

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Андрей Драгомирович Хлудков
Должность: директор
Дата подписания: 23.06.2026 16:54:22
Уникальный программный ключ:
880f7c07c583b07b775f6604c39281b15e9f12

Федеральное государственное бюджетное образовательное учреждение
высшего образования

**РОССИЙСКАЯ АКАДЕМИЯ НАРОДНОГО ХОЗЯЙСТВА и
ГОСУДАРСТВЕННОЙ СЛУЖБЫ при ПРЕЗИДЕНТЕ РОССИЙСКОЙ
ФЕДЕРАЦИИ**

СЕВЕРО-ЗАПАДНЫЙ ИНСТИТУТ УПРАВЛЕНИЯ

Факультет среднего профессионального образования

УТВЕРЖДЕНА
решением цикловой (методической)
комиссии общепрофессиональных
дисциплин и по профессиональным
модулям специальности 09.02.07
Информационные системы и
программирование
Протокол от 31.10.2025 № 2

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

МДК.01.01 Проектирование и разработка баз данных

Специальность – 09.02.11 Разработка и управление программным обеспечением

Профиль – на базе основного общего образования

Квалификация – Программист

Форма обучения – очная

Год набора – 2026

Санкт-Петербург 2025 год

Автор-составитель: Вилков Владислав Евгеньевич, преподаватель ФСПО СЗИУ РАНХиГС.

СОДЕРЖАНИЕ

1. Общие положения	4
1.1. Область применения программы	4
1.2. Место дисциплины в структуре основной профессиональной образовательной программы	4
1.3. Цели и задачи учебной дисциплины	4
1.4. Планируемые результаты обучения по дисциплине	4
2. Структура и содержание дисциплины	11
2.1. Объем учебной дисциплины и виды работ	11
2.2. Тематический план и содержание дисциплины	11
2.3. Регламент распределения видов работ по дисциплине с ДОТ	15
3. Материалы текущего контроля успеваемости и промежуточной аттестации обучающихся	16
3.1. Формы и методы текущего контроля успеваемости обучающихся и промежуточной аттестации.....	16
3.2. Оценочные средства текущего контроля успеваемости обучающихся	17
3.3. Оценочные средства промежуточной аттестации обучающихся	20
4. Методические указания для обучающихся по освоению дисциплины	22
5. Учебная литература и ресурсы информационно-телекоммуникационной сети «Интернет»	22
6. Материально-техническая база, информационные технологии, программное обеспечение и информационные справочные системы	25

1 Общие положения

1.1 Область применения программы

Рабочая программа учебной дисциплины «Проектирование и разработка баз данных» является частью основной профессиональной образовательной программы по специальности СПО 09.02.11 «Разработка и управление программным обеспечением».

1.2 Место дисциплины в структуре основной профессиональной образовательной программы

Учебная дисциплина «Проектирование и разработка баз данных» является частью профессиональной подготовки, входит в общепрофессиональный цикл дисциплин. Базируется на таких дисциплинах, как «Информатика», «Операционные системы и среды», «Основы алгоритмизации и программирования» является основополагающей для такой дисциплины, как «Разработка программных модулей».

Дисциплина изучается на 2 курсе в 4 семестре.

1.3 Цели и задачи учебной дисциплины

Цель дисциплины «Проектирование и разработка баз данных» - формирование у обучающихся системных знаний и практических навыков в области проектирования, разработки, администрирования и оптимизации баз данных, включая владение современными инструментами и технологиями работы с реляционными и NoSQL-системами, а также способность создавать эффективные, безопасные и масштабируемые решения для хранения и обработки данных в профессиональной деятельности.

Задачи дисциплины:

- Овладение понятийным аппаратом теории баз данных (изучение ключевых терминов и определений: сущность, атрибут, связь, нормализация, транзакция и др.; понимание эволюции моделей данных и тенденций развития СУБД; осознание роли баз данных в современных информационных системах).
- Формирование знаний в области моделирования данных (освоение методов анализа предметной области; изучение принципов построения концептуальной, логической и физической моделей данных; работа с нотациями ER-диаграмм; понимание различий реляционной, иерархической, сетевой и NoSQL-моделей).
- Освоение принципов проектирования реляционных баз данных (изучение норм нормализации до 3NF и BCNF; проектирование схем с учётом целостности данных;

определение типов данных, ключей, индексов; разработка ограничений и правил ссылочной целостности).

- Приобретение навыков работы с языком SQL (освоение синтаксиса DDL, DML, DCL и TCL-команд; написание сложных запросов с JOIN, подзапросами, агрегатными функциями; создание представлений, хранимых процедур, функций и триггеров; оптимизация запросов через анализ планов выполнения).

- Развитие компетенций в области администрирования СУБД (установка и настройка серверов баз данных; управление пользователями и правами доступа; резервное копирование и восстановление данных; мониторинг производительности; настройка параметров конфигурации СУБД).

- Овладение методами оптимизации баз данных (проектирование эффективных индексов; денормализация с обоснованием; секционирование и шардирование данных; кэширование; настройка параметров СУБД для повышения производительности).

- Изучение особенностей NoSQL-систем (освоение моделей данных: документо-ориентированные, «ключ-значение», колоночные, графовые; понимание сценариев применения MongoDB, Redis, Cassandra и др.; написание запросов на языках специфических для NoSQL-СУБД; оценка компромиссов между согласованностью, доступностью и устойчивостью — CAP-теорема).

- Формирование навыков обеспечения безопасности баз данных (реализация механизмов аутентификации и авторизации; шифрование данных на уровне столбцов и таблиц; аудит доступа и журналирование; защита от SQL-инъекций; соблюдение стандартов GDPR, PCI DSS в контексте хранения данных).

- Освоение инструментов проектирования и разработки (работа с CASE-средствами для моделирования данных; использование IDE для СУБД; интеграция баз данных с прикладными системами через API и ORM-фреймворки; автоматизация развёртывания и миграции схем).

- Ориентирование в перспективных направлениях развития баз данных (изучение технологий распределённых баз данных и блокчейн-хранилищ; анализ роли ИИ и ML в оптимизации запросов и предсказательной аналитике; понимание тенденций облачных СУБД и Serverless-решений; оценка влияния квантовых вычислений на хранение данных).

1.4 Планируемые результаты обучения по дисциплине

Перечень компетенций

Код ОК, ПК	Уметь	Знать	Владеть навыками
ОК.01	Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам	актуальный профессиональный и социальный контекст, в котором приходится работать и жить; основные источники информации и ресурсы для решения задач и проблем в профессиональном и/или социальном контексте; алгоритмы выполнения работ в профессиональной и смежных областях; методы работы в профессиональной и смежных сферах; структуру плана для решения задач; порядок оценки результатов решения задач профессиональной деятельности	-
ОК.02	Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности	номенклатура информационных источников, применяемых в профессиональной деятельности; приемы структурирования информации; формат оформления результатов поиска информации, современные средства и устройства информатизации; порядок их применения и программное обеспечение в профессиональной деятельности в том числе с использованием цифровых средств	-
ОК.03	Планировать и реализовывать собственное профессиональное и личностное развитие, предпринимательскую	содержание актуальной нормативно-правовой документации; современная научная и профессиональная терминология;	-

