

**МИНОБРНАУКИ РОССИИ**  
**Ярославский государственный университет им. П.Г. Демидова**

Кафедра компьютерной безопасности и математических методов обработки информации

УТВЕРЖДАЮ

Декан математического факультета

\_\_\_\_\_  
Нестеров П.Н.

20 мая 2025 г.

**Рабочая программа дисциплины**

**Базы данных**

Направление подготовки (специальности)  
02.03.01 Математика и компьютерные науки

Направленность (профиль)  
«Программирование, алгоритмы и анализ данных»

Форма обучения очная

Программа рассмотрена  
на заседании кафедры  
от 24.04.2025, протокол № 8

Программа одобрена НМК  
математического факультета  
протокол № 9 от 05.05.2025

## 1. Цели освоения дисциплины

Дисциплина «Базы данных» нацелена на освоение студентами теории баз данных (БД) и приобретение практических навыков построения пользовательских приложений под управлением современных реляционных, реляционно-объектных СУБД и NoSQL баз данных. Курс должен дать фундаментальную подготовку, необходимую для успешного освоения как общепрофессиональных, так и специальных дисциплин, изучение которых связано с созданием информационных систем для различных предметных областей, их анализом, внедрением и сопровождением. Содержание программы дисциплины должно обеспечить базовую подготовку студентов в процессе формирования устойчивых знаний и практических навыков проектирования реляционных баз данных, работы с «настольными» и клиент-серверными СУБД. Материалы курса могут использоваться при изучении основ Web-программирования, основ моделирования и проектирования информационных систем.

Целью воспитания личности при реализации программы дисциплины является формирование таких черт как организованность и умение планировать время для выполнения сложных проектов; умение общаться с людьми в ходе выполнения этапа анализа предметной области и при подготовке рекомендаций по использованию созданных приложений, трудолюбие, ответственность, способность к саморазвитию, повышению своей квалификации и мастерства.

Задачи дисциплины:

- получение основополагающих знаний о принципах проектирования, построения и использования реляционных баз данных;
- получение основополагающих знаний о принципах проектирования, построения и использования NoSQL базы данных;
- изучение основ организации вычислений в распределенных многопользовательских средах.

## 2. Место дисциплины в структуре образовательной программы

Данная дисциплина относится к части образовательной программы, формируемой участниками образовательных отношений.

Изучение её базируется на следующих дисциплинах:

«Алгебра» - основные свойства важнейших алгебраических структур;

«Основы программирования» - формы и способы представления данных в персональном компьютере, классификация современных компьютерных систем, типовые структуры и принципы организации компьютерных сетей; - общие принципы построения и использования современных языков программирования высокого уровня, особенности взаимодействия языков высокого и низкого уровня, организации работы с памятью в скриптовых языках; - базовые структуры данных, оценка сложности алгоритмов, принципы разработки эффективных алгоритмов и программ;

«Операционные системы и оболочки» - принципы построения современных операционных систем и особенности их применения, конфигурирование локальных компьютерных сетей, реализация сетевых протоколов с помощью программных средств.

Знания и практические навыки, полученные при изучении дисциплины «Базы данных», используются студентами при разработке курсовых и дипломных работ, а также дисциплин вариативной части профессионального цикла, предусмотренных примерным учебным планом.

## 3. Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы

Процесс изучения дисциплины направлен на формирование следующих компетенций в соответствии с ФГОС ВО, ООП ВО и приобретения следующих знаний, умений, навыков и (или) опыта деятельности:

Формируемая компетенция (код и формулировка)	Индикатор достижения компетенции (код и формулировка)	Перечень планируемых результатов обучения
<b>Профессиональные компетенции</b>		
<b>ПК-2</b> Способен участвовать в разработке технической документации программных продуктов и программных комплексов.	<b>И-ПК-2.1</b> Знает основные стандарты, нормы и правила разработки технической документации программных продуктов и программных комплексов	<b>Знать:</b> - основные стандарты, нормы и правила разработки технической документации программных продуктов и программных комплексов
	<b>И-ПК-2.2</b> Умеет использовать их при подготовке технической документации программных продуктов	<b>Владеть:</b> - навыками документирования и сопровождения программного обеспечения с учетом повышенных требований к надежности программ и их защищенности от несанкционированного доступа; - методикой сбора и анализа информации по исследуемой предметной области
<b>ПК-4</b> Способен использовать современные методы разработки и реализации конкретных алгоритмов математических моделей на базе языков программирования и пакетов прикладных программ моделирования	<b>И-ПК-4.2</b> Умеет разрабатывать и реализовывать алгоритмы математических моделей на базе языков и пакетов прикладных программ моделирования	<b>Знать:</b> - физическую организацию баз данных; - средства поддержания целостности в БД; - особенности управления данными в системах распределенной обработки; - причины нарушения целостности баз данных; - смысл и методы абстрагирования данных; - характеристики и типы систем БД; - области применения СУБД; <b>Уметь:</b> - выделять сущности и связи предметной области; - отображать предметную область на конкретную модель данных; - нормализовывать отношения при проектировании реляционной БД; - проектировать базы данных с использованием CASE-средств; - производить установку, наладку, тестирование и обслуживание систем управления базами данных на различных платформах
	<b>И-ПК-4.3</b> Имеет практический опыт разработки и реализации алгоритмов на базе языков и пакетов прикладных программ моделирования.	<b>Знать:</b> - этапы проектирования баз данных; - порядок эксплуатации баз данных; - способы достижения безопасности баз данных <b>Уметь:</b> - строить информационную модель базы данных, определять требования к

		проектируемой базе данных, осуществлять физическое проектирование базы данных; - разрабатывать программы на языках программирования четвертого поколения
--	--	---

#### 4. Объем, структура и содержание дисциплины

Общая трудоемкость дисциплины составляет **3** зачетных единиц, **108** акад. часов.

