

Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Ястребов Олег Александрович  
Должность: Ректор  
Дата подписания: 22.05.2026 11:38:35  
Уникальный программный ключ:  
ca953a0120d891083f939673078ef1a989dae18a

**Федеральное государственное автономное образовательное учреждение высшего образования  
«Российский университет дружбы народов имени Патриса Лумумбы»  
Факультет искусственного интеллекта**

(наименование основного учебного подразделения (ОУП) – разработчика ОП ВО)

## **РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ**

### **SQL И NOSQL БАЗЫ ДАННЫХ**

(наименование дисциплины/модуля)

**Рекомендована МССН для направления подготовки/специальности:**

### **02.04.02 ФУНДАМЕНТАЛЬНАЯ ИНФОРМАТИКА И ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ**

(код и наименование направления подготовки/специальности)

**Освоение дисциплины ведется в рамках реализации основной профессиональной образовательной программы высшего образования (ОП ВО):**

### **УПРАВЛЕНИЕ ДАННЫМИ И ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ**

(наименование (профиль/специализация) ОП ВО)

**2026 г.**

## 1. ЦЕЛЬ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Дисциплина «SQL и NoSQL базы данных» входит в программу магистратуры «Управление данными и искусственный интеллект» по направлению 02.04.02 «Фундаментальная информатика и информационные технологии» и изучается в 1 семестре 1 курса. Дисциплину реализует Кафедра прикладного искусственного интеллекта. Дисциплина состоит из 3 разделов и 26 тем и направлена на изучение формирования у студентов глубокого понимания современных принципов, архитектур и технологий управления данными в масштабируемых, высоконагруженных информационных системах. Особое внимание уделяется выбору оптимальной парадигмы хранения для решения задач искусственного интеллекта и Data Science, интеграции реляционных и нереляционных СУБД в комплексные приложения, обеспечению надежности, согласованности, безопасности и высокой производительности работы с большими и разнообразными структурами данных. Целью освоения дисциплины является освоить современные и специализированные методы проектирования, использования, масштабирования и интеграции SQL и NoSQL баз данных в прикладные и исследовательские ИИ-системы и инфраструктуру больших данных; научиться обосновывать выбор конкретной модели и архитектуры БД под задачу вне зависимости от объемов и типов данных, грамотно внедрять комбинированные решения, обеспечивать отказоустойчивость, мониторинг и безопасность хранения данных.

## 2. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Освоение дисциплины «SQL и NoSQL базы данных» направлено на формирование у обучающихся следующих компетенций (части компетенций):

Таблица 2.1. Перечень компетенций, формируемых у обучающихся при освоении дисциплины (результаты освоения дисциплины)

Шифр	Компетенция	Индикаторы достижения компетенции (в рамках данной дисциплины)
УК-7	Способен: искать нужные источники информации и данные, воспринимать, анализировать, запоминать и передавать информацию с использованием цифровых средств, а также с помощью алгоритмов при работе с полученными из различных источников данными с целью эффективного использования полученной информации для решения задач; проводить оценку информации, ее достоверность, строить логические умозаключения на основании поступающих информации и данных	УК-7.2 Умеет применять цифровые технологии для поиска, обработки, анализа, хранения и представления информации в области профессиональной деятельности; УК-7.3 Владеет навыками применения цифровых технологий и методов поиска, обработки, анализа, хранения и представления информации в области профессиональной деятельности;
ОПК-1	Способен находить, формулировать и решать актуальные проблемы прикладной математики, фундаментальной информатики и информационных технологий	ОПК-1.2 Умеет осуществлять первичный сбор и анализ материала, интерпретировать различные математические объекты;
ОПК-3	Способен проводить анализ математических моделей, создавать инновационные методы решения прикладных задач профессиональной деятельности в области информатики и математического моделирования	ОПК-3.1 Знает методы теории алгоритмов, методы системного и прикладного программирования, основные положения и концепции в области математических, информационных и имитационных моделей; ОПК-3.2 Умеет соотносить знания в области программирования, интерпретацию прочитанного, определять и создавать информационные ресурсы

