, страна, турист, путевка. Составьте запросы, позволяющие выбрать: туристов, посещавших в прошлом году Италию и Францию; турфирмы, продающие туры в Египет и Турцию; туристов, побывавших только в одной стране; туристов, пользовавшихся услугами двух и более турфирм; количество путевок, проданных каждой из турфирм за прошлый год; среднюю цену путевки в Тунис; туристов, побывавших во Франции более пяти раз; туристов, побывавших во всех странах, в которые предлагаются туры; пары туристов, которые всегда путешествуют вместе.
4	Создайте схему отношений для хранения следующих сведений: студент, группа, дисциплина, лабораторная работа, рейтинг за сданную лабораторную работу. Составьте запросы, позволяющие выбрать: максимальный рейтинг, который может получить студент за работу №8 по БД; работы и рейтинги, сданные и полученные конкретным студентом; дисциплины, у которых есть лабораторные работы с одинаковыми названиями; количество работ, сданных каждым студентом по БД; студентов, у которых средний рейтинг за сданные лабораторные работы по БД превышает 4; студентов, не сдавших ни одной работы по БД; лабораторные по БД, которые нужно досдать Сыроежкину из группы 4000; студентов, получивших одинаковый рейтинг за все работы по БД; студентов, сдавших все работы по БД.
5	Создайте схему отношений для хранения следующих сведений: студент, группа, факультет, конференция, тема доклада. Составьте запросы, позволяющие выбрать: студентов первого факультета, выступавших на конференции Информатика; темы докладов студентов для заданной группы;

	<p>выступления, подготовленные двумя студентами различных факультетов; количество докладов для каждой конференции; среднее количество докладов, сделанных студентами третьего факультета на конференциях; студентов четвертого факультета не выступавших на конференциях; студентов, выступивших на трех или большем числе конференций; студентов, выступивших на всех конференциях; пары студентов, всегда выступающие вместе.</p>
6	<p>Создайте схему отношений для хранения следующих сведений: компьютерный магазин, модель компьютера, комплектующие. Составьте запросы, позволяющие выбрать: модели компьютеров, в которых используются винчестеры Samsung; модели компьютеров, в которых используются как накопители DVD, так и FDD; количество моделей, продаваемых в каждом из магазинов; модели компьютеров, не имеющие накопителей DVD; магазины, в которых средняя цена компьютера ниже, чем в других; магазины, в которых продается наибольшее количество моделей; магазины, в которых не продаются модели, укомплектованные одновременно оборудованием Intel и Samsung; модели компьютеров, укомплектованные всеми типами периферийных устройств; модели компьютеров, продающиеся во всех магазинах.</p>
7	<p>Создайте схему отношений для хранения следующих сведений: номер маршрута автобуса, остановка, транспортная компания. Составьте запросы, позволяющие выбрать: маршруты, выполняемые заданной компанией; маршруты, которыми можно доехать до Дворцовой площади; маршруты, имеющие общие остановки; количество маршрутов, обслуживаемых каждой компанией; компании, обслуживающие наибольшее число маршрутов; компании, средняя продолжительность маршрутов которых ниже чем у других; компании, маршруты которых не останавливаются на Дворцовой площади; компании, у которых нет маршрутов короче чем 10 остановок; маршруты, которые включают все остановки заданного маршрута.</p>
8	<p>Создайте схему отношений для хранения следующих сведений: театр, спектакль, жанр, актер. Составьте запросы, позволяющие выбрать: спектакли жанра комедия; спектакли, в которых занят заданный актер; спектакли, идущие более чем в одном театре; количество спектаклей для каждого из театров; театры, в которых количество драм превышает число комедий; спектакли, в которых занято наибольшее число актеров; спектакли одного актера; театры, в которых идут спектакли всех жанров; актеров, занятых только в одном театре.</p>
9	<p>Создайте схему отношений для хранения следующих сведений: аптека, медикамент, цена, производитель. Составьте запросы, позволяющие выбрать: аптеки, в которых есть лекарства заданного производителя; аптеки, в которых продается одно и то же лекарство различных производителей; цена аспирина в различных аптеках; количество наименований лекарств, продающихся в каждой из аптек; аптеки, в которых цена аспирина минимальна; средняя стоимость аспирина компании АБВ в аптеках; аптеки, в которых нет медикаментов, заданного производителя; пары производителей, у которых нет ни одного одинакового медикамента; аптеки, в которых есть все лекарства.</p>
10	<p>Создайте схему отношений для хранения следующих сведений: фильм, студия, жанр, актер. Составьте запросы, позволяющие выбрать: список фильмов, снятых заданной студией за заданный период; перечень студий, в фильмах которых играл заданный актер; актеров, снимавшихся как в комедиях, так и в мелодрамах; студии, на которых количество мелодрам превышает число комедий; актеров, снявшихся в десяти фильмах; среднее количество фильмов каждого из жанров, снимающееся на студии; студии, на которых снимаются фильмы только одного жанра; студии, на которых снимаются фильмы всех</p>

	жанров; студии, никогда не выпускавшие ремейков.
11	Создайте схему отношений для хранения следующих сведений: фильм, студия, жанр, актер. Составьте запросы, позволяющие выбрать: список фильмов, снятых заданной студией за заданный период; перечень студий, в фильмах которых играл заданный актер; актеров, снимавшихся как в комедиях, так и в мелодрамах; студии, на которых количество мелодрам превышает число комедий; актеров, снявшихся в десяти фильмах; среднее количество фильмов каждого из жанров, снимающееся на студии; студии, на которых снимаются фильмы только одного жанра; студии, на которых снимаются фильмы всех жанров; студии, никогда не выпускавшие ремейков.
12	Создайте схему отношений для хранения следующих сведений: подразделение, сотрудник, должность, дети. Составьте запросы, позволяющие выбрать: список сотрудников заданного подразделения; подразделения, входящие в состав заданного; сотрудников, у которых есть дети различного пола; среднюю численность подразделений; инженеров, у которых более пяти детей; подразделения, в которых количество техников превышает количество инженеров; подразделения, в которых не работают совместители; подразделения, в которых представлены все должности; сотрудников, у которых все дети одного пола.
13	Создайте схему отношений для хранения следующих сведений: издательство, автор, книга, жанр. Составьте запросы, позволяющие выбрать: перечень книг, выпущенных заданным издательством в прошлом году; авторы, сотрудничающие с несколькими издательствами; книги, написанные в соавторстве; количество книг каждого жанра, выпущенных каждым издательством; авторов, написавших наибольшее количество книг; средний объем книг, выпускаемых заданным издательством; издательства, выпускающие только сказки и детективы; издательства, выпускающие книги всех жанров; издательства, не выпустившие ни одной книги в 2009 году.
14	Создайте схему отношений для хранения следующих сведений: врач, специальность, пациент, прием. Составьте запросы, позволяющие выбрать: список пациентов, принятых терапевтами вчера; врачей, совмещающих различные специальности; пациентов, посещавших и хирурга и кардиолога; количество пациентов, принятых каждым из врачей за прошедший год; врачи, принявшие меньше всего пациентов; врачей, у которых количество принимаемых пациентов превышает среднее; пациентов, которые никогда не посещали хирурга; пациентов, которые посетили всех специалистов; врачи, не совмещающие работу по различным специальностям.
15	Создайте схему отношений для хранения следующих сведений: группа, пара, аудитория, корпус. Составьте запросы, позволяющие выбрать: перечень занятий для заданной группы на завтра; списки аудиторий по корпусам; группы, у которых в один день есть и лабораторные и курсовое проектирование; группы, у которых количество занятий во вторник превышает семь пар; среднее количество пар для заданной группы; аудитории, в которых занимается максимальное количество групп; группы, у которых во вторник все пары проходят в одном корпусе; аудитории, в которых никогда не занимаются студенты четвертого факультета; группы, у которых нет занятий по субботам.
16	Создайте схему отношений для хранения следующих сведений: кафедра, преподаватель, ученая степень, ученое звание, должность. Составьте запросы, позволяющие выбрать: список преподавателей заданной кафедры; преподавателей, совмещающих работу на нескольких кафедрах; кафедры, на которых работают кандидаты наук, не имеющие звания доцента; количество преподавателей, занимающих одинаковые должности для каждой из кафедр;