	деятельность в профессиональной сфере, использовать знания по правовой и финансовой грамотности в различных жизненных ситуациях	возможные траектории профессионального развития и самообразования; основы предпринимательской деятельности; основы финансовой грамотности; правила разработки бизнес-планов; порядок выстраивания презентации; кредитные банковские продукты	
ОК.04	Эффективно взаимодействовать и работать в коллективе и команде	психологические основы деятельности коллектива, психологические особенности личности; основы проектной деятельности	-
ОК.05	Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста	особенности социального и культурного контекста; правила оформления документов и построения устных сообщений	-
ОК.06	Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных российских духовно-нравственных ценностей, в том числе с учетом гармонизации межнациональных и межрелигиозных отношений, применять стандарты антикоррупционного поведения	сущность гражданско-патриотической позиции, общечеловеческих ценностей; значимость профессиональной деятельности по специальности; стандарты антикоррупционного поведения и последствия его нарушения	-
ОК.07	Содействовать сохранению окружающей среды, ресурсосбережению, применять знания об изменении климата, принципы бережливого	правила экологической безопасности при ведении профессиональной деятельности; основные ресурсы, задействованные в	-

	производства, эффективно действовать в чрезвычайных ситуациях	профессиональной деятельности; пути обеспечения ресурсосбережения; принципы бережливого производства; основные направления изменения климатических условий региона	
ОК.08	Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности	роль физической культуры в общекультурном, профессиональном и социальном развитии человека; основы здорового образа жизни; условия профессиональной деятельности и зоны риска физического здоровья для специальности; средства профилактики перенапряжения	-
ОК.09	Пользоваться профессиональной документацией на государственном и иностранном языках	правила построения простых и сложных предложений на профессиональные темы; основные общеупотребительные глаголы (бытовая и профессиональная лексика); лексический минимум, относящийся к описанию предметов, средств и процессов профессиональной деятельности; особенности произношения; правила чтения текстов профессиональной направленности	-
ПК 1.1 Проектировать базы данных	– анализировать предметную область и выделять основные сущности; – определять требования к базе данных;	– основные положения теории баз данных, хранилищ данных, баз знаний; – основные принципы структуризации и нормализации базы данных;	– разработки концептуальной модели базы данных; – разработки инфологической модели базы данных;

	<ul style="list-style-type: none"> – разрабатывать концептуальную, логическую и физическую модели баз данных; – проектировать схему базы данных; – работать с современными case-средствами проектирования баз данных; – определять связи между таблицами; – определять типы данных для полей таблиц; – оформление документации на спроектированную базу данных разработки схемы базы данных, используя NoSQL модели данных, такие как документо-ориентированные, ключ-значение, колоночные и др. 	<ul style="list-style-type: none"> – основные принципы построения концептуальной, логической и физической модели данных; – методы описания схем баз данных в современных системах управления базами данных; – структуру данных систем управления базами данных, основные понятия и принципы проектирования баз данных; – структуру реляционной базы данных; – язык SQL и особенности его реализации в различных системах управления базами данных; – оптимизацию производительности баз данных принципы безопасности хранения данных 	<ul style="list-style-type: none"> – разработки физической модели базы данных; – разработки требований к базе данных – нормализация структуры базы данных – документирования схемы базы данных, включая диаграммы ER и описания таблиц; документирования прав доступа и безопасности базы данных, включая учетные записи пользователей и их роли
<p>ПК 1.2 Разрабатывать объекты баз данных в соответствии с результатами анализа предметной области</p>	<ul style="list-style-type: none"> – разрабатывать объекты баз данных – создавать таблицы, индексы, ограничения и другие объекты базы данных – оптимизировать запросы к базе данных для повышения производительности – разрабатывать хранимые процедуры и триггеры для баз данных; разрабатывать необходимые для различных групп пользователей представления 	<ul style="list-style-type: none"> – основы реляционной модели данных – язык SQL и его основные команды – принципы нормализации баз данных – принципы работы с различными СУБД – общий подход к организации представлений, таблиц, индексов и кластеров; – методы организации целостности данных; способы контроля доступа к данным и управления привилегиями 	<p>работы с различными объектами базы данных</p>

<p>ПК 1.3 Реализовывать базу данных в конкретной системе управления базами данных</p>	<ul style="list-style-type: none"> – разрабатывать объекты базы данных, такие как таблицы, индексы и связи между ними; – программировать и создавать хранимые процедуры, функции и триггеры для обработки данных; – управлять данными в базе данных, включая ввод, обновление и удаление данных; – оптимизировать запросы и проводить мониторинг производительности базы данных; – работать с NoSQL базами данных; – использовать запросы для работы с данными в NoSQL базах данных; оптимизировать производительность NoSQL баз данных. 	<ul style="list-style-type: none"> – основные принципы создания объектов базы данных; – синтаксис и основные приемы работы с SQL; – методы оптимизации запросов и повышения производительности базы данных; – основные принципы управления данными и обслуживания базы данных; – основные принципы работы NoSQL баз данных и их моделей данных; – преимущества и недостатки NoSQL технологий по сравнению с реляционными базами данных; – методы оптимизации производительности NoSQL баз данных; основные принципы управления данными и обслуживания NoSQL баз данных. 	<ul style="list-style-type: none"> – создания таблиц базы данных с определением структуры и типов данных для каждого атрибута; – определения первичных и внешних ключей для установления связей между таблицами; – создания индексов для оптимизации запросов и повышения производительности; – разработки хранимых процедур, функций и триггеров для обработки данных и поддержки бизнес-логики; – ввода, обновления и удаления данных в соответствии с требованиями бизнес-процессов; – оптимизации запросов для повышения производительности системы; – создания баз данных на основе NoSQL технологий – создания запросов для работы с данными в NoSQL базах данных; оптимизации производительности NoSQL баз данных, используя индексы и другие техники
<p>ПК 1.4 Администрировать базы данных</p>	<ul style="list-style-type: none"> – устанавливать и настраивать СУБД; – создавать и удалять базы данных; – создавать пользователей и назначать права доступа; – оптимизировать запросы к базе данных; – обеспечивать безопасность баз данных; 	<ul style="list-style-type: none"> – архитектуру СУБД; – основные принципы администрирования баз данных; – методы мониторинга и оптимизации работы баз данных; – принципы резервного копирования и восстановления баз данных; 	<ul style="list-style-type: none"> – установки и настройки СУБД; – создания и удаления баз данных; – восстановления баз данных; – резервного копирования баз данных; – создания пользователей и назначения прав доступа;

	<ul style="list-style-type: none"> – создавать и настраивать базы данных в соответствии с требованиями бизнеса; – управлять транзакциями и контролировать целостность данных; – обеспечивать безопасность и управлять доступом к данным; – создавать и восстанавливать резервные копии данных; – работать с индексами и оптимизировать производительность запросов; – нормализовать базы данных и проектировать эффективные структуры данных; – мониторить и анализировать производительность баз данных; – работать с нереляционными базами данных и выбирать наиболее подходящий тип базы данных для конкретной задачи 	<ul style="list-style-type: none"> – методы защиты баз данных от внешних угроз; – особенности работы с различными СУБД; – Язык SQL (Structured Query Language); – управление транзакциями и контроль целостности данных; – управление доступом и безопасностью баз данных; – резервное копирование и восстановление данных; – оптимизацию производительности баз данных; – работу с индексами и оптимизация запросов; – мониторинг и анализ производительности; – принципы работы с реляционными базами данных; – принципы работы с нереляционными базами данных 	<ul style="list-style-type: none"> – оптимизации запросов к базе данных мониторинга и обслуживания NoSQL баз данных, включая резервное копирование и восстановление данных.
<p>ПК.1.5 Защищать информацию в базе данных с использованием технологий защиты информации по специальности</p>	<ul style="list-style-type: none"> – разрабатывать и внедрять системы защиты баз данных от несанкционированного доступа; – разрабатывать и внедрять системы резервного копирования и восстановления баз данных; – проводить аудит безопасности баз данных; – устанавливать и настраивать механизмы аутентификации и 	<ul style="list-style-type: none"> – методы защиты баз данных от несанкционированного доступа; – методы создания и восстановления резервных копий баз данных; – особенности работы с различными типами СУБД; – методы проведения аудита безопасности баз данных; – принципы криптографии и методов шифрования данных; 	<ul style="list-style-type: none"> – использования стандартных методов защиты объектов базы данных; – разработки и внедрения систем защиты баз данных от несанкционированного доступа; – разработки и внедрения систем резервного копирования и восстановления баз данных; – аудита безопасности баз данных

