

Практическая работа «Работа с базами данных» по курсу JavaEE.

В рамках данного задания необходимо научиться работать с уровнем данных трехзвенного приложения посредством технологии JDBC или JPA (на усмотрение слушателя).

Постановка задачи будет разбита на несколько этапов, при этом успешно выполненной работой считается та, в которой выполнены все пункты ниже (в некоторых случаях, порядок следования этапов можно менять местами):

Нулевой этап – анализ.

Прежде, чем начать что-либо реализовать, необходимо представить решаемую проблему в голове и постараться декомпозировать задачу на более мелкие. Возможно, есть моменты, которые требуют разъяснения перед началом работ (все вопросы смело уточняйте у Вашего преподавателя). Только после того как предметная область ясна, можно приступать к следующим пунктам, которые позволят поэтапно решить поставленную задачу.

Данный этап можно пропустить, если Вы уже успешно справились с предыдущим заданием «ДЗ_Многоуровневая модель приложения» (далее – ДЗ_1).

Первый этап - проектирование.

Для начала нам понадобится составить *UML-диаграмму классов*, включающую основные сущности изучаемой предметной области (см. ДЗ_1). В дальнейшем, графические артефакты окажутся полезными для отражения этой информации в схеме базы данных (далее - БД), а также в соответствующий код JavaEE-приложения. UML-диаграмму можно смоделировать «на бумаге» или же в понравившемся редакторе UML (предпочтительный способ). Предполагается, что основы языка UML Вам известны или отводятся на самостоятельное изучение.

В данном задании, основным критерием является умение создавать образы и сущности материального и нематериального мира, выявлять их **значимые характеристики и поведение** (умение абстрагироваться и фокусироваться на важных аспектах), а также выявлять *отношения* между ними.

Второй этап – подготовка схемы БД.

От Вас требуется выбрать любую понравившуюся СУБД (MySQL, Oracle, MSSQL, PostgreSQL, DB2, etc), развернуть ее на локальном окружении и создать в ней новую схему и/или базу данных. Используя ранее составленную диаграмму классов, требуется наполнить схему БД всеми таблицами предметной области, колонки которой представляют соответствующие характеристики описываемой сущности (в частности, таблица сотрудников должна содержать поля ИД сотрудника (автоинкрементальное), ФИО, подразделение и город (или внешний ключ на сущность департамента), должность, зарплата (или внешний ключ на сущность должность), логин, пароль/"солированный" хэш (или внешний ключ на сущность оператора))).

В данной части задания, необязательно писать скрипты вручную, можно воспользоваться соответствующими утилитами, предоставляющих возможности администрирования БД через пользовательский интерфейс / UI (для MySQL можно воспользоваться MySQL Workbench, phpMyAdmin, для Oracle – PLSQL Developer, Toad, *SQLDeveloper* и т.д.).

На данном этапе проверяется умение работать с сущностями базы данных.

Третий этап – подготовка Application уровня.

Требуется создать Maven проект веб-приложения (на базе maven-archetype-webapp), который «умеет» соединяться с созданной ранее БД и выполнять DML-операции (CRUD-запросы для JDBC) или осуществлять отображение таблиц БД (JPA – в данном случае, проекту следует добавить поддержку JavaEE Persistence). При желании также можно автоматизировать предыдущий этап,

выполнив его на Application уровне (используя DDL (создание базы, схемы, таблиц схемы) – JDBC или генерируя таблицы фреймворком Hibernate).

Результат выполнения – получить приложение, способное устанавливать соединение с СУБД. То, как подключаться к базе данных, составлять запросы/строить реляционное отображение будет рассказано на вебинаре. Поэтому данное задание позволяет закрепить полученные знания на практике.

Четвертый этап – наполнение таблиц.

Заполнить данными имеющиеся таблицы произвольными статичными данными или данными из некоего внешнего файла (csv, txt, xml - не имеет значение какого типа). Достаточно десятка записей.

Поскольку JavaEE неразрывно связана с JavaSE, то важно проверить насколько хорошо слушатель овладел знаниями из стандартной Java (работа с потоками IO, с файлами, внешними ресурсами). Возможно, есть какие-то пробелы, которые следует локализовать на текущем этапе.

Пятый шаг – поиск, модификация и удаление данных.

После успешного сохранения данных в БД необходимо вывести в консоль/ответ сервлета (или внешний файл result.txt - опционально) информацию с сотрудниками, отсортированную по ИД сотрудника по убыванию. Также потребуются произвести изменение 2 строк данной таблицы, подменив фамилию сотрудника, а также его должность (на данном этапе не обязательно брать во внимание предусмотренные в системе роли: рядовой сотрудник, Бухгалтер, HR-специалист, директор). Удалить произвольные 3 строки из данной таблицы.

Данная часть задания позволяет проверить умение модифицировать/удалять данные из-под аппликейшн уровня.

Шестой шаг – работа с хранимыми процедурами.

Создать хранимую функцию (средствами JDBC или непосредственно в БД), возвращающую фамилия сотрудника, имеющего наибольшую зарплату. Произвести вызов этой процедуры из приложения и вернуть результат в консоль/ответ сервлета или во внешний файл maxSalary.txt

В данном задании ожидается от слушателя научиться работать с хранимыми процедурами и функциями (PL/SQL), а также умение обращаться к ним из-под Java.