**Exercise 10: Employee Management System - Hibernate-Specific Features**

**Business Scenario:**

Leverage Hibernate-specific features to enhance your application's performance and capabilities.

**Instructions:**

1. **Hibernate-Specific Annotations:**
   * Use Hibernate-specific annotations to customize entity mappings.
2. **Configuring Hibernate Dialect and Properties:**
   * Configure Hibernate dialect and properties for optimal performance.
3. **Batch Processing:**
   * Implement batch processing with Hibernate for bulk operations.

 **Hibernate-Specific Annotations Example:**

@DynamicInsert

@DynamicUpdate

public class Employee {

// Entity code

}

 **Hibernate Dialect Configuration:**

spring.jpa.properties.hibernate.dialect=org.hibernate.dialect.MySQL5Dialect

spring.jpa.show-sql=true

 **Batch Processing:**

@Transactional

public void saveAllEmployees(List<Employee> employees) {

for (int i = 0; i < employees.size(); i++) {

entityManager.persist(employees.get(i));

if (i % 20 == 0) { // Flush a batch of inserts and release memory.

entityManager.flush();

entityManager.clear();

}

}

}