	среднее число сотрудников на кафедрах; кафедры, на которых число докторов наук больше чем число кандидатов; кафедры, на которых нет ни одного профессора; кафедры, на которых преподаватели занимают все возможные должности; кафедры, на которых представлены не все ученые степени.
17	Создайте схему отношений для хранения следующих сведений: авиакомпания, авиарейс, тип самолета. Составьте запросы, позволяющие выбрать: список рейсов для заданной авиакомпании; типы самолетов, используемые заданной авиакомпанией; авиакомпании, у которых прямой и обратный рейс выполняют различные типы самолетов; направления, на которых работает более трех авиакомпаний; количество авиарейсов, выполняемых между каждой парой аэропортов; авиакомпании, выполняющие максимальное количество рейсов; авиакомпании, не работающие в Стамбуле; авиакомпании, использующие все типы самолетов; авиакомпании, у которых все самолеты одного производителя.
18	Создайте схему отношений для хранения следующих сведений: город, район, квартира, цена. Составьте запросы, позволяющие выбрать: перечень однокомнатных квартир, продаваемых в Московском районе; квартиры, находящиеся на одной улице, но в различных районах; двух- и трехкомнатные квартиры, имеющие одинаковую площадь; средняя цена однокомнатной квартиры в городе; районы, в которых продается наибольшее число объектов недвижимости; районы, в которых минимальна стоимость квадратного метра; улицы, продолжительность которых ограничивается только одним районом; районы, в которых не продаются однокомнатные квартиры; районы, в которых продаются квартиры всех строительных серий.
19	Создайте схему отношений для хранения следующих сведений: олимпиада, страна, спортсмен, вид спорта, место. Составьте запросы, позволяющие выбрать: страны, принявшие участие в зимней олимпиаде 2008 г.; спортсменов, принявших участие как в летних, так и в зимних олимпиадах; спортсменов, получивших золото по двум или более видам соревнований на одной и той же олимпиаде; среднее число спортсменов, выставяемых каждой страной на каждый из видов олимпиад; страны, количество побед которых на зимних олимпиадах, превышает количество побед на летних; страны, завоевавшие наибольшее количество наград в 2008 г.; страны, никогда не участвовавшие в зимних олимпиадах; страны, не участвовавшие в олимпиадах в период 1991..2011 г.г.; страны, завоевавшие призовые места по всем видам спорта.
20	Создайте схему отношений для хранения следующих сведений: город, ж/д станция, поезд, вагон, место, пассажир, билет. Составьте запросы, позволяющие выбрать: пассажиров, купивших билеты в прошлом месяце из СПб в Москву; пассажиров, купивших в течение месяца и прямые и обратные билеты; поезда, в которых есть и купейные и плацкартные и сидячие вагоны; количество поездов из СПб в Москву; количество билетов, проданных на каждый поезд из СПб в Москву; средняя цена места в купейном вагоне; поезда из СПб, делающие остановку в Окуловке и не делающие в Бологом; поезда, у которых соотношение цены СВ и продолжительности в пути максимально; поезда из СПб в Москву, делающие все остановки.
21	Создайте схему отношений для хранения следующих сведений: оператор сотовой связи, абонент, договор, услуги, тарифы, .опции, начисления, платежи. Составьте запросы, позволяющие выбрать: абонентов, пользующихся опцией АОН по какому-либо тарифу; абонентов, поменявших в прошлом году тариф «нормальный» на тариф «оптимальный»; тарифы, у которых есть одинаковые опции; количество абонентов, пользующихся каждым из тарифов; операторов, у которых средняя цена минуты выше, чем у других; операторов, доходность которых выше чем у АБВ; абонентов, не осуществлявших платежей в прошлом

	квартале; тарифы, включающие все возможные опции; абонентов, которые всегда изменяли тарифы одновременно (в один день).
22	Создайте схему отношений для хранения следующих сведений: Интернет-провайдер, абонент, договор, оборудование, обращение в техподдержку, результат. Составьте запросы, позволяющие выбрать: абонентов, обращавшихся в техподдержку по вопросу неисправности СМ440; абонентов, обращавшихся в техподдержку дважды в прошлом месяце; провайдеров, которые предоставляют доступ по тарифам как на скорости 50, так 70 МБит; провайдеров, предлагающих более семи моделей кабельных модемов; провайдеров, предлагающих наибольшее число тарифов; количество абонентов для каждого провайдера по каждому из тарифов; абонентов, никогда не изменявших тариф; абонентов, пользовавшихся всеми тарифами; абонентов, не обращавшихся в техподдержку более трех раз в год.
23	Создайте схему отношений для хранения следующих сведений: сеть ресторанов, ресторан, меню, состав блюд, бронирование столиков, заказы клиентов. Составьте запросы, позволяющие выбрать: блюда, в состав которых входит говядина; блюда, в состав которых входят одинаковые ингредиенты; рестораны сети, в которых одинаковые блюда имеют различную цену; количество блюд, предлагаемых в сети АБВ; количество блюд, в которые входит каждый из ингредиентов; рестораны, предлагающие в точности столько же блюд, что и заданный; рестораны, в которых нет чека, размер которого превышает 20000; рестораны, в которых средний размер чека минимален; рестораны, выручка которых год от года возрастает.
24	Создайте схему отношений для хранения следующих сведений: почта, письмо, заказное письмо, бандероль, отправитель, получатель, прием/выдача корреспонденции. Составьте запросы, позволяющие выбрать: людей, отправлявших заказные письма в прошлом месяце; людей, отправлявших письма в прошлом месяце дважды по одному и тому же адресу; людей, отправлявших письма в прошлом месяце и получавших ответы; количество писем, пересланных из СПб в Москву в прошлом году; количество корреспонденции каждого из видов между СПб и Москвой; средний вес бандеролей из Москвы в СПб; почтовые отделения, количество корреспонденции в которых больше чем в других; людей, отправляющих письма всегда из одного и того же отделения; людей, отправивших в прошлом году все виды корреспонденции.

Вопросы (задачи) для зачета / дифф. зачета представлены в таблице 16.

Таблица 16 – Вопросы (задачи) для зачета / дифф. зачета

№ п/п	Перечень вопросов (задач) для зачета / дифф. зачета
1	Выбор типа СУБД применительно к особенностям предметной области
2	Объектные базы данных. Проектирование и создание
3	Объектные базы данных. Манипуляции данными
4	Объектно-реляционные базы данных. Проектирование и создание
5	Объектно-реляционные базы данных. Пользовательские типы и операторы
6	Базы данных «Ключ-значение»
7	Структура и создание документных базы данных.
8	Манипуляции данными и поиск в документных базах данных.
9	Графовые базы данных : графы и операции CRUD
10	Базы данных типа «Семейство столбцов». Операции CRUD и администрирование таблиц
11	Базы данных типа «Семейство столбцов». работа с «большими данными»

Перечень тем для курсового проектирования/выполнения курсовой работы представлены в таблице 17.

Таблица 17 – Перечень тем для курсового проектирования/выполнения курсовой работы

№ п/п	Примерный перечень тем для курсового проектирования/выполнения курсовой работы
	Учебным планом не предусмотрено

Вопросы для проведения промежуточной аттестации в виде тестирования представлены в таблице 18.

Таблица 18 – Примерный перечень вопросов для тестов

№ п/п	Примерный перечень вопросов для тестов

Перечень тем контрольных работ по дисциплине обучающихся заочной формы обучения, представлены в таблице 19.