	<p>авторизации пользователей;</p> <ul style="list-style-type: none"> – создавать и управлять ролями и правами доступа к данным; – шифровать данные и обеспечивать их конфиденциальность; – контролировать целостность данных и обнаруживать изменения; – использовать механизмы аудита для отслеживания доступа к данным; – использовать механизмы мониторинга для обнаружения угроз безопасности; – создавать и управлять защищенными соединениями с базой данных; – использовать механизмы защиты от SQL-инъекций и других видов атак; – создавать и управлять бэкапами и резервными копиями данных; обеспечивать безопасность базы данных при использовании облачных сервисов 	<ul style="list-style-type: none"> – стандарты и протоколы безопасности, таких как SSL/TLS, SSH, Kerberos и др.; – методы аутентификации и авторизации пользователей, включая использование паролей, сертификатов и биометрических данных; – методы контроля доступа, включая создание ролей и групп пользователей, управление правами доступа и аудит доступа к данным; – методы обнаружения и предотвращения атак, включая защиту от SQL-инъекций, DoS/DDoS-атак и других угроз безопасности; – методы мониторинга и анализа журналов событий для обнаружения угроз безопасности и анализа производительности базы данных; – методы создания и управления защищенными соединениями с базой данных, включая VPN-туннели и SSL-шифрование; – методы создания и управления бэкапами и резервными копиями данных, включая использование инкрементальных и дифференциальных бэкапов; – методы обеспечения безопасности базы данных при использовании облачных 	
--	--	--	--

		сервисов, включая защиту от утечки данных и управление доступом к облачным ресурсам; законодательство и стандарты безопасности, такие как GDPR, HIPAA, PCI DSS и др.	
--	--	---	--

В результате освоения учебной дисциплины студент должен:

иметь практический опыт	<ul style="list-style-type: none"> – проектирования концептуальных, логических и физических моделей данных; – разработки схем реляционных баз данных (определение сущностей, атрибутов, связей, ключей, индексов); – работы с системами управления базами данных (СУБД) — установка, настройка, администрирование; – написания и оптимизации SQL запросов (DDL, DML, DCL, TCL); – создания и отладки хранимых процедур, функций и триггеров; – нормализации и денормализации структур данных; – проектирования NoSQL решений (документно ориентированные, «ключ значение», колоночные, графовые модели); – настройки механизмов безопасности баз данных (аутентификация, авторизация, шифрование); – резервного копирования и восстановления данных; – работы с CASE средствами для моделирования данных и генерации схем; – интеграции баз данных с прикладными системами (через API, ORM фреймворки); – мониторинга и оптимизации производительности баз данных; – анализа и устранения типовых ошибок и узких мест в работе СУБД; – документирования структур данных и процессов работы с БД.
уметь	<ul style="list-style-type: none"> – проектирования концептуальных, логических и физических моделей данных; – разработки схем реляционных баз данных (определение сущностей, атрибутов, связей, ключей, индексов); – работы с системами управления базами данных (СУБД) — установка, настройка, администрирование; – написания и оптимизации SQL запросов (DDL, DML, DCL, TCL); – создания и отладки хранимых процедур, функций и триггеров; – нормализации и денормализации структур данных; – проектирования NoSQL решений (документно ориентированные, «ключ значение», колоночные, графовые модели); – настройки механизмов безопасности баз данных (аутентификация, авторизация, шифрование); – резервного копирования и восстановления данных; – работы с CASE средствами для моделирования данных и генерации схем; – интеграции баз данных с прикладными системами (через API, ORM фреймворки); – мониторинга и оптимизации производительности баз данных; – анализа и устранения типовых ошибок и узких мест в работе СУБД; – документирования структур данных и процессов работы с БД. –

знать	<ul style="list-style-type: none"> – проектирования концептуальных, логических и физических моделей данных; – разработки схем реляционных баз данных (определение сущностей, атрибутов, связей, ключей, индексов); – работы с системами управления базами данных (СУБД) — установка, настройка, администрирование; – написания и оптимизации SQL запросов (DDL, DML, DCL, TCL); – создания и отладки хранимых процедур, функций и триггеров; – нормализации и денормализации структур данных; – проектирования NoSQL решений (документо ориентированные, «ключ значение», колоночные, графовые модели); – настройки механизмов безопасности баз данных (аутентификация, авторизация, шифрование); – резервного копирования и восстановления данных; – работы с CASE средствами для моделирования данных и генерации схем; – интеграции баз данных с прикладными системами (через API, ORM фреймворки); – мониторинга и оптимизации производительности баз данных; – анализа и устранения типовых ошибок и узких мест в работе СУБД; – документирования структур данных и процессов работы с БД.
-------	---

2 Структура и содержание дисциплины

2.1 Объем учебной дисциплины и виды работ

Виды учебной работы	Объем учебной работы, час.
Учебная нагрузка обучающихся всего, в том числе:	116
лекции	16
практические занятия	72
курсовая работа	20
самостоятельная работа обучающихся	-
консультации	2
промежуточная аттестация	6
Форма промежуточной аттестации	Зачет с оценкой

2.2 Тематический план и содержание дисциплины

№ п/п	Наименование тем (разделов)	Содержание тем (разделов)	Распределение часов			Формируемые компетенции	Формы текущего контроля
			Л	ПР	СРС		
1	Тема 1.1. Язык структурированных запросов	<p>Содержание учебного материала</p> <p>Общий подход к организации представлений, таблиц, индексов и кластеров. Индексы и оптимизация запросов. Понятие индексов. Назначение индексов. Создание индексов. Оптимизация запросов. Анализ производительности запросов. Использование EXPLAIN для анализа выполнения запроса. Понятие хранимой процедуры. Создание и синтаксис хранимых</p>	8	30	-	ОК.01 - ОК.09; ПК 1.1 – ПК 1.5	Т, ПЗ, О

