

Java Persistence API: A Short Course - Part I

Jeferson Souza, MSc.

(a.k.a jefecomp)

jefecomp.official@gmail.com

Last Update: April 17, 2017

1 General Description

This is the first part of a short course addressing the details of the Java Persistence API (JPA), a known Java abstraction which provides a powerful set of operations for transparent persistence of object data models on both relational and non-relational databases. The goal of the *Java Persistence API: A Short Course - Part I* is to present the foundations of the API, introducing the concept of object-relational mapping and the JPA use together with relational databases.

References to additional documentation addressing the topics of this course are provided. In addition, all the source code utilised within this course will be available for download and further analysis.

2 Pre-Requisites and Duration

The mandatory pre-requisite to follow this tutorial is good knowledge of the Java language. The additional list of pre-requisites is presented as follows:

- A computer with the java development kit (JDK) version 7 or later installed;

- Whether a text editor (e.g. *Emacs*) or an Integrated Development Environment (IDE)—e.g. *Eclipse*, *NetBeans*, *IntelliJ*, ...—of your preference, for Java code reading and development;
- Whether a *Git* installation, or the use of an IDE which support access to a *Git* repository—Eclipse IDE with *EGit* plugin.

The duration of the *Java Persistence API: A Short Course - Part I* is initially planned to take **up to 4 hours**, formatted on two sessions of 2 hours each.

3 Outline

The *Java Persistence API: A Short Course - Part I* includes the following topics:

1. What is the *Java Persistence API*?
2. Advantages of *JPA* versus Pure *JDBC*;
3. Concepts of Object-Relational Mapping;
4. The Persistence Provider Abstraction;
5. Oops, I have just got here: A Hello World JPA;
6. The Configuration File and Common Parameters;
7. Essential Mapping Annotations;
8. Understanding Persistence Sessions;
9. Hands-on: Creating an object data model for a small library;
10. Remarks & Conclusion.

3.1 Summary of the content addressed by each topic

3.1.1 What is the Java Persistence API?

In this topic it is explained what the Java Persistence API is, and why it has been changed the way object-oriented models are persisted, providing an intuitive and transparent translation layer to different types of databases.

3.1.2 Advantages of JPA versus Pure JDBC

In this topic the advantages of JPA versus Pure JDBC are presented, which includes writing less code for persisting data, no need of using Structure Query Language (SQL) directly, and automatic data translation between database and the application.

3.2 Concepts of Object-Relational Mapping

In this topic we present the main concepts of object-relational mapping, focusing on the differences among object-oriented and relational data design.

3.3 The Role of the Persistence Provider

The persistence provider is the central component on all the abstraction offered by the JPA specification, and this topic explains details about its role.

3.4 Oops, I have just got here! A Hello World JPA

In this topic a Hello World example is presented, utilising a minimal configuration needed to start using JPA within Java projects.

3.5 The Configuration File and Common Parameters

The structure of the configuration file and common parameters are detailed in this topic.

3.6 Essential Mapping Annotations

This topic covers the essential annotations which are part of the JPA specification such as *@Entity*, *@Id*, *@Column*, *@OneToMany*, *@ManyToOne*, and *@ManyToMany*.

3.7 Understanding Persistence Sessions

In this topic the concept of Persistence Session is presented, highlighting the different “design patterns” existent for establishing persistence sessions within Java applications.

3.8 Hands-on: Creating an object data model for a small library

In this topic an object-oriented data model will be created for a small library, putting in practice all the topics addressed by this course.

3.9 Remarks & Conclusion

At the end we summarise all content presented in this tutorial, listing some references for further consult.