Unit 4 Databases

DB / SQL / Maven / JDBC / MongoDB
Full Stack Java

Tuesday July 14th, 2020

11 of 28

Unit 4

RDBMS

PostgreSQL

MySQL

SQL

Maven

JDBC

MongoDB

Agenda

- Unit 4 Review and Wrap up
    - Prepared Statement
    - DAO Pattern
    - Review Assignment
    - Unit 4 Review Quiz
- MongoDB
- Unit 5 Introduction - Spring Data
- Capstone Project Discussion

Maven Review

Dependencies

Maven Commands (phases - life cycle of the build)

- E.g. compile - Compile source code, classes stored in target / classes

Based on Project Object Model (POM)

Standard way to build projects

Share Java Archives (JARs) across several projects

Building and managing any Java-based project

JDBC Review - Development Process

```
// Get a connection to a database
Connection connection = DriverManager.getConnection(URL, USER, PASS);

// Create a Statement object
Statement statement = connection.createStatement();

// Execute SQL Query
ResultSet resultSet = statement.executeQuery("SELECT * FROM movies;");

// Process the Result Set
while (resultSet.next()) {
   // read data from each row
   String movieName = resultSet.getString("movie_name");
}
```

DAO Pattern

Separate low level data accessing API or operations from high level business services

Data Access Object Interface

    MovieDao.java

Data Access Object Concrete Class

    MovieDaoImpl.java

Model Object or Value Object (POJO)

    Movie.java

Review Assignment

Unit 1-4 Quiz

MongoDB

MongoDB

NoSQL

MongoDB

Open-source document based database

Leading NoSQL database

Written in C++

Database

Physical container for collections

Similar to a Database in RDBMS

Collection

Group of MongoDB documents

Similar to Table in a RDBMS

Related but can have different fields

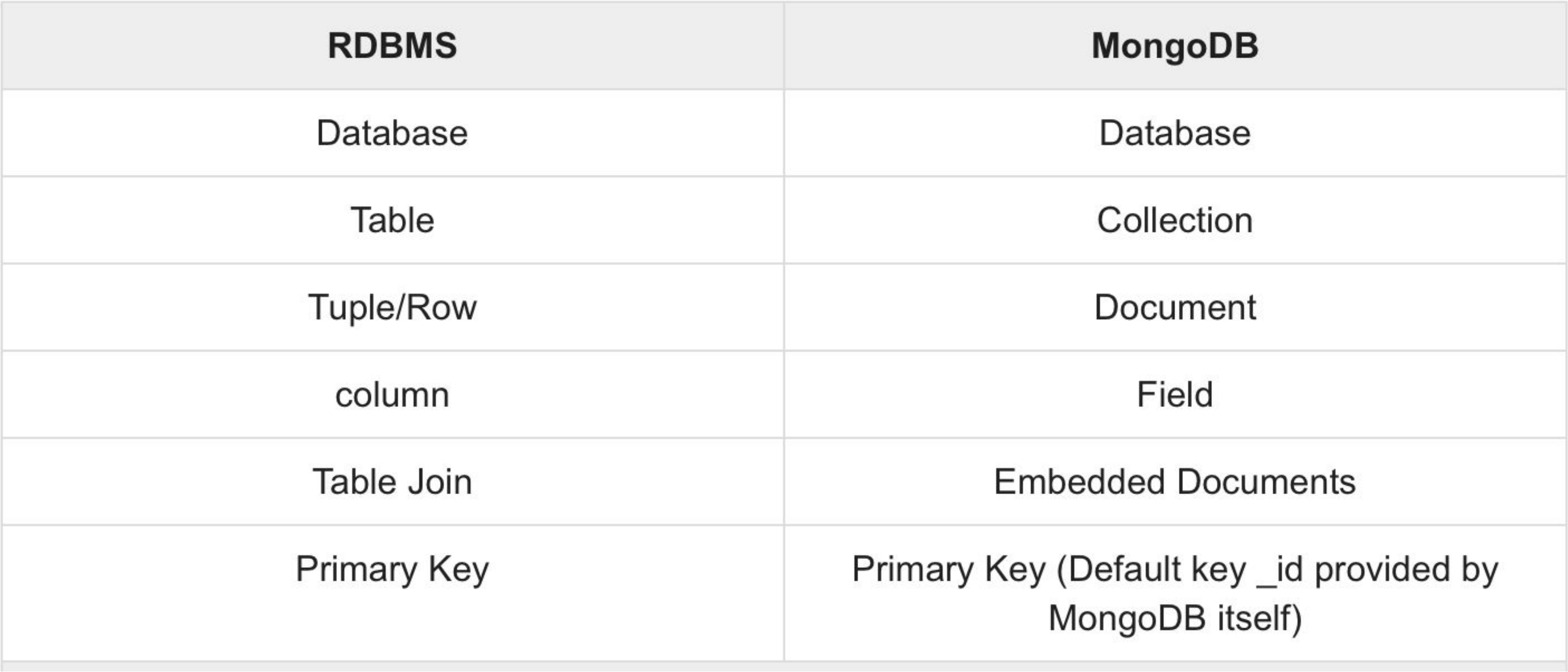
Document

A set of key-value pairs

Dynamic schema - different fields / data

Compared to a Row / Record in RDBMS

RDBMS Comparison



RDBMS	MongoDB
Database	Database
Table	Collection
Tuple/Row	Document
column	Field
Table Join	Embedded Documents
Primary Key	Primary Key (Default key _id provided by MongoDB itself)

Sample Document

```
{
   _id: ObjectId(7df78ad8902c)
   title: 'MongoDB Overview',
   description: 'MongoDB is no sql database',
   by: 'tutorials point',
   url: 'http://www.tutorialspoint.com',
   tags: ['mongodb', 'database', 'NoSQL'],
   likes: 100,
   comments: [
      {
         user:'user1',
         message: 'My first comment',
         dateCreated: new Date(2011,1,20,2,15),
         like: 0
      },
      {
         user:'user2',
         message: 'My second comments',
         dateCreated: new Date(2011,1,25,7,45),
         like: 5
      }
   ]
}
```

Advantages over RDBMS

- Schema less - MongoDB is a document database in which one collection holds different documents. Number of fields, content and size of the document can differ from one document to another.
- Structure of a single object is clear.
- No complex joins.
- Deep query-ability. MongoDB supports dynamic queries on documents using a document-based query language that's nearly as powerful as SQL.
- Tuning.
- Ease of scale-out - MongoDB is easy to scale.
- Conversion/mapping of application objects to database objects not needed.
- Uses internal memory for storing the (windowed) working set, enabling faster access of data.

Why Use MongoDB

Document Oriented Storage - Data is stored in the form of JSON style documents.

Index on any attribute

Replication and high availability

Auto-Sharding

Rich queries

Fast in-place updates

Where to Use MongoDB

Big Data

Content Management and Delivery

Mobile and Social Infrastructure

User Data Management

Data Hub

Install MongoDB

https://docs.mongodb.com/manual/administration/install-community/

Lets Play A Little Bit With MongoDB

MongoDB Exercises

Create a Database

Create a Collection

Insert Document

Query Document

Update Document

Delete Document

MongoDB and Java

Exercise to connect to MongoDB

Spring Boot

Capstone Project

Capstone

Ideas

Teams

Projects

- Java
- Database
- HTML / CSS