№ п/п	Наименование тем (разделов)	Содержание тем (разделов)	Распределение часов			Формируемые компетенции	Формы текущего контроля
			Л	ПР	СРС		
		<p>процедур. Основные конструкции хранимой процедуры: условные конструкции и циклы. Вызов хранимых процедур. Управление хранимыми процедурами. Курсорные операции в хранимых процедурах. Обработка ошибок внутри хранимых процедур. Генерация исключений и сообщений об ошибках. Защита от SQL-инъекций с помощью хранимых процедур. Использование параметризованных запросов. Понятие триггера. Синтаксис создания триггеров. Указание событий, вызывающих срабатывание триггеров: вставка, обновление, удаление. Механизм срабатывания триггера. Доступ к измененным данным. Управление триггерами. Обработка ошибок внутри триггера. Генерация исключений и сообщений об ошибках.</p> <p>Транзакции и блокировка. Понятие транзакции и ACID-принципы. Команды управления транзакциями. Блокировки и уровни изоляции транзакций. Проблемы, связанные с параллелизмом. Управление транзакциями и контроль целостности данных. Отладка и мониторинг транзакций и блокировок. Инструменты для отслеживания состояния транзакций. Анализ блокировок и устранение тупиков.</p> <p>В том числе практических и лабораторных занятий</p> <p>1. Создание и использование индексов для ускорения поиска. Удаление и пересоздание индексов. Оптимизация запросов с использованием EXPLAIN. Применение индексов в сложных запросах. Использование частичных индексов и индексов по выражениям. Работа с составными индексами.</p>					

№ п/п	Наименование тем (разделов)	Содержание тем (разделов)	Распределение часов			Формируемые компетенции	Формы текущего контроля
			Л	ПР	СРС		
		<p>2. Разработка необходимых для различных групп пользователей представления</p> <p>3. Анализ логов ошибок и медленных запросов. Оптимизация запросов. Построение и анализ плана выполнения запросов. Оптимизация структуры таблиц и индексов. Профилирование запросов. Мониторинг и анализ производительности запросов</p> <p>4. Создание и использование простых пользовательских функций. Создание пользовательских функций для работы с текстовыми данными и датами. Вложенные пользовательские функции. Обработка ошибок в пользовательских функциях. Использование пользовательских функций в запросах. Создание пользовательских функций для работы с JSON-данными.</p> <p>5. Создание простой хранимой процедуры для вставки данных. Создание хранимой процедуры для обновления определенного поля в таблице на основании некоторого критерия. Создание хранимой процедуры, принимающую параметры для фильтрации данных и возвращающую результат в виде набора строк. Создание хранимой процедуры с использованием курсора для последовательной обработки записей. Создание хранимой процедуры со встроенной обработкой ошибок. Создание сложной хранимой процедуры с несколькими параметрами, выполняющую несколько операций над данными. Оптимизация хранимых процедур.</p> <p>6. Создание простого триггера для аудита изменений. Проверка корректности данных с помощью триггеров. Автоматическое заполнение полей с помощью триггера. Создание триггера, запрещающий удаление записей из таблицы, если они связаны с</p>					

№ п/п	Наименование тем (разделов)	Содержание тем (разделов)	Распределение часов			Формируемые компетенции	Формы текущего контроля
			Л	ПР	СРС		
		<p>другими таблицами. Создание триггера, который реализует каскадное обновление связанной информации. Создание триггера со сложной логикой, включающей обработку ошибок. Оптимизация триггера с использованием временных таблиц.</p> <p>7. Управление транзакциями. Настройка уровней изоляции транзакций. Анализ и решение проблемы грязного чтения. Неповторяемое чтение и фантомное чтение: диагностика и исправление. Автоматическое и ручное управление блокировками в SQL.</p>					
2	Тема 1.2. NoSQL базы данных	<p>Содержание учебного материала</p> <p>Основные понятия и история развития NoSQL технологий. Преимущества и недостатки NoSQL технологий по сравнению с реляционными базами данных. Типы NoSQL баз данных. Ключ-значение базы данных. Основные принципы работы ключ-значение баз данных. Пример использования Redis: установка, основные команды, типы данных. Применение и сценарии использования ключ-значение баз данных.</p> <p>Документно-ориентированные базы данных. Популярные системы: MongoDB, Couchbase, Firebase. Структура документов и схемы данных. Запросы и индексация в document-oriented базах. Реальные примеры использования.</p> <p>Колоночные базы данных. Архитектура колоночных баз данных. Области применения. Концепции колонок ориентированного подхода. Системы типа Cassandra, HBase.</p> <p>Графовые базы данных. Основные понятия графов: узлы, ребра, свойства. Примеры запросов к графам: язык запросов Cypher. Сценарии использования графовых баз данных.</p>	8	42	-	ОК.01 – ОК.09; ПК 1.1 – ПК 1.5	Т, ПЗ, О

№ п/п	Наименование тем (разделов)	Содержание тем (разделов)	Распределение часов			Формируемые компетенции	Формы текущего контроля
			Л	ПР	СРС		
		Проектирование схем данных в NoSQL. CAP-теорема и её значение. Подходы к денормализации данных. Паттерны проектирования для разных типов NoSQL баз данных. Управление консистентностью и доступностью данных. Методы оптимизации производительности NoSQL систем управления базами данных. Основные принципы управления данными и обслуживания NoSQL систем управления базами данных В том числе практических и лабораторных занятий 8. Работа с различными типами NoSQL систем управления базами данных 9. Создания запросов для работы с данными в NoSQL базах данных 10. Оптимизации производительности NoSQL систем управления баз данных, используя индексы и другие техники 11. Настройка и управление NoSQL системами управления базами данных					
		Итого часов:	16	72	-		

2.3. Регламент распределения видов работ по дисциплине с ДОТ

Данная дисциплина реализуется с применением дистанционных образовательных технологий (ДОТ). Распределение видов учебной работы, форматов текущего контроля представлены в Таблице 2.3.

Таблица 2.3. — Распределение видов учебной работы и текущей аттестации

Вид учебной работы	Формат проведения
Лекционные занятия	Частично с применением ДОТ
Практические занятия	Частично с применением ДОТ
Текущий контроль	Частично с применением ДОТ
Промежуточная аттестация	Контактная аудиторная работа
Формы текущего контроля	Формат проведения

Тестирование	Частично с применением ДОТ
Опрос	Контактная аудиторная работа
Практические задания	Частично с применением ДОТ

Доступ к системе дистанционных образовательных программ осуществляется каждым обучающимся самостоятельно с любого устройства на портале: <https://sziu-de.ranepa.ru>, в соответствии с их индивидуальным паролем и логином к личному кабинету/ профилю.

Текущий контроль, проводимый в системе дистанционного обучения, оцениваются как в системе дистанционного обучения, так и преподавателем вне системы.

Доступ к материалам лекций предоставляется в течение всего семестра по мере прохождения освоения программы. Доступ к каждому виду работ и количество попыток на выполнение задания предоставляется ограниченное время согласно регламенту дисциплины, опубликованному в системе дистанционного обучения. Преподаватель оценивает выполненные обучающимися работы не позднее 14 рабочих дней после окончания срока выполнения.

3 Материалы текущего контроля успеваемости и промежуточной аттестации обучающихся

3.1 Формы и методы текущего контроля успеваемости и промежуточной аттестации обучающихся

Формы текущего контроля успеваемости:

Опрос (О) позволяет выявить правильность ответа по содержанию, его последовательность, самостоятельность суждений и выводов, степень развития логического мышления.

Тестирование (Т) – задания, с вариантами ответов.

Критерии оценивания

Оценки «отлично» заслуживает студент, если он ответил правильно на 90-100% вопросов теста;

Оценки «хорошо» заслуживает студент, если он ответил правильно на часть вопросов 75%-89%;

Оценки «удовлетворительно» заслуживает студент, если он правильно ответил часть вопросов 50%-74%;

Оценки «неудовлетворительно» заслуживает студент, если он правильно ответил менее чем на 50% вопросов или не представлен тест на проверку.

Практическое задание (ПЗ) используется для закрепления теоретических знаний и отработки навыков и умений, способности применять знания при решении конкретных задач.