№ п/п	Темы (разделы) дисциплины, их содержание	Семестр	Виды учебных занятий, включая самостоятельную работу студентов, и их трудоемкость (в академических часах)						Формы текущего контроля успеваемости  Форма промежуточной аттестации (по семестрам)
			Контактная работа					самостоятельная работа	
			лекции	практические	лабораторные	консультации	аттестационные испытания		
1.	Вводная лекция. Информационные системы (ИС) и БД. Архитектура БД. Основные функции системы управления БД (СУБД). Транзакция. Корпоративные и настольные БД.	5	1						
2.	Понятие Базы данных (БД). Модели данных - иерархическая, сетевая, реляционная, постреляцион-ная, многомерная; их особенности. Логическая и физическая структура БД.	5	1						
3.	Проектирование БД: ER-модель, её состав, способ построения.	5	2	2	4	1		8	Контрольная работа «ER-Модель», Индивидуальное задание №1
4.	Процесс нормализации отношений. Функциональная зависимость, транзитивная зависимость, зависимость соединения. Проецирование без потерь. Нормальные формы и их связь с ER-моделью.	5	1	2		1		4	Контрольная работа «Процесс нормализации
5.	Реляционные БД. Понятие отношения. Средства	5	1	2		1		4	Контрольная работа «Реляционная алгебра»

	манипулирования данными. Реляционная алгебра.								
6.	Язык SQL. Оператор CREATE TABLE. MS SQL SERVER. MICROSOFT SQL SERVER MANAGEMENT STUDIO. Создание БД. Создание таблиц. Диаграммы связей.	5	1	2	4			8	Индивидуальное задание №2 Контекст «Создание таблиц и связей»
7.	Язык SQL. Поддержка в SQL операций реляционной алгебры. Типы операторов – DDL, DML, DCL - и их назначение. Структура оператора SELECT. Примеры.	5	1						
8.	Язык SQL. Оператор SELECT, фильтрация, сортировка, виды соединений отношений. Примеры.	5	1	2	4	1		4	Индивидуальное задание №3
9.	Язык SQL. Оператор SELECT с агрегирующими функциями. Группировка. Оконные функции, CTE, рекурсивные запросы. Примеры.	5	1	2				4	Контекст «Команда Select»
10.	Язык SQL. Операторы INSERT, UPDATE, DELETE.	5	1	2		1		4	Контекст «Команды DML»
11.	Язык TSQL. Представления (VIEW). Курсоры. Триггеры (предшествующие, последующие). Хранимые процедуры, функции.	5	3	2	2	1		9	Контрольная работа «Transact SQL» Индивидуальное задание №3-4 Контекст «TSQL»
12.	Журнализация изменений. Восстановление БД после сбоя. Права доступа к объектам БД. Серверные роли. Роли БД. Язык SQL. Оператор GRANT.	5	1		2			8	Контрольная работа «Раздача прав» Индивидуальное задание №5
13.	Жизненный цикл ИС (ГОСТ 12207). Основные, вспомогательные и управляющие процессы. Состав работ процессов и их назначение по ГОСТ 12207	5	1						
							<b>0,3</b>	<b>0,7</b>	<b>зачет</b>

		Всего		16	16	16	6	0,3	53,7	
--	--	-------	--	----	----	----	---	-----	------	--

## 5. Образовательные технологии, в том числе технологии электронного обучения и дистанционные образовательные технологии, используемые при осуществлении образовательного процесса по дисциплине

В процессе обучения используются следующие образовательные технологии:

**Вводная лекция** – дает первое целостное представление о дисциплине и ориентирует студента в системе изучения данной дисциплины. Студенты знакомятся с назначением и задачами курса, его ролью и местом в системе учебных дисциплин и в системе подготовки в целом. Дается краткий обзор курса, история развития науки и практики, достижения в этой сфере, имена известных ученых, излагаются перспективные направления исследований. На этой лекции высказываются методические и организационные особенности работы в рамках данной дисциплины, а также дается анализ рекомендуемой учебно-методической литературы.

**Академическая лекция** (или лекция общего курса) – последовательное изложение материала, осуществляемое преимущественно в виде монолога преподавателя. Требования к академической лекции: современный научный уровень и насыщенная информативность, убедительная аргументация, доступная и понятная речь, четкая структура и логика, наличие ярких примеров, научных доказательств, обоснований, фактов.

Академическая лекция, как правило, состоит из трех частей: вступления (введения), изложения и заключения:

- вступление (введение) определяет тему, план и цель лекции. Оно призвано заинтересовать и настроить аудиторию, сообщить, в чём заключается предмет лекции и (или) её актуальность, основная идея (проблема, центральный вопрос), связь с предыдущими и последующими занятиями, поставить её основные вопросы. Введение должно быть кратким и целенаправленным.

- изложение является основной частью лекции, в которой реализуется научное содержание темы, ставятся все узловые вопросы, приводится вся система доказательств с использованием наиболее целесообразных методических приемов. Каждое теоретическое положение должно быть обосновано и доказано, приводимые формулировки и определения должны быть четкими, насыщенными глубоким содержанием.

