

***HIBERNATE***

**Training Assignment**

|  |  |
| --- | --- |
| Document Code | 25e-BM/HR/HDCV/FSOFT |
| Version | 1.1 |
| Effective Date | 20/11/2012 |

**Hanoi, 09/2019**

RECORD OF CHANGES

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Effective Date | Change Description | Reason | Reviewer | Approver |
| 1 | 25/09/2019 | Create assignments |  | VietTN | VinhNV |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

Contents

[Unit 1. Configuration 4](#_Toc20838758)

[Problem 1: Create a maven project 5](#_Toc20838759)

[Objectives: 5](#_Toc20838760)

[Problem Descriptions: 5](#_Toc20838761)

[Assumptions: 5](#_Toc20838762)

[Technical Requirements: 5](#_Toc20838763)

[Questions to answer: 5](#_Toc20838764)

[Problem 2: Configure Hibernate 6](#_Toc20838765)

[Objectives: 6](#_Toc20838766)

[Problem Descriptions: 6](#_Toc20838767)

[Assumptions: 6](#_Toc20838768)

[Technical Requirements: 6](#_Toc20838769)

[Questions to answer: 6](#_Toc20838770)

[Problem 3: Mapping New Table 7](#_Toc20838771)

[Objectives: 7](#_Toc20838772)

[Problem Descriptions: 7](#_Toc20838773)

[Assumptions: 7](#_Toc20838774)

[Technical Requirements: 7](#_Toc20838775)

[Questions to answer: 7](#_Toc20838776)

|  |  |
| --- | --- |
|  | **CODE: ORM.S.A101**  **TYPE: SHORT**  **LOC: NA**  **DURATION: 90 Minutes (completed in 1 work day)** |

# Unit 1. Configuration

This assignment requires trainees having Java, Maven, SQL, File and Hibernate knowledge. Technical details are:

* Java 8
* [Hibernate 4.3](https://hibernate.org/orm/releases/4.3/)
* [Maven 3](https://maven.apache.org/download.cgi)
* [Apache Derby 10.14](https://db.apache.org/derby/releases/release-10.14.2.0.cgi)

The important note:

* Should **avoid many-to-many** design for entities and have to know the differences between **create** and **update** for the property **hbm2ddl.auto**.
* Java classes must follow naming convention rules (for examples, class names and attribute names must not have \_)
* Tables must have names as required on the design (for examples, Employee table has first\_name column instead of firstname)
* The DAO class should have at least 5 methods as getXxxByID, getAllXxx, updateXxxByID, deleteXxxById, insertXxx where Xxx is the model class of DAO. You could choose other verbs instead of get, update, delete, and insert. However, they should be consistent among DAO classes and meaningful.

Happy coding!

*These assignments should be performed by individuals.*

Problem 1: Create a maven project

Trainees could use Eclipse or Maven commands to create a maven project.

Objectives:

* Understand the Maven project structure (main and test source code folder)
* Able to create a Maven project using Eclipse or commands
* Able to install, test or clean a Maven project by using Eclipse or commands

Problem Descriptions:

The project team needs to create new project structure to use Maven managing libraries. You are assigned to this task.

Assumptions:

* There are no dependencies on the development environment to use Maven.
* Derby 10.14 is downloaded and ready to use.

Technical Requirements:

* Must use Eclipse to build projects.
* Must use tools with versions mentioned above.

Questions to answer:

* Create a Maven project with name as **fa.training.hibernate.day01**.
* Update pom.xml by adding dependencies for **Hibernate 4.3**, **MySQL/MSSQL Serer**/**Derby 10.14** and **Derby Client 10.14**.

Run Maven install to download required jar files without any error.

By completing these tasks, your project must have Maven Dependencies as the picture below:



Problem 2: Configure Hibernate

Objectives:

* Able to configure to use Hibernate in the project

Problem Descriptions:

After creating the project, the team decided to use Hibernate 4.3, MSQL/MSSQL Server/Derby 10.14 to develop their project. Therefore, they could easily perform CRUD operations on database tables.

They want to create a new database with name **fadb**. In the development phase, they want to show SQL queries so they could track and debug if it is necessary.

Assumptions:

* There are no dependencies on the development environment to use Maven.
* MySQL/MSSQL Server/Derby 10.14 is downloaded and ready to use.
* Complete Problem 1 to use the existing project.

Technical Requirements:

* Must use Eclipse to build projects
* Must use tools with versions mentioned above

Questions to answer:

* Create a configuration file hibernate.cfg.xml in src/main/resources
* As the above problems/requirements, update values for these properties: driver\_class, url, dialect, show\_sql, hbm2ddl.auto
* Create a class to test your configuration correct or not (Use SessionFactory and Session in the main method)

The log output should not have any error.

You also should connect to the **fadb** database successful.

Problem 3: Mapping New Table

Objectives:

* Able to configure for model classes by using annotations to generate new tables in the database.

Problem Descriptions:

On the ERD, they want to have Employee table as below:

|  |  |
| --- | --- |
| **Column Name** | **Data Type and Constraints** |
| ID | int, Primary Key |
| First\_Name | varchar(50), not null |
| Last\_Name | varchar(50), not null |

You need to create a model class to create and map with Employee table.

Assumptions:

* There are no dependencies on the development environment to use Maven.
* MySQL/MSSQL Server/Derby 10.14 is downloaded and ready to use.
* Complete Assignment 2 to use the existing project.

Technical Requirements:

* Must use Eclipse to build projects
* Must use tools with versions mentioned above

Questions to answer:

* Create and use annotation in the Employee class. The class should be in a model package, for instance, **fa.training.hibernate.model**
* Create a new Employee object with your selected first name and last name on the main method.
* Use the existing session to save new Employee object.

The log output should not have any error.

You should see your new Employee entity by running ‘**SELECT \* FROM EMPLOYEE**’ after connected to **fadb** database.

**-- THE END --**