Spring MongoDB

~Add the following dependency:-

*Spring Web

*Spring Boot DevTools

*Lombok

*Spring Data MongoDB

Book.java

```
package com.example.demo;
import org.springframework.data.annotation.Id;
import org.springframework.data.mongodb.core.mapping.Document;
import lombok.AllArgsConstructor;
import lombok.Data;
import lombok. NoArgsConstructor;
@Data
@NoArgsConstructor
@AllArgsConstructor
@Document(collection="Book")
public class Book {
@Id
private int id;
private String bookName;
private String authorName;
public Book()
{
}
public Book(int id, String bookName, String authorName)
{
```

```
super();
this.id=id;
this.bookName=bookName;
this.authorName=authorName;
}
public int getId() {

return id;
}
public void setld(int id) { this.id = id;
}
public String getBookName() { return bookName;
}
public void setBookName(String bookName) { this.bookName = bookName;
} public String getAuthorName() { return authorName;
}
public void setAuthorName(String authorName) { this.authorName = authorName;
    }
}
```

BookRepo.java

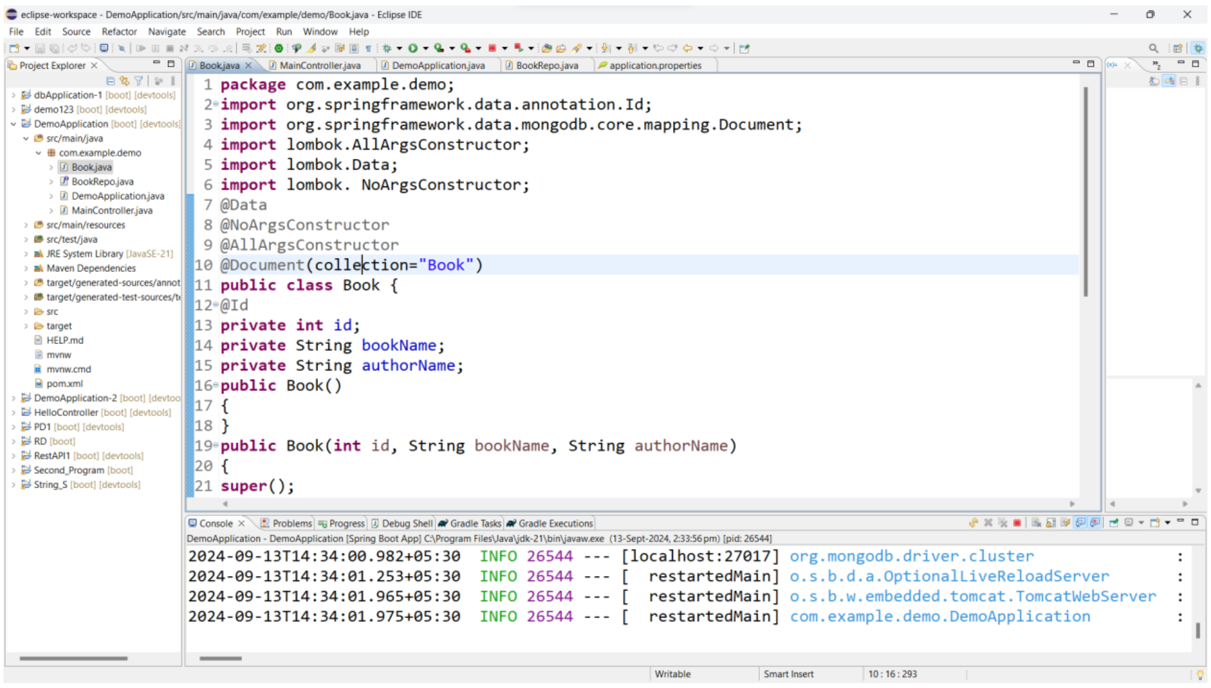
```
package com.example.demo;

import org.springframework.data.mongodb.repository.MongoRepository;
import org.springframework.stereotype.Repository;

@Repository
public interface BookRepo extends MongoRepository<Book, Integer> {
    // No custom methods needed for basic CRUD operations.
    // MongoRepository already provides methods like save(), findAll(), findById(), and deleteById().
}
```

DemoApplication.java

```
package com.example.demo;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication
public class DemoApplication {

        public static void main(String[] args) {
                SpringApplication.run(DemoApplication.class, args);
        }

}
```

