



Java Persistence API

JPA

Prof. Dr. Dieval Guizelini
2018



ORACLE

Menu



Sign In



Country/Region



Call

Oracle Technology Network / Java / Java EE / Technologies

Java SE
Java EE
Java ME
Java SE Subscription
Java Embedded
Java Card
Java TV
Community
Java Magazine

Overview

Downloads

Documentation

Community

Technologies

Training

Java Persistence API

Java Persistence API

The Java Persistence API provides a POJO persistence model for object-relational mapping. The Java Persistence API was developed by the EJB 3.0 software expert group as part of JSR 220, but its use is not limited to EJB software components. It can also be used directly by web applications and application clients, and even outside the Java EE platform, for example, in Java SE applications. See [JSR 220](#).

Downloads

Implementation of Java Persistence API Downloads

To download a fully tested and compatible build, chose the GlassFish v1 FCS branch and download the v1 ur1 b14 binary bundle.

Tutorials

Introduction to the Java Persistence API

Persistence in the Web Tier

Persistence in the EJB Tier

The Java Persistence Query Language

Articles

Java Resources

[Java APIs](#)
[Technical Articles](#)
[Demos and Videos](#)
[Forums](#)
[Java Magazine](#)
[Developer Training](#)
[Tutorials](#)
[Java.com](#)

Java SDKs and Tools

[Java SE](#)
[Java EE and Glassfish](#)
[Java ME](#)
[Java Card](#)
[NetBeans IDE](#)
[Java Mission Control](#)

Fonte: <https://www.oracle.com/technetwork/java/javaee/tech/persistence-jsp-140049.html>

