ОТЧЕТ  
ПО ЛАБОРАТОРНОЙ РАБОТЕ №2  
по дисциплине «Разработка приложений баз данных»  
«Использование технологии ODBC для создания приложений баз данных»  
**Вариант 15**

Факультет: АВТ Преподаватель: Менжулин С. А.

Группа: АВТ-943

Студент: Фокин Е.К.

Цель работы:

Получение практических навыков ODBC PostgreSQL.

Задание:

1. Для базы данных из лабораторной работы №1 разработать набор классов-сущностей, хранящих свое состояние в базе данных, для описания отображения использовать любой доступный способ включая Fluent;
2. Реализовать функции пользовательского интерфейса: просмотр/добавление/редактирование/удаление записи, поиск по всем полям, в том числе частичное совпадение. Запросы реализовать при помощи языка HQL

Задание варианта:

20) БД «Гостиница»

База данных должна содержать сведения о следующих объектах:

Распределение номеров с указанием общего количества мест в номере, количества свободных мест и проживающих.

Гости - фамилия, имя, отчество, пол, адрес, дата рождения, номер паспорта, дата выдачи, учреждение, выдавшее паспорт, номер комнаты, номер стоянки, регистрационный номер автомобиля, дата въезда, дата выезда, список оказанных услуг (наименование услуги, количество, цена).

Адресные данные горничных и расписание их дежурств.

Ход работы:

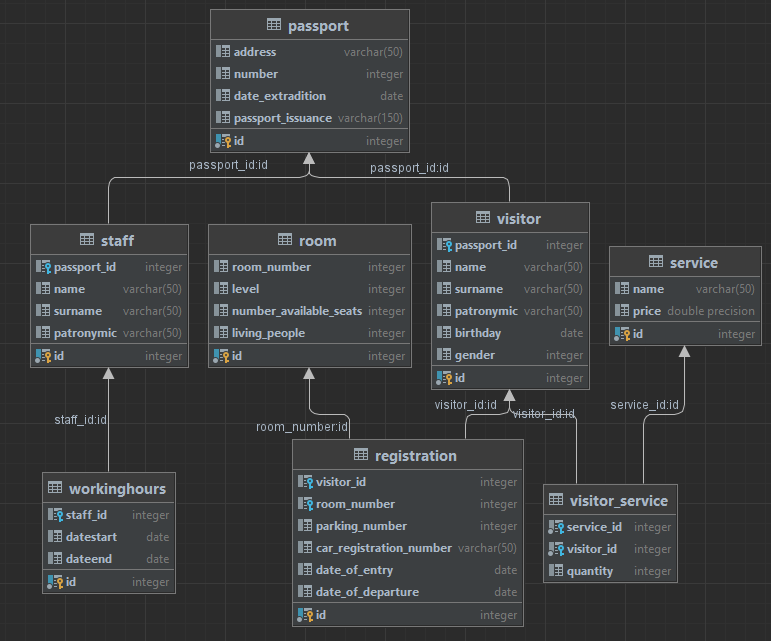


Рис. 1 Схема базы данных

В данной работе будем использовать Java и Hibernate - библиотека для языка программирования Java, предназначенная для решения задач объектно-реляционного отображения.

Для начала определим файл pom.xml:

<dependency>  
 <groupId>org.postgresql</groupId>  
 <artifactId>postgresql</artifactId>  
 <version>42.5.0</version>  
</dependency>  
<dependency>  
 <groupId>org.hibernate.orm</groupId>  
 <artifactId>hibernate-core</artifactId>  
 <version>6.1.5.Final</version>  
</dependency>  
<dependency>

После, создадим подключение к базе данных postgresql, созданной ранее в 1 лабораторной работе.

В файле persistence.xml добавим конфигурацию, в котором укажем параметры для подключения к базе данных, такие как класс драйвера, url базы, логин и пароль:

<properties>  
 <property name="hibernate.connection.url" value="jdbc:postgresql://localhost:5432/hotel"/>  
 <property name="hibernate.connection.driver\_class" value="org.postgresql.Driver"/>  
 <property name="hibernate.connection.username" value="visitor"/>  
 <property name="hibernate.connection.password" value="12345"/>  
 <property name="hibernate.enable\_lazy\_load\_no\_trans" value="true"/>  
</properties>

Тут же будут содержатся mapping файлы, которые будут определять классы, которые являются отображением сущности в базе данных.

Отображения сущности содержатся в виде XML, в которых, также, определены основные запросы на языке HQL.

Добавим также классы отправляющие запросы к базе данных:

FROM Specialization s

FROM Specialization s where upper(cast(s.name as string)) like concat('%', upper(?1), '%')

Весь проект в приложение будет отправлен архивом в Dispace

Вывод:

В ходе лабораторной работы с базой данных из лабораторной работы №1, был разработан набор классов-сущностей, хранящих свое состояние в базе данных, для описания отображения использовался XML.

Разработаны функции пользовательского интерфейса (CLI): просмотр/добавление/редактирование/удаление записи, поиск по всем полям, в том числе частичное совпадение. Запросы реализованы при помощи языка HQL.