Одесский национальный политехнический университет

Кафедра интеллектуальных систем и сетей

Лабораторная работа №11

По предмету «Системное программирование»

На тему «**Язык интегрированных запросов**»

Выполнили

Студенты группы АЕ-145

Михайлов О.М.

Зайцева М.О.

Проверил

Олещук О.В.

Одессы 2017

**Цель лабораторной работы**

Вывести на экран информацию согласно варианту задания:

Варианты заданий

Вариант 1 (класс: город)

Список стран, упорядоченных по суммарному количеству жителей во всех городах.

Название города с площадью территории, наиболее близкой к среднему в списке.

**Ход выполнения работы:**

Файл конфигурации Hibernate:

*<?***xml version='1.0' encoding='utf-8'***?>*

**<!DOCTYPE** **hibernate-configuration** **PUBLIC**

**"-//Hibernate/Hibernate Configuration DTD//EN"**

**"http://www.hibernate.org/dtd/hibernate-configuration-3.0.dtd"*>***

<**hibernate-configuration**>

<**session-factory**>

<**property name="connection.url"**>jdbc:mysql://localhost:3306/Country?serverTimezone=UTC

</**property**>

<**property name="connection.driver\_class"**>com.mysql.jdbc.Driver</**property**>

<**property name="dialect"**>org.hibernate.dialect.MySQL5Dialect</**property**>

<**property name="connection.username"**>root</**property**>

<**property name="connection.password"**>76222377</**property**>

<**property name="hbm2ddl.auto"**>update</**property**>

<**property name="show\_sql"**>true</**property**>

<**property name="format\_sql"**>true</**property**>

<**property name="use\_sql\_comments"**>false</**property**>

<**mapping class="com.hibernate.entity.Cities"**></**mapping**>

</**session-factory**>

</**hibernate-configuration**>

Класс City

**package** com.hibernate.entity;

**import** javax.persistence.Column;

**import** javax.persistence.Entity;

**import** javax.persistence.GeneratedValue;

**import** javax.persistence.GenerationType;

**import** javax.persistence.Id;

**import** javax.persistence.Table;

@Entity

@Table(name = **"cities"**)

**public class** Cities {

@Id

@GeneratedValue(strategy = GenerationType.***IDENTITY***)*//автоинкрементация*

**private int id**;

@Column(name = **"country"**)

**private** String **country**;

@Column(name = **"city"**)

**private** String **city**;

@Column(name = **"population"**)

**private int population**;

@Column(name = **"size"**)

**private int size**;

**public int** getId() {

**return id**;

}

**public void** setId(**int** id) {

**this**.**id** = id;

}

**public** String getCountry() {

**return country**;

}

**public void** setCountry(String country) {

**this**.**country** = country;

}

**public** String getCity() {

**return city**;

}

**public void** setCity(String city) {

**this**.**city** = city;

}

**public int** getPopulation() {

**return population**;

}

**public void** setPopulation(**int** population) {

**this**.**population** = population;

}

**public int** getSize() {

**return size**;

}

**public void** setSize(**int** size) {

**this**.**size** = size;

}

@Override

**public** String toString() {

**return "country='"** + **country** + **'\''** +

**", city='"** + **city** + **'\''** +

**", population="** + **population** +

**", size="** + **size**;

}

}

Запросы к БД:

**package** com.hibernate;

**import** com.hibernate.entity.Cities;

**import** org.hibernate.Session;

**import** org.hibernate.query.Query;

**import** java.util.List;

**public class** Require {

**public** String getMiddleCity(Session session) {

StringBuilder result = **new** StringBuilder();

Query<Double> query = session.createQuery(**"select avg(size) from Cities"**);

Integer middleSize = query.getSingleResult().intValue();

Query<Cities> getCity = session

.createQuery(**"from Cities where size >= "** + middleSize + **" order by size asc"**);

List<Cities> cities = getCity.getResultList();

result.append(cities.get(0).getCity() + **" : "**);

result.append(cities.get(0).getSize());

**return** result.toString();

}

**public void** addCity(String country, String city, **int** population, **int** size, Session session) {

Cities cities = **new** Cities();

cities.setCountry(country);

cities.setCity(city);

cities.setPopulation(population);

cities.setSize(size);

session.save(cities);

}

**public void** sortCountryFromCity(Session session) {

StringBuilder result = **new** StringBuilder();

Query<Double> query = session.createQuery(

**"select c.country, sum(c.population) as total from Cities c group by c.country order by total"**);

List list = query.getResultList();

**for** (**int** i = 0; i < list.size(); i++) {

Object[] o = (Object[]) list.get(i);

result.append(**"{"**);

**for** (Object obj : o) {

result.append(obj.toString() + **" : "**);

}

result.append(**"}"**);

}

}

}

Подключение к БД:

**package** com.hibernate;

**import** org.hibernate.Session;

**import** org.hibernate.SessionFactory;

**import** org.hibernate.Transaction;

**import** org.hibernate.cfg.Configuration;

**public class** HibernateConnection {

**static** Require require = **new** Require();

**public static void** main(String[] args) {

SessionFactory sessionFactory = **new** Configuration()

.configure()

.buildSessionFactory();

Session session = sessionFactory.openSession();

Transaction transaction = session.beginTransaction();

*// require.addCity("USA", "Tyler", 85000, 140, session);*

*// result = require.getMiddleCity(session);*

require.sortCountryFromCity(session);

transaction.commit();*//отправка данных транзакцией в БД*

session.close();

sessionFactory.close();

}

}