Министерство науки и высшего образования Российской Федерации

Муромский институт (филиал)

федерального государственного бюджетного образовательного учреждения высшего образования

«Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»

(МИ ВлГУ)

Кафедра *ИС*

«УТВЕРЖДАЮ»
Заместитель директора по УР
_____Д.Е. Андрианов
19.05.2026

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Разработка систем управления базами данных

Направление подготовки *09.04.02 Информационные системы и технологии*

Профиль подготовки *Системы обработки информации*

Семестр	Трудоемкость, час./зач. ед.	Лекции, час.	Практические занятия, час.	Лабораторные работы, час.	Консультация, час.	Контроль, час.	Всего (контактная работа), час.	СРС, час.	Форма промежуточного контроля (экз., зач., зач. с оц.)
1	**54 / 1,5**	**16**		**32**	**1,6**	**0,25**	**49,85**	**4,15**	**Зач.**
2	**126 / 3,5**	**16**	**14**	**32**	**3,6**	**2,35**	**67,95**	**22,4**	**Экз.(35,65)**
Итого	**180 / 5**	**32**	**14**	**64**	**5,2**	**2,6**	**117,8**	**26,55**	**35,65**

Муром, 2026 г.

1. Цель освоения дисциплины

Целью изучения дисциплины является приобретение студентами знаний и навыков в области разработки СУБД, а также знакомство с возможностями основными структурами и алгоритмами обработки данных.

Задачами дисциплины являются: изучение структур данных, алгоритмов обработки данных, средств многопоточного программирования и синхронизации, средств оптимизации приложений.

2. Место дисциплины в структуре ОПОП ВО

Курс базируется на дисциплине "Управление данными" направления подготовки 09.03.02. Знания, полученные в ходе изучения дисциплины, применяются во время подготовки диссертации.

3. Планируемые результаты обучения по дисциплине

Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения ОПОП (компетенциями и индикаторами достижения компетенций)

Формируемые компетенции (код, содержание компетенции)	Планируемые результаты обучения по дисциплине, в соответствии с индикатором достижения компетенции		Наименование оценочного средства
	Индикатор достижения компетенции	Результаты обучения по дисциплине	
ПК-2 Способен разрабатывать, вводить в действие и обслуживать базы данных; дополнять, модифицировать и совершенствовать базы данных и другие хранилища информации	ПК-2.1 Рзрабатывает, внедряет и обслуживает базы данных и хранилища информации	Разрабатывает, внедряет и обслуживает базы данных и хранилища информации (ПК-2.1) Разрабатывает, внедряет и обслуживает базы данных и хранилища информации (ПК-2.1) Разрабатывает, внедряет и обслуживает базы данных и хранилища информации (ПК-2.1)	Вопросы к устному опросу, вопросы к лабораторным работам,Задания по курсовому проектированию, вопросы по практическим работам
	ПК-2.2 Дополняет, модифицируеть и совершенствует базы данных	Дополняет, модифицирует и совершенствует базы данных (ПК-2.2) Дополняет, модифицирует и совершенствует базы данных (ПК-2.2) Дополняет, модифицирует и совершенствует базы данных (ПК-2.2)	

4. Структура и содержание дисциплины

Общая трудоемкость дисциплины составляет 5 зачетных единиц, 180 часов.

4.1. Форма обучения: очная

Уровень базового образования: высшее.
Срок обучения 2г.

4.1.1. Структура дисциплины

№ п\п	Раздел (тема) дисциплины	Семестр	Контактная работа обучающихся с педагогическим работником							Самостоятельная работа	Форма текущего контроля успеваемости (по неделям семестра), форма промежуточной аттестации(по семестрам)
			Лекции	Практические занятия	Лабораторные работы	Контрольные работы	КП / КР	Консультация	Контроль		
1	Общие сведения	1	16		32					4,15	Устный опрос, лабораторные работы
Всего за семестр		54	16		32			1,6	0,25	4,15	Зач.
2	Архитектура подсистем	2	16		32					17,85	Устный опрос, лабораторные работы
3	Разработка модулей к СУБД	2		14						4,55	Курсовая работа, практические работы
Всего за семестр		126	16	14	32		+	3,6	2,35	22,4	Экз.(35,65)
Итого		180	32	14	64			5,2	2,6	26,55	35,65

4.1.2. Содержание дисциплины

4.1.2.1. Перечень лекций

Семестр 1

Раздел 1. Общие сведения

Лекция 1.

Введение (2 часа).

Лекция 2.

Модели процессов и архитектура компьютеров (2 часа).

Лекция 3.

Потоки СУБД, процессы ОС и связи между ними (2 часа).

Лекция 4.

Параллелизм, модели процессов и координация памяти (2 часа).

Лекция 5.

Контроль допуска и пространства (2 часа).

Лекция 6.

Временной контроль. Буферизация (2 часа).

Лекция 7.

Управление буферами (2 часа).

Лекция 8.

Процессор запросов (2 часа).

Семестр 2

Раздел 2. Архитектура подсистем

Лекция 9.

Переписывание запросов и оптимизатор (2 часа).

Лекция 10.

Выполнение запросов (2 часа).

Лекция 11.

Транзакции и контроль параллельности (2 часа).

Лекция 12.

Менеджер логов (2 часа).

Лекция 13.

Аллокатор памяти (2 часа).

Лекция 14.

Подсистема управления диском (2 часа).

Лекция 15.

Сервисы репликации (2 часа).

Лекция 16.

Хранимые процедуры (2 часа).

4.1.2.2. Перечень практических занятий

Семестр 2

Раздел 3. Разработка модулей к СУБД

Практическое занятие 1

Сборка СУБД Firebird (2 часа).

Практическое занятие 2

Разработка своей UDF для СУБД Firebird (2 часа).

Практическое занятие 3

Добавление системной функции в СУБД Firebird (2 часа).

Практическое занятие 4

Добавление агрегата в СУБД Firebird (2 часа).

Практическое занятие 5

Разработка парсера запросов (2 часа).

Практическое занятие 6

Разработка подсистемы кеширования (2 часа).

Практическое занятие 7

Разработка менеджера блокировок (2 часа).

4.1.2.3. Перечень лабораторных работ

Семестр 1

Раздел 1. Общие сведения

Лабораторная 1.

Сборка СУБД Firebird (4 часа).

Лабораторная 2.

Разработка своей UDF для СУБД Firebird (4 часа).

Лабораторная 3.

Добавление системной функции в СУБД Firebird (4 часа).

Лабораторная 4.

Добавление агрегата в СУБД Firebird (4 часа).

Лабораторная 5.

Разработка парсера запросов (4 часа).

Лабораторная 6.

Разработка подсистемы кеширования (4 часа).

Лабораторная 7.

Разработка менеджера блокировок (4 часа).

Лабораторная 8.

Реализация контроля допуска (4 часа).

Семестр 2

Раздел 2. Архитектура подсистем

Лабораторная 9.

Оптимизация запросов (4 часа).

Лабораторная 10.

Сравнение планов запросов (4 часа).

Лабораторная 11.

Уровни изоляции транзакций (4 часа).

Лабораторная 12.

Менеджер памяти (4 часа).

Лабораторная 13.

Страничный кеш данных (4 часа).

Лабораторная 14.

Синхронная репликация (4 часа).

Лабораторная 15.

Асинхронная репликация (4 часа).

Лабораторная 16.

Отказоустойчивый кластер (4 часа).

4.1.2.4. Перечень тем и учебно-методическое обеспечение самостоятельной работы

Перечень тем, вынесенных на самостоятельное изучение:

1. Оптимизация запросов.
2. Администрирование БД.
3. Резервное копирование и восстановление данных.
4. Многомерная модель данных.
5. Стандарты SQL.

Для самостоятельной работы используются методические указания по освоению дисциплины и издания из списка приведенной ниже основной и дополнительной литературы.

