



Программа курса «BASE: Проектирование и эксплуатация баз данных»

О курсе: научиться проектировать и использовать реляционные и NoSQL-БД, освоить приемы и инструменты оптимизации производительности СУБД на примере PostgreSQL, ClickHouse, Redis, MongoDB и Elasticsearch, Neo4j и NebulaGraph.

Аудитория: системные и бизнес-аналитики, аналитики данных, разработчики и проектировщики информационных систем, которые научатся проектировать и использовать реляционные и NoSQL-БД, а также освоить приемы и инструменты оптимизации производительности СУБД.

Уровень подготовки:

- понимание основ Computer Science
- опыт работы системным аналитиком в ИТ-проектах или разработчиком

Продолжительность курса: 22 академических часа, 4 дня по 4 ак. часа дистанционно

Содержание программы

1. Как устроены базы данных (БД)

- Место БД в архитектуре информационной системы и ее влияние на производительность
- Как связаны БД и СУБД, компоненты СУБД и их назначение
- Как разные СУБД хранят и запрашивают данные: табличные движки, персистентные и резидентные БД, блоки и страницы, колоночное и строковое хранение
- Модели организации данных в БД: особенности работы SQL и NoSQL-хранилища
- OLAP и OLTP-сценарии, нормализация и денормализация схемы данных, HTAP-хранилища
- Оптимизация хранения и доступа к данным на уровне СУБД: индексы, партиционирование таблиц
- Внутренние вычисления (вычисляемые поля, триггеры, хранимые функции и процедуры)
- Доступ к данным в БД извне: соединения, протоколы, интерфейсы
- Как выполняются запросы к БД и как их можно ускорить: планы выполнения запросов и механизмы их оптимизации (порядок выполнения запросов, индексы, сортировка, проекции, представления, хинты для оптимизатора)
- Транзакции и механизмы их выполнения, ACID-требования, блокировки и MVCC
- Распределенные БД, CAP-теорема, ACID и BASE
- Архитектурные приемы повышения производительности и доступности БД: репликация, балансировка нагрузки, шардирование, CQRS, CDC
- Критерии выбора СУБД и мер повышения ее производительности

2. Проектирование реляционной БД на примере PostgreSQL

- Моделирование данных, виды моделей и их назначение
- Алгоритм проектирования схемы для реляционной БД
- **Практика** — разработка концептуальной, логической и физической модели для реляционной БД
- **Практика** — получение DDL-скриптов и их запуск для создания связанных таблиц в PostgreSQL
- Основы SQL, виды операторов
- **Практика** — наполнение БД данными (DML-оператор INSERT), одиночные и пакетные вставки

3. SQL-запросы к БД на примере PostgreSQL

- Выборка, фильтрация, сортировка, дублирование и агрегация данных (SELECT, WHERE, IS NULL, BETWEEN, IN, LIKE и REGEXP, DISTINCT, ORDER BY, GROUP BY, HAVING)
- **Практика** — выполнение простых SQL-запросов к наполненным таблицам своей БД PostgreSQL
- Многотабличные запросы и подзапросы (JOIN, WITH, UNION)

	ООО «Учебный центр «Коммерсант» © «Школа прикладного бизнес-анализа» https://babok-school.ru 2025	
---	---	--

- **Практика** — выполнение многотабличных SQL-запросов к наполненным таблицам своей БД
- Условная логика (CASE, IF), преобразование типов (CAST) и оконные функции (OVER)
- **Практика** — выполнение сложных SQL-запросов к своей БД с использованием операторов условной логики, окон и преобразования типов
- Триггеры, триггерные функции и хранимые процедуры, основы PL/pgSQL
- **Практика** — определение триггерных событий и разработка триггерных функций для своей БД

4. NoSQL

- Колоночные СУБД. Архитектура и принципы работы ClickHouse
- **Практика** — выбор табличного движка, выполнение простых запросов к ClickHouse
- Key-value хранилища. Архитектура и принципы работы Redis
- **Практика** — выбор типа данных для значений, выполнение запросов к Redis
- Документо-ориентированные БД. Архитектура и принципы работы MongoDB и Elasticsearch
- **Практика** — формирование JSON-документов с данными, выполнение запросов к MongoDB/ Elasticsearch
- Графовые БД. Архитектура и принципы работы Neo4j и NebulaGraph
- **Практика** — формирование JSON-документов с данными, выполнение запросов к Neo4j/ NebulaGraph