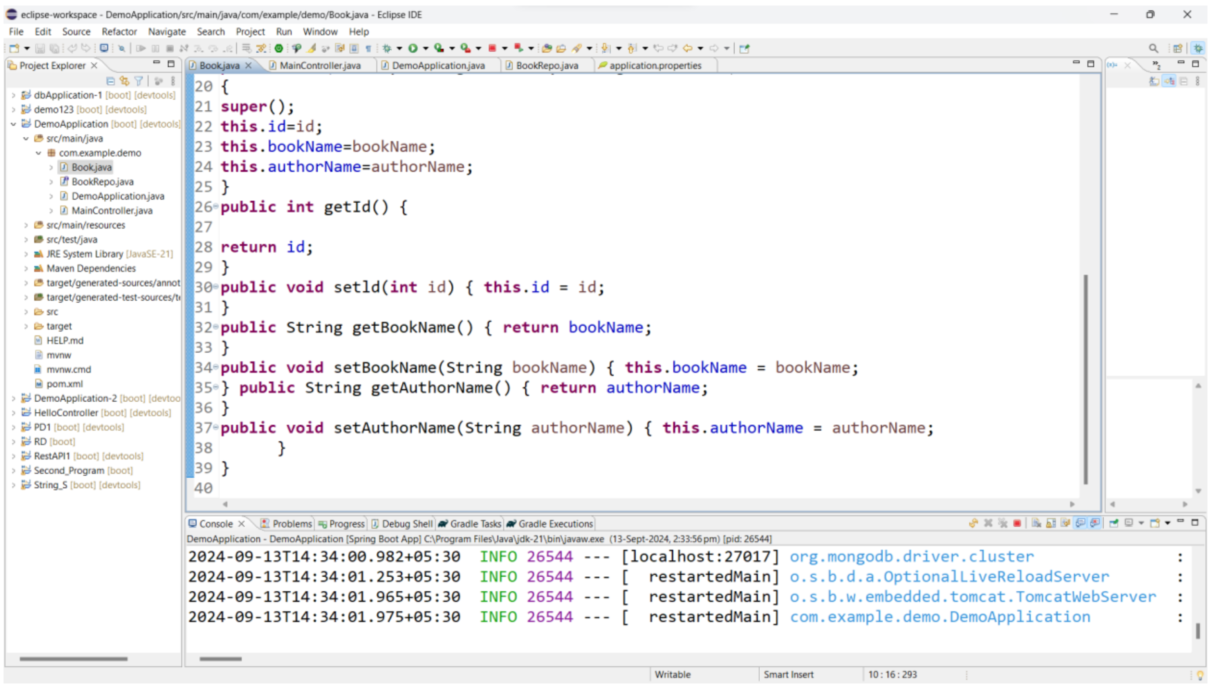
MainController.java

```
package com.example.demo;
import java.util.List;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.web.bind.annotation.DeleteMapping;
import org.springframework.web.bind.annotation.PathVariable;
import org.springframework.web.bind.annotation.PostMapping;
import org.springframework.web.bind.annotation.PutMapping;
import org.springframework.web.bind.annotation.RequestBody;
import org.springframework.web.bind.annotation.RestController;
import org.springframework.web.bind.annotation.GetMapping;

@RestController
public class MainController {
@Autowired
private BookRepo repo;

@PostMapping("/addbook")
public String saveBook(@RequestBody Book book)
{
repo.save(book);
return "Added Successfully";
}

@GetMapping("/findAllBooks")
public List<Book>getBooks()
{
return repo.findAll();
}
@DeleteMapping("/delete/{id}")
public String deleteBook(@PathVariable int id)
{
repo.deleteById(id);
return"deleted Successfully";
}
@PutMapping(""
+ "")
public String updateBook(@RequestBody Book book)
{
repo.save(book);
return "update Successfully";
}
}
```


eclipse-workspace - DemoApplication/src/main/java/com/example/demo/Book.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Project Explorer
dbApplication-1 [boot] [devtools]
demo123 [boot] [devtools]
DemoApplication [boot] [devtools]
src/main/java
com.example.demo
Book.java
BookRepo.java
DemoApplication.java
MainController.java
src/main/resources
src/test/java
JRE System Library [JavaSE-21]
Maven Dependencies
target/generated-sources/annot
target/generated-test-sources/t
src
target
HELP.md
mvnw
mvnw.cmd
pom.xml
DemoApplication-2 [boot] [devtoo
HelloController [boot] [devtools]
PD1 [boot] [devtools]
RD [boot]
RestAPI1 [boot] [devtools]
Second_Program [boot]
String_S [boot] [devtools]
Book.java
MainController.java
DemoApplication.java
BookRepo.java
application.properties
Console
Problems
Progress
Debug Shell
Gradle Tasks
Gradle Executions
Writable
Smart Insert
10 : 16 : 293

