Hibernat`e Interview Questions

1) What is ORM?
2) What are the main features of ORM?
3) why Object Relational Mapping?
4) what is hibernate?

   A) Hibernate is ORM tool. ORM stands for Object relational mapping. Object is one instance of Class. Relation is instance of Table. Both object and Relation are ment for Holding data. To convert one format to another format we required mapping. Mapping solves three problems 1. Mismatch of data types. 2. Mismatch of Reation ships. 3. Storage types (classes and tables).

   Any ORM frame work solves these three Mappings issesues. Mapping can be applied

   In how many ways 1. Hbm files(XML format) 2. Annotations.

   We are using JPA Annotation based mapping. This has flexibility in migrating from one framework to another framework.

What is the Importance Of Wrapper And Primitive Types In Hibernate?

5) Why you are using ORM
6) Benefits of ORM

   1. Reduces the development code.

      No need to register driver, no need to get connection,

      statement, resultset. No need to process resultset, no need to close resources. No need to implement exception handling.

      By using hibernate we can have predefined methods like get, save,update we can reduce our work. It also provides HQL , joins to reduce the code with portability feature.

2. Maintainability

   Less code improves maintainability.

   [fixing bugs, enhancements ,changes.]

3. Improves performance.

   a. Caching.

Caching is temp data that reduces network calls. There are three types of cachings

1. First level caching.

   First level caching is associated with session. It resides b/w your program and Database. It caches all session operations with out interacting database directly. Once Tx has committed then all operations are performed in data base by using batch update by creating query for each operation.

   It improves performance by using batch update that follows few network calls.

   It reduces the number of query generation. Ex: Multiple update methods on same object have single update query.

   Diff b/w flush and commit.

   Flush saves the data into database but not commit the data. Tx.commit method save the data and commits the data into database.

   What is clear, evict mehod. How can you clear the data from fistlevel caching(session.)

   Clear and evict methods are ment for clearing the data from firstlevel(session). Clear method clear all objects.evict method remove specified object only. Session level caching is not configurable. Short lived cahing.

2. Second level caching.

   Second level caching is associated with SessionFactory. It uses secondary memory(hard disk).It long lived caching. It is configurable.

3. Query level caching.

b. Lazy loading.

Lazy loading implemented using proxy Designpattens.

c. Batch update.

s

4. Portability.

7) Hibernate entity Object States.(Life cycle states)

8) What are the dff b/w transient state and persistence state ?

1. No record in the database is available for transient object.
2. In normal cases Transient state don't have ID.

   In case of persistence object there is record and id.

9) What are the dff b/w detached state and persistence state ?

1. Detached object may not in sync with database record.

   In case of persistence object sink is guaranteed

10) What are the different methods available in session.

11) Diff b/w get and load

Load is lazy and raises Object not found exception when record not found in Database.

Get is eager and return null object once it finds no specified record in Database.

12) What is lazy loading ? What is Proxy design pattern how it works in hibernate?

Lazy loading is one of the fetching strategies that load data only when object is using.

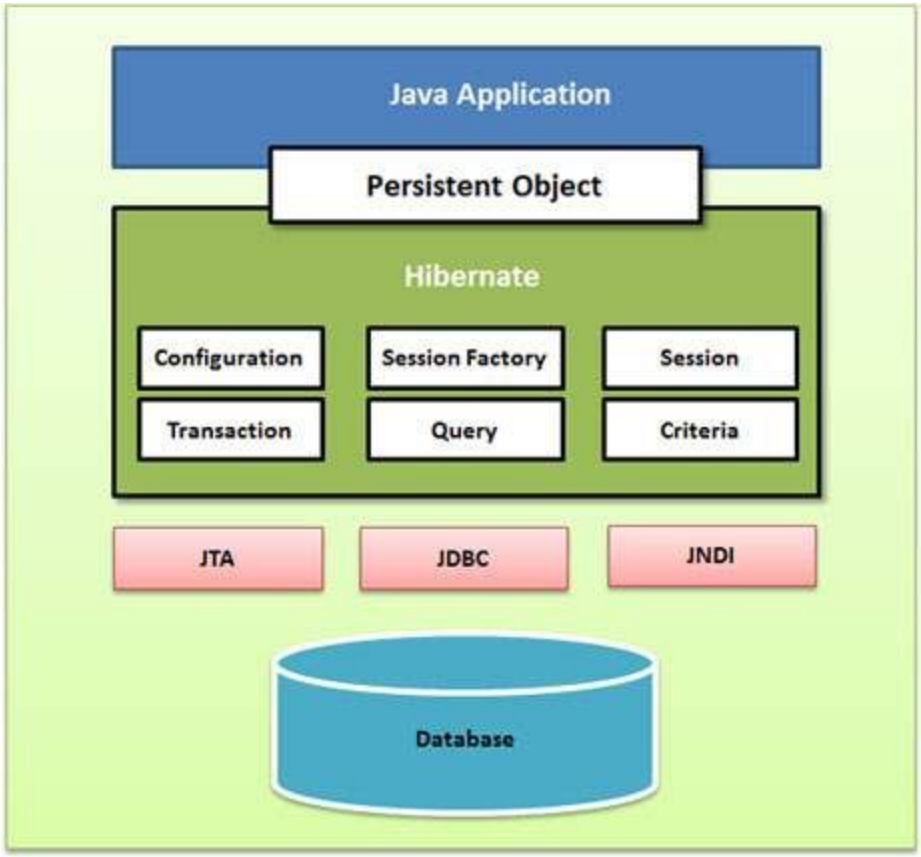
Lazy loading is implemented in Hibernate by using Proxy Design pattern. As per proxy design patterns Hibernate creates subclasses for entity classes and overrides super class methods like getter methods. These methods contain logic to go the database and fetch the information.

