Лабораторная работа №28

25-ПОИТ

Выполнил: Олишкевич Игорь Русланович;

Тема: улучшение интерфейса программы;

Цель: сформировать умения создания объектов ADO.NET в Visual Studio;

Листинг main-формы:

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using MySql.Data.MySqlClient;

using Microsoft.Office.Interop.Excel;

using Word = Microsoft.Office.Interop.Word;

using Microsoft.Office;

using System.Reflection;

namespace WindowsFormsApp1

{

public partial class Form1 : Form

{

public MySqlConnection connection = new MySqlConnection("datasource=127.0.0.1;port=3306;username=root;password=");

public MySqlDataAdapter adapter;

public MySqlCommand command;

public int id\_teachers = 0;

public int id\_subjects = 0;

public int id\_studyLoad = 0;

public Form1()

{

InitializeComponent();

Teachers();

Subjects();

StudyLoad();

}

private void ClearTeachers()

{

textBox1.Clear();

textBox2.Clear();

textBox3.Clear();

textBox4.Clear();

textBox5.Clear();

textBox6.Clear();

textBox7.Clear();

}

private void ClearSubjects()

{

textBox8.Clear();

textBox9.Clear();

textBox10.Clear();

}

public void Teachers()

{

adapter = new MySqlDataAdapter("SELECT \* FROM DistributionOfTrainingLoad.Teachers", connection);

connection.Open();

DataSet ds = new DataSet();

adapter.Fill(ds, "Teachers");

connection.Close();

dataGridView1.DataSource = ds.Tables["Teachers"];

dataGridView1.Columns[0].HeaderText = "ID";

dataGridView1.Columns[1].HeaderText = "Код";

dataGridView1.Columns[2].HeaderText = "Фамилия";

dataGridView1.Columns[3].HeaderText = "Имя";

dataGridView1.Columns[4].HeaderText = "Отчество";

dataGridView1.Columns[5].HeaderText = "Ученая степень";

dataGridView1.Columns[6].HeaderText = "Должность";

dataGridView1.Columns[7].HeaderText = "Стаж";

comboBox2.DataSource = ds.Tables["Teachers"];

comboBox2.DisplayMember = "CodeTeachers";

comboBox2.ValueMember = "CodeTeachers";

}

public void Subjects()

{

adapter = new MySqlDataAdapter("SELECT \* FROM DistributionOfTrainingLoad.Subjects", connection);

connection.Open();

DataSet ds = new DataSet();

adapter.Fill(ds, "Subjects");

connection.Close();

dataGridView2.DataSource = ds.Tables["Subjects"];

dataGridView2.Columns[0].HeaderText = "ID";

dataGridView2.Columns[1].HeaderText = "Код";

dataGridView2.Columns[2].HeaderText = "Название";

dataGridView2.Columns[3].HeaderText = "Тип";

dataGridView2.Columns[4].HeaderText = "Часы";

comboBox3.DataSource = ds.Tables["Subjects"];

comboBox3.DisplayMember = "CodeSubjects";

comboBox3.ValueMember = "CodeSubjects";

}

public void StudyLoad()

{

adapter = new MySqlDataAdapter("SELECT ID\_StudyLoad, Name, Hours, NumberGroups FROM DistributionOfTrainingLoad.StudyLoad, " +

"DistributionOfTrainingLoad.Subjects, DistributionOfTrainingLoad.Teachers WHERE" +

" StudyLoad.CodeTeachers = Teachers.CodeTeachers AND StudyLoad.CodeSubjects = Subjects.CodeSubjects ORDER BY ID\_StudyLoad ASC", connection);

connection.Open();

DataSet ds = new DataSet();

adapter.Fill(ds, "StudyLoad");

connection.Close();

dataGridView3.DataSource = ds.Tables["StudyLoad"];

dataGridView3.Columns[0].HeaderText = "ID";

dataGridView3.Columns[1].HeaderText = "Предмет";

dataGridView3.Columns[2].HeaderText = "Часы";

dataGridView3.Columns[3].HeaderText = "Номер группы";

}

private void button1\_Click(object sender, EventArgs e)

{

if (textBox1.Text == "" || textBox2.Text == "" || textBox3.Text == "" || textBox4.Text == "" || textBox5.Text == "" || textBox6.Text == "" || textBox7.Text == "")

