Работа с Hibernate

№ урока: 5 **Курс:** JDBC & Hibernate

Средства обучения: JDK, Intelij Idea, Maven

Обзор, цель и назначение урока

Работа с выборками добавлением и обновлением в Hibernate. Рассмотрение ID-генератора.

Изучив материал данного занятия, учащийся сможет:

- Научится правильно делать выборку в Hibernate.
- Научиться правильно обновлять данные в Hibernate.
- Научиться делать выборку из нескольких полей.
- Поймет, как работает ID-генератор в Hibernate.

Содержание урока

1. Рассмотрение выборки и добавления в Hibernate.
2. Рассмотрение обновления в Hibernate.
3. Рассмотрения ID-генератора в Hibernate.
4. Рассмотрения выборки некоторых полей.

Резюме

- Логи (лог-файлы) — это файлы, содержащие системную информацию работы сервера или компьютера, в которые заносятся определенные действия пользователя или программы. Иногда также употребляется русскоязычный аналог понятия — журнал.
  Их предназначение — протоколирование операций, выполняемых на машине, для дальнейшего анализа администратором. Регулярный просмотр журналов позволит определить ошибки в работе системы в целом, конкретного сервиса или сайта (особенно скрытые ошибки, которые не выводятся при просмотре в браузере), диагностировать злонамеренную активность, собрать статистику посещений сайта.
- При создании любого приложения – сразу настраивайте логирование! По умолчанию в Hibernate используется реализация jboss-logging–можно «подставить» любую реализацию
  Правильно настроить уровни логирования – насколько детально выводить информацию по разным пакетам java(чтобы не было переизбытка или нехватка логов).
- **@DynamicUpdate и @DynamicInsert** – обновляют только те поля, которые используются. Это аннотации Hibernate, не JPA.
- **Метод flush()** – синхронизация состояния объекта Session (persistence context) и БД,
  все изменения сессия сначала записывает в оперативную память, метод session.flush() – сбрасывает все изменения в БД. Не commit. Может возникнуть ошибка OutOfMemoryException.
- **IDENTITY** – использование поля autoincrement (если поддерживается в СУБД)
- **TABLE** – использовать собственную таблицу (в любом СУБД)
- **SEQUENCE** – специальная таблица (если поддерживается СУБД)
- **AUTO** –JPA библиотека сама выбирает какой способ использовать (зависит от реализации библиотеки).
- Может возникать ошибка «Table 'hibernate_sequence' doesn't exist» при генерации ID.
- Если указано значение параметра hdm2ddl тега property true - создается новая таблица даже с измененными названиями полей.
- При выборке полей необходимо: Использовать интерфейс Selection. В ранних версиях Hibernate API часто использовался Projection, но он для Deprecated API (Criteria). Поля с большим объемом данных –не загружать сразу, а только по требованию.

CyberBionic systematics

CyberBionic Systematics ® 2018
19 Eugene Sverstyuk Str.,5 floor
Kyiv, Ukraine

Tel. 0 800 750 312
E-mail: edu@cbsystematics.com
Site: www.edu.cbsystematics.com | itvdn.com

Закрепление материала

- Как правильно обновлять поля в таблице?
- Зачем нужны аннотации DynamicUpdate и DynamicInsert?
- Какие есть свойства генерации ID в Hibernate и как они работают?
- Как выбрать конкретные поля?

Дополнительное задание

Задание
Из пакета ex_002_insert_and_update создайте в цикле 200 объектов author и сохраните в БД. Значения полей могут быть любыми. Используйте метод flush для каждых 10 объектов. Метод commit выполняйте один раз в конце.

Самостоятельная деятельность учащегося

Задание 1
Выучите основные понятия, рассмотренные на уроке.

Задание 2
Настроить логирование для проекта с дополнительным заданием.

Задание 3
Прочитать: Пример использования @TableGenerator http://www.summa.com/blog/2011/07/29/setting-up-sequential-ids-using-jpa-tablegenerator

Задание 4
К дополнительному заданию добавить метод обновления имени автора по id. (То, что было на уроке, только реализовать это правильно). Аналогично сделать и в классе BookHelper с предыдущего ДЗ.

Задание 5*
В классе BookHelper создать метод, который получает название книг и имя автора.

Рекомендуемые ресурсы

Логи в Hibernate:
http://docs.jboss.org/hibernate/orm/5.2/topical/html_single/logging/Logging.html

MySQL Autoincrement
http://www.mysqltutorial.org/mysql-sequence/

Oracle Autoincrement
https://livesql.oracle.com/apex/livesql/file/content_HAV9MQ5I36RWCUZJR8QQ5VMP.html

Пример использования @TableGenerator
http://www.summa.com/blog/2011/07/29/setting-up-sequential-ids-using-jpa-tablegenerator

CyberBionic systematics

CyberBionic Systematics ® 2018
19 Eugene Sverstyuk Str.,5 floor
Kyiv, Ukraine

Tel. 0 800 750 312
E-mail: edu@cbsystematics.com
Site: www.edu.cbsystematics.com | itvdn.com