4.1.2.5. Перечень тем контрольных работ, рефератов, ТР, РГР, РПР

Не планируется.

4.1.2.6. Примерный перечень тем курсовых работ (проектов)

1. Система мониторинга серверов СУБД Ред База Данных.
2. Реализация геометрических типов данных.
3. Разработка модуля визуализации страниц в БД.
4. Разработка модуля визуализации статистики БД.
5. Сравнительное тестирование PostgreSQL и СУБД Ред База Данных.
6. Анализ совместимости СУБД Ред База Данных с прикладным ПО.
7. Разработка утилиты преобразования SQL-скриптов из MS SQL Server в Ред База Данных.
8. Утилита для тестирования поломок базы.
9. Воспроизводимое тестирование производительности СУБД Ред База Данных.

5. Образовательные технологии

В процессе изучения дисциплины применяется контактная технология преподавания (за исключением самостоятельно изучаемых студентами вопросов). При проведении практических работ применяется имитационный или симуляционный подход. Шаги решения

задач студентам демонстрируются при помощи мультимедийной техники. В дальнейшем студенты самостоятельно решают аналогичные задания.

6. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины.

Фонды оценочных материалов (средств) приведены в приложении.

7. Учебно-методическое и информационное обеспечение дисциплины.

7.1. Основная учебно-методическая литература по дисциплине

1. Системы управления базами данных [Электронный ресурс]: лабораторный практикум/ — Электрон. текстовые данные.— Ставрополь: Северо-Кавказский федеральный университет, 2017.— 148 c.— Режим доступа: http://www.iprbookshop.ru/75595.html.— ЭБС «IPRbooks» - http://www.iprbookshop.ru/75595.html

2. Лазицкас Е.А. Базы данных и системы управления базами данных [Электронный ресурс]: учебное пособие/ Лазицкас Е.А., Загумённикова И.Н., Гилевский П.Г.— Электрон. текстовые данные.— Минск: Республиканский институт профессионального образования (РИПО), 2018.— 268 c.— Режим доступа: http://www.iprbookshop.ru/93382.html.— ЭБС «IPRbooks» - http://www.iprbookshop.ru/93382.html

3. Мирошников А.И. Архитектура систем управления базами данных [Электронный ресурс]: учебное пособие/ Мирошников А.И.— Электрон. текстовые данные.— Липецк: Липецкий государственный технический университет, ЭБС АСВ, 2018.— 94 c.— Режим доступа: http://www.iprbookshop.ru/83189.html.— ЭБС «IPRbooks» - http://www.iprbookshop.ru/83189.html

7.2. Дополнительная учебно-методическая литература по дисциплине

1. Николаев Е.И. Базы данных в высокопроизводительных информационных системах [Электронный ресурс]: учебное пособие/ Николаев Е.И.— Электрон. текстовые данные.— Ставрополь: Северо-Кавказский федеральный университет, 2016.— 163 c.— Режим доступа: http://www.iprbookshop.ru/69375.html.— ЭБС «IPRbooks» - http://www.iprbookshop.ru/69375.html

2. Фатькин Г.А. Распределенные системы управления и последовательные шины передачи данных [Электронный ресурс]: методические указания к лабораторной работе № 4 практикума ТСАНИ/ Фатькин Г.А., Панов А.Н., Орешонок В.В.— Электрон. текстовые данные.— Новосибирск: Новосибирский государственный университет, 2018.— 28 c.— Режим доступа: http://www.iprbookshop.ru/93478.html.— ЭБС «IPRbooks» - http://www.iprbookshop.ru/93478.html

7.3. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем

В образовательном процессе используются информационные технологии, реализованные на основе информационно-образовательного портала института (www.mivlgu.ru/iop), и инфокоммуникационной сети института:

- предоставление учебно-методических материалов в электронном виде;
- взаимодействие участников образовательного процесса через локальную сеть института и Интернет;
- предоставление сведений о результатах учебной деятельности в электронном личном кабинете обучающегося.

Информационные справочные системы:

электронная библиотечная системы "IPRBooks" (http://www.iprbookshop.ru/);

электронная библиотекм "Эврика" (http://elib.mivlgu.local/);

центр информационных технологий (http://citforum.ru).

Программное обеспечение:

РЕД ОС (Соглашение №140/05-21У от 18.05.2021 года о сотрудничестве в области науки, развития инновационной деятельности)

QT Creator ((L)GPL)

GNU Debugger (GPL 3+)

GCC (GNU GPL)

7.4. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

iprbookshop.ru

elib.mivlgu.local

citforum.ru).

mivlgu.ru/iop

8. Материально-техническое обеспечение дисциплины

Лаборатория распределенных систем

12 персональных компьютеров; проектор Nec V300X; экран настенный Lumien Master Picture

9. Методические указания по освоению дисциплины

Для успешного освоения теоретического материала обучающийся: знакомится со списком рекомендуемой основной и дополнительной литературы; уточняет у преподавателя, каким дополнительным пособиям следует отдать предпочтение; ведет конспект лекций и прорабатывает лекционный материал, пользуясь как конспектом, так и учебными пособиями.

На практических занятиях пройденный теоретический материал подкрепляется решением задач по основным темам дисциплины. Занятия проводятся в компьютерном классе, используя специальное программное обеспечение. Каждой подгруппе обучающихся преподаватель выдает задачу, связанную с разработкой и программной реализацией алгоритмов обработки информации. В конце занятия обучающие демонстрируют полученные результаты преподавателю и при необходимости делают работу над ошибками.

До выполнения лабораторных работ обучающийся изучает соответствующий раздел теории. Перед занятием студент знакомится с описанием заданий для выполнения работы, внимательно изучает содержание и порядок проведения лабораторной работы. Лабораторная работа проводятся в компьютерном классе. Обучающиеся выполняют индивидуальную задачу компьютерного моделирования в соответствии с заданием на лабораторную работу. Полученные результаты исследований сводятся в отчет и защищаются по традиционной методике в классе на следующем лабораторном занятии. Необходимый теоретический материал, индивидуальное задание, шаги выполнения лабораторной работы и требование к отчету приведены в методических указаниях, размещенных на информационно-образовательном портале института.

Самостоятельная работа оказывает важное влияние на формирование личности будущего специалиста, она планируется обучающимся самостоятельно. Каждый обучающийся самостоятельно определяет режим своей работы и меру труда, затрачиваемого на овладение учебным содержанием дисциплины. Он выполняет внеаудиторную работу и изучение разделов, выносимых на самостоятельную работу, по личному индивидуальному плану, в зависимости от его подготовки, времени и других условий.

Курсовая работа выполняется в соответствии с методическими указаниями на курсовую работу. Обучающийся выбирает одну из указанных в перечне тем курсовых работ, исходя из своих интересов, наличия соответствующих литературных и иных источников. В ходе выполнения курсовой работы преподаватель проводит консультации обучающегося. На заключительном этапе обучающийся оформляет пояснительную записку к курсовой работе и выполняет ее защиту в присутствии комиссии из преподавателей кафедры.

Форма заключительного контроля при промежуточной аттестации – экзамен. Для проведения промежуточной аттестации по дисциплине разработаны фонд оценочных средств и балльно-рейтинговая система оценки учебной деятельности студентов. Оценка по дисциплине выставляется в информационной системе и носит интегрированный характер, учитывающий результаты оценивания участия студентов в аудиторных занятиях, качества и своевременности выполнения заданий в ходе изучения дисциплины и промежуточной аттестации.