- заключение обобщает в кратких формулировках основные идеи лекции, логически ее завершая. В заключении могут даваться рекомендации о порядке дальнейшего изучения основных вопросов лекции самостоятельно по указанной литературе.

**Лекция с разбором конкретных ситуаций** – это по форме та же лекция-дискуссия, но на обсуждение преподаватель ставит не вопрос, а конкретную ситуацию. Как правило, такая ситуация представляется устно или в очень короткой видеозаписи, поэтому изложение ее должно быть очень кратким, но содержать достаточную информацию для оценки характерного явления и обсуждения. Это, так называемая, микроситуация. Слушатели анализируют и обсуждают ее сообща, всей аудиторией. Преподаватель старается активизировать участие в обсуждении отдельными вопросами, обращенными к отдельным слушателям, выясняет их оценку суждениям коллег, предлагает сопоставить с собственной практикой, «сталкивает» между собой различные мнения и тем развивает дискуссию, стремясь направить ее в нужное русло. Затем, опираясь на правильные высказывания и анализируя неправильные, ненавязчиво, но убедительно подводит аудиторию к коллективному выводу или обобщению.

**Обобщающая лекция** – проводится в завершение изучения раздела или темы для закрепления знаний. На лекции вновь выделяются основные вопросы, используются обобщающие таблицы, схемы, алгоритмы, позволяющие включить усвоенные знания в новые связи и зависимости, переводя их на более высокие уровни усвоения.

**Практическое занятие** – занятие, посвященное освоению конкретных умений и навыков и закреплению полученных на лекции знаний по предложенному алгоритму.

**Практическое занятие на основе кейс-метода («метод кейсов», «кейс-стади»)** – метод, основанный на моделировании ситуации или использования реальной ситуации из научной, производственной, общественной и др. деятельности в целях анализа данного случая, выявления проблем, поиска альтернативных решений и принятия оптимального решения проблем. Данный метод дает возможность изучить сложные или эмоционально значимые вопросы в безопасной обстановке, а не в реальной жизни с ее угрозами, риском, тревогой о неприятных последствиях в случае неправильного решения. Студенты должны проанализировать ситуацию, разобраться в сути проблем, предложить возможные решения и выбрать лучшее из них. Кейсы базируются на реальном фактическом материале или же приближены к реальной ситуации.

**Лабораторная работа** – организация учебной работы с реальными материальными и информационными объектами, экспериментальная работа с аналоговыми моделями реальных объектов.

**Контеcт** - соревнование по программированию, при котором участники отправляют код с решением данных задач на сайт, предназначенный для их автоматического тестирования.

## **6. Перечень лицензионного и (или) свободно распространяемого программного обеспечения, используемого при осуществлении образовательного процесса по дисциплине**

В процессе осуществления образовательного процесса используются:

- для формирования текстов материалов для промежуточной и текущей аттестации - приложение Microsoft Office;
- для разработки презентаций лекций - приложение Microsoft PowerPoint;
- для поиска учебной литературы библиотеки ЯрГУ – Автоматизированная библиотечная информационная система "БУКИ-NEXT" (АБИС "Буки-Next");
- для обучения языку SQL - [www.sql-ex.ru](http://www.sql-ex.ru), [sqlcontest.uniyar.ac.ru](http://sqlcontest.uniyar.ac.ru)
- для отработки практических навыков работы с БД используются СУБД MSSQL, MySQL, Oracle, СУБД MongoDB;
- для разработки интерфейса доступа к данным среды разработки - Visual Studio, Apache+PHP
- CASE-средства проектирования баз данных.

## **7. Перечень современных профессиональных баз данных и информационных справочных систем, используемых при осуществлении образовательного процесса по дисциплине (при необходимости)**

В процессе осуществления образовательного процесса по дисциплине используются:

- Автоматизированная библиотечно-информационная система «БУКИ-NEXT»  
[http://www.lib.uniyar.ac.ru/opac/bk\\_cat\\_find.php](http://www.lib.uniyar.ac.ru/opac/bk_cat_find.php)
- Электронная библиотечная система «Лань» <https://e.lanbook.com>
- Электронная библиотечная система «Юрайт» <https://urait.ru>
- Электронная библиотечная система «Консультант студента»  
<https://www.studentlibrary.ru>

## **8. Перечень основной и дополнительной учебной литературы, ресурсов информационно-телекоммуникационной сети «Интернет» (при необходимости), рекомендуемых для освоения дисциплины**

### **а) основная литература**

1. Власова О.В. Системы управления базами данных: лабораторный практикум – Ярославль: ЯрГУ, 2010. <http://www.lib.uniyar.ac.ru/edocs/iuni/20100201.pdf>
2. Власова О.В. SQL: Учебное пособие – Ярославль: ЯрГУ, 2011. <http://www.lib.uniyar.ac.ru/edocs/iuni/20110206.pdf>
3. Власова О.В. Transact-SQL: методические указания – Ярославль: ЯрГУ, 2013. <http://www.lib.uniyar.ac.ru/edocs/iuni/20130204.pdf>