MessageBox.Show("Заполните все поля!");

else

{

command = new MySqlCommand("INSERT INTO DistributionOfTrainingLoad.Teachers(CodeTeachers, LastName, FirstName, SureName, AcademicDegree, Position, Experience) " +

"VALUES (" + textBox1.Text + ", '" + textBox2.Text + "', '" + textBox3.Text + "', '" + textBox4.Text + "', '" + textBox5.Text + "', '" +

textBox6.Text + "', " + textBox7.Text + ")", connection);

connection.Open();

command.ExecuteNonQuery();

connection.Close();

Teachers();

ClearTeachers();

MessageBox.Show("Запись добавлена!", "Успешно!", MessageBoxButtons.OK);

}

}

private void textBox1\_KeyPress(object sender, KeyPressEventArgs e)

{

char number = e.KeyChar;

if (!Char.IsDigit(number))

e.Handled = true;

}

private void dataGridView1\_CurrentCellChanged(object sender, EventArgs e)

{

try

{

id\_teachers = Convert.ToInt32(dataGridView1.CurrentRow.Cells[0].Value);

textBox1.Text = dataGridView1.CurrentRow.Cells[1].Value.ToString();

textBox2.Text = dataGridView1.CurrentRow.Cells[2].Value.ToString();

textBox3.Text = dataGridView1.CurrentRow.Cells[3].Value.ToString();

textBox4.Text = dataGridView1.CurrentRow.Cells[4].Value.ToString();

textBox5.Text = dataGridView1.CurrentRow.Cells[5].Value.ToString();

textBox6.Text = dataGridView1.CurrentRow.Cells[6].Value.ToString();

textBox7.Text = dataGridView1.CurrentRow.Cells[7].Value.ToString();

}

catch(Exception)

{ }

}

private void button2\_Click(object sender, EventArgs e)

{

if (id\_teachers == 0)

MessageBox.Show("Вы ничего не выбрали!");

else

{

if (textBox1.Text == "" || textBox2.Text == "" || textBox3.Text == "" || textBox4.Text == "" || textBox5.Text == "" || textBox6.Text == "" || textBox7.Text == "")

MessageBox.Show("Заполните все поля!");

else

{

command = new MySqlCommand("UPDATE DistributionOfTrainingLoad.Teachers SET CodeTeachers = " + textBox1.Text + ", LastName = '" + textBox2.Text +

"', FirstName = '" + textBox3.Text + "', SureName = '" + textBox4.Text + "', AcademicDegree = '" + textBox5.Text + "', Position = '" +

textBox6.Text + "', Experience = " + textBox7.Text + " WHERE ID\_Teachers = " + id\_teachers, connection);

connection.Open();

command.ExecuteNonQuery();

connection.Close();

Teachers();

id\_teachers = 0;

ClearTeachers();

MessageBox.Show("Запись изменена!", "Успешно!", MessageBoxButtons.OK);

}

}

}

private void button4\_Click(object sender, EventArgs e)

{

ClearTeachers();

}

private void button3\_Click(object sender, EventArgs e)

{

if (id\_teachers == 0)

MessageBox.Show("Вы ничего не выбрали!");

else

{

command = new MySqlCommand("DELETE FROM DistributionOfTrainingLoad.Teachers WHERE ID\_Teachers = '" + id\_teachers + "'", connection);

connection.Open();

command.ExecuteNonQuery();

connection.Close();

Teachers();

id\_teachers = 0;

ClearTeachers();

MessageBox.Show("Запись удалена!", "Успешно!", MessageBoxButtons.OK);

}

}

private void button5\_Click(object sender, EventArgs e)

{

ClearSubjects();

}

private void dataGridView2\_CurrentCellChanged(object sender, EventArgs e)

{

try

{

id\_subjects = Convert.ToInt32(dataGridView2.CurrentRow.Cells[0].Value);

textBox8.Text = dataGridView2.CurrentRow.Cells[1].Value.ToString();

textBox9.Text = dataGridView2.CurrentRow.Cells[2].Value.ToString();

comboBox1.Text = dataGridView2.CurrentRow.Cells[3].Value.ToString();

textBox10.Text = dataGridView2.CurrentRow.Cells[4].Value.ToString();

}

catch (Exception)

{ }

}

private void button8\_Click(object sender, EventArgs e)

{

if (textBox8.Text == "" || textBox9.Text == "" || textBox10.Text == "")