Программа составлена в соответствии с требованиями ФГОС ВО по направлению *09.04.02 Информационные системы и технологии* и профилю подготовки *Системы обработки информации*
Рабочую программу составил *к.т.н., доцент Симаков Р.А.*_____

Программа рассмотрена и одобрена на заседании кафедры *ИС*

протокол № 21 от 05.05.2026 года.
Заведующий кафедрой *ИС* _____*Андрианов Д.Е.*
(Подпись)

Рабочая программа рассмотрена и одобрена на заседании учебно-методической комиссии факультета

протокол № 9 от 12.05.2026 года.
Председатель комиссии ФИТР _____ *Кутарова Е.И.*
(Подпись) (Ф.И.О.)

Приложение

Фонд оценочных материалов (средств) по дисциплине

Разработка систем управления базами данных

1. Оценочные материалы для проведения текущего контроля успеваемости по дисциплине

1. Темы для устного опроса:

- Каково назначение СУБД?
- Дайте определение приложения, укажите, в каких случаях оно разрабатывается.
- Укажите назначение словаря данных.
- Перечислите функции администратора базы данных.
- Охарактеризуйте основные виды программ, относящихся к СУБД.
- Назовите основные способы работы пользователя с базой данных при решении прикладных задач.
- Дайте характеристику многопользовательским СУБД.
- Перечислите классические модели представления данных.
- Укажите достоинства и недостатки иерархической модели данных.
- Как организуется физическое размещение данных в БД иерархического типа?
- Охарактеризуйте сетевую модель данных.
- Охарактеризуйте реляционную модель данных.
- Дайте определение реляционной модели и назовите составляющие ее элементы.
- Охарактеризуйте составные элементы реляционной модели данных и формы их представления.
- Что представляет собой первичный ключ отношения, для чего он задается?
- Назовите условия, при соблюдении которых таблицу можно считать отношением.
- Что такое индекс, для чего используется индексирование?
- Изобразите схемы одноуровневой и двухуровневой индексаций и дайте им характеристику.
- Что такое вторичный индекс, в чем его отличие от первичного индекса?
- Приведите схему возможной организации связи вторичного индекса с элементами базы данных.
- Опишите действие механизма контроля целостности при манипулировании данными в таблицах.
- Назовите подходы к проектированию структур данных.
- В чем состоит избыточное и неизбыточное дублирование данных?
- Назовите и охарактеризуйте основные виды аномалий.
- Как формируется исходное отношение при проектировании БД?
- Приведите примеры явной и неявной избыточности.
- Назовите основные виды зависимостей между атрибутами отношений.
- Приведите примеры функциональной и частичной функциональной зависимостей.
- Приведите примеры отношений с зависимыми атрибутами.
- Охарактеризуйте нормальные формы.
- Дайте определение первой нормальной формы.
- Дайте определение второй нормальной формы.
- Дайте определение третьей нормальной формы.
- Дайте определение усиленной третьей нормальной формы.
- Поясните на примере используемых в разделе таблиц требования 4НФ.
- Поясните на примере используемых в разделе таблиц требования 5НФ.
- Сформулируйте основное правило создания таблиц сущностей.
- Назовите рекомендации по организации связи сущностей.
- Дайте определение физической и логической целостности БД.
- Приведите примеры ограничений значений и структурных ограничений.
- Поясните понятия внешнего и первичного ключей таблиц.
- Перечислите основные понятия метода сущность-связь.

- Охарактеризуйте понятие ключа сущности.
- Что представляют собой диаграммы ER-экземпляров и диаграммы ER-типа.
- Что определяет степень связи между сущностями?
- Каким может быть класс принадлежности?
- Приведите пример диаграммы ER-экземпляров со степенью связи между сущностями 1:1 и обязательным классом принадлежности двух сущностей.
- Как на диаграммах ER-типа обозначаются степень связи, обязательное и необязательное участие в связи экземпляров сущности?
- Приведите пример диаграммы ER-экземпляров для связи типа 1:M варианта H-O.
- Назовите этапы проектирования базы данных.
- Как осуществляется формирование отношений для связи 1:1?
- Сформулируйте правило формирования отношений, если степень связи 1:1 и класс принадлежности обеих сущностей является необязательным.
- Сформулируйте правило формирования отношения для случая степени связи между сущностями 1:M (M:1) и обязательного класса принадлежности M-связной сущности.
- Укажите правила формирования отношений для связи M:M.
- Покажите, что полученные в прикладном примере из раздела отношения находятся в нормальной форме Бойса-Кодда.

2. Задания для выполнения лабораторных работ:

- Создать БД с именем 'DB_[Фамилия студента].FDB' с помощью диалогового окна. Зарегистрировать БД в IBExpert. Проверить доступность БД и подключиться к ней. Извлечь все метаданные из БД и сохранить в файл сценария с именем SCRIPT_[Фамилия студента].SQL. Изучить содержимое файла сценария. Удалить созданную БД. Создать БД, используя полученный ранее файл сценария.
- Создать не менее 5 таблиц в БД, содержащих не менее 5 полей. При описании полей использовать как стандартные типы данных, так и домены (при необходимости создать требуемые домены). В каждой таблице должно быть как минимум 1 вычисляемое поле. В каждой таблице объявить "непустое" поле целого типа, которое в дальнейшем будет использоваться как уникальный идентификатор записи. Для произвольных полей определить условия проверки значений. Изменить атрибуты произвольных полей. Для каждого действия получить исполняемый сценарий. Заполнить созданные таблицы произвольными значениями.
- Для всех таблиц БД добавить ограничения первичного ключа. Для произвольных таблиц БД добавить ограничения внешнего ключа с указанием действия при удалении записи основной таблицы. Для произвольных таблиц добавить контрольные ограничения. Для произвольных таблиц создать ограничения уникальности.
- Написать SQL-запрос для выборки данных из БД EMPLOYEE: выбрать всех сотрудников, принятых на работу в 1991 году. Отобразить полное имя и дату приема на работу. Отсортировать данные по дате приема на работу в порядке убывания.
- Написать SQL-запрос для выборки данных из БД EMPLOYEE: определить количество сотрудников, занимающих разные должности в разных странах. Отобразить страну, код должности, класс должности и количество сотрудников. Отсортировать данные по стране в порядке возрастания, по коду должности в порядке возрастания, по классу должности в порядке убывания.
- Написать SQL-запрос для выборки данных из БД EMPLOYEE: выбрать всех сотрудников из страны "USA" с зарплатой от 70000 до 100000. Отобразить полное имя, страну и зарплату. Отсортировать данные в порядке убывания зарплаты.
- Написать SQL-запрос для выборки данных из БД EMPLOYEE: выбрать список стран, в которых более одного клиента. Отобразить страну и количество клиентов.
- Написать SQL-запрос для выборки данных из БД EMPLOYEE: выбрать коды всех сотрудников, у которых в 1993 году было снижение зарплаты. Отобразить код сотрудника, дату изменения зарплаты, предыдущую зарплату, процент изменения, новую зарплату. Отсортировать данные по проценту изменения в порядке убывания.

- Написать SQL-запрос для выборки данных из БД EMPLOYEE: выбрать все заказы, по которым была оплата, но заказ не доставлен. Отобразить номер заказа, дату заказа, статус оплаты, статус заказа, количество дней просрочки доставки заказа (с даты заказа по текущую дату). Отсортировать данные в порядке уменьшения количества дней просрочки.

- Создать представления для выборки данных из БД EMPLOYEE: выбрать все отделы предприятия с указанием номера телефона. Ото-бразить название отдела и номер телефона в виде "[Название отдела]: [Номер телефона]".

- Создать представления для выборки данных из БД EMPLOYEE: выбрать всех сотрудников и отсортировать список по дате приема со-трудника в порядке возрастания. Отобразить код должности, полное имя со-трудника ("[Код должности]: [Полное имя]"), дату приема, зарплату.

- Создать представления для выборки данных из БД EMPLOYEE: выбрать все недоставленные заказы. Отобразить номер заказа, дату заказа, статус заказа, отметку об оплате, тип продукции. Реализовать триггер, позволяющий менять тип продукции для неоплаченных заказов и запрещающий менять другие данные в таблице – источнике.

