

Chapter 1 : Spring Notes | teche

Enter your email address to subscribe to this blog and receive notifications of new posts by email.

Hibernate is a persistence framework which is used to persist data from Java environment to database. Persistence is a process of storing the data to some permanent medium and retrieving it back at any point of time even after the application that had created the data ended. **Hibernate Architecture** The above diagram shows minimal architecture of Hibernate. It creates a layer between Database and the Application. It loads the configuration details like Database connection string, entity classes, mappings etc. Hibernate creates persistent objects which synchronize data between application and database. The above diagram shows a comprehensive architecture of Hibernate. In order to persist data to a database, Hibernate create an instance of entity class Java class mapped with database table. This object is called Transient object as they are not yet associated with the session or not yet persisted to a database. To persist the object to database, the instance of SessionFactory interface is created. SessionFactory is a singleton instance which implements Factory design pattern. More details in following section and with the help of TransactionFactory and ConnectionProvider implements all the configuration settings on a database. Each database connection in Hibernate is created by creating an instance of Session interface. Session represents a single connection with database. Session objects are created from SessionFactory object. Each transaction represents a single atomic unit of work. One Session can span through multiple transactions. SessionFactory A thread-safe, immutable cache of compiled mappings for a single database. A factory for org. A client of org. Optionally maintains a second level cache of data that is reusable between transactions at a process or cluster level. Session A single-threaded, short-lived object representing a conversation between the application and the persistent store. Wraps a JDBC java. They are associated with exactly one org. Session is closed, they will be detached and free to use in any application layer for example, directly as data transfer objects to and from presentation. They may have been instantiated by the application and not yet persisted, or they may have been instantiated by a closed org. Transaction Optional A single-threaded, short-lived object used by the application to specify atomic units of work. Session might span several org. Transactions in some cases. However, transaction demarcation, either using the underlying API or org. Transaction, is never optional. It abstracts the application from underlying javax. TransactionFactory Optional A factory for org. **Hibernate Configuration** Hibernate configuration is managed by an instance of org. An instance of org. Configuration is used to build an immutable org. Hibernate provides following types of configurations hibernate. Programmatic configuration " This is the manual approach. The configuration can be defined in Java class. Below is the sample hibernate. In this case hibernate. Create a file hibernate. Configuration instance by instantiating it directly and specifying XML mapping documents. If the mapping files are in the classpath, use addResource. This approach eliminates any hardcoded filenames. Configuration also allows you to specify configuration properties. **Building a SessionFactory** Once the instance of org. Configuration is created using any of the above method, the singleton instance of SessionFactory can be created as follow: This is useful if you are using more than one database. Getting Session instance As noted above, Session represents a communication channel between database and application. Each session represents a factory of transactions. Session can be created from SessionFactory as follows:

Chapter 2 : Spring MVC in Nutshell

The Spring Material which are available here of Durga sir class www.nxgvision.com Spring Material is Clear and good www.nxgvision.com Spring Material covered by all the classes of the Durga www.nxgvision.com is complete notes for Spring www.nxgvision.com sir is expert in the www.nxgvision.com sir also Explain well in the class.

Chapter 3 : nataraj sir class notes | nataraznotespoint

DOWNLOAD PDF SPRING NOTES BY NAVEEN

java spring java spring framework tutorial spring tutorial spring framework tutorial for beginners with examples spring mvc framework tutorial spring framework by durga spring framework interview.

Chapter 4 : MyKnowledge4Share: DURGA Spring Material

Java Notes by Durga Sir, Nataraj Sir and Naresh Technology. 1- Core Java Notes 2- Advance Java Notes 3- Spring Notes 4- Hibernate Notes 5- Struts-2 Notes 6- SCJP/OCJP Notes Durga Sir.

Chapter 5 : Spring Tutorial

Sign in now to see your channels and recommendations! Sign in. Watch Queue Queue.

Chapter 6 : DURGA SOFTWARE SOLUTIONS

Spring notes 1. Spring:Framework is a piece of software. This resolves the commonly and repeatedly occurred problems in multiple www.nxgvision.com framework is an open source Java platform and it was initially written by RodJohnson and was first released under the Apache license in June Spring is more flexible framework when we compared with struts.

Chapter 7 : Naveen Andrews - Wikipedia

8/19/10 Jeanne Boyarsky's Spring 3.X Certification Study Notes Note: Organized based on the July study guide to the exam posted online by Spring Source.

Chapter 8 : Introduction to Hibernate framework. Hibernate Architecture tutorial

spring notes by nataraz www.nxgvision.com - Free ebook download as PDF File (.pdf), Text File (.txt) or read book online for free.

Chapter 9 : www.nxgvision.com | Tumblr

Nataraj a Sun Certified Java Programmer working as Senior executive trainer with the highest track record, "Every 7 students out of 10 who is successfully placed in industry after undergoing training in Java technologies are by Nataraz, from Ameerpet" stated by a Local News gathering agency, by the year