**Java Employee Management System**

**📄 Overview**

This project showcases two different approaches to building a basic **Employee Management System** using Java and MySQL:

1. hibernate-example/ – Implementation using **Hibernate (without Spring)**
2. springdatajpa-example/ – Implementation using **Spring Boot with Spring Data JPA**

Both modules demonstrate how to perform CRUD operations by persisting employee information in a MySQL database using **JPA annotations**.

**📚 Table of Contents**

* Project Features
* Technologies Used
* MySQL Database Setup
* Running Instructions
  + Hibernate-based Version
  + Spring Boot Version
* Sample Output
* Comparison: Hibernate vs Spring Data JPA
* References
* Author

**✅ Key Features**

* Allows insertion of employee records into a MySQL database
* Demonstrates manual Hibernate session handling vs Spring-managed repository abstraction
* Uses standardized JPA annotations (@Entity, @Id, etc.)

**🛠 Technologies Used**

* Java 17
* Hibernate 5+
* Spring Boot 3+
* Spring Data JPA
* MySQL 8
* Maven
* IDE: VS Code or Eclipse

**🗃️ MySQL Database Setup**

Before running the applications, ensure the following MySQL databases are created:

sql

CopyEdit

CREATE DATABASE hibernatedb;

CREATE DATABASE springdb;

Both databases will contain an employee table created automatically based on the entity classes using JPA/Hibernate.

**▶️ Running the Projects**

**🔸 1. Hibernate (without Spring)**

* Navigate to the hibernate-example/ directory
* Ensure your MySQL credentials are correctly set in hibernate.cfg.xml
* Compile and run using:

bash

CopyEdit

mvn clean install

java -cp target/hibernate-example-1.0-SNAPSHOT.jar com.example.EmployeeApp

**🔸 2. Spring Boot + Spring Data JPA**

* Navigate to the springdatajpa-example/ directory
* Update application.properties with your MySQL credentials
* Run using:

bash

CopyEdit

mvn spring-boot:run

**📤 Sample Output**

**Hibernate Version:**

Console Output:

pgsql

CopyEdit

Hibernate: insert into employee (department, name) values (?, ?)

Employee saved with ID: 1

**Spring Boot Version:**

Console Output:

pgsql

CopyEdit

Hibernate: insert into employee (department, name) values (?, ?)

Employee saved!

**MySQL Database Result:**

sql

CopyEdit

SELECT \* FROM employee;

| **id** | **name** | **department** |
| --- | --- | --- |
| 1 | Alice | HR |

**🔍 Comparison: Hibernate vs Spring Data JPA**

| **Feature** | **Hibernate** | **Spring Data JPA** |
| --- | --- | --- |
| Setup Complexity | Requires manual config and XML | Auto-configured using Spring Boot |
| Session Handling | Manually handled | Abstracted by Spring repositories |
| Speed of Development | Slower | Faster with annotations and boilerplate reduction |
| Learning Value | Good for understanding internals | Great for modern enterprise dev |

**🔗 References**

* [Spring Boot Documentation](https://docs.spring.io/spring-boot/docs/current/reference/html/)
* Hibernate User Guide
* JPA Specification

**👨‍💻 Author**

Prepared by: **Chirag N**  
GitHub: [chirag276](https://github.com/chirag276)