Шифр	Компетенция	Индикаторы достижения компетенции (в рамках данной дисциплины)
		глобальных сетей, образовательного контента, средств тестирования систем; ОПК-3.3 Имеет практический опыт применения и разработки программного обеспечения, тестирования программных продуктов;
ОПК-4	Способен оптимальным образом комбинировать существующие информационно-коммуникационные технологии для решения задач в области профессиональной деятельности с учетом требований информационной безопасности	ОПК-4.1 Знает принципы сбора и анализа информации, создания информационных систем на стадиях жизненного цикла; ОПК-4.3 Имеет практический опыт анализа и интерпретации информационных систем;

### 3. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОП ВО

Дисциплина «SQL и NoSQL базы данных» относится к обязательной части блока 1 «Дисциплины (модули)» образовательной программы высшего образования. В рамках образовательной программы высшего образования обучающиеся также осваивают другие дисциплины и/или практики, способствующие достижению запланированных результатов освоения дисциплины «SQL и NoSQL базы данных».

*Таблица 3.1. Перечень компонентов ОП ВО, способствующих достижению запланированных результатов освоения дисциплины*

Шифр	Наименование компетенции	Предшествующие дисциплины/модули, практики*	Последующие дисциплины/модули, практики*
УК-7	Способен: искать нужные источники информации и данные, воспринимать, анализировать, запоминать и передавать информацию с использованием цифровых средств, а также с помощью алгоритмов при работе с полученными из различных источников данными с целью эффективного использования полученной информации для решения задач; проводить оценку информации, ее достоверность, строить логические умозаключения на основании поступающих информации и данных		Методы машинного обучения (продвинутый курс); Глубокое обучение для NLP; Генеративный искусственный интеллект; Преддипломная практика;
ОПК-1	Способен находить, формулировать и решать актуальные проблемы прикладной математики, фундаментальной информатики и информационных технологий		Обучение с подкреплением; Глубокое обучение для NLP; Машинное обучение на больших данных;
ОПК-3	Способен проводить анализ математических моделей, создавать инновационные		Глубокое обучение для NLP; Методы оптимизации; Глубокое обучение в

Шифр	Наименование компетенции	Предшествующие дисциплины/модули, практики*	Последующие дисциплины/модули, практики*
	методы решения прикладных задач профессиональной деятельности в области информатики и математического моделирования		компьютерном зрении;
ОПК-4	Способен оптимальным образом комбинировать существующие информационно-коммуникационные технологии для решения задач в области профессиональной деятельности с учетом требований информационной безопасности		Глубокое обучение в компьютерном зрении; Генеративный искусственный интеллект;

\* - заполняется в соответствии с матрицей компетенций и СУП ОП ВО

\*\* - элективные дисциплины /практики

#### 4. ОБЪЕМ ДИСЦИПЛИНЫ И ВИДЫ УЧЕБНОЙ РАБОТЫ

Общая трудоемкость дисциплины «SQL и NoSQL базы данных» составляет «4» зачетные единицы.

Таблица 4.1. Виды учебной работы по периодам освоения образовательной программы высшего образования для очной формы обучения.

Вид учебной работы	ВСЕГО, ак.ч.		Семестр(-ы)
			1
Контактная работа, ак.ч	51		51
Лекции (ЛК)	17		17
Лабораторные работы (ЛР)	0		0
Практические/семинарские занятия (СЗ)	34		34
Самостоятельная работа обучающихся, ак.ч.	66		66
Контроль (экзамен/зачет с оценкой), ак.ч.	27		27
Общая трудоемкость дисциплины ак.ч.	ак.ч.	144	144
	зач.ед.	4	4

## 5. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Таблица 5.1. Содержание дисциплины (модуля) по видам учебной работы\*