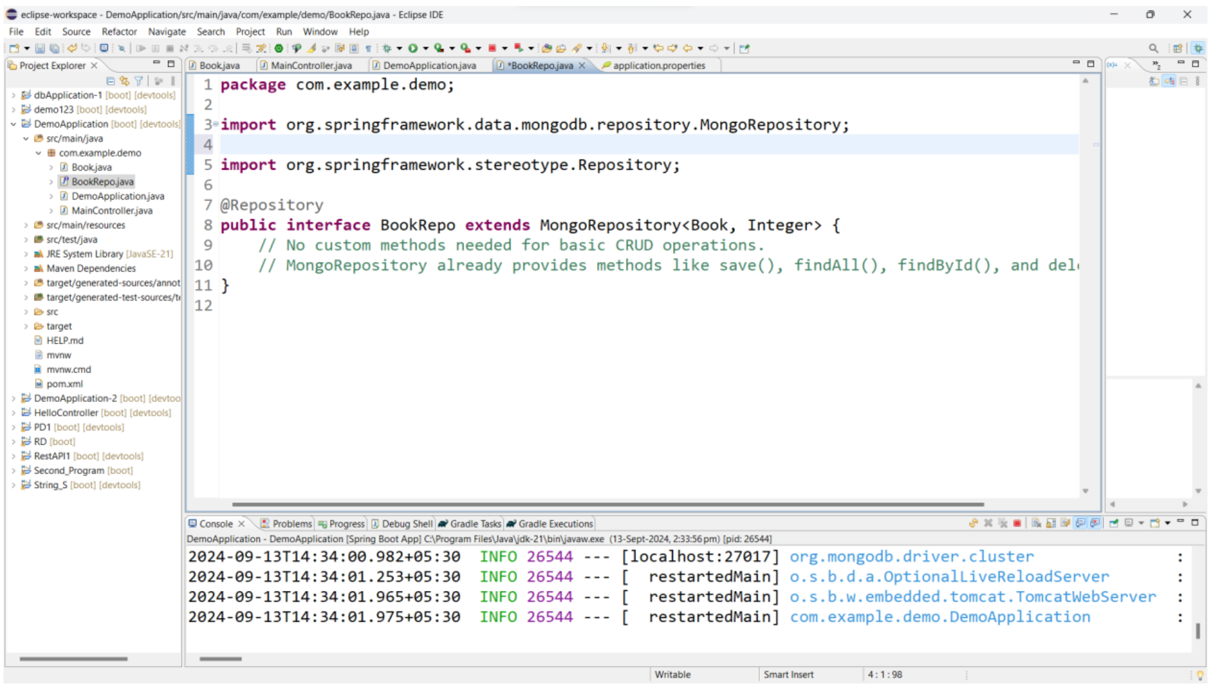
```java
package com.example.demo;
import org.springframework.data.annotation.Id;
import org.springframework.data.mongodb.core.mapping.Document;
import lombok.AllArgsConstructor;
import lombok.Data;
import lombok. NoArgsConstructor;
@Data
@NoArgsConstructor
@AllArgsConstructor
@Document(collection="Book")
public class Book {
@Id
private int id;
private String bookName;
private String authorName;
public Book()
{
}
public Book(int id, String bookName, String authorName)
{
super();
```

```
DemoApplication - DemoApplication [Spring Boot App] C:\Program Files\Java\jdk-21\bin\javaw.exe  (13-Sept-2024, 2:33:56 pm) [pid: 26544]
2024-09-13T14:34:00.982+05:30  INFO 26544 --- [localhost:27017] org.mongodb.driver.cluster              :
2024-09-13T14:34:01.253+05:30  INFO 26544 --- [  restartedMain] o.s.b.d.a.OptionalLiveReloadServer      :
2024-09-13T14:34:01.965+05:30  INFO 26544 --- [  restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer :
2024-09-13T14:34:01.975+05:30  INFO 26544 --- [  restartedMain] com.example.demo.DemoApplication        :
```



