

SAN DIEGO COMMUNITY COLLEGE DISTRICT
CONTINUING EDUCATION
COURSE OUTLINE

SECTION I

SUBJECT AREA AND COURSE NUMBER

COMP 691

COURSE TITLE

NOSQL DOCUMENT DATABASES

TYPE COURSE

NON-FEE

VOCATIONAL

CATALOG COURSE DESCRIPTION

MongoDB is a free and open source document database. It is a NoSQL database that uses a JavaScript Object Notation (JSON)-like schema. It is cross platform and easy to deploy on the cloud or on a server, and it's because of this flexibility that companies like Uber, Lyft, Coinbase, Accenture, Adobe, Cisco, and more choose it over traditional relational database solutions. This course is designed to get students up to speed on MongoDB. Students will learn the basics of the technology, how to work with MongoDB in the cloud, how to create databases and collections, and how to perform basic and advanced create, retrieve, update, and delete (CRUD) operations, and more. (FT)

LECTURE/LABORATORY HOURS

60

ADVISORIES

NONE

RECOMMENDED SKILL LEVEL

Possess a 12th grade reading level; ability to communicate effectively in the English language.

INSTITUTIONAL STUDENT LEARNING OUTCOMES

1. Social Responsibility
SDCE students demonstrate interpersonal skills by learning and working cooperatively in a diverse environment.
2. Effective Communication
SDCE students demonstrate effective communication skills.

NOSQL DOCUMENT DATABASES

PAGE 2

3. Critical Thinking
SDCE students critically process information, make decisions, and solve problems independently or cooperatively.
4. Personal and Professional Development
SDCE students pursue short term and life-long learning goals, mastering necessary skills and using resource management and self -advocacy skills to cope with changing situations in their lives.

COURSE GOALS

1. Introduce MongoDB and understand the role of NoSQL document databases
2. Learn how to set up a MongoDB cluster on the cloud
3. Understand the difference between a cluster, a database, collections, and documents
4. Understand how to load sample data into a MongoDB database as well as manually add data into a database
5. Introduce the MongoDB shell and use the shell to perform simple create, retrieve, update, and delete (CRUD) operations
6. Learn to perform advanced CRUD operations with comparison, logic, expression, and array operators
7. Understand how to query arrays and sub-documents
8. Learn about the MongoDB Aggregation Framework and how to use pipelines to streamline complex CRUD operations
9. Introduce Indexes and Data Modeling in MongoDB

COURSE OBJECTIVES

Upon successful completion, the student will be able to:

1. Describe the role of a NoSQL document database and explain how it differs from a traditional relational database
2. Set up a free MongoDB cluster on the cloud using MongoDB Atlas
3. Describe the difference between a cluster, a database, collections, and documents
4. Use the MongoDB Explorer to load data into a database
5. Work with the MongoDB shell to perform insert, find, update, and delete commands within a MongoDB database
6. Use advanced comparison, logic, expression, and array operators to precisely find and narrow down queries issued to a MongoDB database
7. Query arrays and sub-documents within a database document
8. Create a pipeline using the MongoDB Aggregation Framework to streamline more complex CRUD operations
9. Use data modeling techniques to organize fields in a document to support an application's performance and querying capabilities

SECTION II

COURSE CONTENT AND SCOPE

1. Introduction to MongoDB
 - 1.1. Introduction to MongoDB
 - 1.2. The MongoDB URI
 - 1.3. Setting Up MongoDB on the Cloud with Atlas
 - 1.3.1. Creating an Atlas organization
 - 1.3.2. Creating an Atlas project
 - 1.3.3. Creating a free Cluster
 - 1.3.4. Loading sample data set
 - 1.3.5. Setting up connection security
 - 1.4. Using Compass
 - 1.4.1. Connecting to a database
 - 1.4.2. Reading sample data
2. Creating, Updating, and Deleting Documents
 - 2.1. Inserting Documents
 - 2.1.1. insertMany
 - 2.1.2. InsertValidation
 - 2.1.3. Insert
 - 2.2. Removing documents with drop
 - 2.3. Updating documents
 - 2.3.1. Document replacement
 - 2.3.2. Using update operators
 - 2.3.3. Upserts
 - 2.3.4. Updating multiple documents
 - 2.3.5. Returning updated documents
3. Querying
 - 3.1. Introduction to find
 - 3.1.1. Specifying which keys to return
 - 3.1.2. Limitations
 - 3.2. Query Criteria
 - 3.2.1. Query conditionals
 - 3.2.2. OR Queries
 - 3.2.3. \$not
 - 3.3. Type-Specific queries
 - 3.3.1. Null
 - 3.3.2. Regular expressions
 - 3.3.3. Querying Arrays
 - 3.3.4. Querying on embedded documents
 - 3.4. \$where Queries
 - 3.5. Cursors
 - 3.6. Limits, Skips, and Sorts
 - 3.7. Avoiding large Skips
 - 3.8. Immortal cursors
4. Indexes
 - 4.1. Introduction to indexes
 - 4.1.1. Creating an index