Критерии оценивания

Оценки «отлично» заслуживает студент, обнаруживший глубокое знание материала, умение свободно выполнять задания, понимающий взаимосвязь основных понятий темы;

Оценки «хорошо» заслуживает студент, обнаруживший полное знание материала; успешно выполняющий предусмотренные задания; и допустивший незначительные ошибки: неточность фактов, стилистические ошибки;

Оценки «удовлетворительно» заслуживает студент, обнаруживший знания основного материала в объеме, необходимом для дальнейшего изучения дисциплины. Справляющийся с выполнением заданий; допустивший погрешности в ответе, но обладающий необходимыми знаниями для их устранения под руководством преподавателя;

Оценки «неудовлетворительно» заслуживает студент, обнаруживший существенные пробелы в знании основного материала; не справляющийся с выполнением заданий, допустивший серьезные погрешности в ответах, нуждающийся в повторении основных разделов курса под руководством преподавателя.

Формы текущего контроля

№ п/п	Название темы	Формы текущего контроля успеваемости
1	Тема 1.1. Язык структурированных запросов	Т, ПЗ, О
2	Тема 1.2. NoSQL базы данных	Т, ПЗ, О

Примечание. В столбце «Форма текущего контроля успеваемости, промежуточной аттестации» перечисляются все используемые в учебном процессе по данной дисциплине формы контроля освоения материала. (Т – тестирование; ПЗ – практическое задание, О - опрос). Промежуточная аттестация проходит в виде собеседования по вопросам из перечня по дисциплине.

3.2 Оценочные средства текущего контроля успеваемости обучающихся

Тематика курсовых проектов (работ)

1. Проектирование и разработка базы данных для интернет-магазина

2. Проектирование и разработка базы данных для управления библиотечным фондом
3. Проектирование и разработка базы данных для системы управления университетом
4. Проектирование и разработка базы данных для системы бронирования гостиниц
5. Проектирование и разработка базы данных для ведения учета рабочего времени сотрудников
6. Проектирование и разработка базы данных для системы онлайн-курсов
7. Проектирование и разработка базы данных для управления складами
8. Проектирование и разработка базы данных для медицинской информационной системы
9. Проектирование и разработка базы данных для системы учета заявок и обращений клиентов
10. Проектирование и реализация NoSQL базы данных для проекта с большими данными
11. Проектирование и разработка графовой базы данных для социальной сети

Примеры типовых заданий для практических работ

Тема 1.1. Язык структурированных запросов

1. **Создание и оптимизация индексов.** Задание: в учебной базе данных (например, схема интернет-магазина) выполните:
 - создайте 3–4 индекса разных типов: простой, составной, частичный, индекс по выражению;
 - подготовьте 5–6 запросов, которые должны ускориться за счёт этих индексов;
 - для каждого запроса получите план выполнения до и после создания индекса (через EXPLAIN или аналог);
 - сравните метрики (оценочная стоимость, количество строк, тип доступа).
 Представьте таблицу: «Индекс», «Тип», «Запрос», «План до», «План после», «Эффект». Кратко (3–4 предложения) сформулируйте выводы: какие типы индексов оказались наиболее эффективными и в каких случаях.
2. **Разработка представлений для разных групп пользователей.** Задание: создайте 3 представления (VIEW) для ролей «менеджер по продажам», «аналитик», «администратор склада»:
 - ограничьте видимость колонок и строк в соответствии с правами доступа;

- используйте подзапросы и агрегации (SUM, COUNT, AVG) там, где это уместно;
- проверьте, что пользователь без прав не может обойти ограничения через JOIN с базовыми таблицами.

Приложите SQL-код представлений и примеры запросов от имени разных ролей с результатами. Кратко (2–3 предложения) объясните, как представления упрощают контроль доступа и унификацию отчётов.

3. **Анализ и оптимизация производительности запросов.** Задание: на основе логов медленных запросов (или искусственно замедленных запросов) выберите 3 проблемных запроса. Для каждого выполните:

- получите план выполнения (EXPLAIN ANALYZE);
- выявите узкие места (полное сканирование таблицы, неэффективные JOIN, отсутствие индексов);
- предложите и реализуйте улучшения (новые индексы, переформулировка запроса, изменение структуры таблицы);
- повторно замерьте производительность.

Представьте «до/после» в виде таблицы и краткого отчёта (5–7 предложений) с итоговым приростом производительности и рекомендациями по регулярному мониторингу.

4. **Разработка пользовательских функций.** Задание: создайте 3 пользовательские функции:

- функция для форматирования телефонных номеров (нормализация строки);
- функция для расчёта возраста по дате рождения с обработкой NULL и некорректных дат;
- функция для извлечения и валидации JSON-полей (например, получение адреса из JSON-колонки).

Включите обработку ошибок и генерацию понятных сообщений. Протестируйте функции на наборе тестовых данных (включая граничные и ошибочные значения). Приложите код функций, тесты и результаты. Кратко (3–4 предложения) обсудите, когда использование функций оправдано, а когда лучше использовать встроенные возможности или логику на уровне приложения.

5. **Создание хранимых процедур с разной логикой.** Задание: реализуйте 3 хранимые процедуры:

- простая вставка данных с валидацией входных параметров;
- обновление записей по критерию с возвратом количества изменённых строк;

- сложная процедура с курсором для последовательной обработки записей, обработкой ошибок и транзакционным контролем.

Протестируйте каждую процедуру, зафиксируйте сценарии успеха и ошибок. Приложите код, описание тестов и результаты. Кратко (3–4 предложения) сравните использование процедур и обычных запросов: в каких случаях процедуры повышают безопасность и удобство, а в каких — усложняют поддержку.

6. **Работа с триггерами для аудита и контроля целостности.** Задание: создайте 3 триггера:

- триггер для автоматического заполнения полей `created_at`, `updated_at`;
- триггер-аудит, который пишет изменения в отдельную таблицу истории (кто, когда, какие поля изменились);
- триггер, запрещающий удаление записей, если они связаны с активными заказами (проверка внешних зависимостей).

Протестируйте все сценарии: вставка, обновление, попытка удаления. Приложите код триггеров, примеры запросов и результаты. Кратко (3–4 предложения) оцените влияние триггеров на производительность и сложность отладки.

7. **Управление транзакциями и решение проблем параллелизма.** Задание: смоделируйте 3 сценария параллельного доступа:

- «грязное чтение» (чтение незакоммиченных данных) при разных уровнях изоляции;
- «неповторяемое чтение» (изменение строки между двумя чтениями в одной транзакции);
- «фантомное чтение» (появление новых строк между выборками).

Для каждого сценария:

- настройте нужный уровень изоляции;
- воспроизведите проблему;
- зафиксируйте поведение СУБД и результаты запросов;
- опишите, как избежать проблемы (выбор уровня изоляции, явные блокировки, паттерны приложения).

Представьте описание сценариев, SQL-команды и выводы. Кратко (4–5 предложений) сформулируйте рекомендации по выбору уровней изоляции для OLTP и аналитических нагрузок.

Тема 1.2. NoSQL базы данных

1. **Знакомство с Redis (ключ-значение).** Задание: установите Redis (локально или в контейнере) и выполните:

- основные команды: SET, GET, INCR, EXPIRE, HGET, HSET, LPUSH, RPOP;
- реализуйте простой кэш для результатов запросов: ключ — хеш параметров, значение — JSON-строка результата, TTL — 5 минут;
- продемонстрируйте работу кэша: первый запрос — «холодный», последующие — из кэша;
- добавьте обработку промахов кэша и инвалидацию при изменении данных.