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы*
Раздел 1	Продвинутые методы и архитектуры реляционных БД для Data Science и AI	1.1	Новые стандарты SQL и архитектуры транзакционных СУБД в условиях Big Data и распределённых вычислений	Стандарты SQL:2016 и SQL:2023 привнесли поддержку работы с JSON-данными, временными таблицами и полиморфными табличными функциями. Гибридные архитектуры HTAP (Hybrid Transactional/Analytical Processing) позволяют одновременно обрабатывать транзакционные и аналитические нагрузки в рамках единой системы. Протоколы распределённых транзакций — двухфазный коммит (2PC), алгоритмы консенсуса Paxos и Raft — обеспечивают согласованность данных в распределённых средах. Архитектуры NewSQL-систем (CockroachDB, TiDB, YugabyteDB) объединяют горизонтальную масштабируемость NoSQL с гарантиями ACID-транзакций классических реляционных СУБД. Сценарии применения данных технологий в условиях больших данных и территориально распределённых кластеров.	ЛК
		1.2	Индексация, партиционирование и оптимизация запросов для аналитических, IoT- и ML-нагрузок	Механизмы индексации: колоночные индексы, bitmap-индексы, специализированные структуры GIN и GiST для полнотекстового поиска и работы с многомерными данными. Стратегии партиционирования таблиц — range, list, hash и composite — и их влияние на производительность запросов при работе с большими объёмами данных. Материализованные представления и подходы к их инкрементальному обновлению для ускорения сложных аналитических запросов. Принципы cost-based оптимизации запросов, использование статистик и подсказок (hints) для управления планировщиком. Специфика оптимизации для IoT-нагрузок с непрерывными потоками временных рядов и для ML-задач, требующих эффективного извлечения признаков из больших массивов данных.	ЛК
		1.3	Интеграция реляционных данных в экосистему AI/DS: federated queries, хранилища, DataLake	Механизмы федеративных запросов: PostgreSQL Foreign Data Wrappers, движки распределённых запросов Trino и Presto для выполнения SQL-запросов к разнородным источникам данных. Архитектуры корпоративных хранилищ данных (Data Warehouse) на примере Snowflake и Amazon Redshift. Концепция Data Lake с использованием открытых табличных форматов Delta Lake, Apache Iceberg и Apache Hudi, обеспечивающих транзакционную целостность и версионирование данных в озёрах данных. Архитектура Lakehouse, объединяющая преимущества хранилищ и озёр данных. Способы интеграции SQL-данных в пайплайны машинного обучения через SQL-интерфейсы и коннекторы.	ЛК
		1.4	Проектирование многотабличных, аналитических и временных структур для ИИ-проектов	Классические схемы OLAP-моделирования — звезда (Star) и снежинка (Snowflake). Методология Data Vault 2.0, обеспечивающая гибкость и историчность хранения данных. Временные таблицы (temporal tables) и концепция битемпоральных данных, позволяющая отслеживать как время действия факта, так и время его регистрации в системе. Паттерны медленно меняющихся измерений (Slowly Changing Dimensions) различных типов и их применение в аналитических хранилищах. Проектирование схем для Feature Store — централизованного хранилища признаков для подготовки данных к обучению моделей машинного обучения.	СЗ

		1.5	Практикум по сложной индексации, шардированию	Создание и управление составными, частичными и функциональными индексами с анализом их влияния на планы выполнения запросов. Горизонтальное и вертикальное	СЗ
--	--	-----	---	--	----