- В БД EMPLOYEE разработать хранимую процедуру, вычисляющую общую стоимость всех заказов для указанного клиента за указанный диапазон дат.

- В БД EMPLOYEE разработать хранимую процедуру, которая меняет зарплату сотрудников по следующим правилам: если общая стоимость заказов, оформленная сотрудником за указанный год, менее 50000, уменьшить на 5%; если общая стоимость заказов от 50000 до 100000, увеличить на 10%; если общая стоимость заказов от 100000 до 250000, увеличить на 15%; если общая стоимость заказов более 250000, увеличить на 25%.

- В БД EMPLOYEE разработать хранимую процедуру для выборки статистики по отделам за указанный год: код отдела, название отдела, количество сотрудников, бюд-жет отдела, количество заказов отдела, общая сумма заказов, сумма по выполненным заказам, сумма по невыполненным заказам.

- Создать в БД три пользователя с именами: Admin; User; Guest. Создать в БД три роли и назначить им права: Administrators – полный доступ ко всем объектам БД; Users – полный доступ к трем таблицам БД; разрешение чтения из трех таблиц БД; разрешение выполнения одной хранимой процедуры; запрет на доступ к оставшимся объектам БД; Guests – разрешение только на чтение из трех таблиц БД. Назначить созданным пользователям роли: Admin – роль Administrators; User – роли Administrators и Users; Guest – роль Guests. Подключиться к БД под именем пользователя, который не является владельцем БД. Проверить правильность функционирования ограничения доступа.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: вычисление площади треугольника по координатам трех вершин (фор-мула Герона).

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: определение принадлежности указанного числа к палиндромам.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: кодирование и декодирование строки по ключу (шифр Цезаря).

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: определение последнего дня месяца указанной даты.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: преобразование целого числа в форму записи римскими цифрами.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: подсчет количества слов в строке.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: преобразование строки в код Морзе.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: перевод числа из десятичной системы счисления в двоичную.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: удаление лишних пробелов в строке (пробелы в начале и в конце строки, а также любые не единичные пробелы между словами).

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: определение принадлежности указанного года к високосным.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: сложение двух комплексных чисел. Исходные числа и результат должны быть записаны в виде "A+B*i".

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: определение количества разных букв, содержащихся в строке.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: вычисление даты Пасхи по григорианскому календарю для указанного года (метод Гаусса).

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: генерация простого числа, заданной длины.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: удаление из строки всех символов, кроме букв.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: вычисление стажа в формате "X г. Y мес. Z дн." по дате приема на работу и текущей дате.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: генерация случайного числа, распределенному по нормальному закону.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: поиск самого длинного слова в строке.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: вычисление количества дней до истечения срока годности по дате производства, сроку годности товара и текущей дате.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: генерация случайного числа, распределенному по экспоненциальному закону.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: определение количества вхождений подстроки в строку.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: определить процентное отношение строчных и прописных букв в строке.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: приведение значения угла к диапазону [0, 360] или [0, 2pi] в зависимости от переданного параметра единиц измерения угла.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: удаление из строки слов, длина которых меньше заданной.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: преобразование даты к формату "ДД [название месяца] ГГГГ".

- Ознакомиться со списком компонентов MS Visual Studio из группы "Данные", их свойствами и методами. Разработать приложение с графическим интерфейсом для работы с созданной БД. В приложении реализовать различные способы отображения данных (табличное представление, табличное представление связанных данных, представление данных в виде формы, отображение значений из справочников), а также разработать процедуры сохранения модифицированных данных (новых, отредактированных и удаленных записей) в БД.

- Разработать приложение для печати отчетов из БД. Для создаваемых отчетов выполнить дополнительные настройки шрифтов, выравнивания текста, заголовков и т.п. в дизайнере отчетов. Реализовать построение диаграмм, различные виды группировки и фильтрации данных.

Общее распределение баллов текущего контроля по видам учебных работ для студентов

Рейтинг-контроль 1	Устный опрос (2 вопроса)	15
Рейтинг-контроль 2	Устный опрос (2 вопроса)	15
Рейтинг-контроль 3	Устный опрос (2 вопроса)	15
Посещение занятий студентом	Отметка в журнале посещений	5

Дополнительные баллы (бонусы)		-
Выполнение семестрового плана самостоятельной работы	Защита лабораторных работ	10

2. Промежуточная аттестация по дисциплине

Перечень вопросов к экзамену / зачету / зачету с оценкой.
Перечень практических задач / заданий к экзамену / зачету / зачету с оценкой (при наличии)

1. Темы для устного опроса:

- Каково назначение СУБД?
- Дайте определение приложения, укажите, в каких случаях оно разрабатывается.
- Укажите назначение словаря данных.
- Перечислите функции администратора базы данных.
- Охарактеризуйте основные виды программ, относящихся к СУБД.
- Назовите основные способы работы пользователя с базой данных при решении прикладных задач.
- Дайте характеристику многопользовательским СУБД.
- Перечислите классические модели представления данных.
- Укажите достоинства и недостатки иерархической модели данных.
- Как организуется физическое размещение данных в БД иерархического типа?
- Охарактеризуйте сетевую модель данных.
- Охарактеризуйте реляционную модель данных.
- Дайте определение реляционной модели и назовите составляющие ее элементы.
- Охарактеризуйте составные элементы реляционной модели данных и формы их представления.
- Что представляет собой первичный ключ отношения, для чего он задается?
- Назовите условия, при соблюдении которых таблицу можно считать отношением.
- Что такое индекс, для чего используется индексирование?
- Изобразите схемы одноуровневой и двухуровневой индексаций и дайте им характеристику.
- Что такое вторичный индекс, в чем его отличие от первичного индекса?
- Приведите схему возможной организации связи вторичного индекса с элементами базы данных.
- Опишите действие механизма контроля целостности при манипулировании данными в таблицах.
- Назовите подходы к проектированию структур данных.
- В чем состоит избыточное и неизбыточное дублирование данных?
- Назовите и охарактеризуйте основные виды аномалий.
- Как формируется исходное отношение при проектировании БД?
- Приведите примеры явной и неявной избыточности.
- Назовите основные виды зависимостей между атрибутами отношений.
- Приведите примеры функциональной и частичной функциональной зависимостей.
- Приведите примеры отношений с зависимыми атрибутами.
- Охарактеризуйте нормальные формы.
- Дайте определение первой нормальной формы.
- Дайте определение второй нормальной формы.
- Дайте определение третьей нормальной формы.
- Дайте определение усиленной третьей нормальной формы.
- Поясните на примере используемых в разделе таблиц требования 4НФ.
- Поясните на примере используемых в разделе таблиц требования 5НФ.
- Сформулируйте основное правило создания таблиц сущностей.

- Назовите рекомендации по организации связи сущностей.
- Дайте определение физической и логической целостности БД.
- Приведите примеры ограничений значений и структурных ограничений.
- Поясните понятия внешнего и первичного ключей таблиц.
- Перечислите основные понятия метода сущность-связь.
- Охарактеризуйте понятие ключа сущности.
- Что представляют собой диаграммы ER-экземпляров и диаграммы ER-типа.
- Что определяет степень связи между сущностями?
- Каким может быть класс принадлежности?
- Приведите пример диаграммы ER-экземпляров со степенью связи между сущностями 1:1 и обязательным классом принадлежности двух сущностей.
- Как на диаграммах ER-типа обозначаются степень связи, обязательное и необязательное участие в связи экземпляров сущности?
- Приведите пример диаграммы ER-экземпляров для связи типа 1:M варианта H-O.
- Назовите этапы проектирования базы данных.
- Как осуществляется формирование отношений для связи 1:1?
- Сформулируйте правило формирования отношений, если степень связи 1:1 и класс принадлежности обеих сущностей является необязательным.
- Сформулируйте правило формирования отношения для случая степени связи между сущностями 1:M (M:1) и обязательного класса принадлежности M-связной сущности.
- Укажите правила формирования отношений для связи M:M.
- Покажите, что полученные в прикладном примере из раздела отношения находятся в нормальной форме Бойса-Кодда.

