Федеральное агентство связи

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Сибирский государственный университет

телекоммуникаций и информатики»

кафедра ПМ и К

РГР ПО ВИЗУАЛЬНОМУ ПРОГРАММИРОВАНИЮ И ЧЕЛОВЕКО-МАШИННОМУ ВЗАИМОДЕЙСТВИЮ

Вариант 4

«Автошкола»

Выполнил: студент гр. ИП-814

Якунин А. В.

Проверил: ассистент кафедры ПМ и К

Суходоева Н. Н.

Новосибирск 2020г.

**Содержание**

1. Задание 3

2. Описание структуры баз данных 4

3.Описание работы приложения 5

4.Распечатки отчетов 7

5.Скриншоты работы программы 8

6.Вывод 15

7.Листинг программы 16

**1.Задание**

Написать программу для Автошколы, реализующую следующие функции.

Программа должна содержать:

* данные о группе: название, дата начала занятий, дата окончания занятий, ФИО преподавателя ПДД, ФИО преподавателей вождения (5 человек), Гос. номер и марку автомобилей (5шт.)
* данные о курсантах: ФИО, ФИО в дательном падеже (для печати свидетельства), паспортные данные, место работы, адрес места работы, № медицинской справки, дата ее выдачи и наименование мед. учреждения, № и дата протокола сдачи выпускного экзамена, № и дата выданного свидетельства.

Программа должна:

* помнить данные автошколы: название, ФИО директора, реквизиты ГИБДД, фамилию начальника отдела ГИБДД, в которой курсанты автошколы сдают экзамен;
* позволять выбирать ФИО преподавателей, Гос. номер и марки автомобилей.
* Осуществлять поиск в базе курсантов по ФИО, номеру протокола и номеру выданного свидетельства.
* Печать заявления на имя директора автошколы на принятие курсанта (Текущая дата и ФИО директора - из базы) - П1.
* Печать заявки на регистрацию учебных групп (название и адрес автошколы, ФИО директора – из базы) - П2.
* Печать Приказа о зачислении учащихся (номер и дату приказа – вводить, название автошколы, ФИО директора, список группы, преподавателей, гос. номер и марки машин– из базы) – П3.
* Печать Карточки вождения. (Название автошколы и ФИО курсанта – из базы) – П5.
* Печать платежного поручения за сдачу экзамена в ГИБДД: извещение и квитанция (ФИО и адрес курсанта – из базы) – П6
* Печать экзаменационной карточки по ПДД (ФИО, дата рождения и категория – из базы) – П7
* Печать экзаменационного листа (ФИО, дата рождения и категория – из базы) – П9
* Печать карточки – результаты экзаменов, виды удостоверения – П12

**2.Описание структуры баз данных**

База данных состоит из 2 таблиц: Группа, Курсант

Группа:

ID

Название

Дата начало занятий

Дата окончания занятий

ФИО преподавателей ПДД

ГОС номер

Марка автомобиля

Курсант:

ID\_K

ФИО

ФИО в дательном падеже

Паспортные данные

Место работы

Адрес места работы

№ медицинской справки

Дата выдачи медицинской справки

Наименование мед учреждения

№ протокола сдачи выпускного экзамена

Дата сдачи выпускного экзамена

Дата выданного свидетельства

№ выданного свидетельства

База данных состоит из 1 таблицы: Автошкола

Директор

Адрес

ФИО

Название

Сформированных связей между таблицами нет; выбор определенных данных происходит в запросах к базе.

**3.Описание работы приложения**

При запуске приложения открывается главное окно с двумя таблицами данных. С верху располагается меню, в котором находится кнопки About, Автошкола и Печать (где находятся все разновидности печати). В низу окна находится 4 текстовых поля, которые нужны для ввода ID\_K курсанта и с помощью оставшихся 3 текстовых полей можно найти соответствующих курсантов. И так же имеются 3 кнопки запросов, при нажатии на которых, будет выполнен запрос, а результат выведется в таблицу.

Пункты меню:

Меню ->

About – выводит сообщение об авторе данной программы

Автошкола – выводит данные об автошколе

Печать ->

1.Печать заявления на имя директора автошколы на принятие курсанта - перед печатью, заполните поле ID\_K

2.Печать заявки на регистрацию учебных групп – перед печатью ничего заполнять не нужно

3.Печать Приказа о зачислении учащихся – перед печатью заполните поля Номер приказа и Дата приказа

5.Печать Карточки вождения – перед печатью, заполните поле ID\_K

6.Печать платежного поручения за сдачу экзамена в ГИБДД: извещение и квитанция - перед печатью, заполните поле ID\_K

7.Печать экзаменационной карточки по ПДД - перед печатью заполните ID\_K

9,Печать экзаменационного листа – перед печатью заполните поле ID\_K

12.Печать карточки – перед печатью заполните поле ID\_K

Текстовое поле «Номер приказа» - служит для ввода Номера приказа

Текстовое поле «Дата приказа» - служит для ввода Даты приказа

Текстовое поле «ID\_K» - служит для ввода ID\_K таблицы курсанта

Текстовое поле «Поиск по номеру выданного свидетельства» - служит для ввода № выданного свидетельства курсанта

Текстовое поле «Поиск по номеру протокола» - служит для ввода № протокола сдачи выпускного экзамена курсанта

Текстовое поле «Поиск по ФИО» - служит для ввода ФИО курсанта

Кнопка «Обновить» - выполняется запрос в таблицу, результатом которого является вывод все таблицы

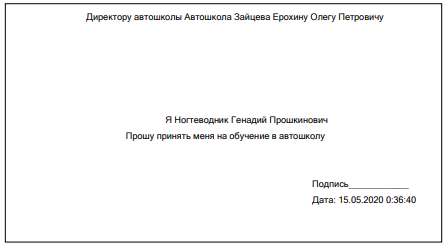
Кнопка «Преподаватель» - выполняется запрос в таблицу, результатом которого является вывод столбца преподавателей