Приведите команды, код клиента (на любом языке) и описание тестов. Кратко (3–4 предложения) укажите, в каких сценариях Redis даёт наибольший выигрыш и какие риски нужно учитывать.

2. **Работа с MongoDB (документо-ориентированная СУБД).** Задание: разверните MongoDB и выполните:

- создайте коллекцию с образцами документов (товары, заказы, клиенты) с разной структурой;
- напишите запросы на поиск, агрегацию, группировку (используя агрегационный конвейер: \$match, \$group, \$lookup);
- спроектируйте индексы для частых запросов и оцените их эффект через explain();
- продемонстрируйте денормализацию: хранение адреса доставки прямо в документе заказа вместо отдельной таблицы.

Приложите JSON-документы, запросы и результаты. Кратко (3–4 предложения) сравните нормализацию в реляционных БД и денормализацию в MongoDB: плюсы и минусы для производительности и целостности.

3. **Работа с Cassandra (колоночная СУБД).** Задание: запустите Cassandra (в Docker или тестовом кластере) и:

- спроектируйте таблицу для временных рядов событий (timestamp, device_id, metric_value, tags);
- учтите паттерн запросов: частые выборки по device_id и диапазону времени;
- определите первичный ключ и индексы, объясните, почему выбран именно такой ключ;
- выполните вставки и выборки, замерьте задержку.

Приложите CQL-код, схему таблицы и результаты тестов. Кратко (3–4 предложения) поясните, как CAP-теорема влияет на выбор Cassandra для таких задач.

4. **Работа с графовой базой данных (Neo4j / Cypher).** Задание: разверните Neo4j и:

- смоделируйте простой граф: узлы «Пользователь», «Товар», «Заказ», рёбра «купил», «рекомендует», «посмотрел»;
- напишите 3 запроса на Cypher: поиск рекомендаций, поиск путей между пользователями, подсчёт степени узлов;
- оцените производительность запросов при увеличении числа узлов (можно на синтетических данных).

Приложите схему графа, запросы на Cypher и результаты. Кратко (3–4 предложения) опишите, когда графовые БД предпочтительнее реляционных и NoSQL.

5. **Сравнение подходов и проектирование схемы под нагрузку.** Задание: для одной предметной области (например, каталог товаров и заказов) спроектируйте 3 варианта хранения:

- реляционная схема (таблицы, внешние ключи, индексы);
- документо-ориентированная схема (MongoDB) с денормализацией;
- колоночная схема (Cassandra) для аналитики и временных рядов.

Сравните по критериям: простота запросов, целостность данных, масштабирование, стоимость поддержки. Представьте схемы (ER-диаграмма, JSON-структура, CQL-таблица) и сравнительную таблицу. Кратко (5–7 предложений) сформулируйте, как выбор СУБД зависит от требований к консистентности, доступности и типу нагрузки (OLTP/OLAP).

6. **Оптимизация производительности NoSQL.** Задание: для выбранной NoSQL-системы (Redis, MongoDB или Cassandra) выполните:

- включите и проанализируйте логи медленных операций;
- найдите и устраните 2–3 узких места (отсутствие индексов, неэффективные запросы, неправильная модель данных);
- замерьте эффект до и после.

Приложите запросы, настройки, метрики и выводы. Кратко (3–4 предложения) предложите чек-лист регулярного обслуживания выбранной СУБД.

Примеры тестовых заданий

Часть 1. Задания с выбором одного правильного ответа

1. Что показывает вывод команды EXPLAIN в SQL?
 - а) Список всех таблиц в базе данных
 - б) План выполнения запроса, включая тип доступа к таблицам и оценочную стоимость
 - в) Историю изменений схемы базы данных
 - г) Список пользователей и их привилегий

2. В чём основное преимущество использования хранимых процедур?
 - а) Они всегда выполняются быстрее обычных запросов
 - б) Позволяют централизовать бизнес-логику, упростить контроль доступа и защитить от SQL-инъекций
 - в) Автоматически создают индексы для ускорения запросов
 - г) Гарантируют отсутствие блокировок при параллельном доступе

3. Что означает принцип ACID в контексте транзакций?
 - а) Автоматизация, контроль, интеграция, доступность
 - б) Атомарность, согласованность, изолированность, долговечность
 - в) Аутентификация, контроль, идентификация, документирование
 - г) Адаптивность, совместимость, инкапсуляция, детерминизм

Часть 2. Задания на установление соответствия

4. Установите соответствие между типом индекса и его назначением:

Тип индекса	Назначение
1. Простой индекс	А. Ускоряет поиск по одному полю
2. Составной индекс	Б. Ускоряет запросы, фильтрующие по нескольким полям одновременно
3. Частичный индекс	В. Покрывает только строки, удовлетворяющие условию (например, is_active = true)
4. Индекс по выражению	Г. Ускоряет поиск по результату вычисления (например, по LOWER(email))

Часть 3. Задания на последовательность действий

5. Установите правильную последовательность шагов при оптимизации медленного запроса:

- а) Получить план выполнения запроса (EXPLAIN)
- б) Выявить узкие места (сканирование таблиц, неэффективные соединения)
- в) Сформулировать гипотезы об улучшениях (индексы, переформулировка)
- г) Применить изменения и протестировать
- д) Замерить производительность до и после, зафиксировать эффект

Правильный порядок: а, б, в, г, д.

Часть 4. Ситуационные задачи

6. У вас есть таблица `orders` (10 млн строк) и частый запрос: `SELECT * FROM orders WHERE customer_id = ? AND status = 'new'`. Запрос выполняется медленно. Опишите пошаговый план оптимизации (не менее 5 шагов) и укажите, какие индексы и параметры запроса вы будете использовать.

7. Требуется реализовать аудит всех изменений в таблице `employees` (кто, когда, старое/новое значение). Предложите архитектуру решения: какие объекты БД (таблицы, триггеры, представления) нужны, как хранить историю, как обеспечить производительность при высокой нагрузке. Приведите краткий SQL-план (основные таблицы и логику триггера).

8. Для аналитической нагрузки (агрегации по временным рядам) нужно выбрать между реляционной БД и Cassandra. Перечислите 3–4 аргумента «за» и «против» каждого варианта и сформулируйте итоговое решение с учётом требований к консистентности и скорости чтения.

Примеры типовых вопросов для устного опроса

1. Объясните, как индексы влияют на производительность `SELECT` и на скорость операций модификации (`INSERT/UPDATE/DELETE`). Приведите пример, когда индекс может замедлить работу.

2. Расскажите, зачем нужны представления (`VIEW`) и в каких случаях их использование особенно полезно.

3. Опишите синтаксис и назначение основных конструкций хранимой процедуры: условные операторы, циклы, курсоры, обработка ошибок.

4. Что такое триггер и какие сценарии его применения вы считаете оправданными, а какие — рискованными?

5. Раскройте смысл принципов ACID и приведите пример, как каждый из них проявляется в реальной транзакции.
6. Какие проблемы параллелизма (грязное чтение, неповторяемое чтение, фантомы) существуют и как уровни изоляции транзакций помогают их решать?
7. В чём основные отличия реляционных и NoSQL баз данных? Приведите по 2–3 примера задач, где каждый тип СУБД будет оптимальным.
8. Объясните CAP-теорему и как она влияет на выбор NoSQL-решения для конкретного проекта.
9. Какие типы NoSQL СУБД вы знаете и для каких задач каждый из них лучше всего подходит?
10. Как подходы к проектированию схем различаются для реляционных, документо-ориентированных и колоночных баз данных? Приведите примеры.

