Chapter 14: Java Hibernate

- Hibernate was introduced by Gavin King in 2001.
- Hibernate is an open source, lightweight framework for ORM (Object Relational Mapping).
- ORM means it is a programming technique that maps the object to the data stored in the database.

Hibernate Benefits:

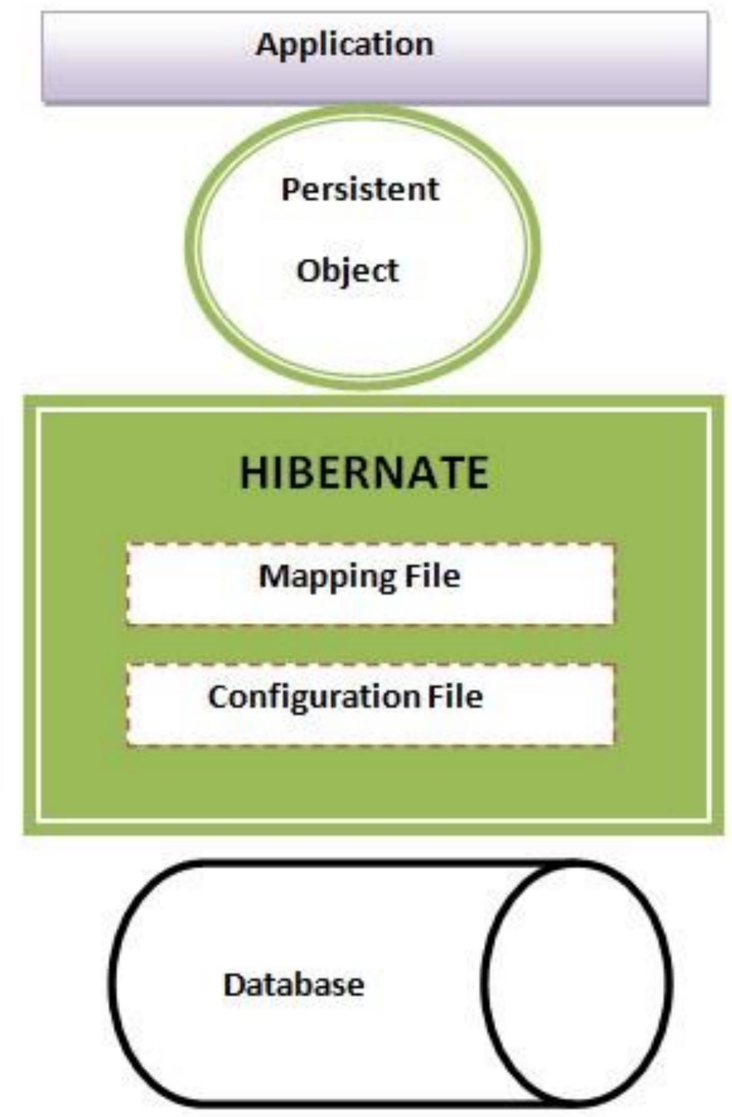
- **Object Mapping:**

In JDBC, you need to write code to map the object model's data representation to a relational model and its corresponding schema. Hibernate itself maps Java classes to database tables using XML

- **Database Independent:**

Hibernate's code is database independent because you do not need to change the HQL queries when you change databases like MySQL, Oracle, etc. Hence, it is easy to migrate to a new database.

Hibernate Architecture:



Steps to use Hibernate Framework:

Step1: Create Persistent Class

Eg: Student.java

Step2: Create a mapping file for Persistent Class

Eg: student.hbm.xml

Mapping file contains mapping from a pojo class name to a table name and pojo class variable names to table column names.

Step3: Create Configuration file.

Eg: hibernate.cfg.xml

Configuration file contains: hibernate properties, connection properties and mapping file names.

Step4: Create class that performs CRUD operations.

Eg: HibOperations.java

- **Configuration** loads the details of mapping files and db details .
- **SessionFactory** is the factory for the session objects. We use Configuration object to build the SessionFactory.
- **Session** acts as a main bridge between Java application and Hibernate.
- **sessionFactory.openSession()** is used to obtain the session from SessionFactory
- **session.beginTransaction()** is used to begin the transaction
- **tx.commit()** is used to commit the transaction
- **tx.rollback()** is used to roll back the transaction if any exception is thrown
- **session.close()** is used to close the session.

Step5: Compile and Run the application with the required jar Files.

Eg: use –cp for javac and java. Classpath would refer to directory containing all the jar files needed.

HQL:

- HQL or Hibernate Query Language is the object-oriented query language of Hibernate Framework.
- HQL is very similar to SQL except that we use Objects instead of table names, that makes it more close to object oriented programming.

- **Order by Id ascending order:**

"from Employee order by id"

- **Order by id descending order:**

"from Employee order by id desc"

- **Find id:**

"from Employee where id = " + id

- **Find name:**

"from Employee where name like '%" + name + "%'"

- **Count employee:**

```
Query q = session.createQuery("select count(*) from Employee");
long count = (Long)(q.uniqueResult());
```

- **Max salary:**

```
emp = session.createQuery("select max(salary) from Employee").list();
System.out.println(emp.get(0));
```

- **Sum of salary:**

```
emp = session.createQuery("select sum(salary) from Employee").list();
System.out.println(emp.get(0));
```

- **Avg salary:**

```
emp = session.createQuery("select avg(salary) from Employee").list();
System.out.println(emp.get(0));
```