Кнопка «Гос номер» - выполняется запрос в таблицу, результатом которого является вывод столбца ГОС номер

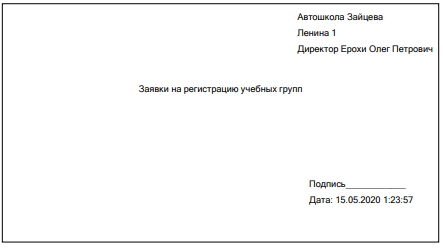
Кнопка «Марка» - выполняется запрос в таблицу, результатом которого является вывод столбца Марка автомобиля

**4.Распечатки отчетов**

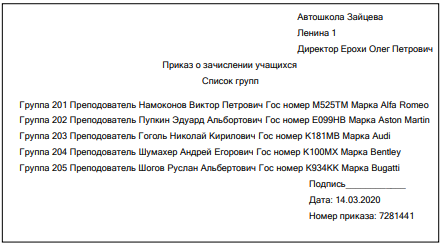
1.Печать заявления на имя директора автошколы на принятие курсанта



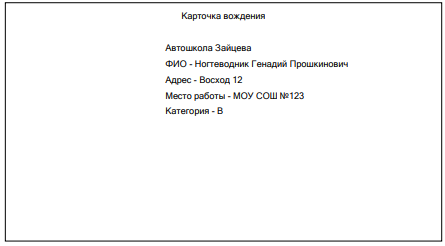
2.Печать заявки на регистрацию учебных групп



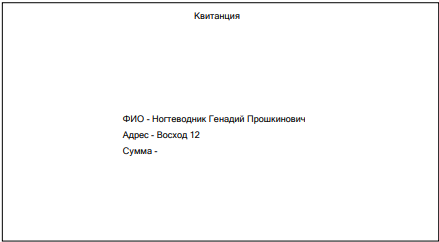
3.Печать Приказа о зачислении учащихся



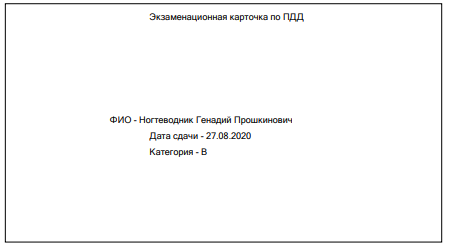
5.Печать Карточки вождения



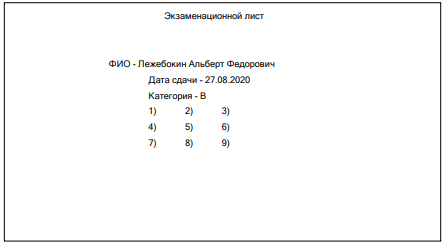
6.Печать платежного поручения за сдачу экзамена в ГИБДД: извещение и квитанция



7.Печать экзаменационной карточки по ПДД



9, Печать экзаменационного листа



12.Печать карточки



**5.Скриншоты работы программы**

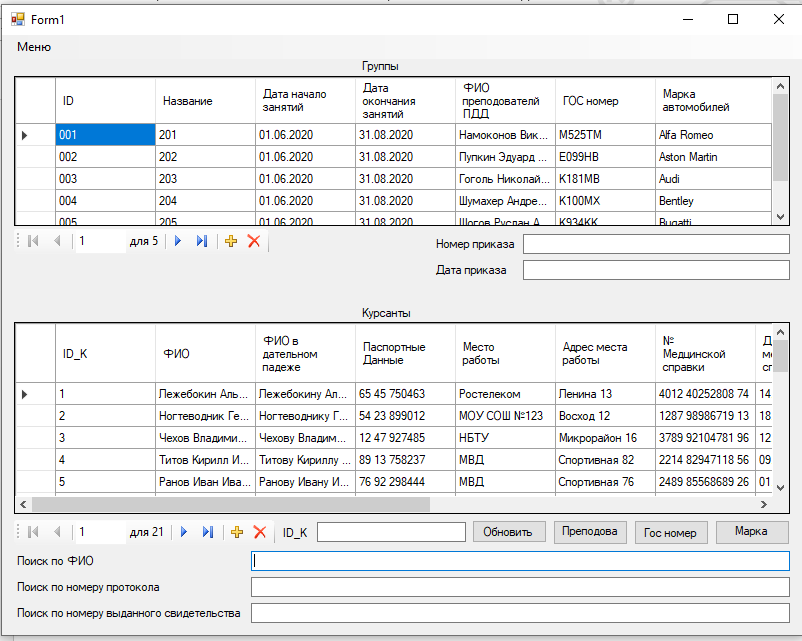


Рис 1(Начало работы программы)

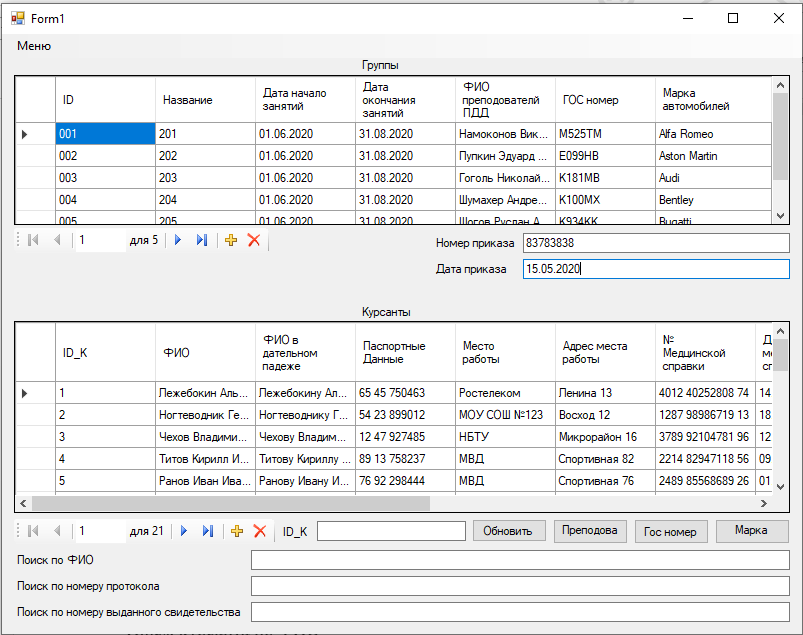


Рис 2(Заполнение «Номер приказа» и «Дата приказа»)

