

Guía de persistencia de JPA

Franco Bulgarelli, Gastón Prieto

Versión 1.2

Mayo 2021

Distribuido bajo licencia [Creative Commons Share-alike](https://creativecommons.org/licenses/by-sa/4.0/)

Anotaciones principales

Aclaración: los atributos de la anotación marcados con ? son opcionales

Anotación	Comentarios
<code>@Entity</code>	
<code>@Id</code>	
<code>@EmbeddedId</code>	Sólo tiene utilidad para mapear PK's compuestas
<code>@MappedSuperclass</code>	Sirve para que utilice las anotaciones de esa clase, pero no es una entidad persistente
<code>@GeneratedValue</code>	
<code>@OneToOne(fetch?, mappedBy?, optional?)</code>	Especifica la cardinalidad de la relación. FetchType puede ser LAZY o EAGER
<code>@OneToMany(fetch?, mappedBy?)</code>	
<code>@ManyToOne(fetch?, mappedBy?)</code>	
<code>@ManyToMany(fetch?, mappedBy?)</code>	
<code>@OrderColumn(name?)</code>	Nombre de la columna donde va a guardar los índices de los elementos.
<code>@Inheritance(strategy)</code>	InheritanceType puede ser JOINED, SINGLE_TABLE, TABLE_PER_CLASS
<code>@Embedded</code>	Va sobre el atributo de la clase que quiero embeber
<code>@Embedable</code>	Va sobre la clase que puede ser embebida
<code>@DiscriminatorValue(value)</code>	Especifica el valor discriminador para la entidad anotada, si no se quiere utilizar el generado
<code>@DiscriminatorColumn(name)</code>	Si no se especifica nombre para la columna discriminador, es DTYPE

<i>@Table(name)</i>	Especifica el nombre de la tabla si no es compatible con la convención
<i>@Column(name?, nullable?)</i>	Especifica el nombre de la columna si no es compatible con la convención. También sirve para generar constraints not-null en la base
<i>@PrePersist/@PreUpdate/@PostLoad</i>	Permite anotar métodos para que se comporten como callbacks de JPA
<i>@Enumerated</i>	
<i>@Transient</i>	Especifica que no queremos que el atributo anotado se persista
<i>@JoinColumn(fk)</i>	Especifica la FK que se utilizará para realizar el join entre dos entidades. Si se aplica sobre una relación <i>@OneToMany</i> , evita la generación de una tabla intermedia
<i>@Convert</i>	Permite establecer una conversión manual de tipo de dato entre un objeto y una columna de la base. Útil si se quiere serializar colecciones
<i>@ElementCollection</i>	Permite persistir una lista de valores

Interfaz del EntityManager

<code>find(entityClass, entityPK)</code>	Busca una entidad en el contexto de persistencia
<code>remove(entity)</code>	Elimina una entidad del contexto de persistencia
<code>persist(entity)</code>	Agrega una entidad al contexto de persistencia
<code>createQuery(string)</code>	Crea una consulta en JQL