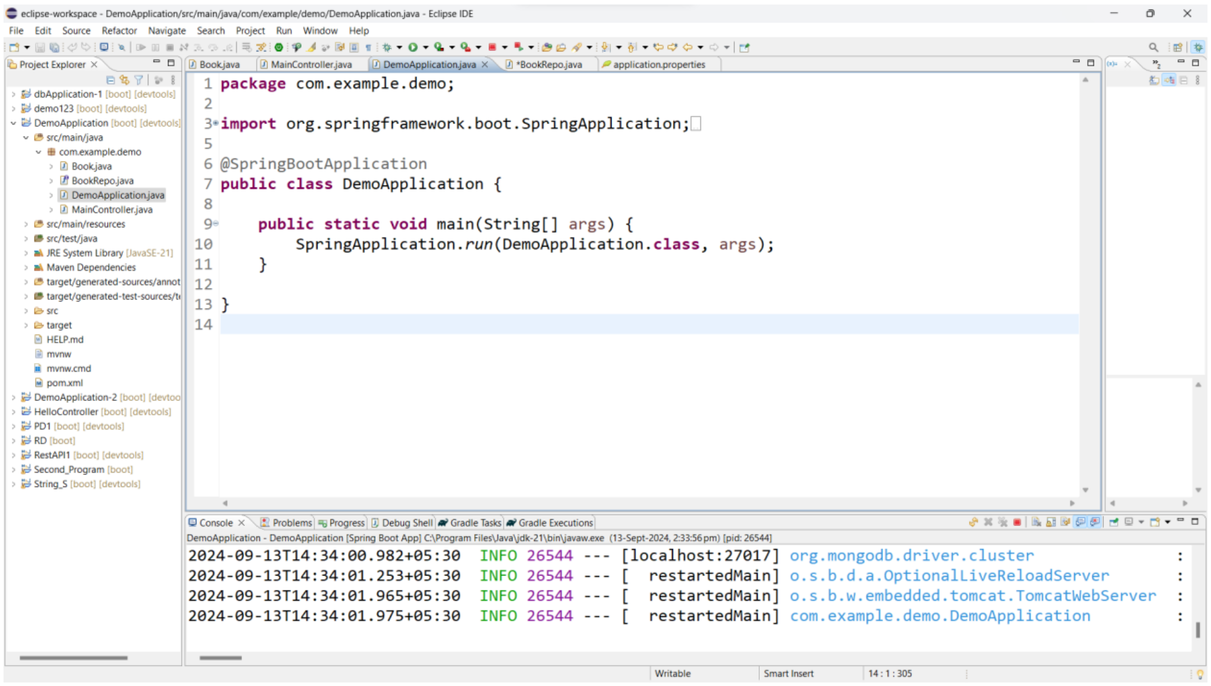
```
{
super();
this.id=id;
this.bookName=bookName;
this.authorName=authorName;
}
public int getId() {

return id;
}
public void setId(int id) { this.id = id;
}
public String getBookName() { return bookName;
}
public void setBookName(String bookName) { this.bookName = bookName;
} public String getAuthorName() { return authorName;
}
public void setAuthorName(String authorName) { this.authorName = authorName;
        }
}
```

```
DemoApplication - DemoApplication [Spring Boot App] C:\Program Files\Java\jdk-21\bin\javaw.exe (13-Sept-2024, 2:33:56 pm) [pid: 26544]
2024-09-13T14:34:00.982+05:30  INFO 26544 --- [localhost:27017] org.mongodb.driver.cluster               :
2024-09-13T14:34:01.253+05:30  INFO 26544 --- [  restartedMain] o.s.b.d.a.OptionalLiveReloadServer       :
2024-09-13T14:34:01.965+05:30  INFO 26544 --- [  restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer  :
2024-09-13T14:34:01.975+05:30  INFO 26544 --- [  restartedMain] com.example.demo.DemoApplication         :
```



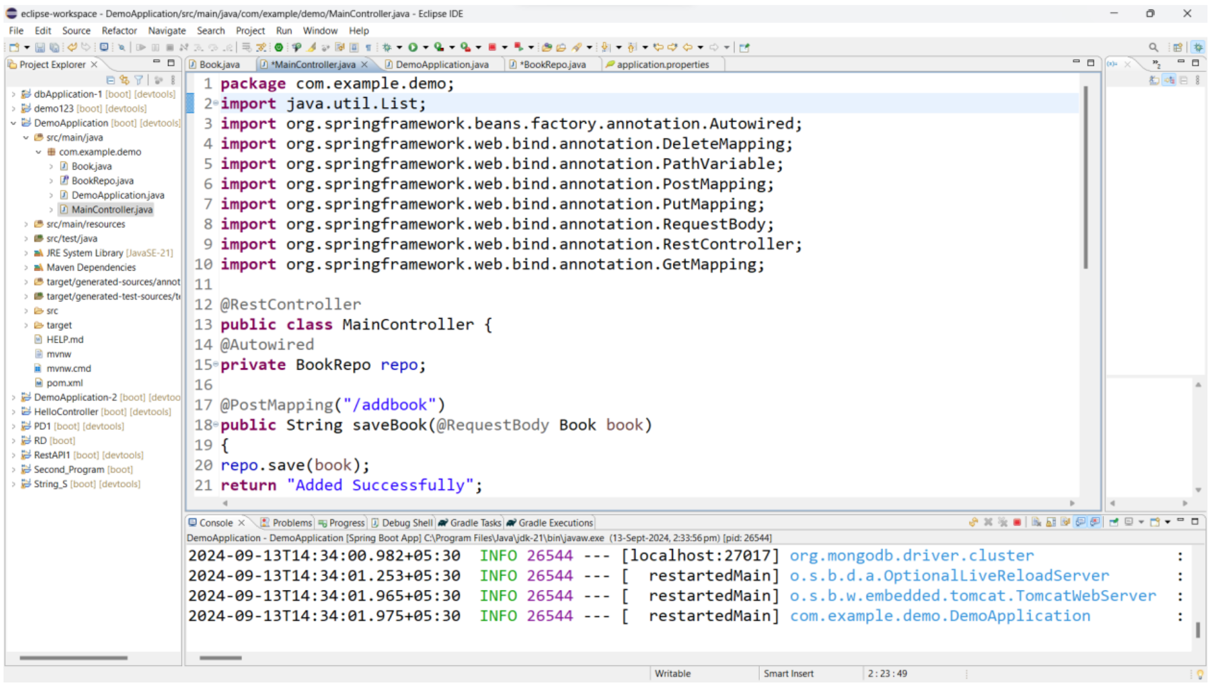
```
package com.example.demo;

import org.springframework.data.mongodb.repository.MongoRepository;

import org.springframework.stereotype.Repository;

@Repository
public interface BookRepo extends MongoRepository<Book, Integer> {
    // No custom methods needed for basic CRUD operations.
    // MongoRepository already provides methods like save(), findAll(), findById(), and del
}

