Q1.What is ORM in Hibernate?

Ans : ORM stands for Object-Relational Mapping (ORM) is a programming technique for converting data between relational databases and object oriented programming languages such as Java.

Q2.What are the advantages of Hibernate over JDBC?

Ans : - The advantages of Hibernate over JDBC are:  
  
1. Hibernate code will work well for all databases, for ex: Oracle, MySQL, etc. where as JDBC is database specific.  
  
2. No knowledge of SQL is needed because Hibernate is a set of objects and a table is treated as an object, where as to work with JDBC, one need to know SQL.  
  
3. Query tuning is not required in Hibernate. The query tuning is automatic in hibernate by using criteria queries, and the result of performance is at its best. Where as in JDBC the query tuning is to be done by the database authors.  
  
4. With the support of cache of hibernate, the data can be placed in the cache for better performance. Where as in JDBC the java cache is to be implemented.

Q3.What are some of the important interfaces of Hibernate framework?

Ans : The core interfaces of Hibernate framework are:

* Configuration.
* SessionFactory.
* Session.
* Query.
* Criteria.
* Transaction.

Q4.What is a Session in Hibernate?

Ans : The [*Session*](https://docs.jboss.org/hibernate/orm/3.5/javadocs/org/hibernate/Session.html) interface is the main tool used to communicate with Hibernate. It provides an API enabling us to create, read, update, and delete persistent objects. The *session* has a simple lifecycle. We open it, perform some operations, and then close it.

When we operate on the objects during the *session*, they get attached to that *session*. The changes we make are detected and saved upon closing. After closing, Hibernate breaks the connections between the objects and the session.

Q5.What is a SessionFactory?

Ans : SessionFactory:

* SessionFactory is an Interface which is present in org.hibernate package and it is used to create Session Object.
* It is immutable and thread-safe in nature.

Q6.What is HQL?

Ans : Hibernate Query Language (HQL) is an object-oriented query language, similar to SQL, but instead of operating on tables and columns, HQL works with persistent objects and their properties. HQL queries are translated by Hibernate into conventional SQL queries, which in turns perform action on database.

Q7.What are Many to Many associations?

Ans : Many-to-Many mapping is usually implemented in database using a Join Table. For example we can have Cart and Item table and Cart\_Items table for many-to-many mapping. Every cart can have multiple items and every item can be part of multiple carts, so we have a many to many mapping here.

Q8.What is hibernate caching?

Ans : Hibernate caching improves the performance of the application by pooling the object in the cache. It is useful when we have to fetch the same data multiple times. There are mainly two types of caching: First Level Cache, and. Second Level Cache.

Q9.What is the difference between first level cache and second level cache?

Ans :

| Sr. No. | Key | First level cache | Second level cache |
| --- | --- | --- | --- |
| 1 | Basic | First level cache is a session level cache and it is always associated with session level object | Second level cache is session factory level cache and it is available across all sessions |
| 2 | Enabled | It is enabled by default. | It is not enabled by default. |
| 3 | Availability | It is available for a session | It is available across all session. |
| 4 | Configuration | No Extra configuration required | We have to decide  which concurrency strategy to use and also need to configure cache expiration and physical cache attributes. |

Q10.What can you tell about Hibernate Configuration File?

Ans : Hibernate Configuration File(cfg file) is the file loaded into an hibernate application when working with hibernate.Hibernate uses this file to establish connection to the database server.It is an XML file which is used to define below information.Standard name for this file is hibernate.cfg.xml.

* DataBase connection details: Driver class, URL, username and Password.
* Hibernate properties: Dialect, show\_sql, second\_level\_cache .etc
* Mapping files names.

There must be one configuration file for each database used in the application, suppose if we want to connect with 2 databases, like Oracle, MySql, then we must create 2 configuration files with different names, like oracle.cfg.xml fr Oracle DB and mysql.cfg.xml for mysql database.