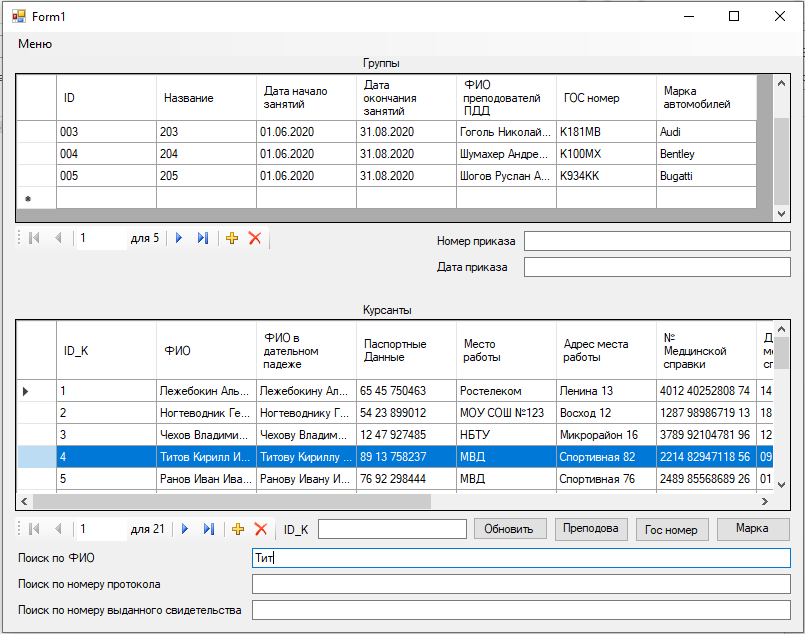


Рис 3 (Ищем курсанта по ФИО)

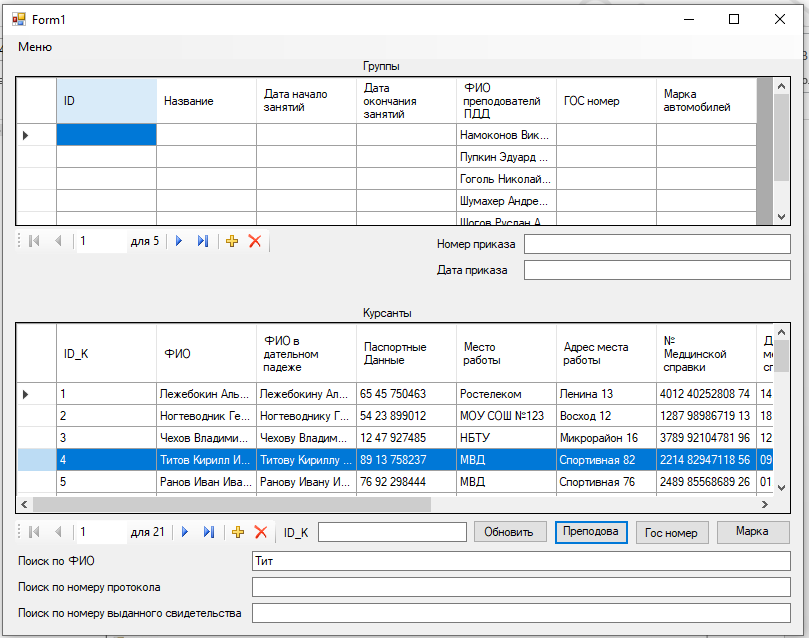


Рис 4(Выполняем запрос «Преподаватель»)

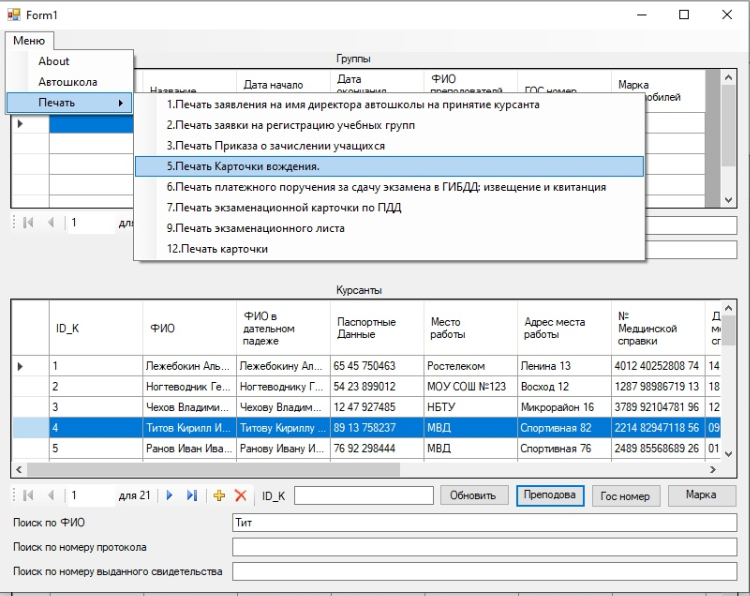


Рис 5(Выбираем интересующий нас пункт печати)