### **б) дополнительная литература**

1. Волк, В. К. Базы данных. Проектирование, программирование, управление и администрирование: учебник для вузов: С.-Пб. Лань, 2022. <https://reader.lanbook.com/book/193373>
2. Парфенов, Ю. П. Постреляционные хранилища данных: учебное пособие для вузов: Издательство Юрайт, 2022. <https://urait.ru/viewer/postrelyacionnye-hranilisha-dannyh-492609>
- Стружкин, Н. П. Базы данных: проектирование: учебник для вузов: М. Издательство Юрайт, 2022. <https://urait.ru/viewer/bazy-dannyh-proektirovanie-489099>
4. Маркин, А. В. Программирование на SQL в 2 ч. Часть 2: учебник и практикум для вузов: М. Издательство Юрайт, 2022. <https://urait.ru/viewer/programmirovanie-na-sql-v-2-ch-chast-2-490104#page/1>

### **в) ресурсы сети «Интернет» (при необходимости)**

1. Интерактивный учебник по SQL (<http://www.sql-tutorial.ru/>)
2. Практическое владение языком SQL (<http://www.sql-ex.ru/>)
3. SQL.RU (<http://www.sql.ru/>)
4. Microsoft Developer Network (MSDN) (<https://msdn.microsoft.com/ru-ru/>)
5. НОУ «ИНТУИТ» (<http://www.intuit.ru/>)
6. The MongoDB 3.0 Manual (<http://docs.mongodb.org/manual/>)

## **9. Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине**

Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине включает в свой состав специальные помещения:

- учебные аудитории для проведения занятий лекционного типа
- учебные аудитории для проведения групповых и индивидуальных консультаций,
- учебные аудитории для проведения текущего контроля и промежуточной аттестации;
- помещения для самостоятельной работы;
- помещения для хранения и профилактического обслуживания технических средств обучения.

Специальные помещения укомплектованы средствами обучения, служащими для представления учебной информации большой аудитории.

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа к электронной информационно-образовательной среде ЯрГУ.

### **Автор(ы):**

Старший преподаватель кафедры компьютерной безопасности и математических методов обработки информации

О.В. Власова



**Приложение № 1 к рабочей программе дисциплины  
«Базы данных»**

**Фонд оценочных средств  
для проведения текущего контроля успеваемости  
и промежуточной аттестации студентов  
по дисциплине**

**1. Типовые контрольные задания и иные материалы,  
используемые в процессе текущего контроля успеваемости**

***Контрольная работа «ER-Модель»( И-ПК-4.3)***

*Пример контрольного задания*

Вы работаете в страховой компании. Вашей задачей является отслеживание её финансовой деятельности.

Компания имеет различные филиалы по всей стране. Каждый филиал характеризуется названием, адресом, телефоном и т.д. Деятельность компании организована следующим образом: к Вам обращаются различные лица с целью заключения договора о страховании. В зависимости от принимаемых на страхование объектов и страхуемых рисков договор заключается по определенному виду страхования (например, страхование автотранспорта от угона, страхование домашнего имущества, добровольное медицинское страхование). При заключении договора Вы фиксируете дату заключения, страховую сумму, вид страхования, тарифную ставку и филиал, в котором заключен договор. Нужно учесть, что договоры заключают страховые агенты. Помимо информации об агентах (Фамилия, имя, отчество, паспортные данные, адрес и т.д.), нужно хранить филиал, в котором работают агенты. Нужно иметь возможность рассчитать заработную плату агентам. Заработная плата составляет некоторый процент от страхового платежа (страховой платеж – это страховая сумма, умноженная на тарифную ставку). Процент зависит от вида страхования, по которому заключен договор.

Постройте модель «сущность-связь».

***Контрольная работа «Реляционная алгебра» (И-ПК-4.3)***

*Пример контрольного задания*

Схема БД состоит из четырех отношений:

Product(maker, model, type)

PC(code, model, speed, ram, hd, cd, price)

Laptop(code, model, speed, ram, hd, screen, price)

Printer(code, model, color, type, price)

Написать формулы реляционного исчисления и выражения реляционной алгебры для получения данных

1. Найдите пары моделей PC, имеющих одинаковые скорость и RAM. В результате каждая пара указывается только один раз, т.е. (i,j), но не (j,i), Порядок вывода: модель с большим номером, модель с меньшим номером, скорость и RAM.

2. Найдите производителя ПК блокнотов, выпускающего также цветные принтеры.

### Контрольная работа «Процесс нормализации» ( И-ПК-4.3)

#### Пример контрольного задания

1. Имеется отношение **Flight**, содержащее информацию об авиарейсах. Ключевые атрибуты выделены подчеркиванием.

Известно, что номер рейса **определяет Место отправления, Место назначения и Время вылета**. **Бортовой номер самолета** определяет **модель самолета**, которая, в свою очередь, определяет **число мест** в салоне. В разные даты рейс может обслуживаться различными самолетами.

Необходимо нормализовать отношение **Flight** до 3-й НФ таким образом, чтобы число полученных отношений было минимальным. Из приведенных вариантов решения выберите лучший. В фигурных скобках приводятся заголовки отношений, ключевые атрибуты выделяются подчеркиванием, FK в скобках указывает на то, что это атрибут - внешний ключ.