3.3 Оценочные средства по дисциплине для промежуточной аттестации

Вопросы для подготовки к зачету с оценкой

1. Что такое индекс в базе данных? Назовите его основные функции.
1. В чём разница между кластерным и некластерным индексом?
2. Как создать составной индекс в SQL? Приведите пример синтаксиса.
3. Для чего используется команда EXPLAIN? Какие данные она выводит?
4. Перечислите основные этапы оптимизации SQL-запроса.
5. Что такое хранимая процедура? Каковы её преимущества перед обычными запросами?
6. Напишите синтаксис создания простой хранимой процедуры (на примере любой СУБД).
7. Какие конструкции управления потоком (условные, циклы) можно использовать в хранимых процедурах?
8. Как вызвать хранимую процедуру с параметрами? Приведите пример.
9. Что такое курсор в контексте хранимых процедур? Когда его целесообразно использовать?
10. Как обработать ошибку внутри хранимой процедуры? Приведите пример кода.
11. Каким образом хранимые процедуры помогают защититься от SQL-инъекций?
12. Что такое параметризованный запрос? Приведите пример.
13. Что такое триггер? Перечислите типы событий, вызывающих его срабатывание.
14. Напишите синтаксис создания триггера на событие INSERT.
15. Как в триггере получить доступ к новым/старым значениям полей?
16. Приведите пример триггера для аудита изменений в таблице.

17. Как отключить или удалить триггер?
18. Что означает аббревиатура ACID? Раскройте каждый компонент.
19. Перечислите команды управления транзакциями в SQL.
20. Что такое уровень изоляции транзакций? Назовите основные уровни.
21. В чём отличие READ COMMITTED от REPEATABLE READ?
22. Что такое «грязное чтение» (dirty read)? Как его предотвратить?
23. Что такое «фантомное чтение» (phantom read)? Приведите пример.
24. Как СУБД обнаруживает и разрешает мёртвые блокировки (deadlocks)?
25. Какие инструменты используются для мониторинга транзакций и блокировок?
26. Как проанализировать текущие блокировки в БД (на примере любой СУБД)?
27. Что такое частичный индекс? Приведите пример его создания.
28. Как оптимизировать запрос с помощью индекса по выражению?
29. Что такое представление (VIEW) в SQL? Для чего оно используется?
30. Раздел 2. NoSQL базы данных
31. В чём ключевые отличия NoSQL от реляционных БД? Назовите 3–4 преимущества и недостатка NoSQL.
32. Перечислите основные типы NoSQL-систем и приведите по одному примеру для каждого.
33. Что такое модель «ключ-значение»? Назовите сферы её применения.
34. Как установить и настроить Redis? Перечислите основные команды для работы с данными.
35. Какие типы данных поддерживает Redis? Приведите примеры использования.
36. Что такое документо-ориентированная БД? В чём её отличие от реляционной?
37. Как устроена структура документа в MongoDB? Что такое BSON?
38. Напишите пример запроса к MongoDB для выборки документов по условию.
39. Как создать индекс в MongoDB? Для чего он нужен?
40. Что такое колоночная БД? Назовите её ключевые особенности.
41. В каких сценариях целесообразно использовать Cassandra или HBase?
42. Что такое графовая БД? Перечислите основные понятия (узлы, рёбра и т. д.).
43. Какой язык запросов используется в графовых БД? Приведите пример запроса на Cypher.
44. В каких задачах применяют графовые БД? Назовите 2–3 примера.
45. Что утверждает CAP-теорема? Как она влияет на выбор NoSQL-системы?
46. Что такое денормализация данных? Почему она важна для NoSQL?
47. Назовите 2–3 паттерна проектирования для документо-ориентированных БД.

48. Как обеспечить консистентность данных в распределённой NoSQL-системе?
49. Перечислите методы оптимизации производительности NoSQL-систем.
50. Как настроить репликации и шардирование в NoSQL-СУБД (на примере любой системы)?

4. Методические указания для обучающихся по освоению дисциплины

Приступая к изучению дисциплины «Проектирование и разработка баз данных», студент должен ознакомиться с содержанием данной «Рабочей учебной программы дисциплины» с тем, чтобы иметь четкое представление о своей работе.

В первую очередь необходимо уяснить цель и задачи изучаемой дисциплины, оценить объем материала, познакомиться с предложенной и подобрать основную и дополнительную литературу, выявить наиболее важные проблемы, стоящие по вопросам изучаемой дисциплины.

Выполнение заданий осуществляется в соответствии с учебным планом и программой. Они должны выполняться в соответствии с методическими рекомендациями, выданными преподавателем, и представлены в установленные преподавателем сроки.

Работая с учебниками и учебными пособиями, целесообразно законспектировать тот материал, который не сообщался студентам на лекциях.

На занятиях лекционного и практического характера студентам для работы требуется тетрадь для записи лекций и заданий.

Для успешного овладения программой дисциплины необходимо выполнять следующие требования:

- посещать все лекционные и практические занятия;
- все рассматриваемые на лекциях и практических занятиях темы и вопросы обязательно фиксировать в тетради;
- в случае пропуска занятий по каким-либо причинам необходимо обязательно самостоятельно изучать соответствующий материал в Moodle, фиксируя записи в тетради, а также выполнять практические задания.

Подготовка к зачету с оценкой осуществляется по представленным в списке основной и дополнительной литературе. Рекомендуемые литература и интернет-ресурсы будут полезны при выполнении практических заданий и для подготовки к тестированиям.

Методические рекомендации по составлению конспекта

Конспект — сложный способ изложения содержания книги или статьи в логической последовательности. Внимательно прочитайте текст. Уточните в справочной литературе непонятные слова. При записи не забудьте вынести справочные данные на поля конспекта.

Выделите главное, составьте план, представляющий собой перечень заголовков, подзаголовков, вопросов, последовательно раскрываемых затем в конспекте.

Законспектируйте материал, четко следуя пунктам плана. При конспектировании старайтесь выразить мысль своими словами. Записи следует вести четко, ясно.

При оформлении конспекта необходимо стремиться к емкости каждого предложения.

Методические рекомендации по составлению опорного конспекта

Опорный конспект — вид внеаудиторной самостоятельной работы студента по созданию краткой информационной структуры, обобщающей и отражающей суть материала лекции, темы учебника.

Опорный конспект — это наилучшая форма подготовки к ответу на вопросы.

Основная цель опорного конспекта — облегчить запоминание. Этапы составления опорного конспекта:

1. Изучить материалы темы, выбрать главное и второстепенное;
2. Установить логическую связь между элементами темы;
3. Представить характеристику элементов в краткой форме;
4. Выбрать опорные сигналы для акцентирования главной информации и отобразить в структуре работы.

Методические рекомендации по прохождению тестирования

Тестирование — это исследовательский метод, который позволяет выявить уровень знаний, умений и навыков, способностей, а также их соответствие определенным нормам усвоения, путем выполнения испытуемым ряда специальных заданий.

Следует понимать, что тестовые задания могут быть представлены в различных формах:

– задания закрытой формы, в которых обучающийся выбирает один или несколько правильных ответов из заданного набора:

– задания на дополнение (открытые задания), требующие самостоятельного получения ответов:

– задания на установления соответствия (с множественным выбором), выполнение которых связано с выявлением соответствия между элементами нескольких множеств:

– задания на установление правильной последовательности, в которых от учащегося требует указать порядок действий или процессов и другие. Этапы подготовки к тестированию:

1. Внимательно прочитайте материал по конспекту, составленному на учебном занятии. Прочитайте тот же материал по учебнику, учебному пособию.