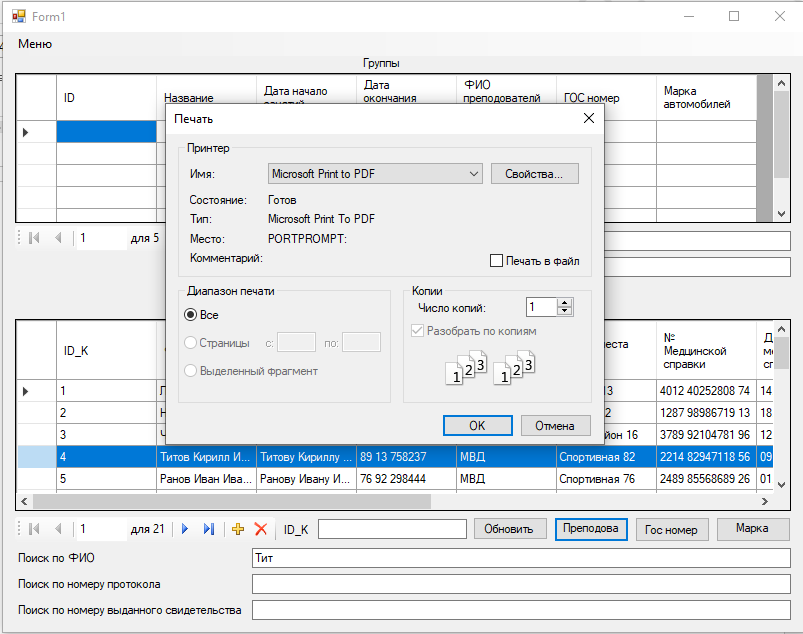


Рис 7(Выбираем формат печати)

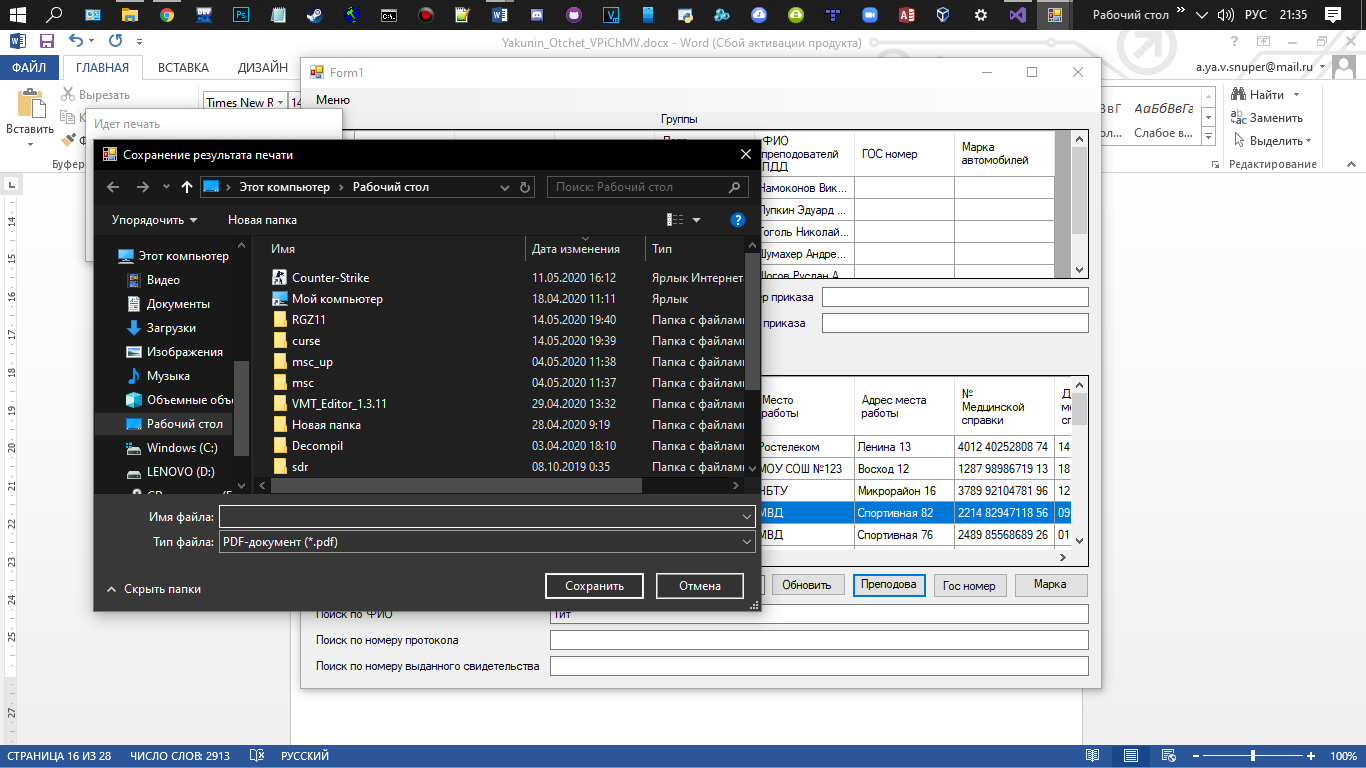


Рис 8(Выбираем имя файла)

**6.Вывод**

В данной РГР была выполнена работа с базой данных: составление таблиц, загрузка базы в приложение, выполнение запросов.

Выполнены: поиск данных по таблице, создание запросов, составление документа.

В приложении были реализованы следующие пункты:

* поиск
* составление документа и его последующая печать
* запросы

К приложению были созданы 2 базы данных, состоящая в совокупности из трех таблиц.