#### Flight

<u>N рейса</u>	<u>Дата вылета</u>	<u>Бортовой N самолета</u>	Модель самолета	Число мест	Место отправки	Место назнач.	Время вылета

A. {N рейса, Дата вылета, Бортовой N самолета (FK), Место отправления, Место назначения, Время вылета},  
{Бортовой N самолета, Модель самолета, Число мест}

B. {N рейса (FK), Дата вылета, Бортовой N самолета (FK)},  
{N рейса, Место отправления, Место назначения, Время вылета},  
{Бортовой N самолета, Модель самолета, Число мест}

C. {N рейса (FK), Дата вылета, Бортовой N самолета (FK)}  
{N рейса, Место отправления, Место назначения, Время вылета},  
{Бортовой N самолета, Модель самолета (FK)},  
{Модель самолета (FK), Число мест}

D. {N рейса (FK), Дата вылета, Бортовой N самолета (FK)}  
{N рейса, Место отправления, Место назначения},  
{N рейса (FK), Время вылета},  
{Бортовой N самолета, Модель самолета (FK)},  
{Модель самолета, Число мест}

1. Приведите отношение к третьей нормальной форме

Предметная область «Учет продаж товаров», например, в аптеках задана описанием следующих атрибутов о количестве и цене проданной фармацевтической продукции:

**(Код аптеки, Название аптеки, Адрес, Телефон, Код товара, Код наименования, Наименование лекарственного препарата, Код группы, Название группы, Форма, Доза, Количество в упаковке, Код страны-производителя, Название страны-производителя, Дата продажи, Цена за уп., Кол-во уп.),**

где

- Название группы обозначает наименование определенной категории фармацевтической продукции, например, «Лекарственные препараты», «Гомеопатические препараты» и т.п.;
- Форма – это «таблетки», «дражже», «ампулы» и т.п.;
- Количество в упаковке – это количество товара данной формы в одной упаковке, например, «анальгин в таблетках по 50 мг №50», где «анальгин» - наименование лекарственного препарата, «таблетки» - форма, «50мг» - доза, «№50» - количество в 1 упаковке;
- Цена за уп. – цена за упаковку товара;
- Кол-во уп. – количество проданных упаковок товара.

При установлении функциональных зависимостей учесть следующее:

- сведения о наименованиях препаратов должны храниться централизованно в едином для всех аптек справочнике со сквозной нумерацией; аналогично для справочника товаров и справочника наименований групп препаратов;
- связь между аптекой и товаром - это связь типа «М:М»;
- срок годности и другие реквизиты фармацевтической продукции, не приведенные в заданном списке атрибутов, можно не учитывать;
- по коду аптеки можно однозначно определить ее название, адрес и телефон;
- по коду наименования можно однозначно определить наименование лекарственного препарата, код группы, название группы;
- по коду группы можно однозначно определить название группы;
- по коду товара можно однозначно определить код наименования, наименование лекарственного препарата, код группы, название группы, форму, дозу, количество в упаковке, код страны-производителя, название страны-производителя;
- по коду страны-производителя можно однозначно определить название страны-производителя;

по коду аптеки, коду товара и дате продажи можно однозначно определить, какова была в этот день цена за одну упаковку данного товара в данной аптеке и сколько упаковок было продано).

или

Предметная область: **Сессия**

Номер Студ. билета	ФИО	Факультет	Декан факультета	Группа	Староста группы	Экзамен в группе	Дата экзамена в группе	Преподаватель, принимающий экзамен	Оценка студента
123456	Иванов	Математический	Нестеров П.Н.	ПМИ 31	123488	БД МА ТФ	10.01 15.01 20.01	Власова О.В. Ухалов А.Ю. Бережной Е.И.	5 4 3
123457	Петров	ИВТ	Чалый Д.Ю.	ИВТ32	123489	МА БД Сети	11.01 16.01 20.01	Богомолов Ю.В. Горбунов О.Е. Носков А.М.	3 4 5

### Тест «SQL» ( ИД-ПК-4.4)

#### Пример контрольного задания

В базе данных имеются 2 таблицы – **Students** и **Results**.

В таблице **Students** столбец **ID** – номер студенческого билета, первичный ключ; столбец **Name** содержит **ФИО** студента.

В таблице **Results** поле **ID** – номер студенческого билета, внешний ключ, ссылающийся на таблицу **Students**; **Mark** – оценка студента за экзамен; первичный ключ таблицы составной, он включает поля **ID**, **Subject**.

Из приведенных вариантов запросов выберите те, которые позволяют отобразить **ФИО** студентов, сдавших хотя бы 1 экзамен на 5.

**Диаграмма**

```
graph TD
    subgraph Students
        ID1((ID))
        Name1[Name]
        ID1 --- Name1
    end
    subgraph Results
        ID2((ID))
        Subject[Subject]
        Mark[Mark]
        ID2 --- Subject
        ID2 --- Subject
    end
    ID1 --> ID2
```

The diagram illustrates a database schema with two tables: **Students** and **Results**.

- Students Table:** Contains attributes **ID** (Primary Key) and **Name**.
- Results Table:** Contains attributes **ID** (Foreign Key), **Subject** (Primary Key), and **Mark**. The **ID** and **Subject** attributes form a composite primary key.

A one-to-many relationship is shown between **Students** and **Results**, with **Students.ID** being the primary key and **Results.ID** being the foreign key.

**Значения**

	ID	Name
1	123	Иванов И.И.
2	124	Петров П.П.
3	125	Сидоров С.С.