- 4.1.2. Compound indexes
- 4.1.3. How MongoDB selects an index
- 4.1.4. Using compound indexes
- 4.1.5. How \$ operators use indexes
- 4.1.6. Indexing Objects and Arrays
- 4.1.7. Index cardinality
- 4.2. The explain function's output
- 4.3. When not to index
- 4.4. Types of Indexes
 - 4.4.1. Unique indexes
 - 4.4.2. Partial indexes
- 4.5. Index administration
 - 4.5.1. Identifying indexes
 - 4.5.2. Changing indexes
- 5. Special Index and Collection Types
 - 5.1. Geospatial indexes
 - 5.1.1. Types of geospatial queries
 - 5.1.2. Using geospatial queries
 - 5.1.3. Compound geospatial queries
 - 5.1.4. 2d indexes
 - 5.2. Indexes for full text search
 - 5.2.1. Creating a text index
 - 5.2.2. Text search
 - 5.2.3. Optimizing full text search
 - 5.2.4. Searching in other languages
 - 5.3. Capped Collections
 - 5.3.1. Creating capped collections
 - 5.3.2. Tailable cursors
 - 5.4. Time-to-live indexes
 - 5.5. Storing files with GridFS
 - 5.5.1. Getting started with GridFS: mongofiles
 - 5.5.2. Working with GridFS from the MongoDB drivers
 - 5.5.3. Under the hood
- 6. The Aggregation Framework
 - 6.1. Pipelines, stages, and turnables
 - 6.2. Getting started with stages: familiar operations
 - 6.3. Expressions
 - 6.4. \$project
 - 6.5. \$unwind
 - 6.6. Array expressions
 - 6.7. Accumulators
 - 6.8. Grouping
 - 6.8.1. The _id field in grouping stages
 - 6.8.2. Group versus project
 - 6.9. Writing aggregation pipeline results to a collection

APPROPRIATE READINGS

Reading assignments may include, but are not limited to: assignments from the textbook, supplemental reading assignments, industry-related periodicals or magazines, manuals, online help pages, articles posted on the Internet, and information from Web sites, online libraries and databases. Topics should be related to NoSQL document database concepts and may include information related to the differences between NoSQL document databases versus traditional relational databases.

WRITING ASSIGNMENTS

Writing assignments may include, but are not limited to: completing assigned reports, providing written answers to assigned questions, performing internet research and reporting on that research. An example would include a case study of how the MongoDB Aggregation Pipeline can be used to streamline how data is matched, sorted, and grouped for quick retrieval.

OUTSIDE ASSIGNMENTS

Assignments may include, but are not limited to: appropriate internet research, reading, preparing reports and studying as needed to perform successfully in class. An appropriate assignment for instance, would include demonstrating how to use the updateMany() command in MongoDB to update many documents in a collection at one time.

APPROPRIATE ASSIGNMENTS THAT DEMONSTRATE CRITICAL THINKING

Assignments which demonstrate critical thinking may include, but are not limited to building a MongoDB cluster within MongoDB's Atlas, loading sample databases into that cluster, creating custom users that have access to the databases, and then using the MongoDB shell to connect to and perform simple CRUD operations within the MongoDB server.

EVALUATION

Evaluation that a student has met the course competencies will include multiple measures of performance related to the course objectives. Evaluation methods may include, but are not limited to performance in a variety of activities and assignments, such as completing a research project individually or in a group, hands-on projects, demonstration of use of the internet, quizzes, class participation, written and practical tests, attendance and punctuality.

Upon successful completion of all courses in the program a Certificate of Program Completion will be issued.

NOSQL DOCUMENT DATABASES

PAGE 6

METHOD OF INSTRUCTION

Methods of instruction, may include, but are not limited to, lecture, in-class and online discussions, hands-on demonstrations, computer-assisted instruction, field trips, and laboratory assignments.

This course, or sections of this course, may be offered through distance education.

TEXTS AND SUPPLIES

Textbooks

MongoDB: The Definitive Guide: Powerful and Scalable Data Storage,
Shannon Bradshaw, Eoin Brazil, and Kristina Chodorow, O'Reilly, current edition

Supplies:

Pen, journal (composition book), notebook paper and a soft 3-ring binder, or a one-subject 110 sheet college ruled notebook, and appropriate storage media such as a USB Drive, external hard drive, or cloud-based storage.

PREPARED BY Zak Ruvalcaba DATE April 7, 2021

REVISED BY _____ DATE _____

Instructors must meet all requirements stated in Policy 3100 (Student Rights, Responsibilities and Administrative Due Process), and the Attendance Policy set forth in the Continuing Education Catalog.

REFERENCES:

San Diego Community College District Policy 3100
California Community Colleges, Title 5, Section 55002
Continuing Education Catalog