¿Qué es Java Hibernate?

**Hibernate** es una herramienta de [Mapeo objeto-relacional](http://es.wikipedia.org/wiki/Mapeo_objeto-relacional) (ORM) para la [plataforma Java](http://es.wikipedia.org/wiki/Plataforma_Java) (y disponible también para [.Net](http://es.wikipedia.org/wiki/.Net) con el nombre de [NHibernate](http://es.wikipedia.org/wiki/NHibernate)) que facilita el mapeo de atributos entre una [base de datos](http://es.wikipedia.org/wiki/Base_de_datos) relacional tradicional y el modelo de [objetos](http://es.wikipedia.org/wiki/Objetos_(programaci%C3%B3n_orientada_a_objetos)) de una aplicación, mediante archivos declarativos ([XML](http://es.wikipedia.org/wiki/XML)) o anotaciones en los [beans](http://es.wikipedia.org/wiki/JavaBean) de las entidades que permiten establecer estas relaciones.

En otras palabras, Hibernate es un Framework que agiliza la relación entre la aplicación y la base de datos. Para poder aprender a utilizarlo es necesario contar con los conocimientos básicos de base de datos y SQL así como manejar el lenguaje Java.

[](http://www.programaenlinea.net/wp-content/uploads/2015/06/HIBER2.png)

¿POR QUE USAR UN FRAMEWORK ORM?

Cuando desarrollamos aplicaciones en muchos casos todo termina siendo un conjunto de ABM (alta, baja y modificaciones de datos) que luego consultamos. Para ello se utiliza una base de datos donde hay muchas tareas repetidas: por cada objeto que quiero persistir debo crear una clase que me permita insertarlo, eliminarlo, modificarlo y consultarlo. Con excepción de consultas especiales, el resto es siempre lo mismo.  Este es el momento dónde un ORM tiene una importancia fundamental. Con solo configurarlo todas estas tareas se ejecutan automáticamente y solo tendrás que preocuparte por las consultas especiales.

[](http://www.programaenlinea.net/wp-content/uploads/2015/06/HIBER.png)

¿COMO FUNCIONA?

El desarrollador deberá configurar en un archivo XML o mediante annotations donde corresponde un atributo de una clase, con una columna de una tabla. Es una tarea simple donde existen herramientas que lo hacen por nosotros.

CARACTERÍSTICAS

Simplicidad y flexibilidad: necesita un único fichero de configuración en tiempo de ejecución y un documento de mapeo para cada aplicación. Este fichero puede ser el estándar de Java (extensión properties) o un fichero XML. También se tiene la alternativa de realizar la configuración de forma programática. El uso de frameworks de persistencia, tales como EJBs hace que la aplicación dependa del framework. Hibernate no crea esa dependencia adicional. Los objetos persistentes en la aplicación no tienen que heredar de una clase de Hibernate u obedecer a una semántica específica. Tampoco necesita un contenedor para funcionar.

[](http://www.programaenlinea.net/wp-content/uploads/2015/06/HIBER3.jpg)

Completo: ofrece todas las características de orientación a objetos, incluyendo la herencia, tipos de usuario y las colecciones. Además, también proporciona una capa de abstracción SQL llamada HQL. Las sentencias HQL son compiladas por el framework de Hibernate y cacheadas para su posible reutilización.

Prestaciones: una de las grandes confusiones que aparecen al utilizar este tipo de frameworks es creer que las prestaciones de la aplicación se ven muy mermadas. Este no es el caso de Hibernate. La clave en este tipo de situaciones es si se realizan el número mínimo de consultas a la base de datos. Muchos frameworks de persistencia actualizan los datos de los objetos incluso cuando no ha cambiado su estado. Hibernate sólo lo hace si el estado de los objetos ha cambiado. El cacheado de objetos juega un papel importante en la mejora de las prestaciones de la aplicación. Hibernate acepta distintos productos de cacheado, tanto de código libre como comercial.