  

	ID	Subject	Mark
1	123	Базы данных	5
2	123	Высшая математика	5
3	124	Высшая математика	3

- A. SELECT DISTINCT Name FROM Students, Results  
WHERE Students.ID=Results.ID AND Mark=5;
- B. SELECT DISTINCT Name  
FROM Students Inner Join Results On Students.ID=Results.ID  
WHERE Mark=5;
- C. SELECT DISTINCT Name  
FROM Students Left Join Results On Students.ID=Results.ID  
WHERE Mark=5;
- D. SELECT DISTINCT Name FROM Students  
WHERE ID In (Select ID From Results Where Mark=5);

Выберите все правильные варианты ответов

### Контрольная работа «Transact SQL» ( ИД-ПК-4.4)

#### Пример контрольного задания

1. Даны отношения:

Сотрудники (Табельный номер, ФИО, Дата рождения, Пол, Паспортные данные, ИНН, Код должности, Адрес, Телефон)

Должности (Код должности, Название должности, Оклад)

Книги (№ контракта, Дата подписи контракта, Менеджер, Название, Цена, Затраты, Авторский гонорар, Дата выхода, Тираж, Ответственный редактор, Остаток тиража)

Авторы(Код автора, ФИО, Паспортные данные, ИНН, Адрес, Телефон)

Книги-Авторы(№ контракта, Код автора, Номер на обложке, Гонорар)

Книги-Редакторы( № контракта, Код редактора)

Заказы ( № заказа, Заказчик, Дата поступления заказа, Адрес заказчика, Дата выполнения заказа)

Строки-заказа ( № заказа, № контракта, Количество)

Написать триггер, который следит за остатком тиража для книг (Изменяет остаток при поступлении заказа на книгу, запрещает заказ превышающий остаток).

## 2. Хранимая процедура

Даны отношения:

R1 (ID, Класс, Предмет, День недели, Номер урока, Учитель);

R2 (Класс, Предмет, Количество уроков в неделю, Стоимость урока);

R3 (Учитель, выходной);

R4 (Класс, Максимальное количество уроков в день, Максимальная стоимость дня).

Написать функцию, которая определяет занятость учителя

Учитель, День недели, Номер первого урока, Номер последнего урока, Количество уроков, Количество окон.

### **Контрольная работа «Раздача прав» ( ИД-ПК-5.2)**

*Пример контрольного задания*

1. В базе данных существуют пользователи Alex и Mike. Периодически пользователю Alex необходимо выполнять ряд действий именно в этой базе данных от имени пользователя Mike. Какую из предложенных инструкций Вы можете применить, но так, чтобы не давать пользователю Alex прочих прав пользователя Mike на уровне экземпляра?

GRANT IMPERSONATE ON USER:: Mike TO Alex;

GRANT IMPERSONATE ON User:: Mike TO LOGIN Alex;

GRANT IMPERSONATE ON USER:: Alex TO Mike;

GRANT IMPERSONATE ON LOGIN:: Mike TO Alex;

GRANT IMPERSONATE ON LOGIN:: Alex TO Mike

2. Вам нужно дать одному из ваших разработчиков (Alexey) права на просмотр определений всех таблиц, представлений, хранимых процедур и прочего в рабочей базе данных (Production), которые принадлежат схеме dbo, при этом Вы не хотите давать ему прав на изменение данных объектов. Выберите, как Вы это сделаете.

USE Production ;

GRANT VIEW DEFINITION ON SHEMA::dbo TO 'Alexey';

USE Production ;

sp\_addrolemember @rolename = 'db\_ddladmin' , @membername = 'Alexey';

USE Production ;

sp\_addrolemember @rolename = 'db\_owner' , @membername = 'Alexey';

USE Production ;

sp\_addrolemember @rolename = 'db\_reader' , @membername = 'Alexey';

GRANT VIEW DEFINITION ON DATABASE::Production TO 'Alexey';

3. Выберите правильное утверждение относительно предназначения ролей базы данных (Database Role) в MS SQL Server 2008.

A. Роль базы данных предназначена для связывания определенных функциональных прав на объекты базы данных с конкретными учетными записями в экземпляре SQL Server

B. Роль базы данных предназначена для временной замены прав определенных для пользователя, на права определенные для данной роли

C. Роль базы данных предназначена для связывания определенных функциональных прав на объекты экземпляра SQL Server

D. Роль базы данных предназначена для группировки определенных объектов с последующим назначением прав пользователю сразу на все объекты, связанные с заданной ролью

F. Роль базы данных предназначена для группировки определенных функциональных прав к объектам базы данных с последующим назначением одной или нескольких ролей конкретной учетной записи пользователя в базе данных

### ***Индивидуальное задание №1 (И-ПК-2.3)***

#### ***Пример задания***

Разработать ER модель. Преобразовать ER модель в реляционную модель данных. Проверить соблюдение требований 3 нормальной формы. Разработать скрипт на языке SQL для создания БД со всеми ограничениями.

#### ***Описание Предметной области***

Информация о врачах: ФИО, телефон, специальность, номер кабинета, для терапевтов – номер участка, график приема(день недели, время начала приема).

Информация об участках: номер, название улицы, список домов.

Каждый врач работает 3 часа в день, время приема одного пациента – 20 мин.

Реализовать:

- Ввод и редактирование данных о врачах и их графике приема;
- Вывод расписания работы врачей по дням недели;
- Ввод информации о талоне, выданном конкретному пациенту к требуемому врачу на требуемое время;
- Просмотр информации о выданных талонах по врачам, дням недели;
- Просмотр списка пациентов, записавшихся к конкретному врачу, по дням недели;
- Получение информации о наличии свободных талонов к нужному врачу по дням недели;
- Получение информации о наличии свободных талонов по адресу пациента к участковому терапевту по дням недели.

## **Индивидуальное задание №2 (И-ПК-4.4)**

### **Пример задания**

Создать разработанную в лабораторной работе №1 БД в СУБД. Заполнить необходимыми данными.