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы*
			и партиционированию	шардирование данных: выбор ключа шардирования и стратегии распределения нагрузки. Стратегии партиционирования для данных временных рядов, характерных для IoT-систем и мониторинга. Методы перебалансировки шардов и миграции данных между партициями без прерывания обслуживания. Работа с реальными наборами данных, анализ метрик производительности и выбор оптимальных стратегий индексации и распределения данных для различных типов нагрузок.	
		1.6	Кейсы масштабирования PostgreSQL/MySQL: от стартапа до корпоративного Data Lake	Этапы масштабирования реляционных СУБД по мере роста проекта на конкретных практических примерах. Механизмы репликации — потоковая (streaming), логическая (logical) и мультимастер (multi-master) — их ограничения и сценарии применения. Инструменты пулинга соединений (PgBouncer, ProxySQL) для эффективного управления большим количеством клиентских подключений. Решения для кластеризации: Patroni для PostgreSQL и Galera Cluster для MySQL, обеспечивающие высокую доступность и автоматический failover. Стратегии миграции с монолитных реляционных баз данных на распределённые системы и озёра данных. Подходы к мониторингу и тюнингу производительности на каждом этапе масштабирования.	СЗ
		1.7	Разбор ошибок и решений при работе с ACID-транзакциями в нестабильных сетях	Уровни изоляции транзакций (Read Uncommitted, Read Committed, Repeatable Read, Serializable) и связанные с ними аномалии — грязное чтение, неповторяющееся чтение, фантомные записи. Проблемы распределённых транзакций при сетевых разделениях (network partitions) и подходы к их решению. Паттерн Saga для управления длительными бизнес-транзакциями через последовательность локальных транзакций с компенсирующими действиями. Стратегии достижения конечной согласованности (eventual consistency) и их применимость в различных сценариях. Принципы идемпотентности операций и механизмы повторных попыток (retry) для корректной обработки сбоев связи.	СЗ
		1.8	Использование современных расширений SQL для ML (PostGIS, MADlib, встроенные ML-модули)	PostGIS как инструмент геопространственной аналитики: пространственные запросы, вычисление расстояний, работа с географическими объектами непосредственно в базе данных. Библиотека MADlib с алгоритмами машинного обучения — регрессия, классификация, кластеризация — для выполнения in-database analytics без выгрузки данных. Процедурные языки PL/Python и PL/R для написания кастомных аналитических функций и интеграции с экосистемой Python и R. Векторные расширения (pgvector) для хранения и индексации эмбедингов, используемых в семантическом поиске. Подходы к feature engineering с использованием чистого SQL.	СЗ
		1.9	Практика интеграции SQL-источников данных с BI, аналитикой и ML-пайплайнами	Настройка коннекторов для BI-инструментов — Tableau, Power BI и Apache Superset — включая оптимизацию запросов для интерактивной визуализации. Интеграция с рабочими окружениями Data Science через Jupyter Notebook, библиотеку pandas и ORM-фреймворк SQLAlchemy. Построение ETL-процессов на основе SQL с использованием Apache Airflow для оркестрации DAG-пайплайнов. Интеграция с MLflow для отслеживания экспериментов, версий данных и моделей. Версионирование данных для обеспечения воспроизводимости экспериментов машинного обучения и аудита изменений в исходных данных.	СЗ

Раздел 2	Модели, архитектуры и инструментарий NoSQL в ИИ и инфраструктуре больших	2.1	Классификация и внутреннее устройство NoSQL-систем: Key-Value,	CAP-теорема, определяющая компромиссы между согласованностью, доступностью и устойчивостью к разделению. Принципы BASE (Basically Available, Soft state, Eventually consistent) как альтернатива ACID-гарантиям. Архитектура Key-Value	ЛК
----------	--	-----	--	--	----