2. Постарайтесь разобраться с непонятным, в частности новыми терминами и конструкциями.

3. Ответьте на контрольные вопросы для самопроверки, имеющиеся в учебнике, конспекте и т. д.

4. Кратко перескажите содержание изученного материала «своими словами».

5. Выучите определения основных понятий, условные обозначения, формулы и конструкции.

Подготовка к практическим занятиям

В ходе подготовки к практическим занятиям необходимо изучить основную литературу, ознакомиться с дополнительной литературой, новыми публикациями в периодических изданиях, ознакомиться с программным обеспечением. Следует дорабатывать свой конспект лекции, делая в нем соответствующие записи из литературы, рекомендованной преподавателем и предусмотренной учебной программой.

Заканчивать подготовку следует закреплением материала с использованием соответствующих программных продуктов.

Все практические задания, предусмотренные рабочей программой, представлены в фонде оценочных средств по дисциплине.

Критерии оценивания выполненных практических работ:

- правильность выполнения работы (отсутствие фактических, логических и других ошибок);
- полнота выполнения работы;
- своевременность выполнения;
- правильность оформления отчета.

За задания, выполненные позже установленного срока или с нарушениями требований к оформлению, оценка на балл снижается.

Порядок организации самостоятельной работы студентов

Самостоятельная работа студентов способствует развитию самостоятельности, ответственности и организованности, творческого подхода к решению проблем учебного и профессионального уровня.

Целью самостоятельной работы студентов является: овладение практическими знаниями, профессиональными умениями и навыками деятельности по специальности, опытом творческой, исследовательской деятельности.

Самостоятельная работа студентов предполагает:

- самостоятельный поиск ответов и необходимой информации в рамках изучаемых тем;
- выполнение заданий для самостоятельной работы, в том числе тестов;
- изучение теоретического и лекционного материала, а также основной и дополнительной литературы при подготовке к практическим занятиям.

5 Учебная литература и ресурсы информационно-телекоммуникационной сети «Интернет»

Основная литература

2. Волк В. К. Базы данных. Проектирование, программирование, управление и администрирование учебник для СПО / В. К. Волк - 3-е изд., стер. - Санкт-Петербург: Лань, 2024 - 340 с. - ISBN 978-5-507-47482-0
3. Домбровская Г., Новиков Б., Бейликова А. Оптимизация запросов в PostgreSQL/ пер. с англ. Д. А. Бейликова. - М.: ДМК Пресс, 2022 - 278 с. - ISBN 978-5-97060-963-7
4. Илюшечкин, В. М. Основы использования и проектирования баз данных : учебник для среднего профессионального образования / В. М. Илюшечкин. — испр. и доп. — Москва : Издательство Юрайт, 2024. — 213 с. — (Профессиональное образование). — ISBN 978-5-534-01283-5.
5. Мамедли Р. Э. Большие данные и NoSQL базы данных: учебное пособие для СПО / Р. Э. Мамедли, Т. Б. Казиахмедов. - Санкт-Петербург: Лань, 2024 - 92 с. - ISBN 978-5-507-49874-1
6. Мамедли Р. Э. Системы управления базами данных: учебник для СПО / Р. Э. Мамедли - Санкт-Петербург: Лань, 2024 - 228 с. - ISBN 978-5-507-48730-1
7. Маркин, А. В. Программирование на SQL : учебное пособие для среднего профессионального образования / А. В. Маркин. — Москва : Издательство Юрайт, 2024. — 435 с. — (Профессиональное образование). — ISBN 978-5-534-11093-7.
8. Молдованова, О. В. Информационные системы и базы данных : учебное пособие для СПО / О. В. Молдованова. — 2-е изд. — Саратов : Профобразование, 2024. — 177 с. — ISBN 978-5-4488-1177-7.
9. Нестеров, С. А. Базы данных : учебник и практикум для среднего профессионального образования / С. А. Нестеров. — 2-е изд. — Москва : Издательство Юрайт, 2024. — 258 с. — (Профессиональное образование). — ISBN 978-5-534-18087-9

10. Полтавцева М. А. Безопасность баз данных : учебник для СПО / М. А. Полтавцева - Санкт-Петербург : Лань, 2024. - 356 с. - (Профессиональное образование). — ISBN 978-5-507-50000-0
11. Стружкин, Н. П. Базы данных: проектирование. Практикум : учебное пособие для среднего профессионального образования / Н. П. Стружкин, В. В. Годин. — Москва : Издательство Юрайт, 2024. — 291 с. — (Профессиональное образование). — ISBN 978-5-534-08140-4.
12. Стружкин, Н. П. Базы данных: Проектирование : учебник для среднего профессионального образования / Н. П. Стружкин, В. В. Годин. — Москва : Издательство Юрайт, 2024. — 477 с. — (Профессиональное образование). — ISBN 978-5-534-11635-9.
13. Федорова Г.Н. Основы проектирования баз данных: учебник / Г.Н. Федорова – 6-е изд., испр. – М.: Образовательно-издательский центр «Академия», 2024. – 224 с. - ISBN 978-5-0054-2120-3
14. Федорова Г.Н. Разработка, администрирование и защита баз данных: учебник / Г.Н. Федорова – 6-е изд., перераб. – М.: Образовательно-издательский центр «Академия», 2024. – 288 с. - ISBN 978-5-0054-1793-0
15. Финкова М.А. Базы данных на примерах. Практика, практика и только практика / М.А. Финкова, Макаренко Н.В. - Москва: Издательство Наука и техника, 2023 - 215с. - ISBN 978-5-907592-10-0.

Интернет-ресурсы

16. Система дистанционного обучения “SQLTest” <https://rgty.ru/sqltest/>
17. Интерактивный курс по SQL <https://sql-academy.org/ru/trainer>
18. Упражнения по SQL <https://www.sql-ex.ru/>

6 Материально-техническая база, информационные технологии, программное обеспечение и информационные справочные системы

Для реализации дисциплины необходимы:

рабочие станции (персональные компьютеры) с характеристиками не ниже: процессор — Intel Core i5 (или аналогичный AMD), ОЗУ — 16 ГБ, SSD — не менее 256 ГБ;

– проекционное оборудование (проектор/интерактивная доска) для демонстрации материалов;

– сетевое подключение со скоростью не менее 100 Мбит/с.

Программное обеспечение

Операционные системы: Windows 10/11, Linux (Ubuntu, CentOS, Astra, Alt)

Офисные пакеты: Microsoft Office 365, LibreOffice

Системы управления базами данных: PostgreSQL, MySQL/MariaDB, Microsoft SQL Server (Express-версия), MongoDB (для работы с NoSQL-данными).

Электронно-библиотечные системы (ЭБС)

1. ЭБС «BOOK.RU». — URL: <https://book.ru/>
2. ЭБС «Znaniium». — URL: <https://znaniium.ru/>
3. ЭБС «Айбукс». — URL: <https://ibooks.ru/>
4. ЭБС «Лань». — URL: <https://e.lanbook.com/>
5. ЭБС «Юрайт». — URL: <https://urait.ru/>
6. Электронные каталоги библиотеки СЗИУ РАНХиГС. — URL: <https://sziu-lib.ranepa.ru/>