

Bootcamp Java Developer

Fase 3 - Java Architect
Módulo 29



Hibernate

Hibernate

Es una herramienta de mapeo objeto-relacional (ORM) para la plataforma Java. Facilita el mapeo de **atributos** entre una **base de datos relacional tradicional** y el **modelo de objetos de una aplicación** mediante archivos declarativos (XML) o anotaciones en los *beans* de las entidades que permiten establecer estas relaciones.



¿Por qué usar un framework ORM?

Cuando se desarrolla una aplicación, en la gran mayoría de los casos, todo termina siendo un conjunto de ABMs (alta, baja, modificación) para luego poder consultar datos.

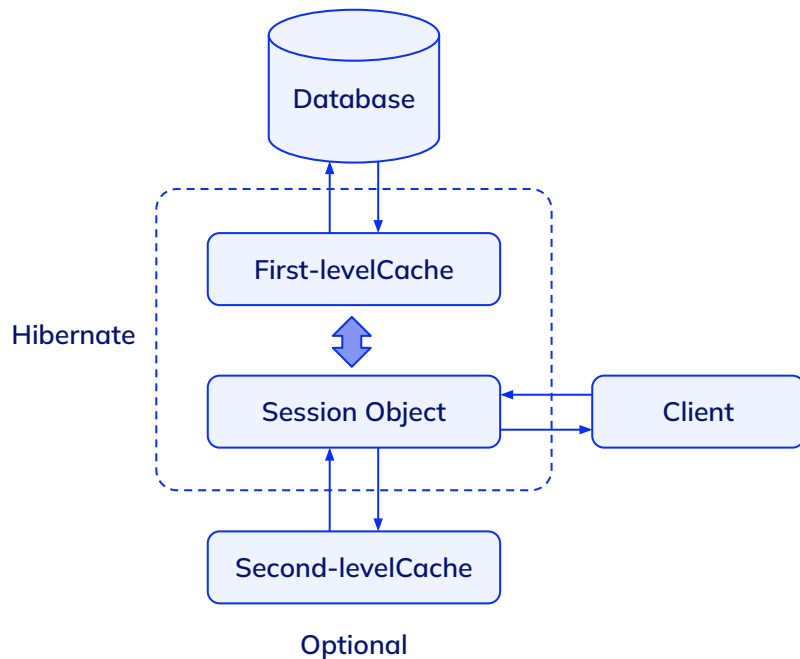
Para ello **se utiliza una base de datos, donde existirán muchas tareas repetidas**: por cada objeto que quiero persistir debo crear una clase que me permita insertarlo, eliminarlo, modificarlo y consultarlo. Salvo aquellas consultas fuera de lo común, el resto es siempre lo mismo.

Aquí es donde entra a jugar el rol importante de **un ORM: con solo configurarlo ya tiene todas esas tareas repetitivas realizadas** y el desarrollador solo tendrá que preocuparse por aquellas consultas fuera de lo normal.



¿Cómo funciona Hibernate?

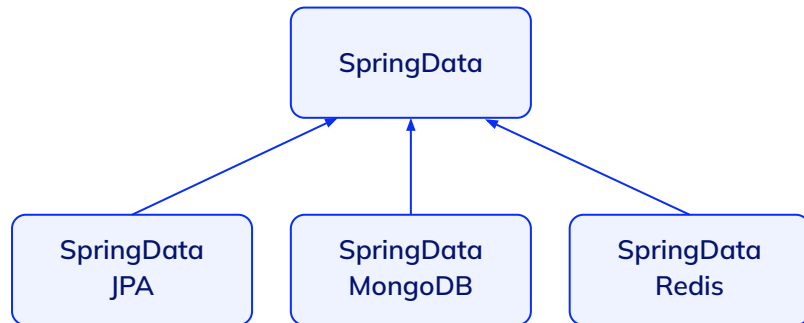
Básicamente, el desarrollador deberá configurar, en un archivo XML o mediante *annotations*, donde corresponde un atributo de una clase, con una columna de una tabla. Es una tarea relativamente sencilla donde existen herramientas que lo hacen por nosotros.



Spring Data

Spring Data

Es uno de los *frameworks* que se encuentra dentro de la plataforma de Spring. **Su objetivo es simplificar la persistencia de datos contra distintos repositorios de información.**



El acceso a bases de datos es una de las tareas más comunes en el desarrollo de *software*, al principio esta tarea se realizaba simplemente haciendo uso de JDBC y, poco a poco, fue evolucionando con patrones de diseño como el DAO (*Data Access Object*) y a través de frameworks y API's tales como Hibernate y JPA. Estos ORM han ayudado a reducir muchas de las tareas que se hacían antes con JDBC.

Spring Data es un módulo de Spring que vamos a utilizar sobre JPA para **hacer las tareas de acceso a base de datos aún más sencillas**. Las ventajas se notarán a simple vista ya que nosotros simplemente crearemos métodos en una interfaz y Spring Data se encargará de hacer las implementaciones por nosotros.

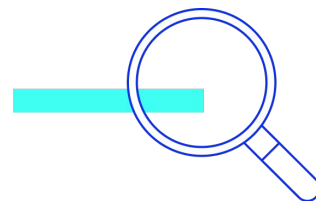
De ese modo, si nombramos un método como "findByName" Spring Data creará la implementación necesaria para buscar en la base de datos a través del nombre, sin que nosotros creamos ni una sola conexión ni procesemos ningún resultado.



Bibliografía

Para realizar los apuntes de este módulo se utilizaron varias fuentes, entre ellas:

- [Antonia Mª Reina Quintero | Universidad de Sevilla](#)
- [Ecu Red](#)
- [Ferestrepoca.github.io](#)
- [ltech.ua.es](#)
- [Coggle.it](#)
- [Academia.edu](#)
- [Ippardo´s Blog](#)



**¡Sigamos
trabajando!**