```

```
DemoApplication - DemoApplication [Spring Boot App] C:\Program Files\Java\jdk-21\bin\javaw.exe (13-Sept-2024, 2:33:56 pm) [pid: 26544]
2024-09-13T14:34:00.982+05:30  INFO 26544 --- [localhost:27017] org.mongodb.driver.cluster               :
2024-09-13T14:34:01.253+05:30  INFO 26544 --- [  restartedMain] o.s.b.d.a.OptionalLiveReloadServer       :
2024-09-13T14:34:01.965+05:30  INFO 26544 --- [  restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer  :
2024-09-13T14:34:01.975+05:30  INFO 26544 --- [  restartedMain] com.example.demo.DemoApplication         :
```



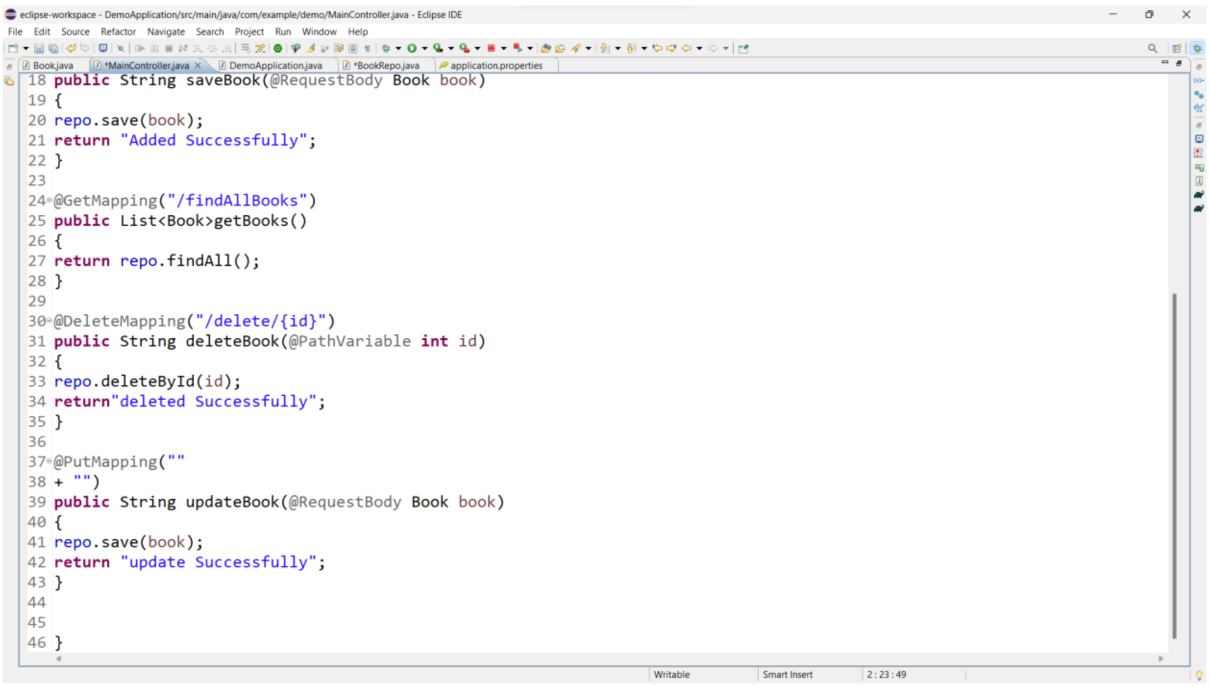
```java
package com.example.demo;

import org.springframework.boot.SpringApplication;

@SpringBootApplication
public class DemoApplication {

    public static void main(String[] args) {
        SpringApplication.run(DemoApplication.class, args);
    }

}
```

```
DemoApplication - DemoApplication [Spring Boot App] C:\Program Files\Java\jdk-21\bin\javaw.exe (13-Sept-2024, 2:33:56 pm) [pid: 26544]
2024-09-13T14:34:00.982+05:30  INFO 26544 --- [localhost:27017] org.mongodb.driver.cluster             :
2024-09-13T14:34:01.253+05:30  INFO 26544 --- [  restartedMain] o.s.b.d.a.OptionalLiveReloadServer     :
2024-09-13T14:34:01.965+05:30  INFO 26544 --- [  restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer :
2024-09-13T14:34:01.975+05:30  INFO 26544 --- [  restartedMain] com.example.demo.DemoApplication       :
```