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы*
	данных		Document, Column Family, Graph DB	хранилищ (Redis, Memcached): структуры данных и механизмы хранения. Документные хранилища с JSON и BSON моделями данных, подходы к индексации и выполнению запросов. Модель Column Family на основе архитектуры BigTable: принципы хранения и извлечения данных. Графовые базы данных: модель свойственных графов (property graphs) и RDF-модель, области применения и преимущества для работы со связанными данными.	
		2.2	Best practices масштабируемости, отказоустойчивости, быстродействия и управления сложными разнородными данными в NoSQL	Стратегии шардирования данных и методы предотвращения горячих точек (hotspots) при неравномерном распределении нагрузки между узлами. Обеспечение отказоустойчивости: стратегии аварийного восстановления (disaster recovery) и резервного копирования. Тюнинг производительности: настройка компактификации, стратегии кэширования, оптимизация использования памяти. Паттерны проектирования схем данных для горизонтальной масштабируемости и минимизации межузловых взаимодействий. Миграция данных между версиями схем и между системами без прерывания обслуживания.	ЛК
		2.3	Практика выбора NoSQL-модели под конкретный ИИ-, IoT- или аналитический проект	Типичные сценарии использования различных моделей данных: рекомендательные системы, системы реального времени, аналитические платформы. Специализированные решения для временных рядов в IoT-проектах — InfluxDB и TimescaleDB — их архитектура и особенности запросов. Паттерны использования Key-Value хранилищ для управления сессиями и кэширования. Документные базы данных в системах управления контентом. Графовые базы данных для моделирования социальных сетей и выполнения сложных графовых запросов. Критерии выбора и обоснование архитектурных решений для каждого сценария.	СЗ
		2.4	Проектирование и оптимизация схем для MongoDB, Cassandra, Redis, Neo4j	MongoDB: стратегии встраивания (embedding) и ссылок (referencing) документов, составные индексы, aggregation pipeline. Cassandra: выбор partition key и clustering columns, моделирование по принципу query-first design. Redis: структуры данных (строки, хэши, списки, множества, sorted sets), стратегии персистентности (RDB, AOF). Neo4j: принципы моделирования графов, оптимизация запросов на языке Cypher, использование индексов. Стратегии миграции схем для каждой из систем.	СЗ
		2.5	Сценарии интеграции NoSQL с ML/Big Data платформами (Spark, Hadoop, Kafka)	Spark-коннекторы для Cassandra и MongoDB для распределённой обработки данных, хранящихся в NoSQL-системах. Kafka Connect как инструмент потоковой интеграции для передачи данных между NoSQL-хранилищами и системами обработки в реальном времени. Интеграция с экосистемой Hadoop (HDFS, MapReduce). Архитектуры сквозных пайплайнов данных: NoSQL как источник данных для Spark-приложений и моделей машинного обучения. Вычисление признаков (features) в реальном времени с использованием потоковых данных из NoSQL-хранилищ.	СЗ

		2.6	Разработка механизмов обеспечения согласованности, мониторинга и восстановления в NoSQL	Паттерны работы с конечной согласованностью (eventual consistency) и управление ожиданиями приложения относительно актуальности данных. Стратегии разрешения конфликтов при одновременной модификации: Last Write Wins (LWW), векторные часы (vector clocks), CRDT-структуры. Мониторинг NoSQL-кластеров: сбор метрик, настройка алертов, визуализация состояния системы. Резервное копирование и восстановление на определённый момент времени (point-in-time recovery). Применение принципов chaos engineering для тестирования отказоустойчивости NoSQL-инфраструктуры.	СЗ
--	--	-----	---	--	----

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы*
		2.7	Анализ security-практик, аудит и контроль доступа в крупных NoSQL-инфраструктурах	Механизмы аутентификации: интеграция с LDAP, Kerberos, OAuth-провайдерами. Модели авторизации — ролевой контроль доступа (RBAC) и атрибутный контроль доступа (ABAC) — их реализация в различных NoSQL-системах. Шифрование данных на уровне хранения (at rest) и при передаче по сети (in transit). Аудит действий пользователей и обеспечение соответствия регуляторным требованиям (compliance). Сетевая безопасность: настройка межсетевых экранов, сегментация сети, использование VPN для защиты NoSQL-кластеров.	СЗ
		2.8	Групповое моделирование архитектур multi-database и polyglot persistence для AI/Data Science	Концепция polyglot persistence — использование различных технологий хранения данных, оптимальных для каждого типа нагрузки. Архитектурный паттерн CQRS (Command Query Responsibility Segregation), разделяющий модели чтения и записи для повышения производительности и масштабируемости. Паттерн Event Sourcing для обеспечения полной истории изменений состояния системы. Проектирование микросервисной архитектуры, в которой каждый сервис использует наиболее подходящий тип базы данных. Стратегии синхронизации данных между различными системами хранения, обеспечение согласованности и управление жизненным циклом данных.	СЗ
Раздел 3	Интеграция, многобазовые решения и будущее data management для AI	3.1	Polyglot persistence, data mesh и orchestration: объединение SQL и NoSQL в корпоративных и исследовательских AI-проектах	Принципы Data Mesh — децентрализованная архитектура данных, основанная на доменном владении, данных как продукте, самообслуживаемой инфраструктуре и федеративном управлении. Архитектурные паттерны polyglot persistence для комбинирования SQL и NoSQL-решений в рамках единой экосистемы данных. Инструменты оркестрации пайплайнов данных — Apache Airflow, Prefect, Dagster — их сравнительные характеристики и сценарии применения. Концепция data contracts и schema registry для обеспечения совместимости данных между компонентами системы. Модели федеративного управления данными, обеспечивающие баланс между автономией доменных команд и соблюдением корпоративных стандартов качества.	ЛК
		3.2	Открытые стандарты и API: обмен, миграция, CDC (change data capture), потоковые среды данных	Подходы к построению API для доступа к данным — REST, GraphQL, gRPC — преимущества, ограничения и критерии выбора. Инструменты Change Data Capture (Debezium, Maxwell) для отслеживания изменений в базах данных и трансляции их в потоковые системы в реальном времени. Потоковые платформы Apache Kafka и Apache Pulsar: архитектура и применение для построения событийно-ориентированных архитектур. Эволюция схем данных и обеспечение обратной и прямой совместимости при изменении структуры сообщений. Формализация data contracts и использование спецификаций OpenAPI для документирования и стандартизации интерфейсов доступа к данным.	ЛК

