

Spring JPA

Nguyễn Anh Tuấn

KTECH
COLLEGE



Nội dung bài giảng

1 Giới thiệu JPA

2 Tính năng của JPA

3 Kiến trúc của JPA

4 Object-Relation Mapping (ORM)

5 Triển khai JPA

Giới thiệu JPA

Giới thiệu JPA

- **Spring Boot JPA:**

- Là một bản ghi chi tiết của Java để quản lý dữ liệu quan hệ trong các ứng dụng Java.
- Cho phép truy cập và lưu trữ dữ liệu giữa các object/class Java và database quan hệ.
- JPA tuân theo **Object-Relation Mapping (ORM)**: Là 1 tập hợp các interface.
- JPA cung cấp một API EntityManager runtime để xử lý các câu query và giao dịch trên các object dựa trên database.
- JPA sử dụng ngôn ngữ truy vấn hướng đối tượng độc lập nền tảng JPQL (Java Persistent Query Language).
- JPA bao gồm ba lĩnh vực:
 - The Java Persistence API
 - Object-Relational metadata
 - Bản thân API, được định nghĩa trong persistence package.
- JPA không phải là một framework. Nó định nghĩa 1 khái niệm có thể được thực thi bởi bất kỳ framework nào.

Tính năng của JPA

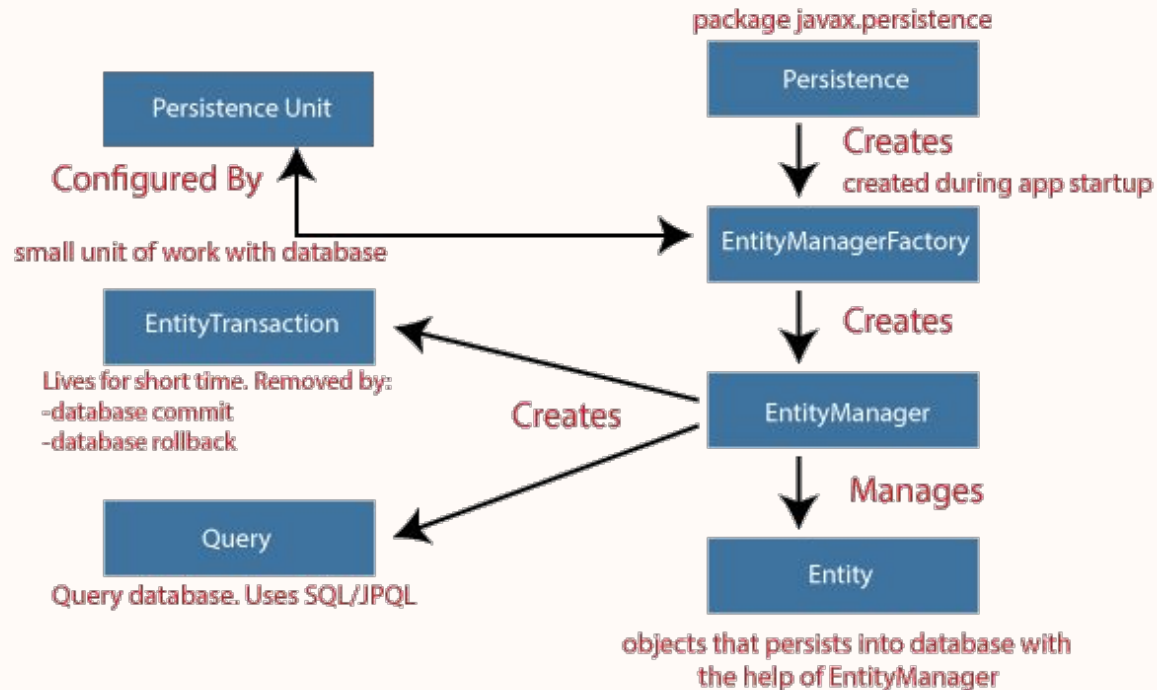
Các tính năng của JPA

- Có các tính năng sau của JPA:
 - Là một repository mạnh mẽ và object-mapping abstraction tùy chỉnh .
 - Hỗ trợ cho cross-store persistence. Tức là một entity có thể được lưu trữ một phần trong MySQL và Neo4j (Hệ thống quản lý cơ sở dữ liệu đồ thị).
 - Tự động tạo các truy vấn từ ngay chính tên các phương thức truy vấn.
 - Các lớp dựa vào tên miền cung cấp các thuộc tính cơ bản.
 - Hỗ trợ kiểm tra quản lý minh bạch.
 - Khả năng tích hợp mã repository tùy chỉnh.
 - Dễ dàng tích hợp với Spring Framework với namespace tùy chỉnh.