2. Задания для выполнения лабораторных работ:

- Создать БД с именем 'DB_[Фамилия студента].FDB' с помощью диалогового окна. Зарегистрировать БД в IBExpert. Проверить доступность БД и подключиться к ней. Извлечь все метаданные из БД и сохранить в файл сценария с именем SCRIPT_[Фамилия студента].SQL. Изучить содержимое файла сценария. Удалить созданную БД. Создать БД, используя полученный ранее файл сценария.
- Создать не менее 5 таблиц в БД, содержащих не менее 5 полей. При описании полей использовать как стандартные типы данных, так и домены (при необходимости создать требуемые домены). В каждой таблице должно быть как минимум 1 вычисляемое поле. В каждой таблице объявить "непустое" поле целого типа, которое в дальнейшем будет использоваться как уникальный идентификатор записи. Для произвольных полей определить условия проверки значений. Изменить атрибуты произвольных полей. Для каждого действия получить исполняемый сценарий. Заполнить созданные таблицы произвольными значениями.
- Для всех таблиц БД добавить ограничения первичного ключа. Для произвольных таблиц БД добавить ограничения внешнего ключа с указанием действия при удалении записи основной таблицы. Для произвольных таблиц добавить контрольные ограничения. Для произвольных таблиц создать ограничения уникальности.
- Написать SQL-запрос для выборки данных из БД EMPLOYEE: выбрать всех сотрудников, принятых на работу в 1991 году. Отобразить полное имя и дату приема на работу. Отсортировать данные по дате приема на работу в порядке убывания.
- Написать SQL-запрос для выборки данных из БД EMPLOYEE: определить количество сотрудников, занимающих разные должности в разных странах. Отобразить страну, код должности, класс должности и количество сотрудников. Отсортировать данные по стране в порядке возрастания, по коду должности в порядке возрастания, по классу должности в порядке убывания.
- Написать SQL-запрос для выборки данных из БД EMPLOYEE: выбрать всех сотрудников из страны "USA" с зарплатой от 70000 до 100000. Отобразить полное имя, страну и зарплату. Отсортировать данные в порядке убывания зарплаты.
- Написать SQL-запрос для выборки данных из БД EMPLOYEE: выбрать список стран, в которых более одного клиента. Отобразить страну и количество клиентов.

- Написать SQL-запрос для выборки данных из БД EMPLOYEE: выбрать коды всех сотрудников, у которых в 1993 году было снижение зарплаты. Отобразить код сотрудника, дату изменения зарплаты, предыдущую зарплату, процент изменения, новую зарплату. Отсортировать данные по проценту изменения в порядке убывания.

- Написать SQL-запрос для выборки данных из БД EMPLOYEE: выбрать все заказы, по которым была оплата, но заказ не доставлен. Отобразить номер заказа, дату заказа, статус оплаты, статус заказа, количество дней просрочки доставки заказа (с даты заказа по текущую дату). Отсортировать данные в порядке уменьшения количества дней просрочки.

- Создать представления для выборки данных из БД EMPLOYEE: выбрать все отделы предприятия с указанием номера телефона. Ото-бразить название отдела и номер телефона в виде "[Название отдела]: [Номер телефона]".

- Создать представления для выборки данных из БД EMPLOYEE: выбрать всех сотрудников и отсортировать список по дате приема со-трудника в порядке возрастания. Отобразить код должности, полное имя со-трудника ("[Код должности]: [Полное имя]"), дату приема, зарплату.

- Создать представления для выборки данных из БД EMPLOYEE: выбрать все недоставленные заказы. Отобразить номер заказа, дату заказа, статус заказа, отметку об оплате, тип продукции. Реализовать триггер, позволяющий менять тип продукции для неоплаченных заказов и запрещающий менять другие данные в таблице – источнике.

- В БД EMPLOYEE разработать хранимую процедуру, вычисляющую общую стоимость всех заказов для указанного клиента за указанный диапазон дат.

- В БД EMPLOYEE разработать хранимую процедуру, которая меняет зарплату сотрудников по следующим правилам: если общая стоимость заказов, оформленная сотрудником за указанный год, менее 50000, уменьшить на 5%; если общая стоимость заказов от 50000 до 100000, увеличить на 10%; если общая стоимость заказов от 100000 до 250000, увеличить на 15%; если общая стоимость заказов более 250000, увеличить на 25%.

- В БД EMPLOYEE разработать хранимую процедуру для выборки статистики по отделам за указанный год: код отдела, название отдела, количество сотрудников, бюд-жет отдела, количество заказов отдела, общая сумма заказов, сумма по выполненным заказам, сумма по невыполненным заказам.

- Создать в БД три пользователя с именами: Admin; User; Guest. Создать в БД три роли и назначить им права: Administrators – полный доступ ко всем объектам БД; Users – полный доступ к трем таблицам БД; разрешение чтения из трех таблиц БД; разрешение выполнения одной хранимой процедуры; запрет на доступ к оставшимся объектам БД; Guests – разрешение только на чтение из трех таблиц БД. Назначить созданным пользователям роли: Admin – роль Administrators; User – роли Administrators и Users; Guest – роль Guests. Подключиться к БД под именем пользователя, который не является владельцем БД. Проверить правильность функционирования ограничения доступа.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: вычисление площади треугольника по координатам трех вершин (фор-мула Герона).

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: определение принадлежности указанного числа к палиндромам.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: кодирование и декодирование строки по ключу (шифр Цезаря).

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: определение последнего дня месяца указанной даты.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: преобразование целого числа в форму записи римскими цифрами.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: подсчет количества слов в строке.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: преобразование строки в код Морзе.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: перевод числа из десятичной системы счисления в двоичную.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: удаление лишних пробелов в строке (пробелы в начале и в конце строки, а также любые не единичные пробелы между словами).

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: определение принадлежности указанного года к високосным.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: сложение двух комплексных чисел. Исходные числа и результат должны быть записаны в виде "A+B*i".

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: определение количества разных букв, содержащихся в строке.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: вычисление даты Пасхи по григорианскому календарю для указанного года (метод Гаусса).

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: генерация простого числа, заданной длины.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: удаление из строки всех символов, кроме букв.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: вычисление стажа в формате "X г. Y мес. Z дн." по дате приема на работу и текущей дате.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: генерация случайного числа, распределенному по нормальному закону.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: поиск самого длинного слова в строке.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: вычисление количества дней до истечения срока годности по дате производства, сроку годности товара и текущей дате.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: генерация случайного числа, распределенному по экспоненциальному закону.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: определение количества вхождений подстроки в строку.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: определить процентное отношение строчных и прописных букв в строке.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: приведение значения угла к диапазону [0, 360] или [0, 2pi] в зависимости от переданного параметра единиц измерения угла.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: удаление из строки слов, длина которых меньше заданной.

- Разработать, зарегистрировать в БД и проверить работоспособность UDF: преобразование даты к формату "ДД [название месяца] ГГГГ".

- Ознакомиться со списком компонентов MS Visual Studio из группы "Данные", их свойствами и методами. Разработать приложение с графическим интерфейсом для работы с созданной БД. В приложении реализовать различные способы отображения данных (табличное представление, табличное представление связанных данных, представление данных в виде формы, отображение значений из справочников), а также разработать процедуры сохранения модифицированных данных (новых, отредактированных и удаленных записей) в БД.

