

En primer lugar tenemos la versión del documento y si queremos que el documento sea único. A continuación, tenemos la etiqueta “hibernate-mapping”, esta etiqueta nos permite anidar varias etiquetas de tipo “class”. Dentro de la etiqueta “hibernate-mapping” tenemos diferentes parámetros o atributos que son:

- auto-import: especifica si podemos utilizar ese nombre de clases no calificados como clases en este mapeo de consultas.
- default-access: es una estrategia de Hibernate para acceder a todas las propiedades.
- default-cascade: es un estilo de cascada por defecto.
- default-lazy: es el valor por defecto para los atributos “lazy” no especificados de mapeo de clase y de colección.

tenemos la etiqueta “class”, que también tiene una serie de atributos o parámetros que son los siguientes:

- catalog: sobrescribe el nombre del catálogo especificado por el elemento raíz “hibernate-mapping”. Nombre de la base de datos.
- dynamic-insert: especifica que el SQL INSERT debe ser generado en tiempo de ejecución y puede contener solamente aquellas columnas cuyos valores no son nulos.
- dynamic-update: especifica que el SQL UPDATE debe ser generado en tiempo de ejecución y puede contener solamente aquellas columnas cuyos valores hayan cambiado.
- mutable: especifica que las instancias de la clase (no) son mutables.
- name: es el nombre completamente calificado de la clase
- optimistic-lock: determina la estrategia optimista de bloqueo.
- polymorphism: determina si se utiliza polimorfismo de consulta implícito o explícito.
- select-before-update: especifica que el hibernante nunca debe realizar un UPDATE SQL a menos de que se tenga certeza de que realmente se haya modificado un objeto.
- table: es el nombre de la tabla en la base de datos

En la etiqueta “id”, encontramos los parámetros:

- name: es el nombre de la propiedad del identificador.
- type: es un nombre que indica el tipo de hibernate.

Esta etiqueta consta de otras dos etiquetas más dentro de ella, que son: “column”, (nombre de la columna) y “generator” (nombra a una clase Java para generar identificadores únicos para instancias de la clase persistente).

En la etiqueta “property”:

podemos encontrar los atributos:

- generated: especifica que este valor de la propiedad es de hecho generado por la base de datos.
- lazy: especifica que se debe recuperar perezosamente esta propiedad cuando se acceda por primera vez a la variable de instancia.
- name: es el nombre de la propiedad.
- optimistic-lock: especifica que las actualizaciones a esta propiedad requieren o no

de la obtención de un bloqueo optimista.

- type: (opcional) es el nombre que indica el tipo de hibernate.
 - unique: activa la generación DDL de una restricción de unicidad para las columnas.
- Y también consta de la etiqueta "column" dentro de la de "property".

la etiqueta "set":

embed-xml:

- fetch: nos especifica que es lo que hay que buscar.
- inverse: nos devuelve la opción inversa, esta opción puede devolver varios valores.
- lazy: especifica que se debe recuperar perezosamente esta propiedad cuando se acceda por primera vez a la variable de instancia.
- mutable: especifica que las instancias de la clase (no) son mutables.
- name: nombre del "set" o sección.
- optimistic-lock: especifica que las actualizaciones a esta propiedad requieren o no de la obtención de un bloqueo optimista.
- sort: nos ordena los diferentes datos que podemos encontrar en la tabla creada en la base de datos.
- table: nombre de la tabla en la base de datos.

Y dentro de esta etiqueta tenemos otras dos llamadas "key" y "one-to-many".