## **Индивидуальное задание №3 ( И-ПК-4.3)**

### **Пример задания части 1**

Составить следующие запросы, сохранить их в БД в виде представлений (VIEW)

- a) Выбрать информацию о свободных талонах на конкретный день недели
- b) Для заданного пациента (ФИО) выдать все его посещения с начала текущего года
- c) Пациент может пройти медосмотр во вторник с 14 до 16 ч. Выбрать всех специалистов, к которым он может попасть
- d) Для каждого врача вывести количество посещений по датам за период с начала текущего месяца
- e) Вывести количество посещений для каждого участка за вчерашний день

### **Задания части 2**

Для своей ПО составить запросы по плану ниже. Формулировки запроса составляете самостоятельно, они должны быть содержательными.

Отчет по лабораторной работе предоставляется в виде документа (Фамилия\_Группа.docx).

В этом документе по каждому заданию необходимо представить: условие запроса, текст SQL-запроса, скрин-копию результата выполнения запроса.

Задания и краткое описание работы:

#### **1. Выборка из одной таблицы.**

1.1 Выбрать из произвольной таблицы данные и отсортировать их по двум произвольным имеющимся в таблице признакам (разные направления сортировки).

1.2 Выбрать из произвольной таблицы те записи, которые удовлетворяют условию отбора (where). Привести 2-3 запроса.

1.3 Привести примеры 2-3 запросов с использованием агрегатных функций (count, max, sum и др.) с группировкой и без группировки.

1.4 Привести примеры подведения подытога с использованием GROUP BY [ALL] [ CUBE | ROLLUP](2-3 запроса)

1.5 Выбрать из таблиц информацию об объектах, в названиях которых нет заданной последовательности букв (LIKE).

#### **2. Выборка из нескольких таблиц.**

2.1 Вывести информацию подчиненной (дочерней) таблицы, заменяя коды (значения внешних ключей) соответствующими символьными значениями из родительских таблиц. Привести 2-3 запроса с использованием классического подхода соединения таблиц (where).

2.2. Реализовать запросы пункта 2.1 через внутреннее соединение inner join.

2.3. Левое внешнее соединение left join. Привести 2-3 запроса.

2.4. Правое внешнее соединение right join. Привести 2-3 запроса

2.5. Привести примеры 2-3 запросов с использованием агрегатных функций и группировки.

2.6. Привести примеры 2-3 запросов с использованием группировки и условия отбора групп (Having).

2.7. Привести примеры 3-4 вложенных (соотнесенных, с использованием IN, EXISTS) запросов.

#### **3. Представления**

3.1 На основе любых запросов из п. 2 создать два представления (VIEW).

3.2 Привести примеры использования общетабличных представлений (СТЕ) (2-3 запроса)

#### **4. Функции ранжирования**

- 4.1 Привести примеры 3-4 запросов с использованием ROW\_NUMBER, RANK, DENSE\_RANK (с PARTITION BY и без)
5. Объединение, пересечение, разность
- 5.1 Привести примеры 3-4 запросов с использованием UNION / UNION ALL, EXCEPT, INTERSECT. Данные в одном из запросов отсортируйте по произвольному признаку.
6. Использование CASE, PIVOT и UNPIVOT.
- 6.1 Привести примеры получения сводных (итоговых) таблиц с использованием CASE
- 6.2 Привести примеры получения сводных (итоговых) таблиц с использованием PIVOT и UNPIVOT.
7. Рекурсивный запрос
- 7.1 Приведите пример рекурсивного запроса с использованием CTE.

#### **Индивидуальное задание №4 (И-ПК-4.4)**

##### *Пример задания*

1. Создать 4 различных хранимых процедуры в соответствии с индивидуальным заданием:
  - a) Процедура без параметров, возвращающая набор данных
  - b) Процедура с входным параметром, возвращающая набор данных
  - c) Процедура с входными и выходными параметрами
  - d) Вложенная процедура
2. 3 пользовательских функции:
  - a) Функция, возвращающая скалярное значение, содержащая блок команд, использующая входные параметры
  - b) Inline-функция, использующая входные параметры
  - c) Multi-statement функция

#### **Индивидуальное задание №5 (И-ПК-4.4)**

##### *Пример задания*

Создать 3 триггера:

- a) Триггер типа INSTEAD на добавление записи в какую-либо из таблиц
- b) Триггер типа AFTER на изменение записи
- c) Триггер типа AFTER на удаление

#### **Индивидуальное задание №6 (И-ПК-4.3)**

##### *Пример задания*

Создать пользователей БД и роли БД с различными правами доступа в соответствии с индивидуальным заданием

## **2. Список вопросов и (или) заданий для проведения промежуточной аттестации**

Список вопросов к зачету

1. История развития БД.
2. Архитектура СУБД.
3. Основные функции СУБД (управление данными во внешней памяти, буферизация, управление транзакциями, журнализация, поддержка языков).
4. Классификация моделей данных.
5. Иерархическая модель данных
6. Сетевая модель данных
7. Модель "Сущность-Связь". Виды связей.