Kiến trúc của JPA

Kiến trúc của JPA

Architecture of Java Persistence API



Kiến trúc của JPA

JPA là một nguồn để lưu trữ các entity doanh nghiệp dưới dạng các entity quan hệ. Nó chỉ ra cách định nghĩa POJO là một entity và cách quản lý các entity quan hệ. Kiến trúc JPA chứa các đơn vị sau đây:

- **Persistence:** Là 1 class chứa các phương thức tĩnh để lấy một instance EntityManagerFactory.
- **EntityManagerFactory:** Là 1 class Factory của EntityManager. Nó tạo và quản lý nhiều instance EntityManager.
- **EntityManager:** Là 1 giao diện kiểm soát các hoạt động liên tục trên các đối tượng Entity.
- **Entity:** Các entity là các đối tượng lưu trữ lâu dài như một bản ghi trong database.
- **Persistence Unit:** Định nghĩa một tập hợp tất cả các class entity. Trong một ứng dụng, các instance EntityManager quản lý nó. Tập hợp các class entity đại diện cho dữ liệu được chứa trong một kho dữ liệu duy nhất.
- **EntityTransaction:** Có mối quan hệ 1:1 với các class EntityManager. Đối với mỗi EntityManager, các hoạt động được duy trì bởi class EntityTransaction.
- **Query:** Là 1 interface được thực hiện bởi mỗi nhà cung cấp JPA để có được các object quan hệ đáp ứng các tiêu chí.

Object-Relation Mapping (ORM)

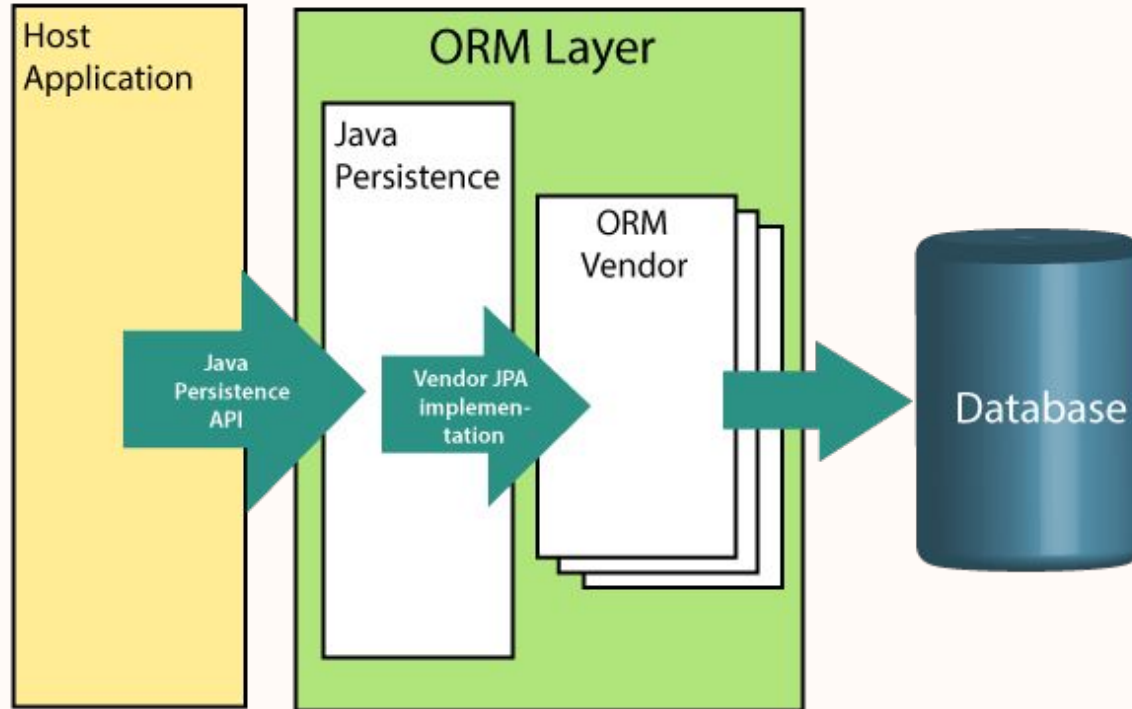
Object-Relation Mapping (ORM)

