

1 of 4 5/3/2011 10:19 PM

My favourite hibernate interview questions with answers? At 2:58 AM on Dec 12, 2007, akpillai wrote:

Fresh Jobs for Developers Post a job opportunity

Q. How will you configure Hibernate?

Reply

3 Answer:

DZone

The configuration files hibernate.cfg.xml (or hibernate.properties) and mapping files *.hbm.xml are used by the Configuration class to create (i.e. configure and bootstrap hibernate) the SessionFactory, which in turn creates the Session instances. Session instances are the primary interface for the persistence service.

- " hibernate.cfg.xml (alternatively can use hibernate.properties): These two files are used to configure the hibernate sevice (connection driver class, connection URL, connection username, connection password, dialect etc). If both files are present in the classpath then hibernate.cfg.xml file overrides the settings found in the hibernate.properties file.
- " Mapping files (*.hbm.xml): These files are used to map persistent objects to a relational database. It is the best practice to store each object in an individual mapping file (i.e mapping file per class) because storing large number of persistent classes into one mapping file can be difficult to manage and maintain. The naming convention is to use the same name as the persistent (POJO) class name. For example Account.class will have a mapping file named Account.hbm.xml. Alternatively hibernate annotations can be used as part of your persistent class code instead of the *.hbm.xml files.

Q. What is a SessionFactory? Is it a thread-safe object?

Answer:

SessionFactory is Hibernate sconcept of a single datastore and is threadsafe so that many threads can access it concurrently and request for sessions and immutable cache of compiled mappings for a single database. A SessionFactory is usually only built once at startup. SessionFactory should be wrapped in some kind of singleton so that it can be easily accessed in an application code.

SessionFactory sessionFactory = new Configuration().configure().buildSessionfactory();

5/3/2011 10:19 PM 2 of 4











1. At 10:34 PM on Dec 20, 2007, akpillai wrote:

Re: My favourite hibernate interview questions with answers?



I have some more on this

Q. What is the difference between the session.get() method and the session.load() method?

Both the session.get(..) and session.load() methods create a persistent object by loading the required object from the database. But if there was not such object in the database then the method session.load(..) throws an exception whereas session.get(&) returns null.

Q. What is the difference between the session.update() method and the session.lock() method?

Both of these methods and saveOrUpdate() method are intended for reattaching a detached object. The session.lock() method simply reattaches the object to the session without checking or updating the database on the assumption that the database in sync with the detached object. It is the best practice to use either session.update(..) or session.saveOrUpdate(). Use session.lock() only if you are absolutely sure that the detached object is in sync with your detached object or if it does not matter because you will be overwriting all the columns that would have changed later on within the same transaction.

Note: When you reattach detached objects you need to make sure that the dependent objects are reatched as well.

Q. How would you reatach detached objects to a session when the same object has already been loaded into the session?

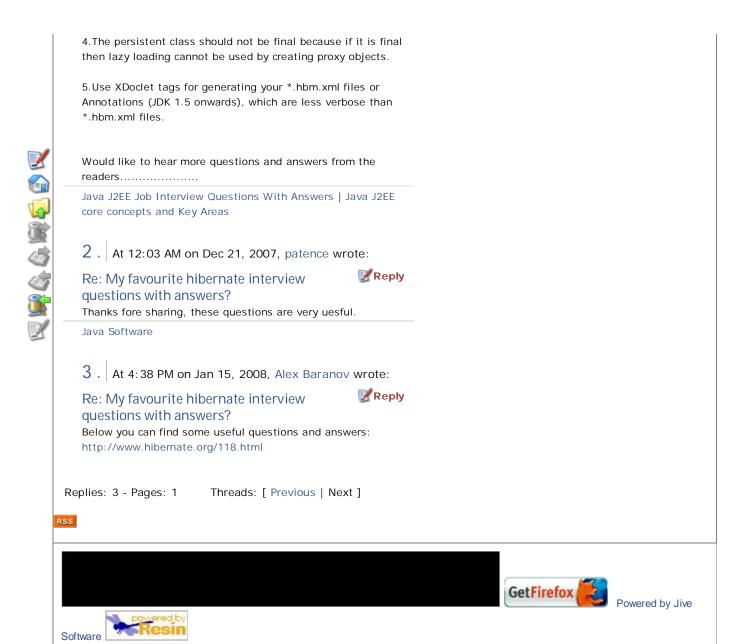
You can use the session.merge() method call.

Q. What are the general considerations or best practices for defining your Hibernate persistent classes?

- 1.You must have a default no-argument constructor for your persistent classes and there should be getXXX() (i.e accessor/getter) and setXXX(i.e. mutator/setter) methods for all your persistable instance variables.
- 2.You should implement the equals() and hashCode() methods based on your business key and it is important not to use the id field in your equals() and hashCode() definition if the id field is a surrogate key (i.e. Hibernate managed identifier). This is because the Hibernate only generates and sets the field when saving the object.
- 3. It is recommended to implement the Serializable interface. This is potentially useful if you want to migrate around a multi-processor cluster.

3 of 4 5/3/2011 10:19 PM





4 of 4 5/3/2011 10:19 PM