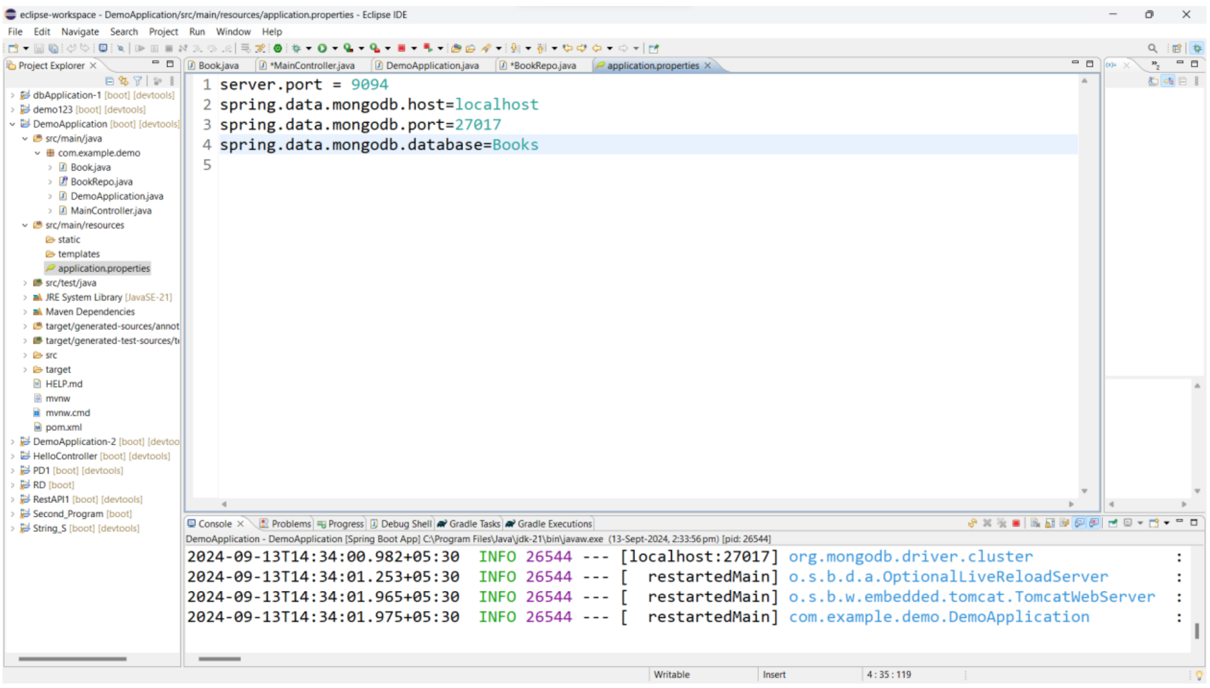


```java
package com.example.demo;
import java.util.List;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.web.bind.annotation.DeleteMapping;
import org.springframework.web.bind.annotation.PathVariable;
import org.springframework.web.bind.annotation.PostMapping;
import org.springframework.web.bind.annotation.PutMapping;
import org.springframework.web.bind.annotation.RequestBody;
import org.springframework.web.bind.annotation.RestController;
import org.springframework.web.bind.annotation.GetMapping;

@RestController
public class MainController {
@Autowired
private BookRepo repo;

@PostMapping("/addbook")
public String saveBook(@RequestBody Book book)
{
repo.save(book);
return "Added Successfully";
```

```
DemoApplication - DemoApplication [Spring Boot App] C:\Program Files\Java\jdk-21\bin\javaw.exe (13-Sept-2024, 2:33:56 pm) [pid: 26544]
2024-09-13T14:34:00.982+05:30  INFO 26544 --- [localhost:27017] org.mongodb.driver.cluster             :
2024-09-13T14:34:01.253+05:30  INFO 26544 --- [  restartedMain] o.s.b.d.a.OptionalLiveReloadServer     :
2024-09-13T14:34:01.965+05:30  INFO 26544 --- [  restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer :
2024-09-13T14:34:01.975+05:30  INFO 26544 --- [  restartedMain] com.example.demo.DemoApplication       :
```



```java
public String saveBook(@RequestBody Book book)
{
repo.save(book);
return "Added Successfully";
}

@GetMapping("/findAllBooks")
public List<Book>getBooks()
{
return repo.findAll();
}

@DeleteMapping("/delete/{id}")
public String deleteBook(@PathVariable int id)
{
repo.deleteById(id);
return"deleted Successfully";
}

@PutMapping(""
+ "")
public String updateBook(@RequestBody Book book)
{
repo.save(book);
return "update Successfully";
}


}
```

