WORKING WITH CASSANDRA (TTDS6776)

Course Code: 100685

Learn how to use Cassandra as your NoSQL database management system to efficiently process large amounts of data.

The Cassandra (C*) database is a massively scalable NoSQL database that provides high availability and fault tolerance, as well as linear scalability when adding new nodes to a cluster. It has many powerful capabilities, such as tunable and eventual consistency that allow it to meet the needs of modern applications. It also introduces a new concept for data modeling and many organizations do not have the expertise to use it in the most efficient manner.

This course provides an in-depth introduction to using Cassandra and creating quality data models. This course is technical and comprehensive, with a focus on the practical aspects of working with C*. It introduces all the important concepts needed to understand Cassandra, including enough coverage of internal architecture so you can make optimal decisions. It also covers CQL (Cassandra Query Language) in depth, as well as covering the Java API for writing Cassandra clients.

After taking this course, you'll have learned what you need to productively work with Cassandra as well as guidelines for using it in an optimal manner. You'll also understand some of the "anti-patterns" that lead to non-optimal C* data models. In the end, you'll be familiar with CQL and with the Java client library and be ready to work on production systems involving Cassandra.

What You'll Learn

Join an engaging hands-on learning environment, where you'll learn to:

- Understand the needs that C* addresses
- Be familiar with the operation and structure of C*
- Be able to install and set up a C* database
- Use the C* tools, including cqlsh, nodetool, and ccm (Cassandra Cluster Manager)
- Be familiar with the C* architecture, and how a C* cluster is structured
- Understand how data is distributed and replicated in a C* cluster
- Understand core C* data modeling concepts, and use them to create

well-structured data models
- Use data replication and eventual consistency intelligently
- Understand and use CQL to create tables and query for data
- Know and use the CQL data types (numerical, textual, uuid, etc.)
- Understand the various kinds of primary keys available (simple, compound, and composite primary keys)
- Use more advanced capabilities like collections, counters, secondary indexes, CAS (Compare and Set), static columns, and batches
- Be familiar with the Java client API
- Use the Java client API to write client programs that work with C*
- Build and use dynamic queries with QueryBuilder
- Understand and use asynchronous queries with the Java API

This course has a 50% hands-on labs to 50% lecture ratio with engaging instruction, demos, group discussions, labs, and project work.

Who Needs to Attend

Experienced Java Developers with database experience.

Prerequisites

Before attending this course, you should:
- Be comfortable with Java
- Have experience working with databases
- Able to navigate the Linux command line
- Have basic knowledge of Linux editors (such as VI/nano) for editing code

WORKING WITH CASSANDRA (TTDS6776)

Course Code: 100685

VIRTUAL CLASSROOM LIVE	$2,984 CAD	3 Day

Virtual Classroom Live Outline

Cassandra Overview

- Why We Need Cassandra
- High level Cassandra Overview
- Cassandra Features
- Basic Cassandra Installation and Configuration

Cassandra Architecture and CQL Overview

- Cassandra Architecture Overview
- Cassandra Clusters and Rings
- Data Replication in Cassandra
- Cassandra Consistency/Eventual Consistency
- Introduction to CQL
- Defining Tables with a Single Primary Key
- Using cqlsh for Interactive Querying
- Selecting and Inserting/Upserting Data with CQL
- Data Replication and Distribution
- Basic Data Types (including uuid, and timeuuid)

Data Modeling and CQL Core Concepts

- Defining a Compound Primary Key
- CQL for Compound Primary Keys
- Partition Keys and Data Distribution
- Clustering Columns
- Overview of Internal Data Organization
- Additional Querying Capabilities
- Result Ordering - ORDER BY and CLUSTERING ORDER BY
- UPDATE and DELETE Queries

- Result Filtering, ALLOW FILTERING
- Batch Queries
- Data Modeling Guidelines
- Denormalization
- Data Modeling Workflow
- Data Modeling Principles
- Primary Key Considerations
- Composite Partition Keys
- Defining with CQL
- Data Distribution with Composite Partition Key
- Overview of Internal Data Organization

Additional CQL Capabilities

- Indexing
- Primary/Partition Keys and Pagination with token()
- Secondary Indexes and Usage Guidelines
- Cassandra Counters
- Counter Structure and Definition
- Using Counters
- Counter Limitations
- Cassandra collections
- Collection Structure and Uses
- Defining Collections (set, list, and map)
- Querying Collections (Including Insert, Update, Delete)
- Limitations
- Overview of Internal Storage Organization
- Static Column: Overview and Usage
- Static Column Guidelines
- Materialized View: Overview and Usage
- Materialized View Guidelines

Data Consistency In Cassandra

- Overview of Consistency in Cassandra
- CAP Theorem
- Eventual (Tunable) Consistency in C* - ONE, QUORUM, ALL
- Choosing CL ONE
- Choosing CL QUORUM
- Achieving Immediate Consistency
- Using other Consistency Levels
- Internal Repair Mechanisms (Read Repair and Hinted Handoff)

Lightweight Transactions (LWT)/Compare and Set (CAS)

- Overview of Lightweight Transactions
- Using LWT, the [applied] Column
- IF EXISTS, IF NOT EXISTS, Other IF conditions
- Basic CAS Internals
- Overhead and Guidelines

Practical Considerations

- Dealing with Write Failure
- Unavailable Nodes and Node Failure
- Requirements for Write Operations
- Key and Row Caches
- Cache Overview
- Usage Guidelines
- Multi-Data Center Support
- Overview
- Replication Factor Configuration
- Additional Consistency Levels - LOCAL/EACH QUORUM
- Deletes
- CQL for Deletion
- Tombstones
- Usage Guidelines

The Java Client API

- API Overview
- Introduction
- Architecture and Features
- Connecting to a Cluster
- Cluster and Cluster.Builder
- Contact Points, Connecting to a Cluster
- Session Overview and API
- Working with Sessions
- The Query API
- Overview
- Dynamic Queries, Statement, SimpleStatement
- Processing Query Results, ResultSet, Row
- PreparedStatement, BoundStatement
- Binding Values and Querying with PreparedStatements
- CQL to Java Type Mapping
- Working with UUIDs
- Working with Time/Date Values
- Working with Batches of SimpleStatement and PreparedStatement
- Dynamic Queries and QueryBuilder
- QueryBuilder Overview and API
- Building SELECT, DELETE, INSERT, and UPDATE Queries
- Creating WHERE Clauses
- Other Query Examples
- Configuring Query Behavior
- Setting LIMIT and TTL
- Working with Consistency
- Using LWT
- Working with Driver Policies

- Load Balancing Policies - RoundRobinPolicy, DCAwareRoundRobinPolicy
- Retry Policies - DefaultRetryPolicy, DowngradingConsistencyRetryPolicy, Other Policies
- Reconnection Policies
- Asynchronous Querying Overview
- Synchronous vs. Asynchronous Querying
- Executing Asynchronous Queries
- java.util.concurrent.Future
- Cassandra ResultSetFuture

WORKING WITH CASSANDRA (TTDS6776)

Course Code: 100685

PRIVATE GROUP TRAINING	3 Day

Visit us at www.globalknowledge.com or call us at 1-866-716-6688.