8. Физическая организация данных (Куча, Хеширование).
9. Физическая организация данных (Индексные файлы).
10. Физическая организация данных (В - деревья).
11. Физическая организация данных (Файлы с плотным индексом, Записи переменной длины, Структуры данных для поиска по не ключевым полям).
12. Модели физической организации данных при бесфайловой организации.
13. Реляционная модель данных. Фундаментальные свойства отношений. Целостность.
14. Манипуляционная часть реляционной модели данных. Реляционная алгебра. Примеры.
15. Манипуляционная часть реляционной модели данных. Реляционное исчисление.  
Примеры.
16. Язык запросов SQL. Простейшие примеры.
17. Язык запросов SQL. Объединение, пересечение, разность.
18. Язык запросов SQL. Примеры (группировка, подзапросы).
19. Язык запросов SQL. Виды соединения таблиц. Примеры.
20. Язык запросов SQL. Функции ранжирования.
21. Язык запросов SQL. CTE, рекурсивные запросы.
22. Язык запросов SQL. Pivot, Unpivot
23. Язык манипулирования данными DML. Вставка, обновление, удаление записей. Примеры.
24. Язык описания данных DDL. Типы ограничений. Примеры.
25. Проектирование баз данных. Нормальные формы (1NF, 2NF, 3NF).
26. Проектирование баз данных. Нормальные формы (BCNF, 4NF, 5NF)
27. Параллельные операции над базами данных. Конфликтные ситуации.
28. Модель транзакций. Алгоритм проверки сериализуемости.
29. Журнализация изменений в базе данных. Индивидуальный откат, мягкий и жесткий сбои.
30. Постреляционная и многомерная модели данных.
31. Представления (View). Виды представлений.
32. Триггеры
33. Хранимые процедуры
34. Хранимые функции
35. CTE (общетабличные представления)
36. Модели доступа к данным
37. Права доступа к данным

## Приложение № 2 к рабочей программе дисциплины «Базы данных»

### Методические указания для студентов по освоению дисциплины

При изучении дисциплины должны решаться следующие образовательные задачи:

- изучение основ организации БД и СУБД, их функций и отличий от файловых систем;
- изучение многоуровневой архитектуры современных СУБД, моделей данных и языковых средств СУБД для различных моделей данных;
- изучение теоретических основ разработки реляционных баз данных и получение навыков их практического применения при проектировании базы данных и разработке приложения баз даны;
- приобретение навыков проектирования структур БД путем нормализации схемы предметной области, знакомство со средствами автоматизации проектирования БД;
- приобретение навыков формирования запросов к БД средствами языка SQL;
- приобретение навыков разработки и реализации законченных приложений баз данных под управлением СУБД MSSQL SERVER.
- приобретение навыков разработки и реализации законченных Windows-приложений b Web–приложений для доступа к БД.

Изучение теоретического материала поддерживается практическими и лабораторными занятиями. Часть вопросов, хорошо обеспеченных литературой и не представляющих сложности для изучения ввиду того, что их содержание основано на теоретическом материале и практическом опыте программирования, полученном в рамках других дисциплин, вынесена на самостоятельное изучение.

Курс призван также повысить общую эрудицию студентов, показать методы создания моделей и применения средств ИКТ в различных областях.

На лекциях используется «проблемный» подход к изложению материала: материал каждой лекции иллюстрируется примерами, рассматриваются нестандартные ситуации, требующие решения с использованием рассматриваемого материала. При этом студенты должны активно участвовать в обсуждении вопросов, выработке решений, предлагаемые студентами решения, обсуждаются, анализируются и оцениваются в ходе лекции. Предлагается рассматривать не только «верные», оптимальные решения, но и решения, приводящие к ошибкам. По каждому рассматриваемому на лекции вопросу следует предложить задачи для самостоятельного решения и вопросы для самостоятельного изучения с использованием разработанных материалов.

На практических занятиях используются следующие методы обучения и контроля усвоения материала:

- 1) выполнение заданий по теме занятия сопровождается контрольным опросом;
- 2) обсуждение различных вариантов решения, предложенных студентами, сравнение решений, анализ возможных ситуаций.

Рекомендуется использовать «защиту» выполненных домашних и контрольных заданий.

Кроме того, рекомендуется рассмотреть примеры разработанных баз данных, демонстрирующие результаты ошибок проектирования (аномалии, являющиеся следствием неликвидированной избыточности), а также различные варианты реализации запросов с использованием оператора SELECT, различных его возможностей.

Получаемые на лекциях и практических занятиях знания о языке *SQL* рекомендуется закреплять с помощью книги «*SQL* в примерах и задачах», сайта [www.sql-ex.ru](http://www.sql-ex.ru).

При создании БД рекомендуется использовать справочную систему СУБД MSSQL SERVER, примеры и рекомендации по решению задач, приведенные в электронных пособиях по курсу, указанных в списке дополнительной литературы.

Студенту рекомендуется следующая схема подготовки к практическому занятию:

- 1) проработать конспект лекций;
- 2) проанализировать основную и дополнительную литературу, рекомендованную по изучаемому разделу;
- 3) при необходимости найти дополнительную информацию в Internet, на сайтах электронных библиотек;
- 4) проанализировать варианты решений, предложенные преподавателем, найденные в дополнительных источниках;
- 5) при затруднениях сформулировать вопросы к преподавателю.

Студенту рекомендуется следующая схема подготовки к лекции:

- 1) проработать конспект лекций;
- 2) изучить материал, предложенный для самостоятельного изучения;
- 3) выполнить предложенные преподавателем задания;
- 4) при затруднениях задать вопросы к преподавателю при проведении индивидуальных консультаций.

Рекомендуется при выполнении домашних заданий и подготовке к контрольным работам рассмотреть возможность защиты предложенных решений, подготовить документацию и «презентацию» работы.