Таблица 19 – Перечень контрольных работ

№ п/п	Перечень контрольных работ
1	Проектирование взаимодействия базы данных и приложения (6 семестр)
1.1	<p>Создать для базы данных программы для рисования графов приложение с двумя способами подсоединения базы данных:</p> <p>1) Соединение через компоненты ADO.NET или его аналога с помощью строки связи. При этом отображаться должны 2 таблицы/элемента данных (не обязательно в форме таблицы, может быть просто текст): одна через DataGridView ,с подключением источника данных, а другая через чтение результатов запроса SqlCommand в цикле.</p> <p>2) С использованием технологии ORM (Object-Relational Mapping) с отображение третьей таблицы /элемента данных</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. Вершины, название/текст которых содержит слово «конец», но не заканчивается им б. Вершины, у которых нет исходящих ребер в. Графы, в которых есть пара вершин, связанных ребрами в обе стороны г. Ширина графа в пикселях (от максимальной сумма координаты по горизонтали с шириной отнять минимальную левую координату) д. Пользователь- автор, графов с максимальных количеством вершин е. Вершины, для которых есть исходящие ребра, ведущие ко всем остальным вершинам ж. Вершина, нет входящих ребер от вершины 2
1.2	<p>Создать для базы данных Садоводства приложение с двумя способами подсоединения базы данных: 1) Соединение через компоненты ADO.NET или его аналога с помощью строки связи. При этом отображаться должны 2 таблицы/элемента данных (не обязательно в форме таблицы, может быть просто текст): одна через DataGridView ,с подключением источника данных, а другая через чтение результатов запроса SqlCommand в цикле.</p> <p>2) С использованием технологии ORM (Object-Relational Mapping) с отображение третьей таблицы /элемента данных</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. номера участков владельцев с отчеством, заканчивающимся на «вна» б. участки, на которых зарегистрировано более1 постройки в. тип взносов, которые пока никто не оплатил г. Владелец (владельцы) участка максимальной площади

	<ul style="list-style-type: none"> д. Владельцы максимального количества участков е. Участки, на которых есть постройки всех типов ж. Участки, на которых нет бань
1.3	<p>Создать для базы данных ТСЖ приложение с двумя способами подсоединения базы данных: 1)Соединение через компоненты ADO.NET или его аналога с помощью строки связи. При этом отображаться должны 2 таблицы/элемента данных (не обязательно в форме таблицы, может быть просто текст): одна через DataGridView ,с подключением источника данных, а другая через чтение результатов запроса SqlCommand в цикле.</p> <p>2) С использованием технологии ORM (Object-Relational Mapping) с отображение третьей таблицы /элемента данных</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. квартиры владельцев, отчества которых заканчиваются на 'ич' б. владельцев, у которых есть квартиры на разных этажах в. квартиры, в которых не записано ни одного показания счетчика газа г. владельцы квартир с минимальной жилой площадью д. этаж, на котором меньше всего квартир находится в собственности е. квартиры, в которых стоят все типы счетчиков ж. владелец, у которого нет трехкомнатных квартир
1.4	<p>Создать для базы данных парка приложение с двумя способами подсоединения базы данных: 1)Соединение через компоненты ADO.NET или его аналога с помощью строки связи. При этом отображаться должны 2 таблицы/элемента данных (не обязательно в форме таблицы, может быть просто текст): одна через DataGridView ,с подключением источника данных, а другая через чтение результатов запроса SqlCommand в цикле.</p> <p>2) С использованием технологии ORM (Object-Relational Mapping) с отображение третьей таблицы /элемента данных</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. аллеи, на которых встречаются разные виды кленов (клен в названии) б. аллеи, состоящие из разных пород деревьев в. породы, не высаженные в парке г. дерево, которое было посажено раньше всех д. порода, деревьев которой меньше всего е. аллея, на которой растут деревья всех пород, которые есть в парке ж. аллея, деревья на которой не высаживались в прошлом году
1.5	<p>Создать для базы данных расписания экзаменов/зачетов приложение с двумя способами подсоединения базы данных: 1)Соединение через компоненты ADO.NET или его аналога с помощью строки связи. При этом отображаться должны 2 таблицы/элемента данных (не обязательно в форме таблицы, может быть просто текст): одна через DataGridView ,с подключением источника данных, а другая через чтение результатов запроса SqlCommand в цикле.</p> <p>2) С использованием технологии ORM (Object-Relational Mapping) с отображение третьей таблицы /элемента данных</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. аудитории, которых проходят экзамены по дисциплинам, имеющим в названии слова «базы данных» б. аудитории, где в один день проходит несколько экзаменов в. преподаватели, не принимающие экзаменов г. дисциплины самого последнего экзамена д. аудитории, в которых проходит больше всего экзаменов

	<p>е. дисциплина, по которой есть экзамены у всех групп</p> <p>ж. преподаватель, не принимающий экзамены у группы Z9431</p>
1.6	<p>Создать для базы данных поликлиники приложение с двумя способами подсоединения базы данных: 1)Соединение через компоненты ADO.NET или его аналога с помощью строки связи. При этом отображаться должны 2 таблицы/элемента данных (не обязательно в форме таблицы, может быть просто текст): одна через DataGridView ,с подключением источника данных, а другая через чтение результатов запроса SqlCommand в цикле.</p> <p>2) С использованием технологии ORM (Object-Relational Mapping) с отображение третьей таблицы /элемента данных</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. пациенты, приходившие на любые процедуры, связанные с электрофорезом б. пациент, приходивший к одному врачу и на прием и на процедуру в. пациенты, не ходившие в поликлинику г. процедуры с наименьшей стоимостью д. пациент, ходивший к наибольшему количеству врачей е. пациент, ходивший на все процедуры ж. пациент, не ходивший на процедуры к Иванову Ивану Ивановичу
1.7	<p>Создать для базы данных личного кабинета студента приложение с двумя способами подсоединения базы данных: 1)Соединение через компоненты ADO.NET или его аналога с помощью строки связи. При этом отображаться должны 2 таблицы/элемента данных (не обязательно в форме таблицы, может быть просто текст): одна через DataGridView ,с подключением источника данных, а другая через чтение результатов запроса SqlCommand в цикле.</p> <p>2) С использованием технологии ORM (Object-Relational Mapping) с отображение третьей таблицы /элемента данных</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. преподаватели, аз которыми закреплены дисциплины, начинающиеся со слова «автоматизирован» б. дисциплина, по которой есть и лабораторные и курсовая работа в. студенты, не прикрепившие ни одной работы г. студент, сдавший курсовую раньше всех д. студент, прикрепивший в этом месяце наибольшее число работ е. дисциплина, по которой есть все типы работ (КР, ЛР, практические) ж. преподаватель, которому не прикрепляли отчетов по курсовому проекту или работе
1.8	<p>Создать для базы данных костюмерной театра приложение с двумя способами подсоединения базы данных: 1)Соединение через компоненты ADO.NET или его аналога с помощью строки связи. При этом отображаться должны 2 таблицы/элемента данных (не обязательно в форме таблицы, может быть просто текст): одна через DataGridView ,с подключением источника данных, а другая через чтение результатов запроса SqlCommand в цикле.</p> <p>2) С использованием технологии ORM (Object-Relational Mapping) с отображение третьей таблицы /элемента данных</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. спектакли, в которых используются костюмы, имеющие в названии слово шут б. костюм, в котором есть и куртка и штаны в. роль, на которую пока нет костюма г. автор, разработавший самый старый из костюмов д. спектаклю, к которому разработано наибольшее число костюмов е. костюм, в котором есть все типы деталей