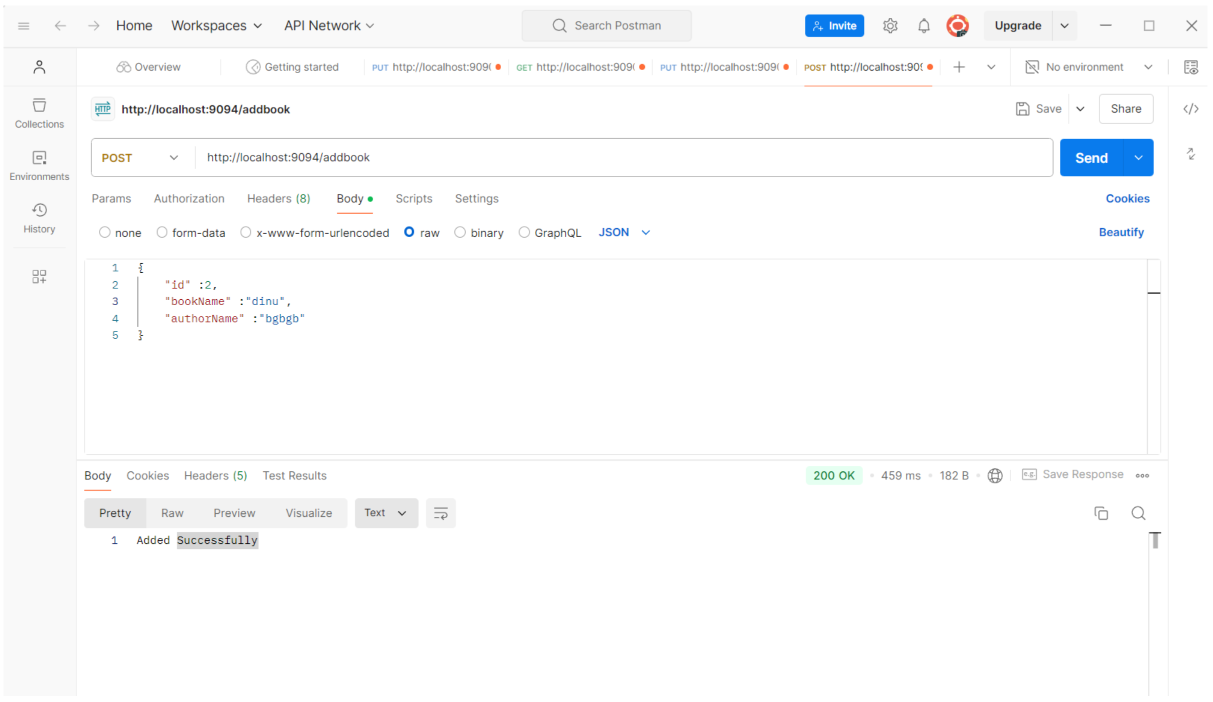


```
server.port = 9094
spring.data.mongodb.host=localhost
spring.data.mongodb.port=27017
spring.data.mongodb.database=Books
```

```
DemoApplication - DemoApplication [Spring Boot App] C:\Program Files\Java\jdk-21\bin\javaw.exe (13-Sept-2024, 2:33:56 pm) [pid: 26544]
2024-09-13T14:34:00.982+05:30  INFO 26544 --- [localhost:27017] org.mongodb.driver.cluster              :
2024-09-13T14:34:01.253+05:30  INFO 26544 --- [  restartedMain] o.s.b.d.a.OptionalLiveReloadServer      :
2024-09-13T14:34:01.965+05:30  INFO 26544 --- [  restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer :
2024-09-13T14:34:01.975+05:30  INFO 26544 --- [  restartedMain] com.example.demo.DemoApplication        :
```