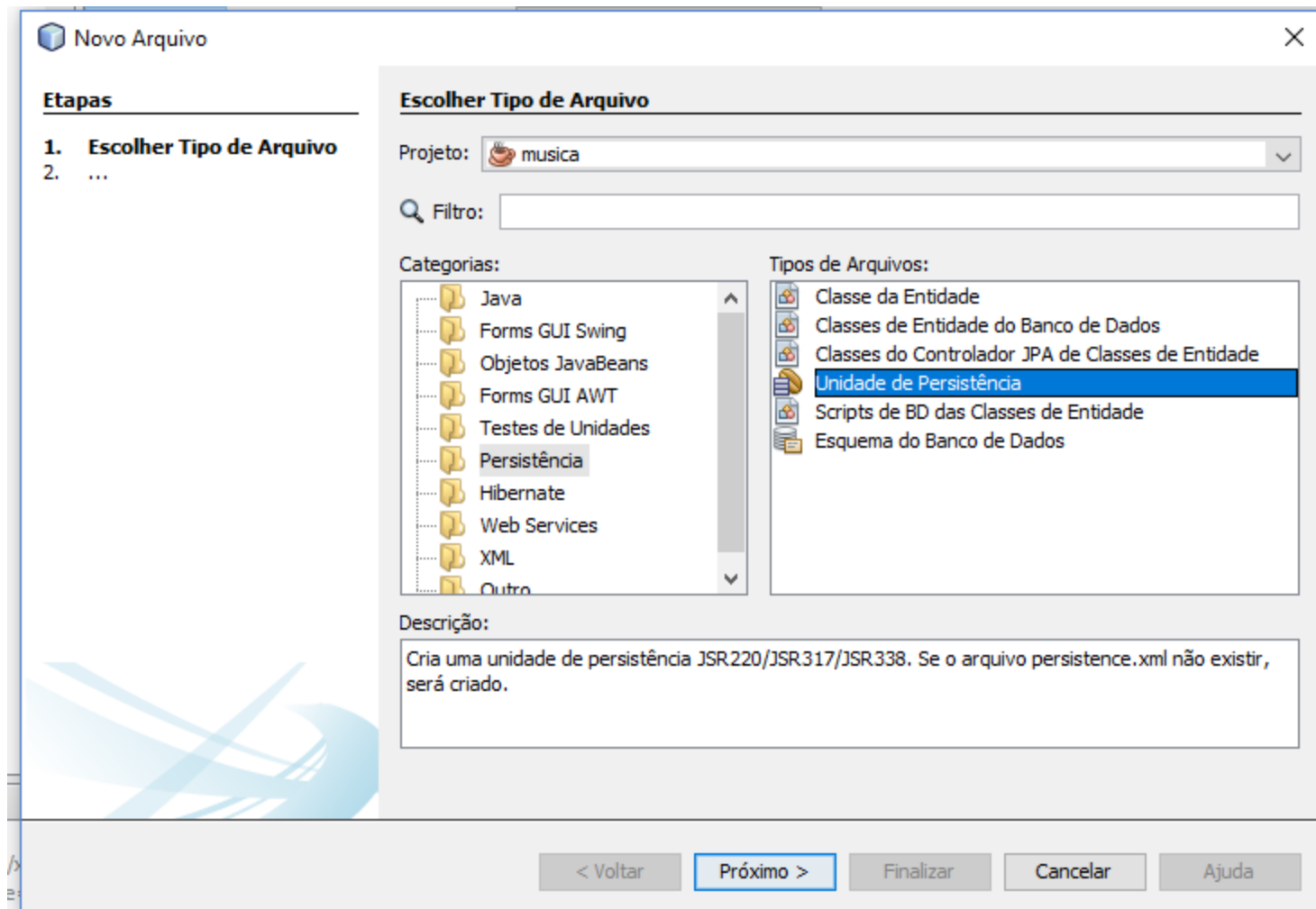


Java Persistence API e Frameworks ORM

- A camada de persistência tende a seguir um padrão e em muitos casos representa um percentual significativo do sistema;
- A camada de persistência basicamente mapeia classes e atributos com tabelas e campos mantidos por SGBDs. Da mesma forma, relaciona objetos com registros.
- Ferramentas que automatizam essa tarefa são conhecidas por **mapeamento objeto-relacional** (ORM).
- O Hibernate é uma ferramenta ORM open source e é a líder de mercado, sendo a inspiração para a especificação **Java Persistence API** (JPA).
- O Hibernate nasceu sem JPA mas hoje em dia é comum acessar o Hibernate pela especificação JPA.
- Como toda especificação, ela deve possuir implementações. Entre as implementações mais comuns, podemos citar: Hibernate da JBoss, EclipseLink da Eclipse Foundation e o OpenJPA da Apache. Apesar do Hibernate ter originado a JPA.
- O EclipseLink é a implementação referencial.



Unidade de Persistência





Unidade de Persistência

New Unidade de Persistência ✕

Etapas

- Escolher Tipo de Arquivo
- Provedor e Banco de Dados**

Provedor e Banco de Dados

Nome da Unidade de Persistência:

Especifique o provedor de persistência e o banco de dados para as classes de entidade.

Biblioteca de Persistência:

Conexão ao Banco de Dados:

Estratégia de Geração de Tabela: ☐ Criar ☐ Eliminar e Criar ☒ Nenhum

< Voltar Próximo > Finalizar Cancelar Ajuda



Unidade de Persistência

Unidades de Persistência

Adicionar

musicaPU

Remover

Geral:

Nome da Unidade de Persistência:

musicaPU

Biblioteca de Persistência:

EclipseLink (JPA 2.1)

▼

Conexão JDBC:

jdbc:h2:tcp://localhost/~ /musicadb [sa em PUBLIC]

▼

☐ Utilizar APIs de Transação Java

Estratégia de Geração de Tabela:

☐ Criar ☐ Eliminar e Criar ☒ Nenhum

Estratégia de Validação:

☒ Automático(a) ☐ Chamada de Retorno ☐ Nenhum

Modo de Cache Compartilhado:

☐ Tudo ☐ Nenhum ☐ Ativação Seletiva ☐ Desativação Seletiva ☒ Não Especificado

☐ Incluir Todas as Classes de Entidade no Módulo "musica"

Incluir Classes de Entidade:

Adicionar Classe...

Remover



Classe Genero

```
public class Genero implements Serializable {  
    private Integer id;  
    private String nome;  
  
    public Genero() {  
        id = -1;  
        nome = null;  
    }  
  
    public Genero(Integer id, String nome) {  
        this.id = id;  
        this.nome = nome;  
    }  
}
```



```
package br.ufpr.tads.musica.model;

import java.io.Serializable;
import java.util.Objects;
import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.Table;

@Entity
@Table(name="genero")
public class Genero implements Serializable {
    @Id
    @GeneratedValue(strategy = GenerationType.AUTO)
    private Integer id;
    private String nome;
    public Genero(Integer id, String nome) {
        this.id = id;
        this.nome = nome;
    }
}
```




Referências

- ORACLE.
Disponível em:
<https://docs.oracle.com/javaee/5/tutorial/doc/bnbqa.html>