**Demonstrate writing Hibernate Query Language and Native Query**

## *Example Setup*

## Entity: Country

@Entity

public class Country {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private Long id;

private String code;

private String name;

private String capital;

private Long population;

}

## Repository Interface Using @Query

import org.springframework.data.jpa.repository.Query;

import org.springframework.data.repository.query.Param;

import org.springframework.data.jpa.repository.JpaRepository;

import java.util.List;

public interface CountryRepository extends JpaRepository<Country, Long> {

@Query("SELECT c FROM Country c WHERE c.population > :pop")

List<Country> findByPopulationGreaterThanJPQL(@Param("pop") Long pop);

@Query("SELECT c FROM Country c LEFT JOIN FETCH c.states WHERE c.name = :name")

List<Country> fetchStatesForCountry(@Param("name") String name);

@Query("SELECT COUNT(c) FROM Country c WHERE c.capital LIKE :prefix%")

long countByCapitalPrefix(@Param("prefix") String prefix);

@Query(value = "SELECT \* FROM country WHERE capital = :capital", nativeQuery = true)

List<Country> findByCapitalNative(@Param("capital") String capital);

@Query(value = "SELECT \* FROM country ORDER BY population DESC LIMIT 3", nativeQuery = true)

List<Country> findTop3PopulousCountries();

}

## Sample Test Calls

**JPQL - Get countries with population > 1 million**

GET /api/countries/greater?pop=1000000

**Native Query - Countries with capital Delhi**

GET /api/countries/capital?capital=Delhi

**Top 3 by population (Native)**

GET /api/countries/top3