		3.3	Новые горизонты: serverless-базы, облачные сервисы хранения данных, автоматизация управлением	Serverless-базы данных (Amazon Aurora Serverless, Azure Cosmos DB): автоматическое масштабирование вычислительных ресурсов в зависимости от нагрузки и тарификация по фактическому потреблению. Облачные сервисы хранения данных — Amazon S3, Azure Blob Storage, Google Cloud Storage — их роль в современных архитектурах данных. Управляемые сервисы баз данных (managed services): снижение операционной нагрузки и риски vendor lock-in. Автоматическое масштабирование и оптимизация затрат на облачную инфраструктуру. Infrastructure as Code (Terraform, CloudFormation) для автоматизации развёртывания и управления инфраструктурой данных.	ЛК
		3.4	Построение пайплайнов	Сравнение парадигм ETL и ELT, их преимущества и недостатки в контексте	СЗ

Номер раздела	Наименование раздела дисциплины	Наименование темы	Содержание темы	Вид учебной работы*
		ETL/ELT для работы с гетерогенными и мультимодальными источниками	современных облачных архитектур. Инструменты построения пайплайнов данных: dbt для трансформации данных в хранилищах, Airbyte и Fivetran для интеграции с различными источниками. Обработка мультимодальных данных (текст, изображения, аудио, видео), стратегии их хранения и индексации. Обеспечение качества данных (data quality): валидация, профилирование, мониторинг аномалий. Стратегии инкрементальной обработки данных для минимизации времени и ресурсов при обновлении пайплайнов.	
		3.5 Практика передачи, миграции и бэкапа в multi-cloud и гибридных инфраструктурах	Стратегии размещения данных в нескольких облачных провайдерах (multi-cloud) для снижения зависимости от одного поставщика и повышения отказоустойчивости. Механизмы репликации данных между облачными платформами: синхронная и асинхронная репликация. Планирование аварийного восстановления (disaster recovery): определение RPO и RTO, тестирование планов восстановления. Технологии обеспечения связности гибридных облачных инфраструктур: VPN, выделенные каналы (Direct Connect, ExpressRoute). Оптимизация затрат на передачу данных между облаками и регионами, стратегии минимизации egress charges при работе с большими объёмами данных.	СЗ
		3.6 Разработка и отладка CDC/Change Streams для stream processing и online-аналитики	Паттерны Change Data Capture (CDC) для отслеживания изменений в базах данных в реальном времени: MySQL binlog, PostgreSQL logical replication, MongoDB Change Streams. Инструменты CDC — Debezium, Maxwell, Kafka Connect — их настройка и интеграция с потоковыми платформами. Построение материализованных представлений на основе потока изменений с использованием Redis и Kafka. Архитектурные паттерны Event Sourcing и CQRS через CDC для разделения моделей чтения и записи. Механизмы обеспечения надёжности потоковой обработки: resume tokens, сохранение позиции для восстановления после сбоя, батчинг для повышения производительности. Применение CDC для синхронизации данных между гетерогенными хранилищами, аудита изменений и построения event-driven архитектур в системах онлайн-аналитики.	СЗ
		3.7 Групповое проектирование data mesh платформ для масштабируемого хранения и AI-аналитики	Принципы Data Mesh: доменное владение данными, данные как продукт с определёнными SLA (freshness, quality, availability), самообслуживаемая инфраструктурная платформа и федеративное вычислительное управление. Проектирование Data Products с публикацией схем в Data Catalog, настройкой политик доступа и мониторингом соблюдения SLA. Построение Self-Serve Data Infrastructure: провиजनинг хранилищ, регистрация в каталоге, автоматическое применение governance-политик. Интеграция Data Mesh с Feature Store для машинного обучения — регистрация признаков, вычисление features на основе Data Products, получение обучающих выборок. Стратегии масштабируемого хранения с tiered storage (hot/warm/cold) и автоматической оптимизацией размещения данных по частоте доступа. Мониторинг и observability Data Products: отслеживание качества данных, проверка completeness, validity and consistency.	СЗ