- JPA là một API mã nguồn mở.
- Có nhiều nhà cung cấp khác nhau như Eclipse, RedHat, Oracle, ... cung cấp các sản phẩm mới bằng cách thêm JPA vào trong sản phẩm.
- Có một số framework triển khai JPA phổ biến như Hibernate, EclipseLink, DataNucleus, v.v. Nó còn được gọi là công cụ Object-Relational Mapping (ORM).

Object-Relation Mapping (ORM)

- Trong ORM, ánh xạ (mapping) các đối tượng Java tới các bảng database và ngược lại được gọi là Object-Relation Mapping. Ánh xạ ORM hoạt động như một cầu nối giữa database quan hệ (bảng và bản ghi) và ứng dụng Java (các class và object).
- Trong hình sau, layer ORM là một layer chuyển đổi. Nó điều chỉnh ngôn ngữ của đồ thị object sang ngôn ngữ của SQL và các bảng quan hệ.
- Layer ORM tồn tại giữa ứng dụng và database. Nó chuyển đổi các class và object Java để chúng có thể được lưu trữ và quản lý trong database quan hệ. Theo mặc định, tên class trở thành tên của bảng và các trường trở thành cột. Khi một ứng dụng thiết lập, mỗi hàng trong bảng tương ứng với một object.

Object-Relation Mapping (ORM)



Triển khai JPA

Thêm dependency Spring Data JPA

The screenshot shows the Spring Initializr web application interface. The browser address bar displays 'start.spring.io'. The page features a sidebar with a hamburger menu icon and the 'spring initializr' logo. The main content area is divided into several sections: 'Project' (Maven Project selected), 'Language' (Java selected), 'Spring Boot' (2.5.0 selected), 'Project Metadata' (Group, Artifact, Name, Description, Package name, Packaging, Java version), and 'Dependencies'. The 'Dependencies' section on the right lists 'Spring Web' (WEB), 'Spring Batch' (I/O), and 'Spring Data JPA' (SQL). The 'Spring Data JPA' entry is highlighted with a red rectangular box. Below the dependencies list, there are buttons for 'GENERATE CTRL + G', 'EXPLORE CTRL + SPACE', and 'SHARE...'. The bottom left corner of the page includes social media icons for GitHub and Twitter.

start.spring.io

spring initializr

Project

☒ Maven Project ☐ Gradle Project

Language

☒ Java ☐ Kotlin ☐ Groovy

Spring Boot

☐ 2.5.1 (SNAPSHOT) ☒ 2.5.0 ☐ 2.4.7 (SNAPSHOT) ☐ 2.4.6 ☐ 2.3.12 (SNAPSHOT) ☐ 2.3.11

Project Metadata

Group

Artifact

Name

Description

Package name

Packaging ☒ Jar ☐ War

Java ☐ 16 ☐ 11 ☒ 8

Dependencies ADD DEPENDENCIES... CTRL + B

Spring Web WEB
Build web, including RESTful, applications using Spring MVC.
Uses Apache Tomcat as the default embedded container.

Spring Batch I/O
Batch applications with transactions, retry/skip and chunk based processing.

Spring Data JPA SQL
Persist data in SQL stores with Java Persistence API using Spring Data and Hibernate.

GENERATE CTRL + G EXPLORE CTRL + SPACE SHARE...

Demo

**Tạo 1 dự án quản lý học viên cơ bản sử dụng
Spring Web và Spring JPA để truy vấn dữ liệu
và trả về kết quả truy vấn.**

ROAD TO KOREA

Nếu có bất kỳ thắc mắc nào, hãy đặt câu hỏi qua

mail@mail.com hoặc Zalo 0xxx xxx xxx