- Разработать приложение для печати отчетов из БД. Для создаваемых отчетов выполнить дополнительные настройки шрифтов, выравнивания текста, заголовков и т.п. в дизайнере отчетов. Реализовать построение диаграмм, различные виды группировки и фильтрации данных.

Методические материалы, характеризующие процедуры оценивания

На основе перечня вопросов к тестированию программным комплексом информационно-образовательного портала МИ ВлГУ формируются в автоматическом режиме тестовые задания для студентов: 8 вопросов из блока 1, 4 вопроса из блока 2 и 3 вопроса из

блока 3. Программный комплекс формирует индивидуальные задания для каждого зарегистрированного в системе студента и устанавливает время прохождения тестирования. Результатом тестирования является балл, рассчитанный на основе количества правильных ответов. С учетом индивидуального семестрового рейтинга студента формируется итоговый балл по курсу.

Максимальная сумма баллов, набираемая студентом по дисциплине равна 100.

Оценка в баллах	Оценка по шкале	Обоснование	***Уровень сформированности компетенций***
Более 80	«Отлично»	Содержание курса освоено полностью, без пробелов, необходимые практические навыки работы с освоенным материалом сформированы, все предусмотренные программой обучения учебные задания выполнены, качество их выполнения оценено числом баллов, близким к максимальному	***Высокий уровень***
66-80	«Хорошо»	Содержание курса освоено полностью, без пробелов, некоторые практические навыки работы с освоенным материалом сформированы недостаточно, все предусмотренные программой обучения учебные задания выполнены, качество выполнения ни одного из них не оценено минимальным числом баллов, некоторые виды заданий выполнены с ошибками	***Продвинутый уровень***
50-65	«Удовлетворительно»	Содержание курса освоено частично, но пробелы не носят существенного характера, необходимые практические навыки работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий, возможно, содержат ошибки	***Пороговый уровень***
Менее 50	«Неудовлетворительно»	Содержание курса не освоено, необходимые практические навыки работы не сформированы, выполненные учебные задания содержат грубые ошибки	***Компетенции не сформированы***

3. Задания в тестовой форме по дисциплине

Примеры заданий:

Перечень вопросов для проведения тестирования:

ОПК-2, ОПК-3, ПК-4

Блок 1 (знать):

1. Какой из языков относится к языкам манипулирования данными?

SQL
DDL
XML
MDX

2. Какие из перечисленных свойств присущи транзакциям?

атомарность
сериализуемость
долговечность
стабильность

3. Какие из перечисленных правил проверяются при контроле целостности связей?

каждой записи основной таблицы соответствует ноль или более записей дополнительной таблицы

в дополнительной таблице нет записей, которые не имеют родительских записей в основной таблице

каждая запись дополнительной таблицы имеет только одну родительскую запись основной таблицы

в дополнительной таблице присутствует более одной записи

4. Какая аномалия проявляется в том, что изменение значения одного данного может повлечь за собой просмотр всей таблицы и соответствующее изменение некоторых других записей таблицы?

аномалия редактирования
аномалия добавления
аномалия удаления
аномалия выборки

5. Какая аномалия проявляется в том, что информацию в таблицу нельзя поместить до тех пор, пока она неполная, либо требует дополнительного просмотра таблицы?

аномалия редактирования
аномалия добавления
аномалия удаления
аномалия выборки

6. Какая аномалия проявляется в том, что при удалении какого-либо данного из таблицы может пропасть и другая информация, которая не связана напрямую с удаляемым данным?

аномалия редактирования
аномалия добавления
аномалия удаления
аномалия выборки

12. Какая из приведенных моделей данных является наиболее продвинутой?

сетевая
реляционная
иерархическая
плоская база данных

13. Какая модель данных изменяется при изменении выбранной СУБД?

физическая

логическая
обе
никакая

14. Какая модель данных основывается на понятии отношения, представляющего собой множество элементов, называемых кортежами?

иерархическая
сетевая
реляционная
постреляционная

15. Какая модель данных позволяет отображать разнообразные взаимосвязи элементов данных в виде произвольного графа?

иерархическая
сетевая
реляционная
постреляционная

16. Какая модель данных позволяет отображать связи между данными с помощью упорядоченного графа (или дерева)?

иерархическая
сетевая
реляционная
постреляционная

17. Какая модель данных снимает ограничение неделимости данных, хранящихся в записях таблиц?

иерархическая
сетевая
реляционная
постреляционная

18. Как в терминах метода "сущность-связь" называется атрибут или набор атрибутов, используемый для идентификации экземпляра сущности?

диаграмма
атрибут
связь
ключ

19. Как в терминах метода "сущность-связь" называется зависимость между атрибутами нескольких сущностей?

диаграмма
атрибут
связь
ключ

20. Как в терминах метода "сущность-связь" называется объект, информация о котором хранится в БД?

сущность
атрибут
связь
ключ

21. Как в терминах метода "сущность-связь" называется свойство сущности?

атрибут
связь
ключ
диаграмма

22. Как в терминах реляционной модели называется атрибут отношения, однозначно идентифицирующий каждый из его кортежей?

первичный ключ
внешний ключ
домен
сущность

23. Как в терминах реляционной модели называется двумерная таблица, содержащая некоторые данные?

отношение
атрибут
домен
сущность

24. Как в терминах реляционной модели называется множество всех возможных значений определенного атрибута отношения?

сущность
первичный ключ
домен
схема отношения

25. Как в терминах реляционной модели называется свойство, характеризующее сущность?

атрибут
отношение
домен
сущность

26. Как в терминах реляционной модели называется список имен атрибутов?

домен
схема отношения
сущность
отношение

27. Как называется некоторая неделимая последовательность операций над данными БД, которая отслеживается СУБД от начала и до завершения?

транзакция
хранимая процедура
функция пользователя
запрос

28. Как называется ситуация в таблицах БД, которая приводит к противоречиям в БД, либо существенно усложняет обработку данных?

аномалия
дублирование
отношение
несоответствие

29. Как называют логическую структуру хранимых в базе данных?

модель данных
структура данных
запись
таблица

30. Как называются данные, которые получаются из других данных путем их обобщения?

метаданные
агрегированные данные
нормализованные данные
упорядоченные данные

31. Как называются данные, описывающие структуру БД?

метаданные
агрегированные данные
нормализованные данные
модель данных

32. Какое свойство транзакции определяет, что выполняются все входящие в транзакцию операции или ни одна?

атомарность
сериализуемость
долговечность
стабильность

33. Какое свойство транзакции определяет, что даже крах системы не приводит к утрате результатов зафиксированной транзакции?

атомарность
сериализуемость
долговечность
стабильность

34. Какое свойство транзакции определяет, что отсутствует взаимное влияние выполняемых в одно и то же время транзакций?

атомарность
сериализуемость
долговечность
стабильность

35. Какое утверждение является верным?

в первичный ключ входят атрибуты, однозначно идентифицирующие сущность
ключ - это атрибут, значение которого не может изменить пользователь
значения атрибутов, входящих в состав ключа уникальны
значения ключей двух сущностей не должны повторяться

36. Какой объект БД используется для хранения данных?

таблица
представление
домен
хранимая процедура

37. Какой объект БД может быть использован для автоматической проверки значений добавляемой записи?

триггер

курсор
индекс
хранимая процедура

38. Какой объект БД может быть использован для обеспечения безопасности хранимых данных?

роль
хранимая процедура
триггер
транзакция

39. Какой объект БД может быть использован для реализации сложных алгоритмов обработки данных?

курсор
хранимая процедура
представление
таблица

40. Какой объект БД позволяет задавать множество допустимых значений полей таблиц?

домен
представление
триггер
транзакция

41. Какой объект БД предназначен для ускорения операции поиска записей в таблице, а следовательно, и других операций, использующих поиск?