Output:-

POST= http://localhost:9094/addbook

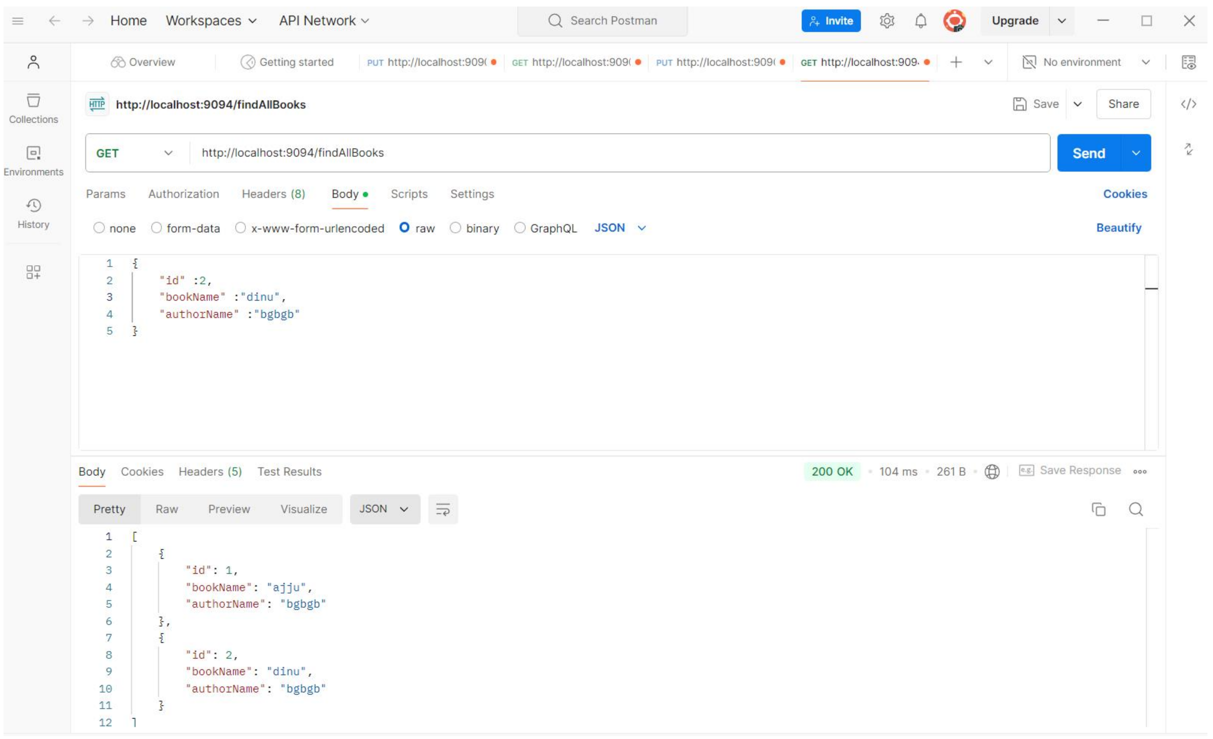

POST http://localhost:9094/addbook

Body raw JSON:
```
{
    "id" :2,
    "bookName" :"dinu",
    "authorName" :"bgbgb"
}
```

Response: 200 OK 459 ms 182 B

```
Added Successfully
```

PUT= http://localhost:9094/findAllBooks

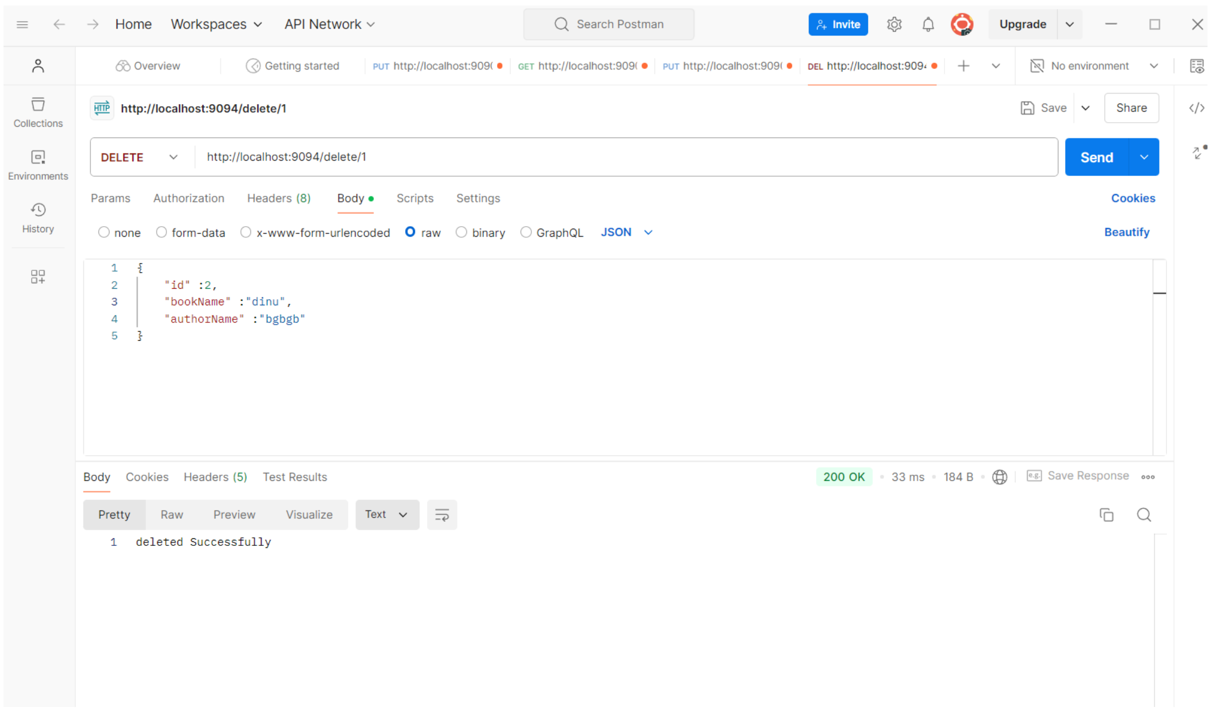

GET http://localhost:9094/findAllBooks

Body raw JSON:
```
{
    "id" :2,
    "bookName" :"dinu",
    "authorName" :"bgbgb"
}
```

Response: 200 OK 104 ms 261 B

```
[
    {
        "id": 1,
        "bookName": "ajju",
        "authorName": "bgbgb"
    },
    {
        "id": 2,
        "bookName": "dinu",
        "authorName": "bgbgb"
    }
]
```

Delete= http://localhost:9094/delete/1



http://localhost:9094/delete/1

DELETE http://localhost:9094/delete/1

```
{
    "id" :2,
    "bookName" :"dinu",
    "authorName" :"bgbgb"
}
```

200 OK · 33 ms · 184 B

```
deleted Successfully
```