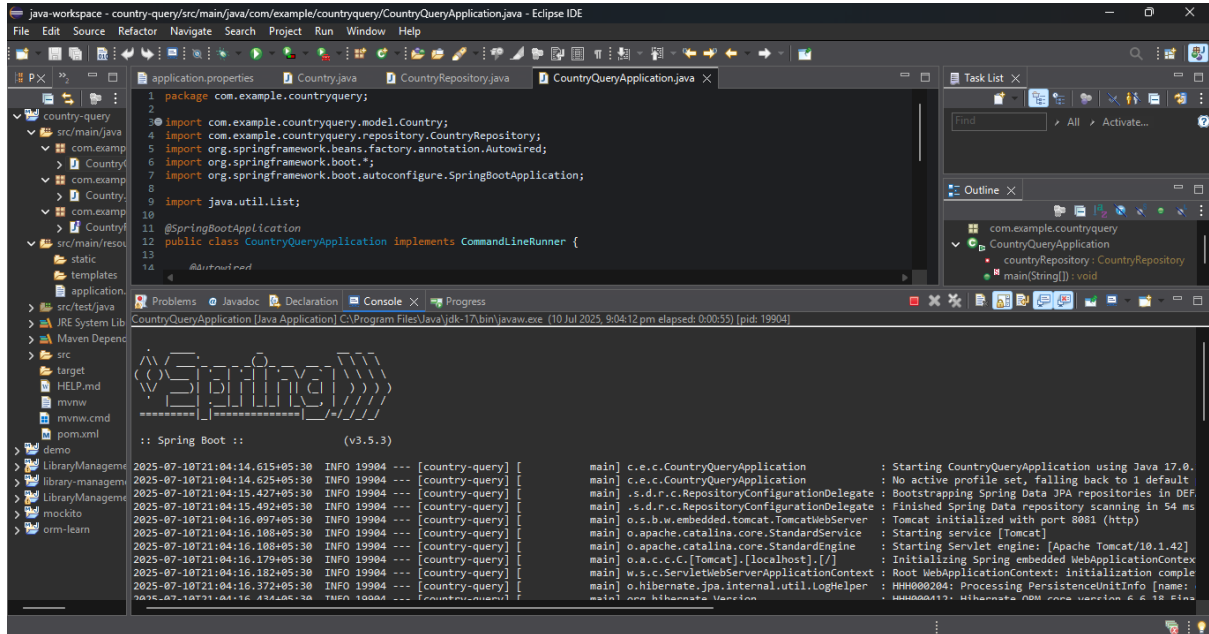


2. spring-data-jpa-handson

Hands on 1

Write queries on country table using Query Methods



The screenshot shows the Eclipse IDE with the `CountryQueryApplication.java` file open. The code defines a `CountryQueryApplication` class that implements `CommandLineRunner`. It uses `CountryRepository` to query the `Country` table. The console output shows the application starting successfully on port 8081.

```
package com.example.countryquery;

import com.example.countryquery.model.Country;
import com.example.countryquery.repository.CountryRepository;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.boot.*;
import java.util.List;

@SpringBootApplication
public class CountryQueryApplication implements CommandLineRunner {

    @Autowired
    CountryRepository countryRepository;

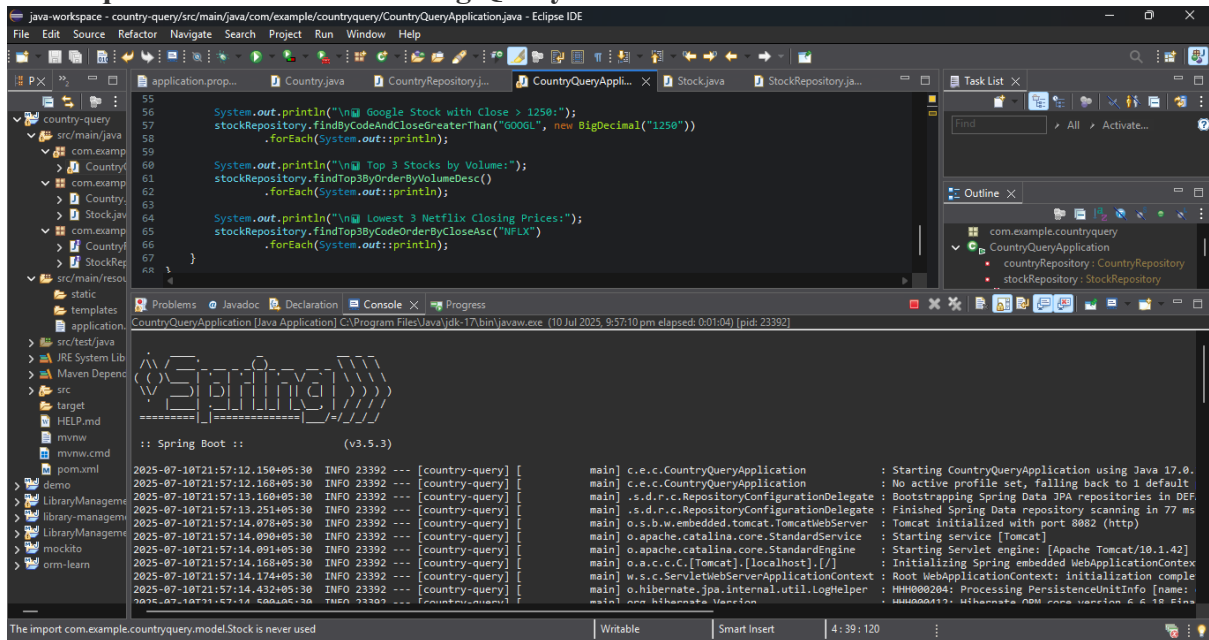
    public void run(String... args) throws Exception {
        List<Country> countries = countryRepository.findAll();
        System.out.println("All Countries:");
        countries.forEach(System.out::println);
    }
}
```

Console Output:

```
2025-07-10T21:04:14.615+05:30 INFO 19904 --- [country-query] [main] c.e.c.CountryQueryApplication : Starting CountryQueryApplication using Java 17.0.
2025-07-10T21:04:14.625+05:30 INFO 19904 --- [country-query] [main] c.e.c.CountryQueryApplication : No active profile set, falling back to 1 default
2025-07-10T21:04:15.427+05:30 INFO 19904 --- [country-query] [main] s.d.r.c.RepositoryConfigurationDelegate : Bootstrapping Spring Data JPA repositories in DEF
2025-07-10T21:04:15.492+05:30 INFO 19904 --- [country-query] [main] s.d.r.c.RepositoryConfigurationDelegate : Finished Spring Data repository scanning in 54 ms