триггер
индекс
курсор
домен

42. Какой объект БД представляет собой виртуальную (логическую) таблицу, заданную в виде поименованного запроса?

представление
хранимая процедура
триггер
домен

43. Как по-другому называется усиленная 3 нормальная форма?

НФ Бойса-Кодда
НФ Шмидта
НФ Кодда
НФ Бойса

44. К какому виду относится связь, образуемая в случае, когда все поля связи основной и дополнительной таблиц являются ключевыми?

1:1
1:M
M:1
M:N

45. К какому виду относится связь, образуемая в случае, когда нескольким записям основной таблицы соответствует несколько записей дополнительной таблицы?

1:1
1:M
M:1
M:N

46. К какому типу относится высокоуровневый непроцедурный язык декларативного типа, предназначенный для описания логической структуры данных?

язык описания данных
язык манипулирования данными
язык модификации данных
язык разметки

47. К какому типу относится язык обеспечивающий выполнение основных операций по работе с данными?

язык описания данных
язык манипулирования данными
язык модификации данных
язык разметки

48. К какому типу СУБД относятся программные комплексы, имеющие развитый интерфейс, позволяющие с помощью команд меню выполнять основные действия с БД?

полнофункциональные СУБД
серверы БД
клиенты БД
средства разработки программ работы с БД

49. К какому типу СУБД относятся программные комплексы предназначенные для организации центров обработки данных в сетях ЭВМ?

полнофункциональные СУБД
серверы БД
клиенты БД
средства разработки программ работы с БД

50. Сколько основных типов аномалий существует?

1
2
3
4

51. Что из перечисленного представляет собой комплекс языковых и программных средств, предназначенный для создания, ведения и совместного использования базы данных многими пользователями?

модель данных
система управления базами данных
информационная система
вычислительная система

52. Что из перечисленного представляет собой совокупность специальным образом организованных данных, хранимых в памяти вычислительной системы и отображающих состояние объектов и их взаимосвязей в рассматриваемой предметной области?

база данных
модель данных
система управления базами данных
информационная система

53. Что присуще подавляющему большинству СУБД?
автоматическая архивация данных
синхронизация данных через Интернет
поддержка языка SQL
поддержка различных форматов импорта и экспорта данных

54. Что является наименьшей единицей структуры данных?
атрибут
сущность
запись
база данных

Блок 2 (уметь):
1. Firebird. Запишите контекстную переменную, возвращающую текущее время

2. Firebird. Запишите контекстную переменную, возвращающую текущую дату

3. Firebird. Запишите контекстную переменную, возвращающую текущую дату и время

4. Firebird. Запишите функцию для вычисления ближайшего большего или равного числа

5. Firebird. Запишите функцию для вычисления ближайшего меньшего или равного числа

6. Firebird. Запишите функцию для вычисления десятичного логарифма

7. Firebird. Запишите функцию для вычисления квадратного корня

8. Firebird. Запишите функцию для вычисления количества символов в строке

9. Firebird. Запишите функцию для вычисления косинуса

10. Firebird. Запишите функцию для вычисления котангенса

11. Firebird. Запишите функцию для вычисления модуля числа

12. Firebird. Запишите функцию для вычисления натурального логарифма

13. Firebird. Запишите функцию для вычисления остатка от деления

14. Firebird. Запишите функцию для вычисления синуса

15. Firebird. Запишите функцию для вычисления степени числа

16. Firebird. Запишите функцию для вычисления тангенса

17. Firebird. Запишите функцию для вычисления экспоненты

18. Firebird. Запишите функцию для извлечения части даты

19. Firebird. Запишите функцию для округления числа

20. Firebird. Запишите функцию для определения кода символа

21. Firebird. Запишите функцию для определения позиции первого вхождения подстроки в строку

22. Firebird. Запишите функцию для определения разницы между датами

23. Firebird. Запишите функцию для определения символа по его коду

24. Firebird. Запишите функцию для получения подстроки заданной длины, начиная с конца строки

25. Firebird. Запишите функцию для получения подстроки заданной длины, начиная с указанной позиции

26. Firebird. Запишите функцию для получения подстроки заданной длины, начиная с начала строки

27. Firebird. Запишите функцию для получения случайного числа

28. Firebird. Запишите функцию для получения строки заданной длины, дополненной слева заданным символом

29. Firebird. Запишите функцию для получения строки заданной длины, дополненной справа заданным символом

30. Firebird. Запишите функцию для преобразования значения в пустое значение (NULL), если оно равно заданному значению

31. Firebird. Запишите функцию для преобразования пустого значения (NULL) в заданное значение

32. Firebird. Запишите функцию для преобразования строки к верхнему регистру

33. Firebird. Запишите функцию для преобразования строки к нижнему регистру

34. Firebird. Запишите функцию для преобразования типов данных

35. Firebird. Запишите функцию для увеличения или уменьшения даты на заданное количество единиц времени

36. Firebird. Запишите функцию для удаления заданных символов слева, справа или с обоих концов строки

37. Язык SQL. Firebird. Запишите оператор вставки записи

38. Язык SQL. Firebird. Запишите оператор выборки данных

39. Язык SQL. Firebird. Запишите оператор редактирования домена

40. Язык SQL. Firebird. Запишите оператор редактирования представления

41. Язык SQL. Firebird. Запишите оператор редактирования роли пользователя

42. Язык SQL. Firebird. Запишите оператор редактирования структуры таблицы

43. Язык SQL. Firebird. Запишите оператор редактирования таблицы

44. Язык SQL. Firebird. Запишите оператор редактирования триггера

45. Язык SQL. Firebird. Запишите оператор редактирования хранимой процедуры

46. Язык SQL. Firebird. Запишите оператор создания домена

47. Язык SQL. Firebird. Запишите оператор создания представления

48. Язык SQL. Firebird. Запишите оператор создания роли пользователя

49. Язык SQL. Firebird. Запишите оператор создания таблицы

50. Язык SQL. Firebird. Запишите оператор создания триггера

51. Язык SQL. Firebird. Запишите оператор создания хранимой процедуры

52. Язык SQL. Firebird. Запишите оператор удаления домена

53. Язык SQL. Firebird. Запишите оператор удаления записи

54. Язык SQL. Firebird. Запишите оператор удаления представления

55. Язык SQL. Firebird. Запишите оператор удаления роли пользователя

56. Язык SQL. Firebird. Запишите оператор удаления таблицы

57. Язык SQL. Firebird. Запишите оператор удаления триггера

58. Язык SQL. Firebird. Запишите оператор удаления хранимой процедуры

59. Язык SQL. Firebird. Запишите предложение оператора SELECT, определяющее условия выборки данных

60. Язык SQL. Firebird. Запишите предложение оператора SELECT, определяющее условия выборки сгруппированных данных

61. Язык SQL. Firebird. Запишите предложение оператора SELECT, отвечающее за установку перекрестной связи с другой таблицей

62. Язык SQL. Firebird. Запишите предложение оператора SELECT, отвечающее за установку левой внешней связи с другой таблицей

63. Язык SQL. Firebird. Запишите предложение оператора SELECT, отвечающее за сортировку данных

64. Язык SQL. Firebird. Запишите предложение оператора SELECT, отвечающее за группировку данных

65. Язык SQL. Firebird. Запишите предложение оператора SELECT, отвечающее за установку правой внешней связи с другой таблицей

66. Язык SQL. Firebird. Запишите предложение оператора SELECT, отвечающее за выборку неповторяющихся записей

67. Язык SQL. Firebird. Запишите предложение оператора SELECT, отвечающее за установку полной связи с другой таблицей

68. Язык SQL. Firebird. Запишите предложение оператора SELECT, отвечающее за установку внутренней связи с другой таблицей

Блок 3 (владеть):

1. Укажите правильную последовательность выполнения команды обновления записей в таблице PERSON для изменения поля город (CITY) с Санкт-Петербурга на Москву.

```
UPDATE
PERSON
SET
CITY = 'Москва'
WHERE
CITY = 'Санкт-Петербург'
```

2. Определите наиболее подходящие типы данных для значений.

```
2012-07-05
10.5
15
'Hello, world!'
```

3. Расположите предложения команды SELECT в правильном порядке.

```
SELECT
столбцы
FROM
таблица
WHERE условие
```

4. Расположите предложения оператора DELETE в правильном порядке.

```
DELETE
FROM
таблица
WHERE условие
```

5. Расположите предложения оператора INSERT в правильном порядке.

```
INSERT
INTO
таблица
VALUES (значения)
```

6. Дана таблица, созданная с помощью SQL-выражения:

```
CREATE TABLE STUDENTS (
    ID INTEGER PRIMARY KEY,
    FIRST_NAME VARCHAR(50) NOT NULL,
    LAST_NAME VARCHAR(50) NOT NULL,
    ADDRESS VARCHAR(100)
);
```

Какие запросы позволят добавить запись в эту таблицу?