	ж. автор, не разрабатывавший костюмы к «Золушке»
1.9	<p>Создать для базы данных охраняемых парковок приложение с двумя способами подсоединения базы данных: 1)Соединение через компоненты ADO.NET или его аналога с помощью строки связи. При этом отображаться должны 2 таблицы/элемента данных (не обязательно в форме таблицы, может быть просто текст): одна через DataGridView ,с подключением источника данных, а другая через чтение результатов запроса SqlCommand в цикле.</p> <p>2) С использованием технологии ORM (Object-Relational Mapping) с отображение третьей таблицы /элемента данных</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. все парковки, расположенные на линиях (не улицах или проспектах)(улица в адресе содержит «линия») б. владелец машины, у которого более одного места под машину в. машины, которые неизвестно когда уедут г. владелец машин, заезжавший раньше всех д. владелец машины, останавливавшийся на минимальном числе парковок е. машина, которая стояла на всех парковках Московского проспекта ж. владелец, не парковавшийся на Невском проспекте
1.10	<p>Создать для базы данных спортзала приложение с двумя способами подсоединения базы данных: 1)Соединение через компоненты ADO.NET или его аналога с помощью строки связи. При этом отображаться должны 2 таблицы/элемента данных (не обязательно в форме таблицы, может быть просто текст): одна через DataGridView ,с подключением источника данных, а другая через чтение результатов запроса SqlCommand в цикле.</p> <p>2) С использованием технологии ORM (Object-Relational Mapping) с отображение третьей таблицы /элемента данных</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. виды спорта, на которые есть безлимитные абонементы (со словом безлимит в названии типа) б. вид спорта, по которому занятия ведут различные тренеры в. вид спорта, по которому не куплено ни одного абонемента г. тренеры, ведущие занятия по виду спорта с минимальным число занятий в абонементе д. тренеры, которые ведут занятия по максимальному количеству видов спорта е. тренеры, не ведущие фитнес ж. абонемент, на все виды спорта, связанные с единоборствами, заканчивающимися на «до»
1.11	<p>Создать для базы данных магазина обоев приложение с двумя способами подсоединения базы данных: 1)Соединение через компоненты ADO.NET или его аналога с помощью строки связи. При этом отображаться должны 2 таблицы/элемента данных (не обязательно в форме таблицы, может быть просто текст): одна через DataGridView ,с подключением источника данных, а другая через чтение результатов запроса SqlCommand в цикле.</p> <p>2) С использованием технологии ORM (Object-Relational Mapping) с отображение третьей таблицы /элемента данных</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. коллекция обоев, начинающаяся со слова Элегия б. производитель, который производит как бумажные, так и флизелиновые обои в. текстура, обоев с которой нет г. производитель с самой большой шириной рулона д. производитель обоев с максимальным количеством коллекций е. производитель, который производит обои из всех материалов

	ж. производитель, у которого нет обоев дороже 3000 рублей
1.12	<p>Создать для базы данных расписания пригородных электричек приложение с двумя способами подсоединения базы данных: 1)Соединение через компоненты ADO.NET или его аналога с помощью строки связи. При этом отображаться должны 2 таблицы/элемента данных (не обязательно в форме таблицы, может быть просто текст): одна через DataGridView ,с подключением источника данных, а другая через чтение результатов запроса SqlCommand в цикле.</p> <p>2) С использованием технологии ORM (Object-Relational Mapping) с отображение третьей таблицы /элемента данных</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. все маршруты, проходящие через остановку , в названии которой есть слово село (царское село, детское село) б. свежестроенная станция, к которой пока не ведет ни один маршрут в. маршрут, проходящий через Купчино и Шушары г. самая поздняя электричка, прибывающая на витебский вокзал д. маршруты с наибольшим количеством электричек е. маршрут, на котором нет электричек с режимом отправления по выходным. ж. маршрут, по которому ходят электрички со всеми режимами движения
1.13	<p>Создать для вакансий базы данных приложение с двумя способами подсоединения базы данных: 1)Соединение через компоненты ADO.NET или его аналога с помощью строки связи. При этом отображаться должны 2 таблицы/элемента данных (не обязательно в форме таблицы, может быть просто текст): одна через DataGridView ,с подключением источника данных, а другая через чтение результатов запроса SqlCommand в цикле.</p> <p>2) С использованием технологии ORM (Object-Relational Mapping) с отображение третьей таблицы /элемента данных</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. вакансии, имеющие в названии SQL б. подозрительные вакансии, где не прописаны обязанности в. работодатели в Санкт-Петербурге, выставившие вакансии программиста и системного администратора г. вакансия с наименьшей зарплатой д. вакансии с минимальным количеством обязательных требований е. требование, присутствующее во всех вакансиях на программиста ж. вакансии, в которых нет требования к опыту работы
1.14	<p>Создать для базы данных калькулятора бюджета физического лица приложение с двумя способами подсоединения базы данных: 1)Соединение через компоненты ADO.NET или его аналога с помощью строки связи. При этом отображаться должны 2 таблицы/элемента данных (не обязательно в форме таблицы, может быть просто текст): одна через DataGridView ,с подключением источника данных, а другая через чтение результатов запроса SqlCommand в цикле.</p> <p>2) С использованием технологии ORM (Object-Relational Mapping) с отображение третьей таблицы /элемента данных</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. все расходы категорий, которые относятся к спорту (содержат слово спорт) б. месяц, в котором были разные статьи дохода в. статья доходов, по которой доходов не было г. категория, по которой наибольшие расходы в текущем году д. категория, по которой не было расходов в январе е. месяц, в котором были траты всех категорий расхода

	ж. месяц, в котором были траты максимального количества статей
1.15	<p>Создать для базы данных книги контактов приложение с двумя способами подсоединения базы данных: 1)Соединение через компоненты ADO.NET или его аналога с помощью строки связи. При этом отображаться должны 2 таблицы/элемента данных (не обязательно в форме таблицы, может быть просто текст): одна через DataGridView ,с подключением источника данных, а другая через чтение результатов запроса SqlCommand в цикле.</p> <p>2) С использованием технологии ORM (Object-Relational Mapping) с отображение третьей таблицы /элемента данных</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. друзья, у которых номер телефона начинается на 8-911, б. Люди без меток, в. люди, которые относятся к соседям и коллегам одновременно г. самые старые люди среди контактов д. месяц, когда есть дни рождения у соседей, но нет у семьи е. метки, к которым относится максимальное количество людей ж. месяц, в котором есть дни рождения у людей со всеми метками
1.16	<p>Создать для базы данных вузов для абитуриентов приложение с двумя способами подсоединения базы данных: 1)Соединение через компоненты ADO.NET или его аналога с помощью строки связи. При этом отображаться должны 2 таблицы/элемента данных (не обязательно в форме таблицы, может быть просто текст): одна через DataGridView ,с подключением источника данных, а другая через чтение результатов запроса SqlCommand в цикле.</p> <p>2) С использованием технологии ORM (Object-Relational Mapping) с отображение третьей таблицы /элемента данных</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. направленности, в которых есть слово автоматизированный б. факультет, не принимающий ни на одно направление в. направление, на которое надо сдавать историю и обществознание г. факультет вуза, принимающий на наименьшее количество направлений д. город, в котором есть все направления е. вуз, с самым ранним началом приемной компании ж. направление, на которое не надо сдавать ЕГЭ по информатике
1.17	<p>Создать для базы данных школьных экскурсий приложение с двумя способами подсоединения базы данных: 1)Соединение через компоненты ADO.NET или его аналога с помощью строки связи. При этом отображаться должны 2 таблицы/элемента данных (не обязательно в форме таблицы, может быть просто текст): одна через DataGridView ,с подключением источника данных, а другая через чтение результатов запроса SqlCommand в цикле.</p> <p>2) С использованием технологии ORM (Object-Relational Mapping) с отображение третьей таблицы /элемента данных</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. экскурсии в музеи (слово музей в любом месте названия) б. экскурсии, относящиеся к литературе и истории в. учитель, который не отвечает за экскурсии г. учащиеся, которые не ездили в музей истории религии д. экскурсия, собравшая наибольшее число участников е. самые дорогие экскурсии ж. учащийся, который был на всех экскурсиях
1.18	Создать для базы данных собачьего питомника приложение с двумя способами

	<p>подсоединения базы данных:</p> <p>1)Соединение через компоненты ADO.NET или его аналога с помощью строки связи. При этом отображаться должны 2 таблицы/элемента данных (не обязательно в форме таблицы, может быть просто текст): одна через DataGridView ,с подключением источника данных, а другая через чтение результатов запроса SqlCommand в цикле.</p> <p>2) С использованием технологии ORM (Object-Relational Mapping) с отображение третьей таблицы /элемента данных</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. породы, названия которых содержат слово «шпиц», но не начинается с него б. собаки без медалей в. собаки, у родителей которых один и тот же владелец г. владелец, у которого есть собаки всех пород д. владелец, у которого есть йоркширские терьеры, но нет мастиффов е. порода, собак которой меньше всего ж. владельцы самых старых собак
1.19	<p>Создать для базы данных службы доставки приложение с двумя способами подсоединения базы данных:</p> <p>1)Соединение через компоненты ADO.NET или его аналога с помощью строки связи. При этом отображаться должны 2 таблицы/элемента данных (не обязательно в форме таблицы, может быть просто текст): одна через DataGridView ,с подключением источника данных, а другая через чтение результатов запроса SqlCommand в цикле.</p> <p>2) С использованием технологии ORM (Object-Relational Mapping) с отображение третьей таблицы /элемента данных</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. все посылки, отправляемые на улицу, в названии которой есть окончание «ая», но это не единственные буквы в нем б. улица, на которую никогда не доставляли посылки в. контактное лицо, получавшее посылки по разным адресам г. адреса, по которым доставили больше всего посылок д. посылка с самым маленьким весом е. контактное лицо, получавшее посылки от всех фирм ж. контактное лицо, никогда не получавшее посылки в январе
1.20	<p>Создать для базы данных туристического путеводителя приложение с двумя способами подсоединения базы данных:</p> <p>1)Соединение через компоненты ADO.NET или его аналога с помощью строки связи. При этом отображаться должны 2 таблицы/элемента данных (не обязательно в форме таблицы, может быть просто текст): одна через DataGridView ,с подключением источника данных, а другая через чтение результатов запроса SqlCommand в цикле.</p> <p>2) С использованием технологии ORM (Object-Relational Mapping) с отображение третьей таблицы /элемента данных</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. достопримечательности, в которых есть слово «мать», но с него название не начинается б. город без достопримечательностей в. улица, на которой есть и памятники, и архитектурные комплексы г. город, в котором нет природных комплексов д. улица, на которой больше всего памятников е. города с самыми старыми достопримечательностями