**7.Листинг программы**

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 System.IO;

using System.Drawing.Printing;

namespace WindowsFormsApp1

{

public partial class Form1 : Form

{

string temp\_text1, temp\_text2, temp\_text3;

string temp1;

string temp2;

string save\_data;

public Form1()

{

InitializeComponent();

}

// Declare a string to hold the entire document contents.

private string documentContents;

// Declare a variable to hold the portion of the document that

// is not printed.

private string stringToPrint;

private void Form1\_Load(object sender, EventArgs e)

{

// TODO: данная строка кода позволяет загрузить данные в таблицу "mainDataSet.Автошкола". При необходимости она может быть перемещена или удалена.

this.автошколаTableAdapter.Fill(this.mainDataSet.Автошкола);

// TODO: данная строка кода позволяет загрузить данные в таблицу "dATAS.Курсант". При необходимости она может быть перемещена или удалена.

this.курсантTableAdapter.Fill(this.dATAS.Курсант);

// TODO: данная строка кода позволяет загрузить данные в таблицу "dATAS.Группа". При необходимости она может быть перемещена или удалена.

this.группаTableAdapter.Fill(this.dATAS.Группа);

}

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

{

MessageBox.Show("Якунин Андрей ИП-814");

}

private void textBox1\_TextChanged(object sender, EventArgs e)

{

string str;

string temp;

int indexOfChar = -1;

for (int i = 0; i < dataGridView2.RowCount - 1; i++)

{

str = dataGridView2.Rows[i].Cells[1].Value.ToString();

temp = textBox1.Text;

indexOfChar = str.IndexOf(temp);

if (indexOfChar >= 0)

{

dataGridView2.Rows[i].Selected = true;

indexOfChar = -1;

}

else dataGridView2.Rows[i].Selected = false;

}

if (textBox1.Text == "") for (int i = 0; i < dataGridView2.RowCount - 1; i++) dataGridView2.Rows[i].Selected = false;

}

private void textBox2\_TextChanged(object sender, EventArgs e)

{

string str;

string temp;

int indexOfChar = -1;

for (int i = 0; i < dataGridView2.RowCount - 1; i++)

{

str = dataGridView2.Rows[i].Cells[9].Value.ToString();

temp = textBox2.Text;

indexOfChar = str.IndexOf(temp);

if (indexOfChar >= 0)

{

dataGridView2.Rows[i].Selected = true;

indexOfChar = -1;

}

else dataGridView2.Rows[i].Selected = false;

}

if (textBox2.Text == "") for (int i = 0; i < dataGridView2.RowCount - 1; i++) dataGridView2.Rows[i].Selected = false;

}

private void textBox3\_TextChanged(object sender, EventArgs e)

{

string str;

string temp;

int indexOfChar = -1;

for (int i = 0; i < dataGridView2.RowCount - 1; i++)

{

str = dataGridView2.Rows[i].Cells[12].Value.ToString();

temp = textBox3.Text;

indexOfChar = str.IndexOf(temp);

if (indexOfChar >= 0)

{

dataGridView2.Rows[i].Selected = true;

indexOfChar = -1;

}

else dataGridView2.Rows[i].Selected = false;

}

if (textBox3.Text == "") for (int i = 0; i < dataGridView2.RowCount - 1; i++) dataGridView2.Rows[i].Selected = false;

}

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

{

try

{

this.группаTableAdapter.FillBy12(this.dATAS.Группа);

}

catch (System.Exception ex)

{

System.Windows.Forms.MessageBox.Show(ex.Message);

}

}

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

{

try

{

this.группаTableAdapter.FillBy1(this.dATAS.Группа);

}

catch (System.Exception ex)

{

System.Windows.Forms.MessageBox.Show(ex.Message);

}

}

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

{

try

{

this.группаTableAdapter.FillBy13(this.dATAS.Группа);

}

catch (System.Exception ex)

{

System.Windows.Forms.MessageBox.Show(ex.Message);

}

}

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

{

MessageBox.Show("Автошкола Зайцева\nДиректор - Ерохи Олег Петрович\nРеквизиты - 182 1 06 04011 02 1000 110\nНачальник отдела ГИБДД - Песков Юрий Андреевич");

}

private void dataGridView2\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

}

private void bindingNavigator1\_RefreshItems(object sender, EventArgs e)

{

}

private void dataGridView1\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

}

private void bindingNavigator2\_RefreshItems(object sender, EventArgs e)

{

}

private void toolStripLabel1\_Click(object sender, EventArgs e)

{

}

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

{

MessageBox.Show("Якунин Андрей ИП-814");

}

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

{

MessageBox.Show("Автошкола Зайцева\nДиректор - Ерохин Олег Петрович\nАдрес - Ленина 1\nРеквизиты - 182 1 06 04011 02 1000 110\nНачальник отдела ГИБДД - Песков Юрий Андреевич");

}

private void ReadDocument()

{

string docName = "direcotr\_name.txt";

printDocument2.DocumentName = docName;

using (FileStream stream = new FileStream(docName, FileMode.Open))

using (StreamReader reader = new StreamReader(stream))

{

documentContents = reader.ReadToEnd();

}

stringToPrint = documentContents;

}

private void printDocument2\_PrintPage(object sender, PrintPageEventArgs e)

{

int charactersOnPage = 0;

int linesPerPage = 0;

e.Graphics.MeasureString(stringToPrint, this.Font,

e.MarginBounds.Size, StringFormat.GenericTypographic,

out charactersOnPage, out linesPerPage);

e.Graphics.DrawString(stringToPrint, this.Font, Brushes.Black,

e.MarginBounds, StringFormat.GenericTypographic);

stringToPrint = stringToPrint.Substring(charactersOnPage);

e.HasMorePages = (stringToPrint.Length > 0);

if (!e.HasMorePages)

stringToPrint = documentContents;

}

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

{

}

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

{

PrintDocument printDocument = new PrintDocument();

// обработчик события печати

printDocument.PrintPage += PrintPageHandler1;

// диалог настройки печати

PrintDialog printDialog = new PrintDialog();

// установка объекта печати для его настройки

printDialog.Document = printDocument;

// если в диалоге было нажато ОК

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

printDialog.Document.Print(); // печатаем

}

private void PrintPageHandler1(object sender, PrintPageEventArgs e)