2025-07-10T21:04:16.097+05:30 INFO 19904 --- [country-query] [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port 8081 (http)
2025-07-10T21:04:16.108+05:30 INFO 19904 --- [country-query] [main] o.apache.catalina.core.StandardService : Starting service [Tomcat]
2025-07-10T21:04:16.108+05:30 INFO 19904 --- [country-query] [main] o.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/10.1.42]
2025-07-10T21:04:16.179+05:30 INFO 19904 --- [country-query] [main] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext
2025-07-10T21:04:16.182+05:30 INFO 19904 --- [country-query] [main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization complete
2025-07-10T21:04:16.372+05:30 INFO 19904 --- [country-query] [main] o.hibernate.jpa.internal.util.LogHelper : HH0000284: Processing PersistenceUnitInfo [name:
2025-07-10T21:04:16.434+05:30 INFO 19904 --- [country-query] [main] org.hibernate.Version : HH0000417: Hibernate ORM core version 5.6.18 Final
```

Hands on 2

Write queries on stock table using Query Methods



The screenshot shows the Eclipse IDE with the `CountryQueryApplication.java` file open. The code has been updated to include queries for the `Stock` table. The console output shows the application starting successfully on port 8082.

```
System.out.println("\n Google Stock with Close > 1250:");
stockRepository.findByCodeAndCloseGreaterThan("GOOGL", new BigDecimal("1250"))
    .forEach(System.out::println);

System.out.println("\n Top 3 Stocks by Volume:");
stockRepository.findTop3ByOrderByVolumeDesc()
    .forEach(System.out::println);

System.out.println("\n Lowest 3 Netflix Closing Prices:");
stockRepository.findTop3ByCodeOrderByCloseAsc("NFLX")
    .forEach(System.out::println);
```

