Учреждение образования

«Белорусский государственный университет информатики и радиоэлектроники»

Факультет компьютерного проектирования

Кафедра экономической информатики

Лабораторная работа №3

**Работа с Entity бинами Java EE**

Выполнил: Проверил:

Студент: гр.714301

Александрова Е.И. Кунцевич А.А.

Минск 2020

**Цель –** разработать веб-ориентированное приложение на базе Java EE, позволяющее при помощи сессионных бинов управлять состоянием приложения (ЛР2), а при помощи entity бинов работать с БД (ЛР3).

**Задание:** разработать приложение на базе Java EE для управления информационной системой банка (управление кредитами). Использовать для организации логики работа сессионные бины, а для работы с данными entity бины.

**Текст программы:**

package com.aleksandrova.entities;  
  
import javax.persistence.\*;  
import java.io.Serializable;  
  
@Entity  
@Table(name = "CUSTOMERS")  
public class Customer implements Serializable {  
 @Id  
 @GeneratedValue(strategy = GenerationType.*IDENTITY*)  
 @Column(name = "id")  
 private int id;  
  
 @Column(name = "name")  
 private String libraryName;  
  
 @Column(name = "readers\_count")  
 private int readersCount;  
  
 @Column(name = "status")  
 private String status;  
  
 public int getId() {  
 return id;  
 }  
  
 public void setId(int id) {  
 this.id = id;  
 }  
  
 public String getLibraryName() {  
 return libraryName;  
 }  
  
 public void setLibraryName(String libraryName) {  
 this.libraryName = libraryName;  
 }  
  
 public int getReadersCount() {  
 return readersCount;  
 }  
  
 public void setReadersCount(int readersCount) {  
 this.readersCount = readersCount;  
 }  
  
 public String getStatus() {  
 return status;  
 }  
  
 public void setStatus(String status) {  
 this.status = status;  
 }  
  
 @Override  
 public String toString() {  
 return "Customer{" +  
 "id=" + id +  
 ", name='" + libraryName + '\'' +  
 ", readersCount=" + readersCount +  
 ", status='" + status + '\'' +  
 '}';  
 }  
}

package com.aleksandrova;  
  
import com.aleksandrova.entities.Customer;  
  
import javax.persistence.EntityManager;  
import javax.persistence.EntityManagerFactory;  
import javax.persistence.Persistence;  
import javax.servlet.annotation.WebServlet;  
import javax.servlet.http.HttpServlet;  
import javax.servlet.http.HttpServletRequest;  
import javax.servlet.http.HttpServletResponse;  
import java.io.IOException;  
import java.io.PrintWriter;  
  
@WebServlet(urlPatterns = {"/customers-servlet"})  
public class CustomerDetails extends HttpServlet {  
 @Override  
 protected void doPost(HttpServletRequest request, HttpServletResponse response) throws IOException {  
  
 EntityManagerFactory emf = Persistence.*createEntityManagerFactory*("PersistenceUnit");  
 EntityManager entityManager = emf.createEntityManager();  
 entityManager.getTransaction().begin();  
 entityManager.joinTransaction();  
 response.setContentType("text/html;charset=UTF-8");  
 PrintWriter out = response.getWriter();  
 out.println("<html>");  
 out.println("<head>");  
 out.println("<title>Servlet CustomerDetails</title>");  
 out.println("</head>");  
 out.println("<body>");  
 out.println("<h1>Search Customer Information</h1>");  
 String customerNr = request.getParameter("customer");  
  
 if ((customerNr != null) && !(customerNr.equals(""))) {  
 Customer client = entityManager.find(Customer.class,Integer.*valueOf*(customerNr));  
 if (client != null) {  
 out.println("Customer's info for № " + customerNr + ": " + client.getLibraryName() + ", "  
 + client.getReadersCount() + ", " + client.getStatus());  
 } else {  
 out.println("Customer not found.");  
 }  
 }  
  
 out.println("<form>");  
 out.println("Customer number: <input type='text' name='customer\_nr'/>");  
 out.println("<input type=submit value=Select />");  
 out.println("</form>");  
 out.println("</body>");  
 out.println("</html>");  
 out.close();  
  
 }  
}

insert into CUSTOMERS(id, name, readers\_count, status) values (1, 'firsLibrary', 120, 'child');  
insert into CUSTOMERS(id, name, readers\_count, status) values (2, 'secondLibrary', 130, 'scince');  
insert into CUSTOMERS(id, name, readers\_count, status) values (3, 'thirdLibrary', 140, 'history');

**Скриншоты работы:**