{

e.Graphics.FillRectangle(Brushes.White, 100, 50, 550, 300);

e.Graphics.DrawRectangle(Pens.Black, 100, 50, 550, 300);

for (int i = 0; i < dataGridView2.RowCount - 1; i++)

{

if (dataGridView2.Rows[i].Cells[3].Value.ToString() == textBox3.Text)

{

save\_data = dataGridView2.Rows[i].Cells[1].Value.ToString();

break;

}

}

for (int i = 0; i < dataGridView2.Rows.Count - 1; i++)

{

if (dataGridView2.Rows[i].Selected == true)

{

e.Graphics.DrawString("Директору автошколы " + dataGridView3.Rows[0].Cells[3].Value.ToString() + " " + dataGridView3.Rows[0].Cells[2].Value.ToString(), dataGridView3.Font, Brushes.Black, 200, 60);

e.Graphics.DrawString("Я" + " " + dataGridView2.Rows[i].Cells[1].Value.ToString(), dataGridView2.Font, Brushes.Black, 300, 190);

e.Graphics.DrawString("Прошу принять меня на обучение в автошколу", dataGridView2.Font, Brushes.Black, 250, 210);

e.Graphics.DrawString(save\_data, dataGridView2.Font, Brushes.Black, 250, 210);

e.Graphics.DrawString("Подпись\_\_\_\_\_\_\_\_\_\_\_\_", dataGridView2.Font, Brushes.Black, 485, 270);

e.Graphics.DrawString("Дата: " + DateTime.Now, dataGridView2.Font, Brushes.Black, 485, 290);

}

}

}

private void printPreviewDialog2\_Load(object sender, EventArgs e)

{

}

private void textBox4\_TextChanged(object sender, EventArgs e)

{

for (int i = 0; i < dataGridView2.RowCount - 1; i++)

{

string str = dataGridView2.Rows[i].Cells[0].Value.ToString();

if (str == textBox4.Text) dataGridView2.Rows[i].Selected = true;

else dataGridView2.Rows[i].Selected = false;

}

}

private void PrintPageHandler2(object sender, PrintPageEventArgs e)

{

e.Graphics.FillRectangle(Brushes.White, 100, 50, 550, 300);

e.Graphics.DrawRectangle(Pens.Black, 100, 50, 550, 300);

for (int i = 0; i < dataGridView2.Rows.Count - 1; i++)

{

if (dataGridView2.Rows[i].Selected == true)

{

e.Graphics.DrawString(dataGridView3.Rows[0].Cells[3].Value.ToString(), dataGridView3.Font, Brushes.Black, 470, 60);

e.Graphics.DrawString(dataGridView3.Rows[0].Cells[1].Value.ToString(), dataGridView3.Font, Brushes.Black, 470, 80);

e.Graphics.DrawString("Директор " + dataGridView3.Rows[0].Cells[0].Value.ToString(), dataGridView3.Font, Brushes.Black, 470, 100);

e.Graphics.DrawString("Заявки на регистрацию учебных групп", dataGridView3.Font, Brushes.Black, 270, 150);

e.Graphics.DrawString(save\_data, dataGridView3.Font, Brushes.Black, 250, 210);

e.Graphics.DrawString("Подпись\_\_\_\_\_\_\_\_\_\_\_\_", dataGridView2.Font, Brushes.Black, 485, 270);

e.Graphics.DrawString("Дата: " + DateTime.Now, dataGridView2.Font, Brushes.Black, 485, 290);

}

}

}

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

{

PrintDocument printDocument = new PrintDocument();

// обработчик события печати

printDocument.PrintPage += PrintPageHandler2;

// диалог настройки печати

PrintDialog printDialog = new PrintDialog();

// установка объекта печати для его настройки

printDialog.Document = printDocument;

// если в диалоге было нажато ОК

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

printDialog.Document.Print(); // печатаем

}

private void textBox5\_TextChanged(object sender, EventArgs e)

{

temp1 = textBox5.Text;

}

private void textBox6\_TextChanged(object sender, EventArgs e)

{

temp2 = textBox6.Text;

}

private void PrintPageHandler3(object sender, PrintPageEventArgs e)

{

e.Graphics.FillRectangle(Brushes.White, 100, 50, 550, 300);

e.Graphics.DrawRectangle(Pens.Black, 100, 50, 550, 300);

for (int i = 0; i < dataGridView2.Rows.Count - 1; i++)

{

if (dataGridView2.Rows[i].Selected == true)

{

e.Graphics.DrawString(dataGridView3.Rows[0].Cells[3].Value.ToString(), dataGridView3.Font, Brushes.Black, 470, 60);

e.Graphics.DrawString(dataGridView3.Rows[0].Cells[1].Value.ToString(), dataGridView3.Font, Brushes.Black, 470, 80);

e.Graphics.DrawString("Директор " + dataGridView3.Rows[0].Cells[0].Value.ToString(), dataGridView3.Font, Brushes.Black, 470, 100);

e.Graphics.DrawString("Приказ о зачислении учащихся", dataGridView3.Font, Brushes.Black, 300, 120);

e.Graphics.DrawString("Список групп", dataGridView3.Font, Brushes.Black, 350, 140);

e.Graphics.DrawString("Группа " + dataGridView1.Rows[0].Cells[1].Value.ToString() + " Преподователь " + dataGridView1.Rows[0].Cells[4].Value.ToString() + " Гос номер " + dataGridView1.Rows[0].Cells[5].Value.ToString() + " Марка " + dataGridView1.Rows[0].Cells[6].Value.ToString(), dataGridView3.Font, Brushes.Black, 120, 170);

e.Graphics.DrawString("Группа " + dataGridView1.Rows[1].Cells[1].Value.ToString() + " Преподователь " + dataGridView1.Rows[1].Cells[4].Value.ToString() + " Гос номер " + dataGridView1.Rows[1].Cells[5].Value.ToString() + " Марка " + dataGridView1.Rows[1].Cells[6].Value.ToString(), dataGridView3.Font, Brushes.Black, 120, 190);