Console Output:

```
2025-07-10T21:57:12.150+05:30 INFO 23392 --- [country-query] [main] c.e.c.CountryQueryApplication : Starting CountryQueryApplication using Java 17.0.
2025-07-10T21:57:12.168+05:30 INFO 23392 --- [country-query] [main] c.e.c.CountryQueryApplication : No active profile set, falling back to 1 default
2025-07-10T21:57:13.251+05:30 INFO 23392 --- [country-query] [main] s.d.r.c.RepositoryConfigurationDelegate : Bootstrapping Spring Data JPA repositories in DEF
2025-07-10T21:57:13.251+05:30 INFO 23392 --- [country-query] [main] s.d.r.c.RepositoryConfigurationDelegate : Finished Spring Data repository scanning in 77 ms
2025-07-10T21:57:14.078+05:30 INFO 23392 --- [country-query] [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port 8082 (http)
2025-07-10T21:57:14.090+05:30 INFO 23392 --- [country-query] [main] o.apache.catalina.core.StandardService : Starting service [Tomcat]
2025-07-10T21:57:14.091+05:30 INFO 23392 --- [country-query] [main] o.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/10.1.42]
2025-07-10T21:57:14.168+05:30 INFO 23392 --- [country-query] [main] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext
2025-07-10T21:57:14.174+05:30 INFO 23392 --- [country-query] [main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization complete
2025-07-10T21:57:14.432+05:30 INFO 23392 --- [country-query] [main] o.hibernate.jpa.internal.util.LogHelper : HH0000284: Processing PersistenceUnitInfo [name:
2025-07-10T21:57:14.434+05:30 INFO 23392 --- [country-query] [main] org.hibernate.Version : HH0000417: Hibernate ORM core version 5.6.18 Final
```

Hands on 3

Create payroll tables and bean mapping

```
CountryQueryApplication.java
42 list3.forEach(c -> System.out.println(c.getCode() + " - " + c.getName()));
43
44 System.out.println("\n Countries starting with 'Z':");
45 List<Country> list3 = countryRepository.findByNameStartingWith("Z");
46 list3.forEach(c -> System.out.println(c.getCode() + " - " + c.getName()));
47
48 // Hands-on 3 logic
49 private void runCountryByCode() {
50     System.out.println("\n Country with Code 'IN':");
51     countryRepository.findByCode("IN")
52         .ifPresentOrElse(
53             country -> System.out.println(country.getCode() + " - " + country.getName()),
54             () -> System.out.println("Country not found")
55         );
56 }
```

```
CountryQueryApplication [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe (10 Jul 2025, 10:02:54 pm elapsed: 0:00:17) [pid: 22320]
:: Spring Boot :: (v3.5.3)
2025-07-10T22:02:56.569+05:30 INFO 22320 --- [country-query] [main] c.e.c.CountryQueryApplication : Starting CountryQueryApplication using Java 17.0.
2025-07-10T22:02:56.576+05:30 INFO 22320 --- [country-query] [main] c.e.c.CountryQueryApplication : No active profile set, falling back to 1 default
2025-07-10T22:02:57.419+05:30 INFO 22320 --- [country-query] [main] s.d.r.c.RepositoryConfigurationDelegate : Bootstrapping Spring Data JPA repositories in DEF
2025-07-10T22:02:57.502+05:30 INFO 22320 --- [country-query] [main] s.d.r.c.RepositoryConfigurationDelegate : Finished Spring Data repository scanning in 68 ms
2025-07-10T22:02:58.134+05:30 INFO 22320 --- [country-query] [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port 8083 (http)
2025-07-10T22:02:58.144+05:30 INFO 22320 --- [country-query] [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Starting service [Tomcat]
2025-07-10T22:02:58.145+05:30 INFO 22320 --- [country-query] [main] o.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/10.1.42]
2025-07-10T22:02:58.228+05:30 INFO 22320 --- [country-query] [main] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext
2025-07-10T22:02:58.223+05:30 INFO 22320 --- [country-query] [main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization comple
2025-07-10T22:02:58.405+05:30 INFO 22320 --- [country-query] [main] o.hibernate.jpa.internal.util.LogHelper : HHH000204: Processing PersistenceUnitInfo [name:
2025-07-10T22:02:58.463+05:30 INFO 22320 --- [country-query] [main] org.hibernate.Version : HHH000417: Hibernate ORM core version 6.6.18.Final
```