	ж. улица, на которой есть достопримечательности всех типов
1.21	<p>Создать для базы данных метрополитена приложение с двумя способами подсоединения базы данных:</p> <p>1)Соединение через компоненты ADO.NET или его аналога с помощью строки связи. При этом отображаться должны 2 таблицы/элемента данных (не обязательно в форме таблицы, может быть просто текст): одна через DataGridView ,с подключением источника данных, а другая через чтение результатов запроса SqlCommand в цикле.</p> <p>2) С использованием технологии ORM (Object-Relational Mapping) с отображение третьей таблицы /элемента данных</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. станции, в названии которых есть слово «площадь», но оно на него не заканчивается а. линии без станций (в проекте) б. пересадочные станции с линии 1 на линию 2 в. линия с самым большим временем проезда г. линия, на которой собраны все стации, открываются раньше 7 утра д. станция с самым ранним открытием е. линия на которой нет перегона больше 3 минут
1.22	<p>Создать для базы данных медпункта организации приложение с двумя способами подсоединения базы данных:</p> <p>1)Соединение через компоненты ADO.NET или его аналога с помощью строки связи. При этом отображаться должны 2 таблицы/элемента данных (не обязательно в форме таблицы, может быть просто текст): одна через DataGridView ,с подключением источника данных, а другая через чтение результатов запроса SqlCommand в цикле.</p> <p>2) С использованием технологии ORM (Object-Relational Mapping) с отображение третьей таблицы /элемента данных</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. все сотрудники, женского пола (отчество, оканчивается на «на») б. сотрудники, никогда не обращавшиеся в медпункт с жалобами в. сотрудники организации, делавшие за последний год прививки от столбняка и кори г. сотрудник, последним сделавший флюорографию д. сотрудник, у которого сделаны все типы прививок е. сотрудник с самым большим количеством жалоб ж. сотрудник, не делавший прививки от энцефалита
1.23	<p>Создать для базы данных по материальной ответственности приложение с двумя способами подсоединения базы данных:</p> <p>1)Соединение через компоненты ADO.NET или его аналога с помощью строки связи. При этом отображаться должны 2 таблицы/элемента данных (не обязательно в форме таблицы, может быть просто текст): одна через DataGridView ,с подключением источника данных, а другая через чтение результатов запроса SqlCommand в цикле.</p> <p>2) С использованием технологии ORM (Object-Relational Mapping) с отображение третьей таблицы /элемента данных</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. любые помещения, где стоит компьютерная мебель (категория мебель и слово «комп») б. помещение, в котором нет никакого имущества в. материально ответственное лицо, у которого подотчетна и компьютерная техника и специальное оборудование г. имущество, списанное последним

	<p>д. помещение, в котором есть имущество всех категорий</p> <p>е. мат. ответственное лицо, ответственное за наибольшее количество имущества</p> <p>ж. помещение, в котором нет компьютерной техники</p>
1.24	<p>Создать для базы данных календаря корпоративных мероприятий приложение с двумя способами подсоединения базы данных:</p> <p>1)Соединение через компоненты ADO.NET или его аналога с помощью строки связи. При этом отображаться должны 2 таблицы/элемента данных (не обязательно в форме таблицы, может быть просто текст): одна через DataGridView ,с подключением источника данных, а другая через чтение результатов запроса SqlCommand в цикле.</p> <p>2) С использованием технологии ORM (Object-Relational Mapping) с отображение третьей таблицы /элемента данных</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. Все мероприятия, текущего года, в названии которых есть слово конференция, но с него с названия не начинаются б. Комната, в которой не проходят мероприятия в. Мероприятия, в которых участвуют сотрудники «подразделения 1» и «подразделения 2» г. Участники самых поздно заканчивающихся мероприятий д. Событие, с самым большим количеством пришедших на него участников.(участники на мероприятие могут быть приглашены и при этом могут придти/не придти). е. Мероприятие, в котором обязательно присутствие всех начальников отделов ж. Сотрудник, не посещавший мероприятий в июле прошлого года
1.25	<p>Создать для базы данных домашних обязанностей приложение с двумя способами подсоединения базы данных:</p> <p>1)Соединение через компоненты ADO.NET или его аналога с помощью строки связи. При этом отображаться должны 2 таблицы/элемента данных (не обязательно в форме таблицы, может быть просто текст): одна через DataGridView ,с подключением источника данных, а другая через чтение результатов запроса SqlCommand в цикле.</p> <p>2) С использованием технологии ORM (Object-Relational Mapping) с отображение третьей таблицы /элемента данных</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. Люди, выполняющие обязанности, имеющие в названии слово уборка, но не заканчивающиеся на него б. Обязанность, которую еще не распределили по выполняющим людям в. Обязанность, которую выполняет более одного человека г. Ежемесячная обязанность, которая выполняется первой по дате (дню) д. Категория, обязанности из которой выполняют все е. Категория, в которой меньше всего обязанностей ж. Человек, который не выполняет обязанности из категории «оплата счетов»
1.26	<p>Создать для базы данных рабочих задач приложение с двумя способами подсоединения базы данных:</p> <p>1)Соединение через компоненты ADO.NET или его аналога с помощью строки связи. При этом отображаться должны 2 таблицы/элемента данных (не обязательно в форме таблицы, может быть просто текст): одна через DataGridView ,с подключением источника данных, а другая через чтение результатов запроса SqlCommand в цикле.</p> <p>2) С использованием технологии ORM (Object-Relational Mapping) с отображение третьей таблицы /элемента данных</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p>

	<ul style="list-style-type: none"> а. Задачи, в названии которых содержится слово «интеграция», но оно не последнее б. Проект без задач в. Задача, относящаяся к 2 различным проектам г. Задача с самым поздним сроком окончания д. Проект с самым большим количеством задач е. Человек, выполняющий задачи всех приоритетов ж. Человек, у которого нет незавершенных задач
2	Триггеры. Обеспечение активной целостности данных базы данных (7 семестр)
2.1	<p>Реализовать для базы данных программы для рисования графов триггеры для всех событий (insert,delete, update) до и после. Часть из которых будет обеспечивать ссылочную целостность, остальные могут иметь другое назначение. Вычисляемые поля можно добавить при необходимости.</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. Вершины, название/текст которых содержит слово «конец», но не заканчивается им б. Вершины, у которых нет исходящих ребер в. Графы, в которых есть пара вершин, связанных ребрами в обе стороны г. Ширина графа в пикселях (от максимальной сумма координаты по горизонтали с шириной отнять минимальную левую координату) д. Пользователь- автор, графов с максимальных количеством вершин е. Вершины, для которых есть исходящие ребра, ведущие ко всем остальным вершинам ж. Вершина, нет входящих ребер от вершины 2
2.2	<p>Реализовать для базы данных Садоводства триггеры для всех событий (insert,delete, update) до и после. Часть из которых будет обеспечивать ссылочную целостность, остальные могут иметь другое назначение. Вычисляемые поля можно добавить при необходимости.</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. номера участков владельцев с отчеством, заканчивающимся на «вна» б. участки, на которых зарегистрировано более1 постройки в. тип взносов, которые пока никто не оплатил г. Владелец (владельцы) участка максимальной площади д. Владельцы максимального количества участков е. Участки, на которых есть постройки всех типов ж. Участки, на которых нет бань
2.3	<p>Реализовать для базы данных ТСЖ триггеры для всех событий (insert,delete, update) до и после. Часть из которых будет обеспечивать ссылочную целостность, остальные могут иметь другое назначение. Вычисляемые поля можно добавить при необходимости.</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. квартиры владельцев, отчества которых заканчиваются на 'ич' б. владельцев, у которых есть квартиры на разных этажах в. квартиры, в которых не записано ни одного показания счетчика газа г. владельцы квартир с минимальной жилой площадью д. этаж, на котором меньше всего квартир находится в собственности е. квартиры, в которых стоят все типы счетчиков ж. владелец, у которого нет трехкомнатных квартир
2.4	<p>Реализовать для базы данных парка триггеры для всех событий (insert,delete, update) до и после. Часть из которых будет обеспечивать ссылочную целостность, остальные могут иметь другое назначение. Вычисляемые поля можно добавить</p>