When we call load method it gives proxy object (subclass of entity), instead of original entity object. When we call getter method that calls proxy object method that goes to database and fetches the information only once per object. If we call another method on same object that uses same data rather than going to database again. of persia

13) .Difference b/w Session Factory and Session

1. Session Factory is factory for Sessions. Session is factory for Transactions, Queries, Criterias.
2. Session Factory is Heavy wait , Session is light wait.
3. Session Factory is thread safe. Session is not thread safe.
4. Session Factory maintains Connection pooling (Datasource). Session maintains connection.
5. Session Factory maintains Second level caching. Session maintains first level caching.
6. Session Factory is One per Database, session is one per request(thread).

14) Can you explain the Hibernate Architecture in detail?



15) Explain about inheritance relationship in hibernate?

16) Explain about Hibernate mappings ?

Two types ..

we are using annotaions … like @entity ……..

17) What is meant by component mapping?

By using Embeded and Embedable we can implement componet mapping.

18) What are the Exceptions you faced while using hibernate in your project?

A. MapppingException,

B. NonUnique

C. Object not found exception .

D. Lazy Initialization exception

E. SQLGrammarException

F.IdentifierGenerationException

G. Session Exception

19) How can we connect more than one database in hibernate applications?

we have to provide hibernate-config file for each database and create SessionFactory for each database. Then use specific session factory for storing data in specific database.

20) Difference between Dialect and Driver?

Deialect is for generating database specific sql queries.

Driver is for connecting specific database.

21) Is it possible to write hibernate application without using hibernate mapping file?

Ans:: using annotations.

22) Is it possible to write hibernate application without using hibernate configuration file ?

yes By creating SessionFactory object programatically.

23) Explain about hibernate major components?

Configuration, SessionFactory, Session, Transaction, Query , Criteria.

24) What is disadvantage using hibernate?

1. Hibernate is very usefull only when we are developing read-write centric applications.
2. Comparatively hibernate is slower than jdbc when we implemnet best pactises of jdbc , batch update, caching and lazy loading.

25) How to define a primary key for a table in hibernate?

@ID

26) I want to change my database from Oracle to MySQL in hibernate. Which configuration changes are required and which classes we have to use?

1. Changes happens only in hibernate configuration file to provide new database details.

27) What are the files we have requires doing a hibernate application?

1. jars 2. configuration file 3. pojo 4. table 5. mappings (Annotations) 6. hibernate code.

28) What is the difference between and merge and update?

Merge compares previous data and current data, if difference found then it generates update query. To compare data , merge method generate select queries.

Update method simply generates update query even then no modifications are present.

Update method causes NonuniqueObjectException if same entity object is available in the same session.

Merge method merges both objects if same entity object is available in the same session without casing NonuniqueObjectException.

29) What is nonuniqueObjectException in hibernate?

When two entity objects referring same record in database , with in the same session then

We face nonuniqueObjectException.

30) What is the use of cascade in hibernate?

Cascade is one mechanism that propagates one operation from one table to related tables.

Ex: if we delete one record in student then delete operation works in qualification table to delete related record.

We can apply operations for cascade in different ways.

Ex: cascade=CascadeType.ALL

@OneToOne(cascade={CascadeType.REMOVE,CascadeType.PERSIST})

cascade=CascadeType. MERGE || PERSIST || REFRESH || REMOVE

31) which version of hibernate is used in your project? 3.5

32) What is the main advantage of using the hibernate Query Language than using the SQL?

1. Portablility 2. Easy to use.

33) How to create composit primary key using hibernate?

using @embeded @Embedable , Embadeble pojo should be serializable. @Id for Embeded property.

