

# Fundamentos de MongoDB

## Practica 7 Programación con Java y MongoDB

### Objetivo

- Aprender a utilizar Java para interactuar con MongoDB

### Requisitos

- La estación de trabajo o PC asignada debe tener instalado JDK 1.8, Maven 3.5 o superior y la última versión del IDE Red Hat CodeReady Studio

### Procedimiento

1. Inicia el servidor mongoDB mediante el demonio mongod.exe y conéctate al servidor usando mongo CLI mongo.exe
2. Crea la base de datos `devmongodb` si es que no existe.
3. Crea la colección people si esta no existe en `devmongodb` para este propósito ejecuta la siguiente sentencia.

```
db.people.insertMany([
  {name:"Mary",gender:"female",size:1.72,weight:54,phone:"+51 2345679", age:25
  ,email:"mary.smith@gmail.com",company:"AWS",isActive:true,address:[{primary:"100
  Boulevard Miami",secondary:"303 St. Geneva Rome"}]},
  {name:"Charles",gender:"male",size:1.86,weight:90,phone:"+86 7345674", age:35
  ,email:"charles.slate@yahoo.com",company:"Redhat",isActive:true},
  {name:"Danny",gender:"male",size:1.91,weight:102,phone:"+1 8445663", age:25
  ,email:"danny.lasalle@growing.com",company:"AWS",isActive:false,address:[{primary:"10
  2 Bronco Texas",secondary:"404 Borbon Street Lousiana"}]},
  {name:"Richard",gender:"male",size:1.82,weight:83,phone:"+86 2545671", age:35
  ,email:"richard.jhonson@gmail.com",company:"Open cloud",isActive:true},
  {name:"Yenny",gender:"female",size:1.75,weight:56,phone:"+51 2345459", age:29
  ,email:"yenny.sullivan@gmail.com",company:"AWS",isActive:false,address:[{primary:"505
  Renfer Madrid",secondary:"345 Republica Barcelona"}]},
  {name:"Rob",gender:"male",size:1.79,weight:85,phone:"+51 7145679", age:35
  ,email:"rob.sax@mshaw.com",company:"Microsoft Inc",isActive:false},
  {name:"Brain",gender:"male",size:1.90,weight:92,phone:"+1 8947679", age:45
  ,email:"brain.dawner@yahoo.com",company:"AWS",isActive:true},
  {name:"Jane",gender:"male",size:1.56,weight:55,phone:"+1 8345663", age:25
  ,email:"jane.gross@growing.com",company:"MongoDB Inc",isActive:true}
```

```
]);
```

4. Crea el proyecto maven MongoDBApp, utiliza el arquetipo `maven-archetype-quickstart`, como groupId usa `org.mongodb.app`
5. Agrega al proyecto MongoDBApp la dependencia maven siguiente:

```
<dependency>
  <groupId>org.mongodb</groupId>
  <artifactId>mongo-java-driver</artifactId>
  <version>2.10.1</version>
</dependency>
```

6. Crea el paquete `org.mongodb.app.controller` y agreg la clase `MongoHelper` con el siguiente contenido

```
package org.mongodb.app.controller;

import java.net.UnknownHostException;

import com.mongodb.DB;
import com.mongodb.DBCollection;
import com.mongodb.MongoClient;

/**
 *
 * @author Daddy
 * Conectarse a una replica
 * MongoClient mongoClient = new MongoClient(Arrays.asList(new
 * ServerAddress("localhost", 27017),
 * new ServerAddress("localhost", 27018),
 * new ServerAddress("localhost", 27019)));
 *
 * Conectarse con autenticacion
 * MongoCredential credential =
 * MongoCredential.createMongoCRCredential(userName, database, password);
 * MongoClient mongoClient = new MongoClient(new ServerAddress(),
 * Arrays.asList(credential));
 */

public class MongoHelper {
    private MongoClient client;
    private String HOST_NAME = "localhost";
    private Integer PORT_NUMBER = new Integer(27017);

    private DB database = null;
    private DBCollection collection = null ;

    public MongoHelper() {
        super();
        try {
            client = new MongoClient( HOST_NAME , PORT_NUMBER );
```

```

        } catch (UnknownHostException e) {
            e.printStackTrace();
        }
    }
    public DB getDatabase() {
        return database;
    }
    public void setDatabase(DB database) {
        this.database = database;
    }

    public void Initialize(String dbName) {
        this.setDatabase(client.getDB( dbName ));
    }
    public DBCollection getCollection(String collectionName) {
        return this.getDatabase().getCollection(collectionName);
    }
    public void setCollection(DBCollection collection) {
        this.collection = collection;
    }

    public void dropDatabase(String dbName) {
        client.dropDatabase(dbName);
    }

    public void dropCollection(String collectionName) {
        this.getDatabase().getCollection(collectionName).drop();
    }
}

```

7. En el paquete org.mongodb.app crea la clase MainApplication asegúrate que se cree con el método public void main(String[] args)
8. En el método main de la clase MainApplication escribe el código para realizar lo siguiente:
  - a. Insertar un documento nuevo en la colección people
  - b. Consultar un documento en la colección people por un criterio definido
  - c. Mostrar todos los documentos de la colección people
  - d. Actualizar un documento de la colección people para agregar nuevos campos o actualizar los existentes
  - e. Eliminar un documento de la colección people dado uno o mas criterios
  - f. Eliminar una colección

En cada uno de los casos prueba el resultado mediante una consulta desde la consola CLI de MongoDB