MessageBox.Show("Заполните все поля!");

else

{

command = new MySqlCommand("INSERT INTO DistributionOfTrainingLoad.Subjects(CodeSubjects, Name, Type, Hours) " +

"VALUES (" + textBox8.Text + ", '" + textBox9.Text + "', '" + comboBox1.Text + "', " + textBox10.Text + ")", connection);

connection.Open();

command.ExecuteNonQuery();

connection.Close();

Subjects();

ClearSubjects();

MessageBox.Show("Запись добавлена!", "Успешно!", MessageBoxButtons.OK);

}

}

private void button7\_Click(object sender, EventArgs e)

{

if (id\_subjects == 0)

MessageBox.Show("Вы ничего не выбрали!");

else

{

if (textBox8.Text == "" || textBox9.Text == "" || textBox10.Text == "")

MessageBox.Show("Заполните все поля!");

else

{

command = new MySqlCommand("UPDATE DistributionOfTrainingLoad.Subjects SET CodeSubjects = " + textBox8.Text + ", Name = '" + textBox9.Text +

"', Type = '" + comboBox1.Text + "', Hours = " + textBox10.Text + " WHERE ID\_Subjects = " + id\_subjects, connection);

connection.Open();

command.ExecuteNonQuery();

connection.Close();

Subjects();

id\_subjects = 0;

ClearSubjects();

MessageBox.Show("Запись изменена!", "Успешно!", MessageBoxButtons.OK);

}

}

}

private void button6\_Click(object sender, EventArgs e)

{

if (id\_subjects == 0)

MessageBox.Show("Вы ничего не выбрали!");

else

{

command = new MySqlCommand("DELETE FROM DistributionOfTrainingLoad.Subjects WHERE ID\_Subjects = '" + id\_subjects + "'", connection);

connection.Open();

command.ExecuteNonQuery();

connection.Close();

Subjects();

id\_subjects = 0;

ClearSubjects();

MessageBox.Show("Запись удалена!", "Успешно!", MessageBoxButtons.OK);

}

}

private void dataGridView3\_CurrentCellChanged(object sender, EventArgs e)

{

try

{

id\_studyLoad = Convert.ToInt32(dataGridView3.CurrentRow.Cells[0].Value);

comboBox2.Text = dataGridView3.CurrentRow.Cells[1].Value.ToString();

comboBox3.Text = dataGridView3.CurrentRow.Cells[2].Value.ToString();

textBox11.Text = dataGridView3.CurrentRow.Cells[3].Value.ToString();

}

catch (Exception)

{ }

}

private void button9\_Click(object sender, EventArgs e)

{

textBox11.Clear();

}

private void button12\_Click(object sender, EventArgs e)

{

if (textBox11.Text == "")

MessageBox.Show("Заполните номер группы!");

else

{

command = new MySqlCommand("INSERT INTO DistributionOfTrainingLoad.StudyLoad(CodeTeachers, CodeSubjects, NumberGroups) " +

"VALUES (" + comboBox2.Text + ", " + comboBox3.Text + ", " + textBox11.Text + ")", connection);

connection.Open();

command.ExecuteNonQuery();

connection.Close();

StudyLoad();

textBox11.Clear();

MessageBox.Show("Запись добавлена!", "Успешно!", MessageBoxButtons.OK);

}

}

private void button11\_Click(object sender, EventArgs e)

{

if (id\_studyLoad == 0)

MessageBox.Show("Вы ничего не выбрали!");

else

{

if (textBox11.Text == "")

MessageBox.Show("Заполните номер группы!");

else

{

command = new MySqlCommand("UPDATE DistributionOfTrainingLoad.StudyLoad SET CodeTeachers = " + comboBox2.Text +

", CodeSubjects = " + comboBox3.Text + ", NumberGroups = " + textBox11.Text + " WHERE ID\_StudyLoad = " + id\_studyLoad, connection);

connection.Open();

command.ExecuteNonQuery();

connection.Close();

StudyLoad();

id\_studyLoad = 0;

textBox11.Clear();

MessageBox.Show("Запись изменена!", "Успешно!", MessageBoxButtons.OK);

}

}

}

private void button10\_Click(object sender, EventArgs e)

{

if (id\_studyLoad == 0)