34) How to store Images using Hibernate?

```
1. CREATE TABLE `slokam`.`avatar` ( `AVATAR_ID` INT(10) UNSIGNED NOT NULL AUTO_INCREMENT,
`IMAGE` BLOB NOT NULL,
PRIMARY KEY (`AVATAR_ID`) USING BTREE) ENGINE=In noDB AUTO_INCREMENT=1 DEFAULT CHARSET=utf8;
```

```
2. pojo: private byte[] image;


3. File file = new File("C:\\mavan-hibernate-image-mysql.gif");
    byte[] bFile = new byte[(int) file.length()];


    try {
            FileInputStream fileInputStream = new FileInputStream(file);
            //convert file into array of bytes
            fileInputStream.read(bFile);
            fileInputStream.close();
    } catch (Exception e) {
            e.printStackTrace();
    }
4.session.save(pojo);
```

35) What is versioning in hibernate and what is the advantage?

1. Versioning in hibernate can be implemented in two ways

   1.versionId

   2. timestamp.

   VersionId is better than timestamp bcz for all updations comparison requires among versioning. So if it is time stamp it takes much time to process.

36) Can you explain generator classes in hibernate?

37) How to pass Runtime values to the database using HQL?

Query qry = new Query ('From Product where name=:name')

Qry.setString("name",value);

38) What are the differences between HQL and Criteria?

1. Hql is faster than Criteria in execution.
2. When we are working with critical searching operations then criteria is Very developer friendly .

39) What are the differences between the named queries and Native SQL?

1. Native sql is database specific. i.e if we change the database , we have to change sql also. HQL query is generic for all databases.
2. Hql first converts to native sql specific to specified database in configuration (dialect). But native sql directly executes in db.

40) How to implement pagination in hibernate?

```
qry.setFirstResult(30);
qry.setMaxResults(10);
```

Query.list()

41) Can you explain how to write joins using Hibernate?

42) What are the jars required to configure hibernate Annotations?

1. hibernate-commons-annotations-3.0.0.ga.jar
2. hibernate-entitymanager-3.3.2.GA.jar
3. persistence-api-1.0.jar
4. other supporting jar files

43) How to integrate hibernate with spring?

44) What are the advantages of Hibernate Template?

45) What differences are between save and persist methods?

1) First difference between save and persist is there return type. Similar to save method persist also INSERT records into database but **return type of persist is void** while return type of save is Serializable object.

2) `persist()` method doesn't guarantee that the identifier value will be assigned to the persistent instance immediately, the assignment might happen at flush time.

3) `persist()` method guarantees that it will not execute an INSERT statement if it is called outside of transaction boundaries. `save()` method does not guarantee the same, it returns an identifier, and if an INSERT has to be executed to get the identifier (e.g. "identity" generator), this INSERT happens immediately, no matter if you are inside or outside of a transaction.

4) Fourth difference between `save` and `persist` method in Hibernate is related to previous difference on save vs persist. Because of its above behavior of persist method outside transaction boundary, its useful in long-running conversations with an extended `Session` context. On the other hand `save` method is not good in a long-running conversation with an extendedSession context.

46) Can you explain hibernate relations using annotations?

47) Explain about Connection pooling in hibernate?

48) How to get JNDI property configured in web logic into Hibernate applications?

```
<property name="connection.datasource">java:comp/env/jdbc/stdb</property>
```

49) How to restrict certain properties of pojo to not to

stored in to the database using hibernate Annototions?

Using @transient we can avoid column in the table for specified property in entity class.

Using updatable = false we can avoid data to be update in the table.

Using insertable = false we can avoid data to be inserted in the table.

50) How to call Stored Procedures from Hibernate?

Stored procedures can be called in two ways in hibernate:

1. Using native sql and anther is using namedqueries.

```
DELIMITER $$

CREATE PROCEDURE `GetStocks`(int_stockcode VARCHAR(20))

BEGIN

  SELECT * FROM stock WHERE stock_code = int_stockcode;

  END $$
```

```
DELIMITER ;

CALL GetStocks('7277');

Query query = session.createSQLQuery(
        "CALL GetStocks(:stockCode)")
        .addEntity(Stock.class)
        .setParameter("stockCode", "7277");

List result = query.list();
for(int i=0; i<result.size(); i++){
        Stock stock = (Stock)result.get(i);
        System.out.println(stock.getStockCode());
}

========================================

@NamedNativeQueries({
        @NamedNativeQuery(
        name = "callStockStoreProcedure",
        query = "CALL GetStocks(:stockCode)",
        resultClass = Stock.class
        )
})

Query query = session.getNamedQuery("callStockStoreProcedure")
        .setParameter("stockCode", "7277");
List result = query.list();
for(int i=0; i<result.size(); i++){
        Stock stock = (Stock)result.get(i);
        System.out.println(stock.getStockCode());
```

}

D

@column (name=”columnName”) is not required if property name

And column are same.

Even both property and column names are same better providing

@column (name=”columnName”) because To know column name

It uses reflection api, using reflection is always degrades

the performance.

54. why serializable for primary key.

55. what are constraints in @column.

updatable=false,length=11,nullable=false,unique="",insertable