Hands on 4

Implement many to one relationship between Employee and Department

```
CountryQueryApplication.java
29 public void run(String... args) {
30     runCountryQueries(); // Hands-on 1
31     runCountryByCode(); // Hands-on 3
32     runStockQueries(); // Hands-on 2
33 }
34
35 private void runCountryQueries() {
36     System.out.println("\n Countries containing 'ou':");
37     List<Country> list1 = countryRepository.findByNameContaining("ou");
38     list1.forEach(c -> System.out.println(c.getCode() + " - " + c.getName()));
39 }
40
41 System.out.println("\n Countries containing 'ou' (sorted):");
42 List<Country> list2 = countryRepository.findByNameContainingOrderByNameAsc("ou");
43 list2.forEach(c -> System.out.println(c.getCode() + " - " + c.getName()));
44 }
```

```
CountryQueryApplication [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe (10 Jul 2025, 10:07:59 pm elapsed: 0:00:18) [pid: 21516]
:: Spring Boot :: (v3.5.3)
2025-07-10T22:08:01.675+05:30 INFO 21516 --- [country-query] [main] c.e.c.CountryQueryApplication : Starting CountryQueryApplication using Java 17.0.
2025-07-10T22:08:01.681+05:30 INFO 21516 --- [country-query] [main] c.e.c.CountryQueryApplication : No active profile set, falling back to 1 default
2025-07-10T22:08:03.794+05:30 INFO 21516 --- [country-query] [main] s.d.r.c.RepositoryConfigurationDelegate : Bootstrapping Spring Data JPA repositories in DEF
2025-07-10T22:08:03.898+05:30 INFO 21516 --- [country-query] [main] s.d.r.c.RepositoryConfigurationDelegate : Finished Spring Data repository scanning in 85 ms
2025-07-10T22:08:04.812+05:30 INFO 21516 --- [country-query] [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port 8084 (http)
2025-07-10T22:08:04.832+05:30 INFO 21516 --- [country-query] [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Starting service [Tomcat]
2025-07-10T22:08:04.833+05:30 INFO 21516 --- [country-query] [main] o.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/10.1.42]
2025-07-10T22:08:04.896+05:30 INFO 21516 --- [country-query] [main] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext
2025-07-10T22:08:04.899+05:30 INFO 21516 --- [country-query] [main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization comple
2025-07-10T22:08:05.148+05:30 INFO 21516 --- [country-query] [main] o.hibernate.jpa.internal.util.LogHelper : HHH000204: Processing PersistenceUnitInfo [name:
2025-07-10T22:08:05.316+05:30 INFO 21516 --- [country-query] [main] org.hibernate.Version : HHH000417: Hibernate ORM core version 6.6.18.Final
```

