Tên: Phạm Dương Minh Nhật

Mã sinh viên: 19IT182

# **Lab 4**

## **StudentModel.cs**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace threelayers

{

internal class StudentModel

{

private string \_id;

private string \_name;

private string \_classid;

public StudentModel(string id, string name, string classid)

{

\_id = id;

\_name = name;

\_classid = classid;

}

public string Id { get => \_id; set => \_id = value; }

public string Name { get => \_name; set => \_name = value; }

public string Classid { get => \_classid; set => \_classid = value; }

}

}

## **DbConnection.cs**

using Npgsql;

using System;

using System.Collections.Generic;

using System.Data;

using System.Data.SqlClient;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Xml.Linq;

namespace threelayers

{

internal class dbConnection

{

private SqlConnection conn;

private SqlDataAdapter adapter;

private DataSet ds;

private string strconn = @"Data Source=SONNE;Initial Catalog=STUDENT\_MANAGERMENT;Integrated Security=true";

public dbConnection()

{

try

{

conn = new SqlConnection(strconn);

conn.Open();

MessageBox.Show("Succesful Connection!");

} catch (Exception e)

{

MessageBox.Show(e.Message);

}

}

public DataSet excuteQuery(string \_query)

{

ds = new DataSet();

try

{

adapter = new SqlDataAdapter(\_query, strconn);

SqlCommandBuilder builder = new SqlCommandBuilder(adapter);

ds = new DataSet();

adapter.Fill(ds, "Student");

}

catch (NpgsqlException e)

{

MessageBox.Show(e.ToString());

}

return ds;

}

public void insertStudent(string id, string name, string classid)

{

int result = 0;

DataRow row = ds.Tables[0].NewRow();

row[0] = id;

row[1] = name;

row[2] = classid;

ds.Tables[0].Rows.Add(row);

try

{

result = adapter.Update(ds, "Student");

}

catch (Exception ex)

{

MessageBox.Show($"Sth went wrong!!,{ex.ToString()}");

}

if (result > 0)

{

MessageBox.Show("Insert Successfully!");

}

else

{

MessageBox.Show("Insert Failled!");

}

}

public void updateStudent(string id, string name, string classid, int position)

{

if (position == -1)

{

MessageBox.Show("No row is selected!");

return;

}

DataRow row = ds.Tables[0].Rows[position];

row.BeginEdit();

row[0] = id;

row[1] = name;

row[2] = classid;

row.EndEdit();

int result = adapter.Update(ds.Tables[0]);

if (result > 0)

{

MessageBox.Show("Update Successfully!");

}

else

{

MessageBox.Show("Update Failled!");

}

}

public void deleteStudent(int position)

{

if (position == -1)

{

MessageBox.Show("No row is selected!"); return;

}

DataRow row = ds.Tables[0].Rows[position]; row.Delete();

int result = adapter.Update(ds.Tables[0]);

if (result > 0)

{

MessageBox.Show("Delete Successfully!");

}

else

{

MessageBox.Show("Delete Failled!");

}

}

}

}

## **StudentDAO.cs**

using System;

using System.Collections.Generic;

using System.Data.SqlClient;

using System.Data;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using Npgsql;

using NpgsqlTypes;

namespace threelayers.DAO

{

internal class StudentDao

{

private dbConnection conn;

public StudentDao()

{

conn = new dbConnection();

}

public DataSet getAll()

{

string query = "select \* from Student";

return conn.excuteQuery(query);

}

public void insertStudent(string id, string name, string classid)

{

conn.insertStudent(id,name,classid);

}

public void updateStudent(string id, string name, string classid, int position)

{

conn.updateStudent(id, name, classid, position);

}

public void deleteStudent(int position)

{

conn.deleteStudent(position);

}

}

}

## **StudentBUS.cs**

using System;

using System.Collections.Generic;

using System.Data;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using threelayers.DAO;

namespace threelayers.BUS

{

internal class StudentBus

{

private StudentDao \_stdDao;

public StudentBus()

{

\_stdDao = new StudentDao();

}

public DataSet getAll()

{

return \_stdDao.getAll();

}

public void insertStudent(string id, string name, string classid)

{

\_stdDao.insertStudent(id, name, classid);

}

public void updateStudent(string id, string name, string classid, int position)

{

\_stdDao.updateStudent(id, name, classid, position);

}

public void deleteStudent(int position)

{

\_stdDao.deleteStudent(position);

}

}

}

Graphical user interface

Description automatically generated