e.Graphics.DrawString("Группа " + dataGridView1.Rows[2].Cells[1].Value.ToString() + " Преподователь " + dataGridView1.Rows[2].Cells[4].Value.ToString() + " Гос номер " + dataGridView1.Rows[2].Cells[5].Value.ToString() + " Марка " + dataGridView1.Rows[2].Cells[6].Value.ToString(), dataGridView3.Font, Brushes.Black, 120, 210);

e.Graphics.DrawString("Группа " + dataGridView1.Rows[3].Cells[1].Value.ToString() + " Преподователь " + dataGridView1.Rows[3].Cells[4].Value.ToString() + " Гос номер " + dataGridView1.Rows[3].Cells[5].Value.ToString() + " Марка " + dataGridView1.Rows[3].Cells[6].Value.ToString(), dataGridView3.Font, Brushes.Black, 120, 230);

e.Graphics.DrawString("Группа " + dataGridView1.Rows[4].Cells[1].Value.ToString() + " Преподователь " + dataGridView1.Rows[4].Cells[4].Value.ToString() + " Гос номер " + dataGridView1.Rows[4].Cells[5].Value.ToString() + " Марка " + dataGridView1.Rows[4].Cells[6].Value.ToString(), dataGridView3.Font, Brushes.Black, 120, 250);

e.Graphics.DrawString(save\_data, dataGridView3.Font, Brushes.Black, 250, 210);

e.Graphics.DrawString("Подпись\_\_\_\_\_\_\_\_\_\_\_\_", dataGridView2.Font, Brushes.Black, 485, 270);

e.Graphics.DrawString("Дата: " + temp2, dataGridView2.Font, Brushes.Black, 485, 290);

e.Graphics.DrawString("Номер приказа: " + temp1, dataGridView2.Font, Brushes.Black, 485, 310);

}

}

}

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

{

PrintDocument printDocument = new PrintDocument();

// обработчик события печати

printDocument.PrintPage += PrintPageHandler3;

// диалог настройки печати

PrintDialog printDialog = new PrintDialog();

// установка объекта печати для его настройки

printDialog.Document = printDocument;

// если в диалоге было нажато ОК

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

printDialog.Document.Print(); // печатаем

}

private void PrintPageHandler4(object sender, PrintPageEventArgs e)

{

e.Graphics.FillRectangle(Brushes.White, 100, 50, 550, 300);

e.Graphics.DrawRectangle(Pens.Black, 100, 50, 550, 300);

for (int i = 0; i < dataGridView2.RowCount - 1; i++)

{

if (dataGridView2.Rows[i].Cells[3].Value.ToString() == textBox3.Text)

{

save\_data = dataGridView2.Rows[i].Cells[1].Value.ToString();

break;

}

}

for (int i = 0; i < dataGridView2.Rows.Count - 1; i++)

{

if (dataGridView2.Rows[i].Selected == true)

{

e.Graphics.DrawString("Карточка вождения", dataGridView2.Font, Brushes.Black, 320, 60);

e.Graphics.DrawString(dataGridView3.Rows[0].Cells[3].Value.ToString(), dataGridView2.Font, Brushes.Black, 300, 100);

e.Graphics.DrawString("ФИО - " + dataGridView2.Rows[i].Cells[1].Value.ToString(), dataGridView2.Font, Brushes.Black, 300, 120);

e.Graphics.DrawString("Адрес - " + dataGridView2.Rows[i].Cells[5].Value.ToString(), dataGridView2.Font, Brushes.Black, 300, 140);

e.Graphics.DrawString("Место работы - " + dataGridView2.Rows[i].Cells[4].Value.ToString(), dataGridView2.Font, Brushes.Black, 300, 160);

e.Graphics.DrawString("Категория - В", dataGridView2.Font, Brushes.Black, 300, 180);

}

}

}

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

{

PrintDocument printDocument = new PrintDocument();

// обработчик события печати

printDocument.PrintPage += PrintPageHandler4;

// диалог настройки печати

PrintDialog printDialog = new PrintDialog();

// установка объекта печати для его настройки

printDialog.Document = printDocument;

// если в диалоге было нажато ОК

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

printDialog.Document.Print(); // печатаем

}

private void PrintPageHandler5(object sender, PrintPageEventArgs e)

{

e.Graphics.FillRectangle(Brushes.White, 100, 50, 550, 300);

e.Graphics.DrawRectangle(Pens.Black, 100, 50, 550, 300);

for (int i = 0; i < dataGridView2.RowCount - 1; i++)

{

if (dataGridView2.Rows[i].Cells[3].Value.ToString() == textBox3.Text)

{

save\_data = dataGridView2.Rows[i].Cells[1].Value.ToString();

break;

}

}

for (int i = 0; i < dataGridView2.Rows.Count - 1; i++)

{

if (dataGridView2.Rows[i].Selected == true)

{

e.Graphics.DrawString("Квитанция", dataGridView2.Font, Brushes.Black, 340, 60);

e.Graphics.DrawString("ФИО - " + dataGridView2.Rows[i].Cells[1].Value.ToString(), dataGridView2.Font, Brushes.Black, 250, 190);

e.Graphics.DrawString("Адрес - " + dataGridView2.Rows[i].Cells[5].Value.ToString(), dataGridView2.Font, Brushes.Black, 250, 210);

e.Graphics.DrawString("Сумма - ", dataGridView2.Font, Brushes.Black, 250, 230);

}

}

}

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

{

PrintDocument printDocument = new PrintDocument();

// обработчик события печати

printDocument.PrintPage += PrintPageHandler5;

// диалог настройки печати

PrintDialog printDialog = new PrintDialog();

