

# SpringBoot MongoDB

## 1.Pom.Xml

```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
    <modelVersion>4.0.0</modelVersion>
    <groupId>com.example</groupId>
    <artifactId>spring-boot-mongodb</artifactId>
    <version>1.0.0</version>
    <packaging>jar</packaging>
    <name>spring-boot-mongodb</name>
    <description>A Spring Boot application using MongoDB</description>
    <properties>
        <java.version>21</java.version>
    </properties>
    <dependencies>
        <dependency>
            <groupId>org.springframework.boot</groupId>
            <artifactId>spring-boot-starter-data-mongodb</artifactId>
        </dependency>
        <dependency>
            <groupId>org.springframework.boot</groupId>
            <artifactId>spring-boot-starter-thymeleaf</artifactId>
        </dependency>
        <dependency>
            <groupId>org.springframework.boot</groupId>
            <artifactId>spring-boot-starter-web</artifactId>
        </dependency>
        <dependency>
            <groupId>org.projectlombok</groupId>
            <artifactId>lombok</artifactId>
            <optional>true</optional>
        </dependency>
        <dependency>
            <groupId>org.springframework.boot</groupId>
            <artifactId>spring-boot-starter-test</artifactId>
            <scope>test</scope>
        </dependency>
    </dependencies>
</project>
```

## 2.Properties

```
spring.application.name=demo-employee-mongo-db

server.port=8080

# If MongoDB runs locally
spring.data.mongodb.host=localhost
spring.data.mongodb.port=27017
spring.data.mongodb.database=employee_db

# If you want to use MongoDB URI style (instead of host/port):
spring.data.mongodb.uri=mongodb://localhost:27017/employee_db

spring.thymeleaf.cache=false
spring.thymeleaf.prefix=classpath:/templates/
spring.thymeleaf.suffix=.html
```

## 3.Entity-(EmployeeDb)

```
package com.example.demo_employee_mongo_db.entity;

import lombok.AllArgsConstructor;
import lombok.Data;
import lombok.NoArgsConstructor;
import org.springframework.data.annotation.Id;
import org.springframework.data.mongodb.core.mapping.Document;

@Document(collection = "employees") 13 usages
@Data
@AllArgsConstructor
@NoArgsConstructor
public class EmployeeDb {
    @Id
    private Long employeeId;
    private String name;
    private String email;
    private String location;
}
```

## 4.Repository

```
package com.example.demo_employee_mongo_db.repository;

import com.example.demo_employee_mongo_db.entity.EmployeeDb;
import org.springframework.data.mongodb.repository.MongoRepository;

import java.util.Optional;

public interface EmployeeDbRepository extends MongoRepository<EmployeeDb, Long> {
    Optional<EmployeeDb> findById(Long employeeId); no usages
}
```

## 5.Service

```
package com.example.demo_employee_mongo_db.service;

import com.example.demo_employee_mongo_db.entity.EmployeeDb;
import com.example.demo_employee_mongo_db.repository.EmployeeDbRepository;
import lombok.AllArgsConstructor;
import lombok.NoArgsConstructor;
import org.springframework.stereotype.Service;

import java.util.List;
import java.util.Optional;

@Service 2 usages
@AllArgsConstructor
public class EmployeeDbService {
    private EmployeeDbRepository employeeDbRepository;

    public EmployeeDb saveEmployee(EmployeeDb employeeDb){ 1 usage
        return employeeDbRepository.save(employeeDb);
    }

    public List<EmployeeDb> getAllEmployees(){ 1 usage
        return employeeDbRepository.findAll();
    }

    public Optional<EmployeeDb> getEmployeeById(Long employeeId){ 1 usage
        return employeeDbRepository.findById(employeeId);
    }
}
```

## 6.Controller

```
package com.example.demo_employee_mongo_db.controller;

import com.example.demo_employee_mongo_db.entity.EmployeeDb;
import com.example.demo_employee_mongo_db.service.EmployeeDbService;
import lombok.AllArgsConstructor;
import org.springframework.http.MediaType;
import org.springframework.stereotype.Controller;
import org.springframework.ui.Model;
import org.springframework.web.bind.annotation.*;

import java.util.List;

@Controller
@AllArgsConstructor
public class EmployeeDbController {
    private EmployeeDbService employeeDbService;

    @GetMapping("/")
    public String indexPage(Model model) {
        model.addAttribute("employeeDb", new EmployeeDb());
        return "index";
    }

    @PostMapping("/employeeSave")
    public String saveEmployee(@ModelAttribute EmployeeDb employeeDb, Model model) {
        employeeDbService.saveEmployee(employeeDb);
        model.addAttribute("message", "Employee Saved Successfully!");
        // return "index";
        return "redirect:/displayAll";
    }
}
```