MessageBox.Show("Вы ничего не выбрали!");

else

{

command = new MySqlCommand("DELETE FROM DistributionOfTrainingLoad.StudyLoad WHERE ID\_StudyLoad = '" + id\_studyLoad + "'", connection);

connection.Open();

command.ExecuteNonQuery();

connection.Close();

StudyLoad();

id\_studyLoad = 0;

textBox11.Clear();

MessageBox.Show("Запись удалена!", "Успешно!", MessageBoxButtons.OK);

}

}

private void newToolStripMenuItem\_Click\_1(object sender, EventArgs e)

{

Microsoft.Office.Interop.Excel.Application ExcelApp = new Microsoft.Office.Interop.Excel.Application();

Microsoft.Office.Interop.Excel.Workbook ExcelWorkBook;

Microsoft.Office.Interop.Excel.Worksheet ExcelWorkSheet;

ExcelWorkBook = ExcelApp.Workbooks.Add(System.Reflection.Missing.Value);

ExcelWorkSheet = (Microsoft.Office.Interop.Excel.Worksheet)ExcelWorkBook.Worksheets.get\_Item(1);

if (tabControl1.SelectedIndex == 0)

{

for (int i = 0; i < dataGridView1.Rows.Count; i++)

{

for (int j = 0; j < dataGridView1.ColumnCount; j++)

{

ExcelApp.Cells[i + 1, j + 1] = dataGridView1.Rows[i].Cells[j].Value;

}

}

}

else if (tabControl1.SelectedIndex == 1)

{

for (int i = 0; i < dataGridView2.Rows.Count; i++)

{

for (int j = 0; j < dataGridView2.ColumnCount; j++)

{

ExcelApp.Cells[i + 1, j + 1] = dataGridView2.Rows[i].Cells[j].Value;

}

}

}

else if (tabControl1.SelectedIndex == 2)

{

for (int i = 0; i < dataGridView3.Rows.Count; i++)

{

for (int j = 0; j < dataGridView3.ColumnCount; j++)

{

ExcelApp.Cells[i + 1, j + 1] = dataGridView3.Rows[i].Cells[j].Value;

}

}

}

ExcelApp.Visible = true;

ExcelApp.UserControl = true;

}

private void exitToolStripMenuItem\_Click(object sender, EventArgs e)

{

System.Windows.Forms.Application.Exit();

}

private void openToolStripMenuItem\_Click\_1(object sender, EventArgs e)

{

SaveFileDialog sfd = new SaveFileDialog();

sfd.Filter = "Word Documents (\*.docx)|\*.docx";

sfd.FileName = "export.docx";

if (sfd.ShowDialog() == DialogResult.OK)

{

if (tabControl1.SelectedIndex == 0)

Export\_Data\_To\_Word(dataGridView1, sfd.FileName);

else if (tabControl1.SelectedIndex == 1)

Export\_Data\_To\_Word(dataGridView2, sfd.FileName);

else if (tabControl1.SelectedIndex == 2)

Export\_Data\_To\_Word(dataGridView3, sfd.FileName);

}

}

public void Export\_Data\_To\_Word(DataGridView DGV, string filename)

{

if (DGV.Rows.Count != 0)

{

int RowCount = DGV.Rows.Count;

int ColumnCount = DGV.Columns.Count;

Object[,] DataArray = new object[RowCount + 1, ColumnCount + 1];

int r = 0;

for (int c = 0; c <= ColumnCount - 1; c++)

for (r = 0; r <= RowCount - 1; r++)

DataArray[r, c] = DGV.Rows[r].Cells[c].Value;

Word.Document oDoc = new Word.Document();

oDoc.Application.Visible = true;

oDoc.PageSetup.Orientation = Word.WdOrientation.wdOrientLandscape;

dynamic oRange = oDoc.Content.Application.Selection.Range;

string oTemp = "";

for (r = 0; r <= RowCount - 1; r++)

for (int c = 0; c <= ColumnCount - 1; c++)

oTemp = oTemp + DataArray[r, c] + "\t";

oRange.Text = oTemp;

object oMissing = Missing.Value;

object Separator = Word.WdTableFieldSeparator.wdSeparateByTabs;

object ApplyBorders = true;

object AutoFit = true;

object AutoFitBehavior = Word.WdAutoFitBehavior.wdAutoFitContent;

