

Schema Design for Distributed SQL with CockroachDB

Duration: 1 Day

Intro:

Gain a comprehensive understanding of schema design in CockroachDB by learning how architecture, indexing, and key selection impact performance and consistency in a distributed SQL environment.

Overview:

This course provides an in-depth exploration of CockroachDB's architectural components and advanced SQL features while emphasizing their critical roles in a distributed and replicated environment. You will gain hands-on insights into how distributed nodes and clusters manage ranges, replicas, and leaseholders; support distributed writes and reads; and operate a robust key-value store with multi-version concurrency control. The course also covers essential SQL aspects such as CRUD operations, primary key selection, and advanced indexing strategies. It explains why optimal primary key design, effective indexing, and efficient JOINS are crucial when data is distributed and replicated across multiple nodes.

Objectives:

By the end of this course, you will be able to:

- Describe CockroachDB's architectural components, including nodes, clusters, ranges, replicas, and leaseholders, and explain their significance in a distributed environment.
- Explain the principles behind distributed writes and reads and how CockroachDB's distributed key-value store and MVCC ensure data integrity and consistency across multiple nodes.

- Evaluate the critical importance of choosing the right primary key for data partitioning and distribution, and compare natural versus artificial keys in a replicated database setting.
- Demonstrate the application of SQL CRUD operations within CockroachDB, highlighting necessary adjustments for managing distributed data.
- Analyze various indexing strategies, including secondary, composite, covering, and partial indexes, and evaluate how they affect performance in a system where data is both distributed and replicated.
- Apply advanced SQL JOIN techniques to efficiently combine data across distributed and replicated tables, and assess the impact of data distribution on query performance.

Prerequisites:

Experience with DDL and SQL

Agenda:

- [CockroachDB Components](#)
 - [Nodes & Clusters](#)
 - [Ranges, Replicas, & Leaseholders](#)
 - [Distributed Writes & Reads](#)
- [Distributed KV and MVCC](#)
 - [Distributed Key-value Store](#)
 - [Multi-version Concurrency Control](#)
- [Primary Keys](#)
 - [Natural Keys](#)
 - [Artificial Keys](#)
- [SQL Features & CRUD](#)
- [Secondary Indexes](#)
 - [Index Considerations](#)
 - [Composite Indexes](#)
 - [Covering Indexes](#)
 - [Partial Indexes](#)
- [SQL JOINS](#)