Close. Alt+Click to Close Others (Ctrl+F4)

```
public class EmployeeDbController {  
    public String saveEmployee(@ModelAttribute EmployeeDb employeeDb, Model model) {  
    }  
    @PostMapping("/employee/api") no usages  
    @ResponseBody  
    public EmployeeDb saveEmployeeApi(@RequestBody EmployeeDb employeeDb) {  
        return employeeDbService.saveEmployee(employeeDb);  
    }  
    |  
    @GetMapping("/displayAll") no usages  
    @ResponseBody  
    public List<EmployeeDb> displayAllEmployeesDb() {  
        return employeeDbService.getAllEmployees();  
    }  
    @GetMapping("/display/{employeeId}") no usages  
    @ResponseBody  
    public EmployeeDb displayEmployeesDb(@PathVariable Long employeeId){  
        return employeeDbService.getEmployeeById(employeeId).orElseThrow(() -> new RuntimeException("Employee Not found"));  
    }  
    // @GetMapping("/displayAll")
```

## 7.Index

```
1 <!DOCTYPE html>
2 <html lang="en" xmlns:th="http://www.thymeleaf.org">
3 <head>
4     <meta charset="UTF-8">
5     <title>Employees Form</title>
6 </head>
7 <body>
8 <h2> Enter Employee Details</h2>
9 <form th:action="@{/employeeSave}" th:object="${employeeDb}" method="post">
10     <label>Employee ID:</label>
11     <input type="text" th:field="*{employeeId}" required/><br>
12
13     <label>Employee Name:</label>
14     <input type="text" th:field="*{name}" required/><br>
15
16     <label>Employee Email:</label>
17     <input type="email" th:field="*{email}" required/><br>
18
19     <label>Employee Location:</label>
20     <input type="text" th:field="*{location}" required/><br>
21     <button type="submit">Submit</button>
22
23 <!-- <p th:text="${message}"></p-->
24 </form>
25 </body>
26 </html>
```

```

      . ____ .      ____ .
/ \ / ___' _ _ _ _ _ _ _ _ \ \ \
( ) \___ | ' | ' | ' | ' \___ | \ \ \
\ \ ___ | | | | | | | | ( | | ) ) )
' | ___ | . | | | | | \___ | / / /
=====|_|=====|___/ _/_/_/_

:: Spring Boot ::                (v3.5.4)

2025-09-19T19:44:00.696+05:30 INFO 15552 --- [demo-employee-mongo-db] [      main] c.e.d.DemoEmployeeMongoDbApplication : Starting DemoEmployeeMongoDbApplication
using Java 24.0.2 with PID 15552 (C:\Users\mrwis\Downloads\java_program_and_learn_and_training_doc\demo-employee-mongo-db\target\classes started by mrwis in
C:\Users\mrwis\Downloads\java_program_and_learn_and_training_doc\demo-employee-mongo-db)

2025-09-19T19:44:00.699+05:30 INFO 15552 --- [demo-employee-mongo-db] [      main] c.e.d.DemoEmployeeMongoDbApplication : No active profile set, falling back to
default profile: "default"

2025-09-19T19:44:02.282+05:30 INFO 15552 --- [demo-employee-mongo-db] [      main] .s.d.r.c.RepositoryConfigurationDelegate : Bootstrapping Spring Data MongoDB
repositories in DEFAULT mode.

2025-09-19T19:44:02.409+05:30 INFO 15552 --- [demo-employee-mongo-db] [      main] .s.d.r.c.RepositoryConfigurationDelegate : Finished Spring Data repository scanning
in 102 ms. Found 1 MongoDB repository interface.

2025-09-19T19:44:03.232+05:30 INFO 15552 --- [demo-employee-mongo-db] [      main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port 8080 (http)

2025-09-19T19:44:03.262+05:30 INFO 15552 --- [demo-employee-mongo-db] [      main] o.apache.catalina.core.StandardService : Starting service [Tomcat]

2025-09-19T19:44:03.263+05:30 INFO 15552 --- [demo-employee-mongo-db] [      main] o.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache
Tomcat/10.1.43]

2025-09-19T19:44:03.341+05:30 INFO 15552 --- [demo-employee-mongo-db] [      main] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded
WebApplicationContext

2025-09-19T19:44:03.342+05:30 INFO 15552 --- [demo-employee-mongo-db] [      main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext:
initialization completed in 2451 ms

2025-09-19T19:44:03.749+05:30 INFO 15552 --- [demo-employee-mongo-db] [      main] org.mongodb.driver.client : MongoClient with metadata {"driver":
{"name": "mongo-java-driver|sync|spring-boot", "version": "5.5.1"}, "os": {"type": "Windows", "name": "Windows 11", "architecture": "amd64", "version": "10.0"},
"platform": "Java (Oracle Corporation/24.0.2/12.5.1)} created with settings MongoClientSettings{readPreference=primary, writeConcern=WriteConcern{w=1, wTimeout=null, ms

```