INSERT INTO STUDENTS(ID, FIRST_NAME, LAST_NAME) VALUES (1, 'Name', 'Surname');
INSERT INTO STUDENTS(ID, FIRST_NAME) VALUES (2, 'Name');
INSERT INTO STUDENTS VALUES (3, 'Name', 'Surname', NULL);
INSERT INTO STUDENTS VALUES (4, NULL, 'Surname', NULL);

7. Какие из определений таблицы гарантируют, что в колонку C1 нельзя поместить значение NULL?

CREATE TABLE T1 (C1 INTEGER DEFAULT 0);
CREATE TABLE T1 (C1 INTEGER PRIMARY KEY);
CREATE TABLE T1 (C1 INTEGER NOT NULL);
CREATE TABLE T1 (C1 INTEGER);

8. Таблица создана с помощью SQL-выражения:

```
CREATE TABLE PUBLISHER (
    NAME VARCHAR(45)
);
```

Какие из запросов корректно отработают?

INSERT INTO PUBLISHER (NAME) VALUES ('OREILLY');
INSERT INTO PUBLISHER (NAME) VALUES ('O Reilly');
INSERT INTO PUBLISHER (NAME) VALUES ('O'Reilly');
INSERT INTO PUBLISHER (NAME) VALUES ('O-Reilly');

9. Дана таблица PEOPLES

Num	Name	Amount
1	Kate	15
2	Misha	NULL
3	Nick	0
4	Larisa	20

Какой будет результат выполнения запроса:

```
SELECT COUNT(AMOUNT) FROM PEOPLE;
```

3
4
Запрос завершится ошибкой, поскольку присутствует NULL
NULL, поскольку присутствует NULL

10. Дана таблица CLIENTS

ID	Name	Summa
1	Sasha	4000
2	Marina	2000
3	Stepan	0
4	Klara	NULL

После обновления таблицы:

```
UPDATE CLIENTS SET SUMMA = SUMMA + 700;
```

Какой будет результат выполнения запроса:

```
SELECT AVG(SUMMA) FROM CLIENTS;
```

2000
2700
3000
Запрос завершится ошибкой из-за присутствия NULL

11. Даны таблицы:

DOGS

Name	Age

Snoopy 4
Benny 2
ANIMALS
Name Age
- -
Поле Name имеет тип VARCHAR(15), а поле Age - INTEGER.
Выберите некорректные INSERT-запросы.

INSERT INTO ANIMALS (AGE, NAME) SELECT AGE, NAME FROM DOGS;
INSERT INTO ANIMALS (AGE, NAME) SELECT * FROM DOGS;
INSERT INTO ANIMALS VALUES SELECT NAME, AGE FROM DOGS;
INSERT INTO ANIMALS SELECT * FROM DOGS;

12. Какие из перечисленных SQL-запросов являются верными?

INSERT INTO students SELECT VALUES (95, 'Bill', 'Enzo');
INSERT INTO students (id, first_name, last_name);
INSERT INTO students (id, first_name, last_name) (118, 'Ellen', 'Moran');
INSERT INTO students (id, first_name, last_name) VALUES (100, 'Kim', 'Matheson');

13. Как выбрать все записи из таблицы "Persons", где значение поля "FirstName" равно "Peter"?

SELECT * FROM Persons WHERE FirstName <> 'Peter'
SELECT * FROM Persons WHERE FirstName = 'Peter'
SELECT [all] FROM Persons WHERE FirstName = 'Peter'
SELECT [all] FROM Persons WHERE FirstName LIKE 'Peter'

14. Как выбрать все записи из таблицы "Persons", где значение поля "FirstName" равно "Peter" и "LastName" равно "Jackson"?

SELECT * FROM Persons WHERE FirstName = 'Peter' AND LastName = 'Jackson'
SELECT FirstName = 'Peter', LastName = 'Jackson' FROM Persons
SELECT * FROM Persons WHERE FirstName <> 'Peter' AND LastName <> 'Jackson'

15. Как выбрать все записи из таблицы "Persons", где значение поля "LastName" в алфавитном порядке находится между значениями "Hansen" и "Pettersen"?

SELECT * FROM Persons WHERE LastName BETWEEN 'Hansen' AND 'Pettersen'
SELECT * FROM Persons WHERE LastName > 'Hansen' AND LastName < 'Pettersen'
SELECT LastName > 'Hansen' AND LastName < 'Pettersen' FROM Persons

16. Как выбрать все записи из таблицы "Persons", упорядоченных по полю "FirstName" в обратном порядке?

SELECT * FROM Persons ORDER BY FirstName DESC
SELECT * FROM Persons ORDER FirstName DESC
SELECT * FROM Persons SORT BY 'FirstName' DESC
SELECT * FROM Persons SORT 'FirstName' DESC

17. Выберите правильный SQL запрос для вставки новой записи в таблицу "Persons".

INSERT ('Jimmy', 'Jackson') INTO Persons
INSERT VALUES ('Jimmy', 'Jackson') INTO Persons
INSERT INTO Persons VALUES ('Jimmy', 'Jackson')

18. Как изменить значение "Hansen" на "Nilsen" в колонке "LastName", таблицы Persons?

MODIFY Persons SET LastName = 'Nilsen' WHERE LastName = 'Hansen'
UPDATE Persons SET LastName = 'Hansen' INTO LastName = 'Nilsen'
UPDATE Persons SET LastName = 'Nilsen' WHERE LastName = 'Hansen'
MODIFY Persons SET LastName = 'Hansen' INTO LastName = 'Nilsen

19. Как удалить записи, где значение поля "FirstName" равно "Peter"?
DELETE FROM Persons WHERE FirstName = 'Peter'
DELETE ROW FirstName = 'Peter' FROM Persons
DELETE FirstName = 'Peter' FROM Persons

20. Как вывести количество записей, хранящихся в таблице "Persons"?
SELECT COUNT() FROM Persons
SELECT COLUMNS() FROM Persons
SELECT COLUMNS(*) FROM Persons
SELECT COUNT(*) FROM Persons

21. Верно ли синтаксически составлен запрос:
SELECT EMP_NAME, SUM(SAL) FROM EMPLOYEE

22. Дана таблица, созданная с помощью SQL-выражения:

```
CREATE TABLE PEOPLE (
    ID INTEGER PRIMARY KEY,
    NAME VARCHAR(45)
);
```

Можно ли гарантированно удалить все записи из этой таблицы, выполнив запрос:
DELETE FROM PEOPLE WHERE ID > = 0

Полный перечень тестовых заданий с указанием правильных ответов, размещен в банке вопросов на информационно-образовательном портале института по ссылке https://www.mivlgu.ru/iop/question/edit.php?courseid=547

Оценка рассчитывается как процент правильно выполненных тестовых заданий из их общего числа.