Program.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Bank

{

    static class Program

    {

        public static string appPath = Application.StartupPath.Replace("\\bin\\Debug", "\\Data");

        public static string str = $@"Data Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename={appPath}\Bank.mdf;Integrated Security=True;";

        public static bool isgovernor = false;

        public static string emppas, fam, name, sname, curepas, curcpas, curcont;

        public static bool FIO(string s)

        {

            return s.All(c => Char.IsLetterOrDigit(c) || c == '\_');

        }

        public static string trun(string s) //rem last char typed

        {

            return s = s.Truncate(s.Length-1);

        }

        /// <summary>

        /// The main entry point for the application.

        /// </summary>

        [STAThread]

        static void Main()

        {

            Application.EnableVisualStyles();

            Application.SetCompatibleTextRenderingDefault(false);

            Application.Run(new Login());

        }

    }

}

Login.cs

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.Data.SqlClient;

using System.Text.RegularExpressions;

namespace Bank

{

    //Для ограничения длины ввода

   public partial class Login : Form

    {

        public Login()

        {

            InitializeComponent();

        }

        public class LogPas

        {

            public int sid { get; set; } //id

            public string slogin { get; set; } //логин

            public string spass { get; set; } //пароль

            public bool sactive { get; set; } //активный ли аккаунт

        }

        public string replace(string s) //все попытки sql injections убираем

        {

            return Regex.Replace(s, "[-']", "");

        }

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

        {

            if (textBox1.Text.Length < 2 || textBox2.Text.Length < 2)

            {

                label1.Text = ("Слишком короткий логин или пароль");

                label1.Visible = true;

                return;

            }

            var conn = new SqlConnection();

            conn.ConnectionString = Program.str;

            replace(textBox1.Text);

            replace(textBox2.Text);

            LogPas[] slogpass;

            //Login

            using (SqlCommand StrQuer = new SqlCommand("SELECT \* FROM Login WHERE login=@login AND password=@password", conn)) //берем всю запись

            {

                StrQuer.Parameters.AddWithValue("@login", textBox1.Text);

                StrQuer.Parameters.AddWithValue("@password", textBox2.Text);

                conn.Open();

                SqlDataReader dr = StrQuer.ExecuteReader();

                using (dr) //для проверки на регистр в логине/пароле

                {

                    var list = new List<LogPas>();

                    while (dr.Read())

                        list.Add(new LogPas {sid = dr.GetInt32(0), slogin = dr.GetString(1), spass = dr.GetString(2), sactive = dr.GetBoolean(3) });

                    slogpass = list.ToArray();

                }

            }

            if (!(slogpass.GetLength(0) == 0) && String.Equals(textBox1.Text, slogpass[0].slogin) && String.Equals(textBox2.Text, slogpass[0].spass))

            {

                if (!slogpass[0].sactive) //не активен

                {

                    label1.Text = ("Этот аккаунт закрыт");

                    label1.Visible = true;

                    textBox2.Clear();

                    conn.Close();

                    return;

                }

                //Задаем ФИО и права

                using (SqlCommand StrQuer = new SqlCommand("SELECT emppas, fam, name, sname, job FROM emp WHERE logid=@logid", conn))

                {

                    StrQuer.Parameters.AddWithValue("@logid", slogpass[0].sid);

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    using (dr)

                    {

                        while (dr.Read())

                        {

                            Program.emppas = dr.GetString(0);

                            Program.fam = dr.GetString(1);

                            Program.name = dr.GetString(2);

                            Program.sname = dr.GetString(3);

                            Program.isgovernor = (dr.GetInt32(4) == 1 ? true : false); //управляющий ли?

                        }

                    }

                }

                    this.Hide();

                //MessageBox.Show($"loginned as {textBox1.Text}!");

                Меню tmp = new Меню();

                tmp.Show();

            }

            else

            {

                label1.Text = ("Неверное имя пользователя или пароль");

                label1.Visible = true;

                textBox2.Clear();

            }

            conn.Close();

        }

        //Для проверки ввода

        void press (object sender, KeyPressEventArgs e)

        {

            label1.Visible = false;

            base.OnKeyPress(e);

            char c = e.KeyChar;

            if (e.KeyChar == 22) //запрет на вставку чего-либо в текстбокс

            {

                e.Handled = true;

            }

            if (!(char.IsDigit(c) || char.IsControl(c) || (c >= 'A' && c <= 'Z') || (c >= 'a' && c <= 'z') || (c == '\_')))

            {

                e.Handled = true;

            }

            if (textBox1.Text.Length > 32 || textBox2.Text.Length > 32) //проверка на длину

            {

                label1.Text = ("Слишком длинный ввод");

                label1.Visible = true;

                e.Handled = true;

            }

        }

        //VISUAL

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

        {

            Regex.Replace(textBox1.Text, @"\s+", "");

            if (textBox1.Text == "")

            {

                textBox1.Text = "Имя пользователя";

                textBox1.ForeColor = Color.DarkGray;