		3.8	Практика интеграции REST/GraphQL/Streaming API с базами данных в приложениях AI/IoT	REST API интеграция с NoSQL-хранилищами на примере FastAPI и MongoDB: CRUD-операции, агрегированная аналитика, кэширование результатов запросов в Redis с настройкой TTL. GraphQL как гибкий интерфейс доступа к данным: построение схем с типами, запросами, мутациями и подписками (subscriptions) для получения данных из	СЗ
--	--	-----	---	--	----

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы*
				MongoDB в реальном времени. Streaming API через WebSocket для передачи IoT-данных с сенсоров в реальном времени, управление пулом активных соединений и маршрутизация сообщений. Паттерны кэширования на уровне API с использованием Redis для снижения нагрузки на основную базу данных. Интеграция MongoDB Change Streams с GraphQL Subscriptions для построения реактивных приложений. Критерии выбора типа API (REST, GraphQL, gRPC, WebSocket) в зависимости от требований IoT- и AI-приложений к латентности, объёму данных и модели взаимодействия.	
		3.9	Аналитика трендов будущего: автоматизация, self-tuning DBs, AI-driven DataOps	Self-tuning базы данных: автоматическая настройка параметров конфигурации, адаптивная оптимизация запросов и автоматическое создание/удаление индексов на основе анализа рабочей нагрузки — OtterTune, Amazon Aurora Auto Tuning, Oracle Autonomous Database. AI-driven DataOps как концепция применения методов машинного обучения для автоматизации полного жизненного цикла данных — от ingestion и трансформации до мониторинга качества и управления инцидентами. Интеллектуальное управление ресурсами: предиктивное масштабирование кластеров баз данных на основе прогнозирования нагрузки, автоматическая балансировка шардов и оптимизация стоимости облачной инфраструктуры. Автоматическое обнаружение аномалий в данных и дрейфа схем с использованием ML-моделей для превентивного реагирования на деградацию качества данных. Автоматизация управления метаданными: ML-driven классификация данных, автоматическое тегирование, обнаружение PII и построение data lineage. Тренды развития: natural language интерфейсы к базам данных (Text-to-SQL на основе LLM), автономные базы данных с минимальным участием администратора, конвергенция операционных и аналитических хранилищ, vector databases для AI-приложений и интеграция генеративного ИИ в инструменты управления данными.	СЗ

\* - заполняется только по ОЧНОЙ форме обучения: ЛК – лекции; ЛР – лабораторные работы; СЗ – практические/семинарские занятия.

## 6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Таблица 6.1. Материально-техническое обеспечение дисциплины

Тип аудитории	Оснащение аудитории	Специализированное учебное/лабораторное оборудование, ПО и материалы для освоения дисциплины (при необходимости)
Лекционная	Аудитория для проведения занятий лекционного типа, оснащенная комплектом специализированной мебели; доской (экраном) и техническими средствами мультимедиа презентаций.	
Семинарская	Аудитория для проведения занятий семинарского типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, оснащенная комплектом специализированной мебели и техническими средствами мультимедиа презентаций.	
Для самостоятельной работы	Аудитория для самостоятельной работы обучающихся (может использоваться для проведения семинарских занятий и консультаций), оснащенная комплектом специализированной мебели и компьютерами с доступом в ЭИОС.	