	<p>при необходимости.</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ol style="list-style-type: none"> а. аллеи, на которых встречаются разные виды кленов (клен в названии) б. аллеи, состоящие из разных пород деревьев в. породы, не высаженные в парке г. дерево, которое было посажено раньше всех д. порода, деревьев которой меньше всего е. аллея, на которой растут деревья всех пород, которые есть в парке ж. аллея, деревья на которой не высаживались в прошлом году
2.5	<p>Реализовать для базы данных расписания экзаменов/зачетов триггеры для всех событий (insert,delete, update) до и после. Часть из которых будет обеспечивать ссылочную целостность, остальные могут иметь другое назначение. Вычисляемые поля можно добавить при необходимости.</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ol style="list-style-type: none"> а. аудитории, которых проходят экзамены по дисциплинам, имеющим в названии слова «базы данных» б. аудитории, где в один день проходит несколько экзаменов в. преподаватели, не принимающие экзаменов г. дисциплины самого последнего экзамена д. аудитории, в которых проходит больше всего экзаменов е. дисциплина, по которой есть экзамены у всех групп ж. преподаватель, не принимающий экзамены у группы Z9431
2.6	<p>Реализовать для базы данных поликлиники триггеры для всех событий (insert,delete, update) до и после. Часть из которых будет обеспечивать ссылочную целостность, остальные могут иметь другое назначение. Вычисляемые поля можно добавить при необходимости.</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ol style="list-style-type: none"> а. пациенты, приходившие на любые процедуры, связанные с электрофорезом б. пациент, приходивший к одному врачу и на прием и на процедуру в. пациенты, не ходившие в поликлинику г. процедуры с наименьшей стоимостью д. пациент, ходивший к наибольшему количеству врачей е. пациент, ходивший на все процедуры ж. пациент, не ходивший на процедуры к Иванову Ивану Ивановичу
2.7	<p>Реализовать для базы данных личного кабинета студента триггеры для всех событий (insert,delete, update) до и после. Часть из которых будет обеспечивать ссылочную целостность, остальные могут иметь другое назначение. Вычисляемые поля можно добавить при необходимости.</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ol style="list-style-type: none"> а. преподаватели, аз которыми закреплены дисциплины, начинающиеся со слова «автоматизирован» б. дисциплина, по которой есть и лабораторные и курсовая работа в. студенты, не прикрепившие ни одной работы г. студент, сдавший курсовую раньше всех д. студент, прикрепивший в этом месяце наибольшее число работ е. дисциплина, по которой есть все типы работ (КР, ЛР, практические) ж. преподаватель, которому не прикрепляли отчетов по курсовому проекту или работе
2.8	<p>Реализовать для базы данных костюмерной театра триггеры для всех событий (insert,delete, update) до и после. Часть из которых будет обеспечивать ссылочную</p>

	<p>целостность, остальные могут иметь другое назначение. Вычисляемые поля можно добавить при необходимости.</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ol style="list-style-type: none"> а. спектакли, в которых используются костюмы, имеющие в названии слово шут б. костюм, в котором есть и куртка и штаны в. роль, на которую пока нет костюма г. автор, разработавший самый старый из костюмов д. спектаклю, к которому разработано наибольшее число костюмов е. костюм, в котором есть все типы деталей ж. автор, не разрабатывавший костюмы к «Золушке»
2.9	<p>Реализовать для базы данных охраняемых парковок триггеры для всех событий (insert,delete, update) до и после. Часть из которых будет обеспечивать ссылочную целостность, остальные могут иметь другое назначение. Вычисляемые поля можно добавить при необходимости.</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ol style="list-style-type: none"> а. все парковки, расположенные на линиях (не улицах или проспектах)(улица в адресе содержит «линия») б. владелец машины, у которого более одного места под машину в. машины, которые неизвестно когда уедут г. владелец машин, заезжавший раньше всех д. владелец машины, останавливавшийся на минимальном числе парковок е. машина, которая стояла на всех парковках Московского проспекта ж. владелец, не парковавшийся на Невском проспекте
2.10	<p>Реализовать для базы данных спортзала триггеры для всех событий (insert,delete, update) до и после. Часть из которых будет обеспечивать ссылочную целостность, остальные могут иметь другое назначение. Вычисляемые поля можно добавить при необходимости.</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ol style="list-style-type: none"> з. виды спорта, на которые есть безлимитные абонементы (со словом безлимит в названии типа) и. вид спорта, по которому занятия ведут различные тренеры к. вид спорта, по которому не куплено ни одного абонемента л. тренеры, ведущие занятия по виду спорта с минимальным числом занятий в абонементе м. тренеры, которые ведут занятия по максимальному количеству видов спорта н. тренеры, не ведущие фитнес о. абонемент, на все виды спорта, связанные с единоборствами, заканчивающимися на «до»
2.11	<p>Реализовать для базы данных магазина обоев триггеры для всех событий (insert,delete, update) до и после. Часть из которых будет обеспечивать ссылочную целостность, остальные могут иметь другое назначение. Вычисляемые поля можно добавить при необходимости.</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ol style="list-style-type: none"> з. коллекция обоев, начинающаяся со слова Элегия и. производитель, который производит как бумажные, так и флизелиновые обои к. текстура, обоев с которой нет л. производитель с самой большой шириной рулона м. производитель обоев с максимальным количеством коллекций н. производитель, который производит обои из всех материалов о. производитель, у которого нет обоев дороже 3000 рублей
2.12	<p>Реализовать для базы данных расписания пригородных электричек триггеры</p>

	<p>для всех событий (insert,delete, update) до и после. Часть из которых будет обеспечивать ссылочную целостность, остальные могут иметь другое назначение. Вычисляемые поля можно добавить при необходимости.</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ol style="list-style-type: none"> а. все маршруты, проходящие через остановку , в названии которой есть слово село (царское село, детское село) б. свежестроенная станция, к которой пока не ведет ни один маршрут в. маршрут, проходящий через Купчино и Шушары г. самая поздняя электричка, прибывающая на витебский вокзал д. маршруты с наибольшим количеством электричек е. маршрут, на котором нет электричек с режимом отправления по выходным. ж. маршрут, по которому ходят электрички со всеми режимами движения
2.13	<p>Реализовать для вакансий базы данных триггеры для всех событий (insert,delete, update) до и после. Часть из которых будет обеспечивать ссылочную целостность, остальные могут иметь другое назначение. Вычисляемые поля можно добавить при необходимости.</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ol style="list-style-type: none"> а. вакансии, имеющие в названии SQL б. подозрительные вакансии, где не прописаны обязанности в. работодатели в Санкт-Петербурге, выставившие вакансии программиста и системного администратора г. вакансия с наименьшей зарплатой д. вакансии с минимальным количеством обязательных требований е. требование, присутствующее во всех вакансиях на программиста ж. вакансии, в которых нет требования к опыту работы
2.14	<p>Реализовать для базы данных калькулятора бюджета физического лица триггеры для всех событий (insert,delete, update) до и после. Часть из которых будет обеспечивать ссылочную целостность, остальные могут иметь другое назначение. Вычисляемые поля можно добавить при необходимости.</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ol style="list-style-type: none"> а. все расходы категорий, которые относятся к спорту (содержат слово спорт) б. месяц, в котором были разные статьи дохода в. статья доходов, по которой доходов не было г. категория, по которой наибольшие расходы в текущем году д. категория, по которой не было расходов в январе е. месяц, в котором были траты всех категорий расхода ж. месяц, в котором были траты максимального количества статей
2.15	<p>Реализовать для базы данных книги контактов триггеры для всех событий (insert,delete, update) до и после. Часть из которых будет обеспечивать ссылочную целостность, остальные могут иметь другое назначение. Вычисляемые поля можно добавить при необходимости.</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ol style="list-style-type: none"> а. друзья, у которых номер телефона начинается на 8-911, б. Люди без меток, в. люди, которые относятся к соседям и коллегам одновременно г. самые старые люди среди контактов д. месяц, когда есть дни рождения у соседей, но нет у семьи е. метки, к которым относится максимальное количество людей ж. месяц, в котором есть дни рождения у людей со всеми метками