// установка объекта печати для его настройки

printDialog.Document = printDocument;

// если в диалоге было нажато ОК

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

printDialog.Document.Print(); // печатаем

}

private void PrintPageHandler6(object sender, PrintPageEventArgs e)

{

e.Graphics.FillRectangle(Brushes.White, 100, 50, 550, 300);

e.Graphics.DrawRectangle(Pens.Black, 100, 50, 550, 300);

for (int i = 0; i < dataGridView2.RowCount - 1; i++)

{

if (dataGridView2.Rows[i].Cells[3].Value.ToString() == textBox3.Text)

{

save\_data = dataGridView2.Rows[i].Cells[1].Value.ToString();

break;

}

}

for (int i = 0; i < dataGridView2.Rows.Count - 1; i++)

{

if (dataGridView2.Rows[i].Selected == true)

{

e.Graphics.DrawString("Экзаменационная карточка по ПДД ", dataGridView2.Font, Brushes.Black, 280, 60);

e.Graphics.DrawString("ФИО - " + dataGridView2.Rows[i].Cells[1].Value.ToString(), dataGridView2.Font, Brushes.Black, 230, 190);

e.Graphics.DrawString("Дата сдачи - " + dataGridView2.Rows[i].Cells[10].Value.ToString(), dataGridView2.Font, Brushes.Black, 280, 210);

e.Graphics.DrawString("Категория - B ", dataGridView2.Font, Brushes.Black, 280, 230);

}

}

}

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

{

PrintDocument printDocument = new PrintDocument();

// обработчик события печати

printDocument.PrintPage += PrintPageHandler6;

// диалог настройки печати

PrintDialog printDialog = new PrintDialog();

// установка объекта печати для его настройки

printDialog.Document = printDocument;

// если в диалоге было нажато ОК

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

printDialog.Document.Print(); // печатаем

}

private void PrintPageHandler7(object sender, PrintPageEventArgs e)

{

e.Graphics.FillRectangle(Brushes.White, 100, 50, 550, 300);

e.Graphics.DrawRectangle(Pens.Black, 100, 50, 550, 300);

for (int i = 0; i < dataGridView2.RowCount - 1; i++)

{

if (dataGridView2.Rows[i].Cells[3].Value.ToString() == textBox3.Text)

{

save\_data = dataGridView2.Rows[i].Cells[1].Value.ToString();

break;

}

}

for (int i = 0; i < dataGridView2.Rows.Count - 1; i++)

{

if (dataGridView2.Rows[i].Selected == true)

{

e.Graphics.DrawString("Экзаменационной лист", dataGridView2.Font, Brushes.Black, 300, 60);

e.Graphics.DrawString("ФИО - " + dataGridView2.Rows[i].Cells[1].Value.ToString(), dataGridView2.Font, Brushes.Black, 230, 120);

e.Graphics.DrawString("Дата сдачи - " + dataGridView2.Rows[i].Cells[10].Value.ToString(), dataGridView2.Font, Brushes.Black, 280, 140);

e.Graphics.DrawString("Категория - B ", dataGridView2.Font, Brushes.Black, 280, 160);

e.Graphics.DrawString("1) " + "2) " + "3) ", dataGridView2.Font, Brushes.Black, 280, 180);

e.Graphics.DrawString("4) " + "5) " + "6) ", dataGridView2.Font, Brushes.Black, 280, 200);

e.Graphics.DrawString("7) " + "8) " + "9) ", dataGridView2.Font, Brushes.Black, 280, 220);

}

}

}

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

{

PrintDocument printDocument = new PrintDocument();

// обработчик события печати

printDocument.PrintPage += PrintPageHandler7;

// диалог настройки печати

PrintDialog printDialog = new PrintDialog();

// установка объекта печати для его настройки

printDialog.Document = printDocument;

// если в диалоге было нажато ОК

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

printDialog.Document.Print(); // печатаем

}

private void PrintPageHandler8(object sender, PrintPageEventArgs e)

{

e.Graphics.FillRectangle(Brushes.White, 100, 50, 550, 300);

e.Graphics.DrawRectangle(Pens.Black, 100, 50, 550, 300);

for (int i = 0; i < dataGridView2.RowCount - 1; i++)

{

if (dataGridView2.Rows[i].Cells[3].Value.ToString() == textBox3.Text)

{

save\_data = dataGridView2.Rows[i].Cells[1].Value.ToString();

break;

}

}

for (int i = 0; i < dataGridView2.Rows.Count - 1; i++)

{

if (dataGridView2.Rows[i].Selected == true)

{

e.Graphics.DrawString("Карточка", dataGridView2.Font, Brushes.Black, 350, 60);

e.Graphics.DrawString("ФИО - " + dataGridView2.Rows[i].Cells[1].Value.ToString(), dataGridView2.Font, Brushes.Black, 230, 180);

e.Graphics.DrawString("Дата сдачи - " + dataGridView2.Rows[i].Cells[10].Value.ToString(), dataGridView2.Font, Brushes.Black, 280, 200);

e.Graphics.DrawString("Категория - B ", dataGridView2.Font, Brushes.Black, 280, 220);

e.Graphics.DrawString("Результат - Сдал", dataGridView2.Font, Brushes.Black, 360, 240);

}

}

}

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

{

PrintDocument printDocument = new PrintDocument();

// обработчик события печати

printDocument.PrintPage += PrintPageHandler8;

// диалог настройки печати

PrintDialog printDialog = new PrintDialog();

// установка объекта печати для его настройки

printDialog.Document = printDocument;

// если в диалоге было нажато ОК

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

printDialog.Document.Print(); // печатаем

}

private void button1\_Click\_1(object sender, EventArgs e)

{

try

{

this.группаTableAdapter.FillBy(this.dATAS.Группа);

}

catch (System.Exception ex)

{

System.Windows.Forms.MessageBox.Show(ex.Message);

}

}

}

}