\* - аудитория для самостоятельной работы обучающихся указывается **ОБЯЗАТЕЛЬНО!**

## 7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

### Основная литература:

1. Моргунов, Е. П. PostgreSQL. Профессиональный SQL: учеб. пособие / Е. П. Моргунов; под ред. Е. В. Рогова. — М.: ДМК Пресс, 2025. — 444 с.

ISBN 978-5-93700-373-7

2. Жиянов, Антон. Оконные функции SQL. Анализ данных на практике / Антон Жиянов. - Москва: Издательство АСТ, 2024. - 254 с. – (Программирование для всех). ISBN 978-5-17-158845-8

3. Мартишин, С. А. Базы данных. Практическое применение СУБД SQL и NoSQL-типа для проектирования информационных систем: учебное пособие / С.А. Мартишин, В.Л. Симонов, М.В. Храпченко. — Москва: ФОРУМ: ИНФРА-М, 2024. — 368 с. — (Высшее образование). - ISBN 978-5-8199-0946-1. - Текст: электронный. - URL: <https://znanium.ru/catalog/product/2096940>

4. Мамедли, Р. Э. Большие данные и NoSQL базы данных: учебное пособие для вузов / Р. Э. Мамедли, Т. Б. Казиахмедов. - Санкт-Петербург: Лань, 2024. - 92 с. - ISBN 978-5-507-49873-4. Книга из коллекции Лань - Информатика

### Дополнительная литература:

1. Жао, Элис. SQL: [pocket guide]: MySQL, Oracle, PostgreSQL, SQL Server, SQLite / Элис Жао; [перевел с английского В. Дмитрущенко]. - 4-е изд. - Астана: Спринт Бук, 2024. - 315, [2] с.: ил. ISBN 978-601-08-3728-7

2. Kaufmann, Michael. SQL and NoSQL Databases: Modeling, Languages, Security and Architectures for Big Data Management / Kaufmann, Michael; Meier, Andreas. - Ed. 2. - Springer Nature, 2023.; ISBN 978-3-031-27908-9; 978-3-031-27907-2

### Ресурсы информационно-телекоммуникационной сети «Интернет»:

1. ЭБС РУДН и сторонние ЭБС, к которым студенты университета имеют доступ на основании заключенных договоров

- Электронно-библиотечная система РУДН – ЭБС РУДН <https://mega.rudn.ru/MegaPro/Web>
- ЭБС «Университетская библиотека онлайн» <http://www.biblioclub.ru>
- ЭБС «Юрайт» <http://www.biblio-online.ru>
- ЭБС «Консультант студента» [www.studentlibrary.ru](http://www.studentlibrary.ru)



- ЭБС «Знаниум» <https://znanium.ru/>
- 2. Базы данных и поисковые системы
  - Sage <https://journals.sagepub.com/>
  - Springer Nature Link <https://link.springer.com/>
  - Wiley Journal Database <https://onlinelibrary.wiley.com/>
  - Наукометрическая база данных Lens.org <https://www.lens.org>

*Учебно-методические материалы для самостоятельной работы обучающихся при освоении дисциплины/модуля\*:*

1. Курс лекций по дисциплине «SQL и NoSQL базы данных».

\* - все учебно-методические материалы для самостоятельной работы обучающихся размещаются в соответствии с действующим порядком на странице дисциплины **в ТУИС!**

## РАЗРАБОТЧИКИ

---

Должность

**РУКОВОДИТЕЛЬ БУП**

Заведующий кафедрой

---

Должность

**РУКОВОДИТЕЛЬ ОП ВО**

Заведующий кафедрой прикладного  
искусственного интеллекта

---

Должность

---

Подолько П.М.

Фамилия И.О

---

Подолько П.М.

Фамилия И.О

---

Подолько П.М.

Фамилия И.О