2.16	<p>Реализовать для базы данных вузов для абитуриентов триггеры для всех событий (insert,delete, update) до и после. Часть из которых будет обеспечивать ссылочную целостность, остальные могут иметь другое назначение. Вычисляемые поля можно добавить при необходимости.</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. направленности, в которых есть слово автоматизированный б. факультет, не принимающий ни на одно направление в. направление, на которое надо сдавать историю и обществознание г. факультет вуза, принимающий на наименьшее количество направлений д. город, в котором есть все направления е. вуз, с самым ранним началом приемной компании ж. направление, на которое не надо сдавать ЕГЭ по информатике
2.17	<p>Реализовать для базы данных школьных экскурсий триггеры для всех событий (insert,delete, update) до и после. Часть из которых будет обеспечивать ссылочную целостность, остальные могут иметь другое назначение. Вычисляемые поля можно добавить при необходимости.</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. экскурсии в музеи (слово музей в любом месте названия) б. экскурсии, относящиеся к литературе и истории в. учитель, который не отвечает за экскурсии г. учащиеся, которые не ездили в музей истории религии д. экскурсия, собравшая наибольшее число участников е. самые дорогие экскурсии ж. учащийся, который был на всех экскурсиях
2.18	<p>Реализовать для базы данных собачьего питомника триггеры для всех событий (insert,delete, update) до и после. Часть из которых будет обеспечивать ссылочную целостность, остальные могут иметь другое назначение. Вычисляемые поля можно добавить при необходимости.</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. породы, названия которых содержат слово «шпиц», но не начинается с него б. собаки без медалей в. собаки, у родителей которых один и тот же владелец г. владелец, у которого есть собаки всех пород д. владелец, у которого есть йоркширские терьеры, но нет мастиффов е. порода, собак которой меньше всего ж. владельцы самых старых собак
2.19	<p>Реализовать для базы данных службы доставки триггеры для всех событий (insert,delete, update) до и после. Часть из которых будет обеспечивать ссылочную целостность, остальные могут иметь другое назначение. Вычисляемые поля можно добавить при необходимости.</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. все посылки, отправляемые на улицу, в названии которой есть окончание «ая», но это не единственные буквы в нем б. улица, на которую никогда не доставляли посылки в. контактное лицо, получавшее посылки по разным адресам г. адреса, по которым доставили больше всего посылок д. посылка с самым маленьким весом е. контактное лицо, получавшее посылки от всех фирм ж. контактное лицо, никогда не получавшее посылки в январе

2.20	<p>Реализовать для базы данных туристического путеводителя триггеры для всех событий (insert,delete, update) до и после. Часть из которых будет обеспечивать ссылочную целостность, остальные могут иметь другое назначение. Вычисляемые поля можно добавить при необходимости.</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. достопримечательности, в которых есть слово «мать», но с него название не начинается б. город без достопримечательностей в. улица, на которой есть и памятники, и архитектурные комплексы г. город, в котором нет природных комплексов д. улица, на которой больше всего памятников е. города с самыми старыми достопримечательностями ж. улица, на которой есть достопримечательности всех типов
2.21	<p>Реализовать для базы данных метрополитена триггеры для всех событий (insert,delete, update) до и после. Часть из которых будет обеспечивать ссылочную целостность, остальные могут иметь другое назначение. Вычисляемые поля можно добавить при необходимости.</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. станции, в названии которых есть слово «площадь», но оно на него не заканчивается б. линии без станций (в проекте) в. пересадочные станции с линии 1 на линию 2 г. линия с самым большим временем проезда д. линия, на которой собраны все стации, открываются раньше 7 утра е. станция с самым ранним открытием ж. линия на которой нет перегона больше 3 минут
2.22	<p>Реализовать для базы данных медпункта организации триггеры для всех событий (insert,delete, update) до и после. Часть из которых будет обеспечивать ссылочную целостность, остальные могут иметь другое назначение. Вычисляемые поля можно добавить при необходимости.</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. все сотрудники, женского пола (отчество, оканчивается на «на») б. сотрудники, никогда не обращавшиеся в медпункт с жалобами в. сотрудники организации, делавшие за последний год прививки от столбняка и кори г. сотрудник, последним сделавший флюорографию д. сотрудник, у которого сделаны все типы прививок е. сотрудник с самым большим количеством жалоб ж. сотрудник, не делавший прививки от энцефалита
2.23	<p>Реализовать для базы данных по материальной ответственности триггеры для всех событий (insert,delete, update) до и после. Часть из которых будет обеспечивать ссылочную целостность, остальные могут иметь другое назначение. Вычисляемые поля можно добавить при необходимости.</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. любые помещения, где стоит компьютерная мебель (категория мебель и слово «комп») б. помещение, в котором нет никакого имущества в. материально ответственное лицо, у которого подотчетна и компьютерная техника и специальное оборудование г. имущество, списанное последним д. помещение, в котором есть имущество всех категорий е. мат. ответственное лицо, ответственное за наибольшее количество имущества ж. помещение, в котором нет компьютерной техники

2.24	<p>Реализовать для базы данных календаря корпоративных мероприятий триггеры для всех событий (insert,delete, update) до и после. Часть из которых будет обеспечивать ссылочную целостность, остальные могут иметь другое назначение. Вычисляемые поля можно добавить при необходимости.</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. Все мероприятия, текущего года, в названии которых есть слово конференция, но с него с названия не начинаются б. Комната, в которой не проходят мероприятия в. Мероприятия, в которых участвуют сотрудники «подразделения 1» и «подразделения 2» г. Участники самых поздно заканчивающихся мероприятий д. Событие, с самым большим количеством пришедших на него участников.(участники на мероприятие могут быть приглашены и при этом могут придти/не придти). е. Мероприятие, в котором обязательно присутствие всех начальников отделов ж. Сотрудник, не посещавший мероприятий в июле прошлого года
2.25	<p>Реализовать для базы данных домашних обязанностей триггеры для всех событий (insert,delete, update) до и после. Часть из которых будет обеспечивать ссылочную целостность, остальные могут иметь другое назначение. Вычисляемые поля можно добавить при необходимости.</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. Люди, выполняющие обязанности, имеющие в названии слово уборка, но не заканчивающиеся на него б. Обязанность, которую еще не распределили по выполняющим людям в. Обязанность, которую выполняет более одного человека г. Ежемесячная обязанность, которая выполняется первой по дате (дню) д. Категория, обязанности из которой выполняют все е. Категория, в которой меньше всего обязанностей ж. Человек, который не выполняет обязанности из категории «оплата счетов»
2.26	<p>Реализовать для базы данных рабочих задач триггеры для всех событий (insert,delete, update) до и после. Часть из которых будет обеспечивать ссылочную целостность, остальные могут иметь другое назначение. Вычисляемые поля можно добавить при необходимости.</p> <p>База данных должна иметь структуру позволяющую реализовать следующие запросы:</p> <ul style="list-style-type: none"> а. Задачи, в названии которых содержится слово «интеграция», но оно не последнее б. Проект без задач в. Задача, относящаяся к 2 различным проектам г. Задача с самым поздним сроком окончания д. Проект с самым большим количеством задач е. Человек, выполняющий задачи всех приоритетов ж. Человек, у которого нет незавершенных задач

10.4. Методические материалы, определяющие процедуры оценивания индикаторов, характеризующих этапы формирования компетенций, содержатся в локальных нормативных актах ГУАП, регламентирующих порядок и процедуру проведения текущего контроля успеваемости и промежуточной аттестации обучающихся ГУАП.