            }

        }

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

        {

            Regex.Replace(textBox2.Text, @"\s+", "");

            if (textBox2.Text == "")

            {

                textBox2.Text = "Пароль";

                textBox2.UseSystemPasswordChar = false;

                textBox2.ForeColor = Color.DarkGray;

            }

        }

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

        {

            if (textBox1.Text == "Имя пользователя")

            {

                textBox1.Text = null;

            }

            textBox1.ForeColor = Color.Black;

        }

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

        {

            if (textBox2.Text == "Пароль")

            {

                textBox2.UseSystemPasswordChar = true;

                textBox2.Text = null;

            }

            textBox2.ForeColor = Color.Black;

        }

        //Remove char if count  > 32

        private void removeChar(object sender, KeyEventArgs e)

        {

            if (textBox1.Text.Length > 32 || textBox2.Text.Length > 32)

            {

                textBox1.Text = textBox1.Text.Truncate(32);

                textBox2.Text = textBox2.Text.Truncate(32);

            }

        }

    }

    //Truncate

    public static class StringExt

    {

        public static string Truncate(this string value, int maxLength)

        {

            if (string.IsNullOrEmpty(value)) return value;

            return value.Length <= maxLength ? value : value.Substring(0, maxLength);

        }

    }

}

Menu.cs

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;

namespace Bank

{

    public partial class Меню : Form

    {

        public Меню()

        {

            InitializeComponent();

        }

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

        {

            Поиск tmp = new Поиск();

            tmp.Show();

            return;

        }

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

        {

            List<Form> openForms = new List<Form>();

            foreach (Form f in Application.OpenForms)

                openForms.Add(f);

            foreach (Form f in openForms)

            {

                if (!(String.Equals(f.Name, "Login") || String.Equals(f.Name, "Меню")))

                    f.Close();

            }

            Login tmp = new Login();

            tmp.Show();

            this.Hide();

        }

        private void Меню\_FormClosed(object sender, FormClosedEventArgs e)

        {

            Application.Exit();

        }

        private void Меню\_VisibleChanged(object sender, EventArgs e) //показ имени пользователя

        {

            ФИО.Text = $"{Program.fam} {Program.name} {Program.sname}";

            if (Program.isgovernor)

            {

                ФИО.Text = ФИО.Text + ", Управляющий";

                addempb.Visible = true;

                label3.Visible = true;

            }

        }

        private void toolStripMenuItem1\_Click(object sender, EventArgs e) //личные данные

        {

            Program.curepas = Program.emppas; //чтобы при отмене не зануляло

            empinfo tmp = new empinfo();

            tmp.Show();

            tmp.empdata(Program.emppas);

            tmp.button2.PerformClick();

        }

        private void addempb\_Click(object sender, EventArgs e) //добавление сотрудника

        {

            empinfo tmp = new empinfo();

            tmp.Show();

            Program.curepas = null;

            tmp.button5.PerformClick();

        }

        private void addclientb\_Click(object sender, EventArgs e) //добавление клиента

        {

            clieinfo tmp = new clieinfo();

            tmp.Show();

            Program.curcpas = null;

            tmp.button5.PerformClick();

        }

    }

}

Addcard.cs

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.Data.SqlClient;

namespace Bank

{

    public partial class addcard : Form

    {

        public addcard()

        {

            InitializeComponent();

        }

        Dictionary<int, string> test = new Dictionary<int, string>();

        private void addcard\_VisibleChanged(object sender, EventArgs e)

        {

            if (Program.isgovernor)

            {

                deletecon.Visible = true; //управляющий может удалять

            }

        }

        public void loadinfocard (string contnum) //загрузка данных по ид договора

        {

            Program.curcont = contnum;

            var conn = new SqlConnection();

            conn.ConnectionString = Program.str;

            using (SqlCommand StrQuer = new SqlCommand("SELECT \* " +

                "FROM contcard INNER JOIN prcard ON contcard.prcard = prcard.Id " +

                $"WHERE contcard.contid = {contnum}", conn))

            {

                conn.Open();

                SqlDataReader dr = StrQuer.ExecuteReader();

                using (dr) //для проверки на регистр в логине/пароле

                {

                    while (dr.Read())

                    {  //вставка полей в textbox

                        textBox1.Text = dr.GetInt32(0).ToString(); //ид карты

                        textBox2.Text = dr.GetInt32(1).ToString(); //ид договора

                        textBox3.Text = dr.GetDateTime(2).Date.ToString("d"); //заканчивается

                        comboBox1.SelectedIndex = comboBox1.FindStringExact(test[dr.GetInt32(3)]); //код программы карты

                        textBox5.Text = dr.GetDateTime(4).Date.ToString("d"); //заключение

                        textBox6.Text = dr.GetString(5); //паспорт клиента

                        textBox7.Text = dr.GetString(6); //паспорт сотрудника

                        textBox8.Text = dr.GetInt32(7).ToString(); // номер счета

                        textBox9.Text = dr.GetDecimal(10).ToString(); // 10 loan lim

                        textBox10.Text = dr.GetDecimal(11).ToString(); // 11 loan max

                    }

                }

                conn.Close();

