Hibernate FrameWork

**What is Hibernate?**

Hibernate is an open source object relational mapping (ORM) tool that provides a framework to map object-oriented domain models to relational databases for web applications.

Object relational mapping is based on the containerization of objects and the abstraction that provides that capacity. Abstraction makes it possible to address, access and manipulate objects without having to consider how they are related to their data sources.

The Hibernate ORM framework guides mapping Java classes to database tables and Java data types to SQL data types and provides querying and retrieval.

**Benefits of Hibernate**

Any changes made are encapsulated in the data source itself, so that when those sources or their application programming interfaces (APIs) change, the applications that use ORM don't have to make changes or even be aware of that information. Similarly, programmers can have a consistent view of objects over time, although the sources that deliver them, the sinks that receive them and the applications that access them may change.

Hibernate is freely available to download and is licensed under the open source GNU Lesser General Public License (LGPL).

**How does Hibernate work?**

Hibernate is an open source Object-Relational Persistence and Query service for any Java Application. Hibernate maps Java classes to database tables and from Java data types to SQL data types and relieves the developer from most common data persistence related programming tasks.

Hibernate sits between traditional Java objects and database server to handle all the works in persisting those objects based on the appropriate O/R mechanisms and patterns.

**Why use Hibernate?**

Hibernate reduces lines of code by maintaining object-table mapping itself and returns result to application in form of Java objects. It relieves programmer from manual handling of persistent data, hence reducing the development time and maintenance cost.

**Hibernate and JPA vs. JDBC**

Java Database Connectivity (JDBC) is an API packaged with the Java SE edition that makes it possible to standardize and simplify the process of connecting Java applications to external, relational database management systems (RDBMS).

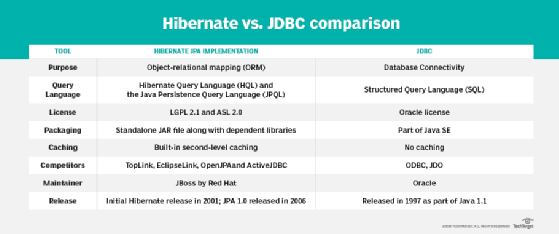
**Hibernate vs JDBC**

Hibernate and JDBC are two complimentary Java technologies for database integration.

Fundamentally, applications written in Java perform logic. The Java language provides facilities for performing iterative logic with looks, conditional logic with if statements and object-oriented analysis through the use of classes and interfaces. But Java applications do not store data persistently.

Data persistence is typically delegated to NoSQL databases such as MongoDB and Cassandra, or to relational databases such as IBM's DB2 or Microsoft's SQL Server or the popular open source database MySQL.

To help address the object-relational impedance mismatch, a number of frameworks exist that simplify the task of moving data between a relational database and a Java program. Popular object-relational mapping (ORM) frameworks include Hibernate, TopLink and DataNucleus. While each framework has its own set of unique capabilities, all of them comply with the Java Persistence API standard, which is now part of the Java EE/Jakarta EE specification.



**History of Hibernate**

Hibernate was started in 2001 by Gavin King with colleagues from Cirrus Technologies as an alternative to using EJB2-style entity beans. The original goal was to offer better persistence capabilities than those offered by EJB2.

In early 2003, the Hibernate development team began Hibernate2 releases. In 2005, Hibernate version 3.0 was released. In December 2018, Hibernate ORM 5.4.0 Final was released.