11. Методические указания для обучающихся по освоению дисциплины
(Ниже приводятся рекомендации по составлению данного раздела)

11.1. Методические указания для обучающихся по освоению лекционного материала

Основное назначение лекционного материала – логически стройное, системное, глубокое и ясное изложение учебного материала. Назначение современной лекции в рамках дисциплины не в том, чтобы получить всю информацию по теме, а в освоении фундаментальных проблем дисциплины, методов научного познания, новейших достижений научной мысли. В учебном процессе лекция выполняет методологическую, организационную и информационную функции. Лекция раскрывает понятийный аппарат конкретной области знания, её проблемы, дает цельное представление о дисциплине, показывает взаимосвязь с другими дисциплинами.

Планируемые результаты при освоении обучающимся лекционного материала:

- получение современных, целостных, взаимосвязанных знаний, уровень которых определяется целевой установкой к каждой конкретной теме;
- получение опыта творческой работы совместно с преподавателем;
- развитие профессионально–деловых качеств, любви к предмету и самостоятельного творческого мышления.
- появление необходимого интереса, необходимого для самостоятельной работы;
- получение знаний о современном уровне развития науки и техники и о прогнозе их развития на ближайшие годы;
- приобретение навыков методической обработки материалов (выделение главных мыслей и положений, формулировка конкретных выводов);
- получение точного понимания всех необходимых терминов и понятий.

Лекции обеспечивают теоретическое изучение дисциплины «Проектирование баз данных». На лекциях излагаются теоретические основы баз данных, методология их создания, а так же начальные сведения и указание источников для изучения основных возможностей языка SQL.

Структура предоставления лекционного материала:

- Раздел 1. Введение в проектирование баз данных:
 - Тема 1.1. История развития и причины появления СУБД;
 - Тема 1.2. Модели данных;
 - Тема 1.3 Нормализация и денормализация баз данных;
 - Тема 1.4 Роль баз данных в программных системах
- Раздел 2. Введение в базы данных:
 - Тема 2.1. Отношения и их свойства, ключи отношений, абстрактные операции манипулирования данными;
 - Тема 2.2. Реализация отношений в базах данных, типы данных в языке SQL, операторы языка SQL для создания, удаления, модификации таблиц базы данных, манипулирования данными;
 - Тема 2.3. Реляционная алгебра;
- Раздел 3. Введение в язык SQL;
 - Тема 3.1. Оператор выборки в языке SQL, агрегатные функции;
 - Тема 3.2. Запросы с подзапросами, экзистенциальные запросы, объединение, пересечение, разность запросов;
 - Тема 3.3.Использование представлений, управляющих конструкций в языке SQL;
 - Тема 3.4. Хранимые процедуры.
 - Тема 3.5. Триггеры, обеспечение активной целостности
- Раздел 4. . Объектные и объектно-реляционные базы данных
 - Тема 4.1. Выбор типа СУБД применительно к особенностям предметной области
 - Тема 4.2. Объектно-реляционные базы данных
 - Тема 4.3. Объектные базы данных
- Раздел 5. NoSQL базы данных
 - Тема 5.1. Базы данных «Ключ-значение»
 - Тема 5.2. Документные базы данных

Тема 5.3. Графовые базы данных

Тема 5.4. Базы данных «Семейство столбцов»

Лекционный материал сопровождается демонстрацией слайдов.

11.2. Методические указания для обучающихся по выполнению лабораторных работ

В ходе выполнения лабораторных работ обучающийся должен углубить и закрепить знания, практические навыки, овладеть современной методикой и техникой эксперимента в соответствии с квалификационной характеристикой обучающегося. Выполнение лабораторных работ состоит из экспериментально-практической, расчетно-аналитической частей и контрольных мероприятий.

Выполнение лабораторных работ обучающимся является неотъемлемой частью изучения дисциплины, определяемой учебным планом, и относится к средствам, обеспечивающим решение следующих основных задач обучающегося:

- приобретение навыков исследования процессов, явлений и объектов, изучаемых в рамках данной дисциплины;
- закрепление, развитие и детализация теоретических знаний, полученных на лекциях;
- получение новой информации по изучаемой дисциплине;
- приобретение навыков самостоятельной работы с лабораторным оборудованием и приборами.

Структура и форма отчета о лабораторной работе

В состав отчета должны входить:

- Титульный лист
- Текст задания (с запросами).
- Модель базы данных.
- Тестовые данные (при задании связанном с выполнением кода на компьютере)
- Содержательная часть отчета, определяемая заданием
- Скриншоты результата выполнения кода (при задании связанном с выполнением кода на компьютере)

Требования к оформлению отчета о лабораторной работе

Отчет о выполнении лабораторной работы должен быть представлен в электронном виде в файле формата pdf,, подготовленном в текстовом редакторе, в соответствии с правилами http://guap.ru/guap/standart/titl_main.shtml.

Подробные методические указания по прохождению лабораторных работ, структуре и оформлению отчета приведены в Методических указаниях по выполнению лабораторных работ, расположенных на сервере кафедры 43, доступном из лабораторий кафедры по адресу:
z:/Методическое обеспечение каф 43/Проектирование баз данных/МУЛР ПБД 2 сем.pdf

11.3. Методические указания для обучающихся по прохождению практических занятий

Практическое занятие является одной из основных форм организации учебного процесса, заключающаяся в выполнении обучающимися под руководством преподавателя комплекса учебных заданий с целью усвоения научно-теоретических основ учебной дисциплины, приобретения умений и навыков, опыта творческой деятельности.

Целью практического занятия для обучающегося является привитие обучающимся умений и навыков практической деятельности по изучаемой дисциплине.

Планируемые результаты при освоении обучающимся практических занятий:

- закрепление, углубление, расширение и детализация знаний при решении конкретных задач;
- развитие познавательных способностей, самостоятельности мышления, творческой активности;
- овладение новыми методами и методиками изучения конкретной учебной дисциплины;
- выработка способности логического осмысления полученных знаний для выполнения заданий;
- обеспечение рационального сочетания коллективной и индивидуальной форм обучения.

11.4. Методические указания для обучающихся по прохождению самостоятельной работы

В ходе выполнения самостоятельной работы, обучающийся выполняет работу по заданию и при методическом руководстве преподавателя, но без его непосредственного участия.

Самостоятельная работа студента предполагает знакомство и использование источников информации, размещенных в сети Internet ведущими корпорациями – производителями современного программного обеспечения (таблица 9).

11.5. Методические указания для обучающихся по прохождению промежуточной аттестации.

Промежуточная аттестация обучающихся предусматривает оценивание промежуточных и окончательных результатов обучения по дисциплине. Она включает в себя:

- Экзамен (6 семестр) – форма оценки знаний, полученных обучающимся в процессе изучения всей дисциплины или ее части, навыков самостоятельной работы, способности применять их для решения практических задач. Экзамен, как правило, проводится в период экзаменационной сессии и завершается аттестационной оценкой «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

Вопросы и задачи для проведения экзамена представлены в таблице 15.

- зачет (7 семестр) – это форма оценки знаний, полученных обучающимся в ходе изучения учебной дисциплины в целом или промежуточная (по окончании семестра) оценка знаний обучающимся по отдельным разделам дисциплины с аттестационной оценкой «зачтено» или «не зачтено».

Вопросы и задачи для проведения зачета представлены в таблице 16.

Система оценок при проведении промежуточной аттестации осуществляется в соответствии с требованиями Положений «О текущем контроле успеваемости и промежуточной аттестации студентов ГУАП, обучающихся по программы высшего образования» и «О модульно-рейтинговой системе оценки качества учебной работы студентов в ГУАП».

Лист внесения изменений в рабочую программу дисциплины

Дата внесения изменений и дополнений. Подпись внесшего изменения	Содержание изменений и дополнений	Дата и № протокола заседания кафедры	Подпись зав. кафедрой