

Расширенные возможности Hibernate

№ урока: 6 **Курс:** JDBC & Hibernate

Средства обучения: JDK, IntelliJ Idea, Maven

Обзор, цель и назначение урока

Работа с генерацией классов в IntelliJ Idea. Рассмотрение выборки по условию и удаления. Получение связей между таблицами.

Изучив материал данного занятия, учащийся сможет:

- Научится генерировать классы в IntelliJ Idea.
- Научиться делать выборку по условию.
- Научиться удалять данные.
- Научиться работать со связями таблиц.

Содержание урока

1. Рассмотрение средств генерации классов в IntelliJ Idea.
2. Рассмотрение выборки при условии.
3. Рассмотрение удаления.
4. Рассмотрения связей между таблицами.

Резюме

- Для генерации классов:
 - Используются специальные скрипты библиотеки Hibernate
 - Разные IDE предоставляют визуальные способы работы с этими скриптами, вместо работы в консоли (например, NetBeans имеет удобный инструмент работы с Hibernate)
 - Разные IDE предоставляют разные возможности и инструменты для генерации (кстати, в NetBeans в данный момент реализована более удобная работа с Hibernate)
- **Пакетное удаление** – возможность удалять несколько записей при условии.
- Существует еще удаление **конкретного** объекта.
- **@ManyToOne (объект)** -у многих книг может быть один автор
- **@OneToMany (коллекция)** -автор может иметь много книг
- **@OneToOne (объект)** -человек и документ
- **@ManyToMany (коллекция)** –у книги может быть несколько авторов, автор может иметь много книг.
- В JPA есть 2 типа загрузки (FetchType): EAGER and LAZY. EAGER загрузка заставляет ORM загружать связанные сущности и коллекции сразу, вместе с корневой сущностью. LAZY загрузка означает, что ORM загрузит сущность или коллекцию отложено, при первом обращении к ней из кода.
- FetchType в JPA говорит когда мы хотим, чтоб связанная сущность или коллекция была загружена. По умолчанию JPA провайдер загружает связанные коллекции (отношения один-ко-многим и многие-ко-многим) отложено (lazy loading). В большинстве случаев отложенная загрузка — оптимальный вариант. Нет смысла инициализировать все связанные коллекции, если к ним не будет обращений.
- JPA предоставляет две основных стратегии загрузки: SELECT и JOIN. Когда выбрана стратегия загрузки SELECT, ORM загружает связанные коллекции отдельным SQL запросом. Иногда эта стратегия может негативно повлиять на производительность, особенно, когда в результирующей выборке большое количество элементов. Эту проблему часто называют «N+1 selects».

- Стратегия JOIN указывает ORM, что загружать связанные коллекции необходимо и одним SQL запросе с корневой сущностью, используя оператор LEFT JOIN в сгенерированном SQL запросе. Часто эта стратегия лучше с точки зрения производительности, особенно, когда в результирующей выборке большое количество элементов. Конечно, при условии, что в дальнейшем к загруженным коллекциям будут обращения в коде. Есть несколько способов указать ORM использовать стратегию загрузки JOIN: JPQL оператор JOIN FETCH, метод fetch класса Root (JPA Criteria), entity graph, добавленные в JPA 2.1

Закрепление материала

- Как сгенерировать класс в IntelliJ Idea?
- Как сделать выборку по условию?
- Какие есть способы удаления объектов?
- Какие есть аннотации для связывания объектов как таблиц?
- Что за свойства аннотаций FetchType.LAZY и FetchType.EAGER?

Дополнительное задание

Задание

Из пакета ex_002_select_where написать отдельный метод для выборки по поиску выражения и в пакете ex_003_delete методы createCriteria и createCriteriaLogic переписать правильно.

Самостоятельная деятельность учащегося

Задание 1

Выучите основные понятия, рассмотренные на уроке.

Задание 2

Обновить поле name для всех записей, у которых длина значения поля last_name больше 7

В поле name записать значение «1».

* Задание на самостоятельный поиск решения.

Задание 3

В класс BookHelper пакета ex_004_relations дописать методы удаления книги по id и по автору.

Задание 4

Используя MySQL Workbench переписать базу данных так, чтобы одну книгу могли написать несколько авторов, также один автор может написать несколько книг. Реализовать связь многие ко многим.

Рекомендуемые ресурсы

Пример использования CriteriaUpdate

<http://dreamand.me/java/jpa-update-selected-fields/>

Стратегии загрузки коллекций в JPA

<https://dou.ua/lenta/articles/jpa-fetch-types/>