oRange.ConvertToTable(ref Separator, ref RowCount, ref ColumnCount,

Type.Missing, Type.Missing, ref ApplyBorders,

Type.Missing, Type.Missing, Type.Missing,

Type.Missing, Type.Missing, Type.Missing,

Type.Missing, ref AutoFit, ref AutoFitBehavior, Type.Missing);

oRange.Select();

oDoc.Application.Selection.Tables[1].Select();

oDoc.Application.Selection.Tables[1].Rows.AllowBreakAcrossPages = 0;

oDoc.Application.Selection.Tables[1].Rows.Alignment = 0;

oDoc.Application.Selection.Tables[1].Rows[1].Select();

oDoc.Application.Selection.InsertRowsAbove(1);

oDoc.Application.Selection.Tables[1].Rows[1].Select();

oDoc.Application.Selection.Tables[1].Rows[1].Range.Bold = 1;

oDoc.Application.Selection.Tables[1].Rows[1].Range.Font.Name = "Tahoma";

oDoc.Application.Selection.Tables[1].Rows[1].Range.Font.Size = 14;

for (int c = 0; c <= ColumnCount - 1; c++)

{

oDoc.Application.Selection.Tables[1].Cell(1, c + 1).Range.Text = DGV.Columns[c].HeaderText;

}

oDoc.Application.Selection.Tables[1].Rows[1].Select();

oDoc.Application.Selection.Cells.VerticalAlignment = Word.WdCellVerticalAlignment.wdCellAlignVerticalCenter;

foreach (Word.Section section in oDoc.Application.ActiveDocument.Sections)

{

Word.Range headerRange = section.Headers[Word.WdHeaderFooterIndex.wdHeaderFooterPrimary].Range;

headerRange.Fields.Add(headerRange, Word.WdFieldType.wdFieldPage);

headerRange.Text = tabControl1.SelectedTab.Text;

headerRange.Font.Size = 16;

headerRange.ParagraphFormat.Alignment = Word.WdParagraphAlignment.wdAlignParagraphCenter;

}

oDoc.SaveAs(filename, ref oMissing, ref oMissing, ref oMissing,

ref oMissing, ref oMissing, ref oMissing, ref oMissing, ref oMissing,

ref oMissing, ref oMissing, ref oMissing, ref oMissing, ref oMissing,

ref oMissing, ref oMissing);

}

}

private void textSettingsToolStripMenuItem\_Click\_1(object sender, EventArgs e)

{

AboutBox1 aboutBox = new AboutBox1();

aboutBox.ShowDialog(this);

}

}

}

Листинг пользовательского компонента «О программе»:

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Drawing;

using System.Linq;

using System.Reflection;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace WindowsFormsApp1

{

partial class AboutBox1 : Form

{

public AboutBox1()

{

InitializeComponent();

this.Text = String.Format("О программе {0}", AssemblyTitle);

this.labelProductName.Text = AssemblyTitle;

this.labelVersion.Text = String.Format("Версия {0}", AssemblyVersion);

this.labelCopyright.Text = "Олишкевич И.Р. - " + AssemblyCopyright;

this.labelCompanyName.Text = AssemblyCompany + "ГКБиП";

this.textBoxDescription.Text = AssemblyDescription;

}

#region Методы доступа к атрибутам сборки

public string AssemblyTitle = "Учебная нагрузка";

public string AssemblyVersion

{

get

{

return Assembly.GetExecutingAssembly().GetName().Version.ToString();

}

}

public string AssemblyDescription

{

get

{

object[] attributes = Assembly.GetExecutingAssembly().GetCustomAttributes(typeof(AssemblyDescriptionAttribute), false);

if (attributes.Length == 0)

{

return "";

}

return ((AssemblyDescriptionAttribute)attributes[0]).Description;

}

}

public string AssemblyProduct

{

get

{

object[] attributes = Assembly.GetExecutingAssembly().GetCustomAttributes(typeof(AssemblyProductAttribute), false);

if (attributes.Length == 0)

{

return "";

}

return ((AssemblyProductAttribute)attributes[0]).Product;

}

}

public string AssemblyCopyright

{

get

{

object[] attributes = Assembly.GetExecutingAssembly().GetCustomAttributes(typeof(AssemblyCopyrightAttribute), false);

if (attributes.Length == 0)

{

return "";

}

return ((AssemblyCopyrightAttribute)attributes[0]).Copyright;