Hands on 5

Implement one to many relationship between Employee and Department

```
CountryQueryApplication.java - country-query/src/main/java
55  system.out.println("Country with code 'IN' : ");
56  countryRepository.findById("IN")
57  .ifPresentOrElse(
58      country -> System.out.println(country.getCode() + " - " + country.getName()),
59      () -> System.out.println("Country not found")
60  );
61  }
62  // Hands-on 5
63  private void runCountryUpdate() {
64      System.out.println("\n% Updating Country name for 'IN' to 'Bharat'");
65      countryService.updateCountryName("IN", "Bharat");
66      countryRepository.findById("IN").ifPresent(c ->
67          System.out.println("✓ Updated: " + c.getCode() + " - " + c.getName())
68      );
69  }
```

```
CountryQueryApplication [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe (10 Jul 2025, 10:11:29 pm elapsed: 0:00:16) [pid: 22736]
com.example.countryquery.model.Stock@7d0893d
com.example.countryquery.model.Stock@69cc3370

Google Stock with Close > 1250:
Hibernate: select s1_0.st_id,s1_0.st_close,s1_0.st_code,s1_0.st_date,s1_0.st_open,s1_0.st_volume from stock s1_0 where s1_0.st_code=? and s1_0.st_close=?
com.example.countryquery.model.Stock@7c794986
com.example.countryquery.model.Stock@32d3b965
com.example.countryquery.model.Stock@7fb2645b

Top 3 Stocks by Volume:
Hibernate: select s1_0.st_id,s1_0.st_close,s1_0.st_code,s1_0.st_date,s1_0.st_open,s1_0.st_volume from stock s1_0 order by s1_0.st_volume desc limit ?
com.example.countryquery.model.Stock@51d41b97
com.example.countryquery.model.Stock@7fd3c38b
com.example.countryquery.model.Stock@27260f9c

Lowest 3 Netflix Closing Prices:
Hibernate: select s1_0.st_id,s1_0.st_close,s1_0.st_code,s1_0.st_date,s1_0.st_open,s1_0.st_volume from stock s1_0 where s1_0.st_code=? order by s1_0.st_close limit ?
com.example.countryquery.model.Stock@21ebf9be
com.example.countryquery.model.Stock@7d50b4c4
com.example.countryquery.model.Stock@7860a014
```

Hands on 6

Implement many to many relationship between Employee and Skill

```
CountryQueryApplication.java - country-query/src/main/java
55  system.out.println("Country with code 'IN' : ");
56  countryRepository.findById("IN")
57  .ifPresentOrElse(
58      country -> System.out.println(country.getCode() + " - " + country.getName()),
59      () -> System.out.println("Country not found")
60  );
61  }
62  // Hands-on 5
63  private void runCountryUpdate() {
64      System.out.println("\n% Updating Country name for 'IN' to 'Bharat'");
65      countryService.updateCountryName("IN", "Bharat");
66      countryRepository.findById("IN").ifPresent(c ->
67          System.out.println("✓ Updated: " + c.getCode() + " - " + c.getName())
68      );
69  }
```

```
CountryQueryApplication [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe (10 Jul 2025, 10:26:08 pm elapsed: 0:00:31) [pid: 26386]
IN - Bharat

Updating Country name for 'IN' to 'Bharat'
Hibernate: select c1_0.co_code,c1_0.co_name from country c1_0 where c1_0.co_code=?
Hibernate: select c1_0.co_code,c1_0.co_name from country c1_0 where c1_0.co_code=?
✓ Updated: IN - Bharat

Facebook Stock (Sept 2019):
Hibernate: select s1_0.st_id,s1_0.st_close,s1_0.st_code,s1_0.st_date,s1_0.st_open,s1_0.st_volume from stock s1_0 where s1_0.st_code=? and s1_0.st_date between ? and ?
com.example.countryquery.model.Stock@3f910f36
com.example.countryquery.model.Stock@6421bc7e
com.example.countryquery.model.Stock@2e2546bf

Google Stock with Close > 1250:
Hibernate: select s1_0.st_id,s1_0.st_close,s1_0.st_code,s1_0.st_date,s1_0.st_open,s1_0.st_volume from stock s1_0 where s1_0.st_code=? and s1_0.st_close=?
com.example.countryquery.model.Stock@76ccde41
com.example.countryquery.model.Stock@5f26fc4d
com.example.countryquery.model.Stock@3e8eea59

Top 3 Stocks by Volume:
Hibernate: select s1_0.st_id,s1_0.st_close,s1_0.st_code,s1_0.st_date,s1_0.st_open,s1_0.st_volume from stock s1_0 order by s1_0.st_volume desc limit ?
```