Session 7 (Page 2)

CSC 285

EmployeeManager (MySQL) Part 2

EmployeeDAO

Modifications On This Page

//package finalproject;

import java.sql.\*;

import java.util.List;

import java.util.ArrayList;

import java.util.logging.Logger;

public class EmployeeDAO {

private static Logger log = Logger.getLogger(EmployeeDAO.class.getName());

public EmployeeDAO() throws Exception {

// Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

//String url = "jdbc:odbc:Driver={Microsoft Access Driver (\*.mdb)};DBQ=C:\\EmployeeManager\\employees.mdb;";

String url = "jdbc:odbc:Driver={Microsoft Access Driver (\*.mdb)};DBQ=C:\\EmployeeManager\\employees.mdb;";

// String url = "jdbc:odbc:Driver={Microsoft Access Driver (\*.mdb)};DBQ=C:\\test1\\employee.mdb;";

String username = "anonymous";

String password = "guest";

// Load the driver to allow connection to the database

try {

Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

Connection connection = DriverManager.getConnection(url, username, password);

} catch (ClassNotFoundException cnfex) {

System.err.println("Failed to load JDBC/ODBC driver.");

cnfex.printStackTrace();

System.exit(1); // terminate program

} catch (SQLException sqlex) {

System.err.println("Unable to connect");

sqlex.printStackTrace();

}

}

public List<Employee> getAllEmployees() throws Exception {

log.fine("getAllEmployees called");

Connection connection = null;

Statement statement = null;

ResultSet resultSet = null;

try {

connection = getConnection();

statement = connection.createStatement();

resultSet = statement.executeQuery("SELECT \* FROM EMPLOYEES ORDER BY name");

return resultSetToEmployees(resultSet);

} finally {

close(resultSet, statement, connection);

}

}

public void addEmployee(Employee employee) throws Exception {

log.fine("addEmployee called");

Connection connection = null;

PreparedStatement statement = null;

try {

connection = getConnection();

statement = connection.prepareStatement("INSERT INTO EMPLOYEES (name, address, hours, rate, sex, age, active, ssn,phone) VALUES (?,?,?,?,?,?,?,?,?)");

int i = 1;

statement.setString(i++, employee.getName());

statement.setString(i++, employee.getAddress());

statement.setDouble(i++, employee.getHours());

statement.setDouble(i++, employee.getRate());

statement.setString(i++, String.valueOf(employee.getSex()));

statement.setInt(i++, employee.getAge());

statement.setBoolean(i++, employee.isActive());

statement.setString(i++, employee.getSSN());

statement.setString(i++, employee.getPhone());

statement.executeUpdate();

} finally {

close(null, statement, connection);

}

}

public void updateEmployee(Employee employee) throws Exception {

log.fine("updateEmployee called");

Connection connection = null;

PreparedStatement statement = null;

try {

connection = getConnection();

statement = connection.prepareStatement("UPDATE EMPLOYEES SET name = ?, address = ?, hours = ?, rate = ?, sex = ?, age = ?, active = ?, ssn = ?, phone = ? WHERE id = ?");

int i = 1;

statement.setString(i++, employee.getName());

statement.setString(i++, employee.getAddress());

statement.setDouble(i++, employee.getHours());

statement.setDouble(i++, employee.getRate());

statement.setString(i++, String.valueOf(employee.getSex()));

statement.setInt(i++, employee.getAge());

statement.setBoolean(i++, employee.isActive());

statement.setString(i++, employee.getSSN());

statement.setString(i++, employee.getPhone());

statement.setInt(i++, employee.getId());

statement.executeUpdate();

} finally {

close(null, statement, connection);

}

}

public void deleteEmployee(Employee employee) throws Exception {

log.fine("deleteEmployee called");

Connection connection = null;

PreparedStatement statement = null;

try {

connection = getConnection();

statement = connection.prepareStatement("DELETE FROM EMPLOYEES WHERE id = ?");

statement.setInt(1, employee.getId());

statement.executeUpdate();

} finally {

close(null, statement, connection);

}

}

private List<Employee> resultSetToEmployees(ResultSet resultSet) throws Exception {

log.fine("resultSetToEmployees called");

ArrayList<Employee> employees = new ArrayList<Employee>();

while (resultSet.next()) {

Employee employee = new Employee();

employee.setId(resultSet.getInt("id"));

employee.setName(resultSet.getString("name"));

employee.setAddress(resultSet.getString("address"));

employee.setHours(resultSet.getDouble("hours"));

employee.setRate(resultSet.getDouble("rate"));

String sex = resultSet.getString("sex");

if (sex != null && sex.length() > 0)

employee.setSex(sex.charAt(0));

employee.setAge(resultSet.getInt("age"));

employee.setActive(resultSet.getBoolean("active"));

employee.setSSN(resultSet.getString("ssn"));

employee.setPhone(resultSet.getString("phone"));

employees.add(employee);

}//while

return employees;

}

private void close(ResultSet resultSet, Statement statement, Connection connection) {

if (resultSet != null) {

try {

resultSet.close();

} catch (SQLException e) {

e.printStackTrace();

}

}

if (statement != null) {

try {

statement.close();

} catch (SQLException e) {

e.printStackTrace();

}

}

if (connection != null) {

try {

connection.close();

} catch (SQLException e) {

e.printStackTrace();

}

}

}

private Connection getConnection() throws Exception {

log.fine("getConnection called");

//c:\windows\syswow64\odbcad32

//ideally these connection would come from a connection pool...

// return DriverManager.getConnection("jdbc:odbc:employees");

//return DriverManager.getConnection("jdbc:odbc:MS ACCESS DataBase"+";DBQ=c:\\EmployeeManager\\employees.mdb");

return DriverManager.getConnection("jdbc:odbc:Driver={Microsoft Access Driver (\*.mdb)};DBQ=C:\\EmployeeManager\\employees.mdb");

}

}

C’est Finis!!!!!!