## Spring Data JPA – Country Management Hands-on

Hands-on 5: Implement Services for Managing Country

**Task Description:**

Implement CRUD and search operations for the Country entity using Spring Data JPA.

**Configuration in** application.properties**:**

spring.datasource.url=jdbc:mysql://localhost:3306/orm\_learnspring.datasource.username=rootspring.datasource.password=yourpasswordspring.jpa.hibernate.ddl-auto=validatespring.jpa.show-sql=true

**Country.java:**

@Entity@Table(name = "country")public class Country {

@Id

private String coCode;

private String coName;

// Getters, setters, toString()

}

**CountryRepository.java:**

public interface CountryRepository extends JpaRepository<Country, String> {

List<Country> findByCoNameContaining(String name);

}

**Output:** MySQL Workbench with country data inserted.

### Hands-on 6: Find Country by Code

**Task Description:**

Retrieve a country using its country code and handle the case when it's not found.

**CountryNotFoundException.java:**

public class CountryNotFoundException extends Exception {

public CountryNotFoundException(String message) {

super(message);

}

}

**Service Method in CountryService.java:**

@Transactionalpublic Country findCountryByCode(String countryCode) throws CountryNotFoundException {

Optional<Country> result = countryRepository.findById(countryCode);

if (!result.isPresent()) {

throw new CountryNotFoundException("Country not found");

}

return result.get();

}

**Test Method in OrmLearnApplication.java:**

private static void getAllCountriesTest() {

LOGGER.info("Start");

Country country = countryService.findCountryByCode("IN");

LOGGER.debug("Country:{}", country);

LOGGER.info("End");

}

**Output:** showing fetched country with code "IN".

### Hands-on 7: Add a New Country

**Task Description:**

Add a new country using a service method.

**Service Method in CountryService.java:**

@Transactionalpublic void addCountry(Country country) {

countryRepository.save(country);

}

**Test Code in OrmLearnApplication.java:**

Country newCountry = new Country();

newCountry.setCoCode("ZZ");

newCountry.setCoName("Testland");

countryService.addCountry(newCountry);

**Verification Code:**

Country result = countryService.findCountryByCode("ZZ");

LOGGER.debug("Added Country: {}", result);

Output: showing newly added country with code "ZZ".