}

}

public string AssemblyCompany

{

get

{

object[] attributes = Assembly.GetExecutingAssembly().GetCustomAttributes(typeof(AssemblyCompanyAttribute), false);

if (attributes.Length == 0)

{

return "";

}

return ((AssemblyCompanyAttribute)attributes[0]).Company;

}

}

#endregion

}

}

Тестирование:

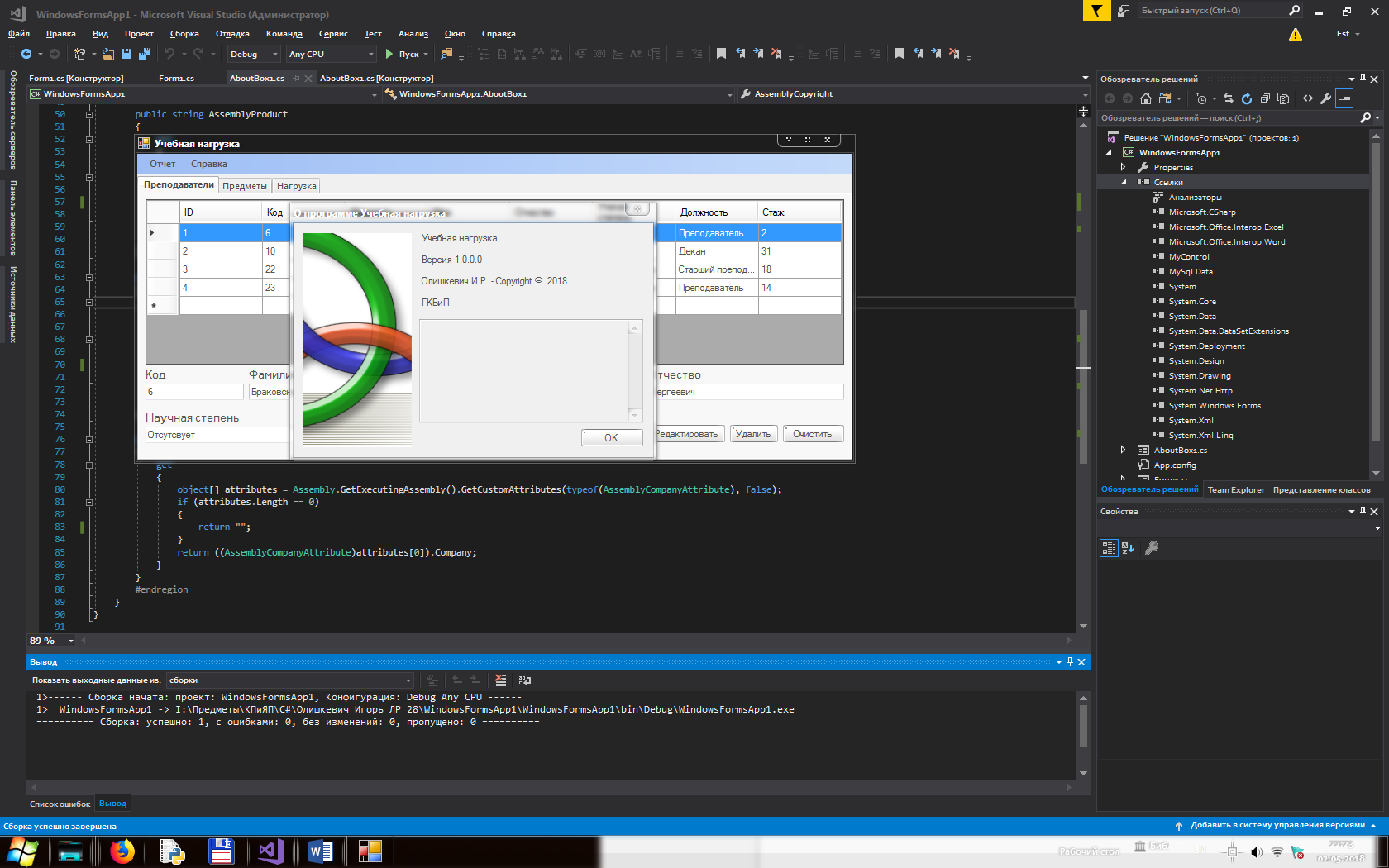


Рисунок 1 – Готовое приложение