            }

        }

        private void addcard\_Load(object sender, EventArgs e) //загрузка словаря для комбобокса из ДБ

        {

            var conn = new SqlConnection();

            conn.ConnectionString = Program.str;

            using (SqlCommand StrQuer = new SqlCommand("SELECT \* " +

                "FROM prcard ", conn))

            {

                conn.Open();

                SqlDataReader dr = StrQuer.ExecuteReader();

                using (dr) //для проверки на регистр в логине/пароле

                {

                    while (dr.Read())

                    {  //вставка полей в textbox

                        int tmp;

                        string tmps;

                        tmp = dr.GetInt32(0); //ид программы

                        tmps = dr.GetString(1); //ид договора

                        test.Add(tmp, tmps);

                    }

                }

                conn.Close();

            }

            comboBox1.DataSource = new BindingSource(test, null);

            comboBox1.DisplayMember = "Value";

            comboBox1.ValueMember = "Key";

        }

        #region texbox

        //Для блока текстбоксов

        public void ronly()

        {

            textBox1.ReadOnly = true;

            textBox2.ReadOnly = true;

            textBox3.ReadOnly = true;

            comboBox1.Enabled = false;

            textBox5.ReadOnly = true;

            textBox6.ReadOnly = true;

            textBox7.ReadOnly = true;

            textBox8.ReadOnly = true;

        }

        public void write()

        {

            textBox1.ReadOnly = false;

            textBox2.ReadOnly = false;

            textBox3.ReadOnly = false;

            comboBox1.Enabled = true;

            textBox5.ReadOnly = false;

            textBox6.ReadOnly = false;

            textBox7.ReadOnly = false;

            textBox8.ReadOnly = false;

        }

        public void wipe()

        {

            textBox1.Text = null;

            textBox2.Text = null;

            textBox3.Text = null;

            textBox5.Text = null;

            textBox6.Text = null;

            textBox7.Text = null;

            textBox8.Text = null;

            textBox9.Text = null;

            textBox10.Text = null;

        }

        #endregion

        //удаление

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

        {

            DialogResult dialogResult = MessageBox.Show($"Удаление договора {Program.curcont}, уверены?", "Внимание", MessageBoxButtons.YesNo);

            if (dialogResult == DialogResult.Yes)

            {

                var conn = new SqlConnection();

                conn.ConnectionString = Program.str;

                using (SqlCommand StrQuer = new SqlCommand("BEGIN TRANSACTION " +

                    $"DELETE FROM contract WHERE Id = '{Program.curcont}'; " +

                    $"DELETE FROM contcard WHERE contid = '{Program.curcont}';" +

                    $"COMMIT", conn))

                {

                    conn.Open();

                    SqlDataReader dr = StrQuer.ExecuteReader();

                }

                ronly();

                wipe();

                edit.Enabled = true;

                create.Enabled = true;

                save.Visible = false;

                revert.Visible = false;

                MessageBox.Show("Удаление завершено");

            }

            else if (dialogResult == DialogResult.No)

            {

                return;

            }

        }

        //Редактирование

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

        {

            create.Enabled = false;

            save.Visible = true;

            revert.Visible = true;

            write();

        }

        //сохранение

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

        {

            var conn = new SqlConnection();

            conn.ConnectionString = Program.str;

            if (edit.Enabled == true)  //редакт

            {

                //Получение ном дог и карты текущ договора

                string cardid = "-1", conid = "-1";

                using (SqlCommand StrQuer = new SqlCommand("SELECT Id, contid " +

                    "FROM contcard " +

                    $"WHERE contid = {Program.curcont}", conn))

                {

                    conn.Open();

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    using (dr)

                    {

                        while (dr.Read())

                        {  //вставка полей в textbox

                            cardid = dr.GetInt32(0).ToString(); //ид карты

                            conid = dr.GetInt32(1).ToString(); //ид договора

                        }

                    }

                    conn.Close();

                }

                //проверка текущего номера договора и карты на повторение (кроме себя самого)

                bool k = true; ;

                using (SqlCommand StrQuer = new SqlCommand("SELECT contcard.Id, contract.Id " +

                    "FROM contcard INNER JOIN contract ON contcard.contid = contract.Id " +

                    $"WHERE contcard.contid = '{textBox1.Text}' " +

                    $"OR contract.Id = '{textBox2.Text}'", conn))

                {

                    conn.Open();

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    using (dr)

                    {

                        bool found = false;

                        while (dr.Read())

                        {

                            found = true;

                            string tcardid = "-1", tid = "-1";

                            tcardid = dr.GetInt32(0).ToString(); //номер карты

                            tid = dr.GetInt32(1).ToString(); //номер договора

                            //Если нашли старую запись - скип

                            if (String.Equals(cardid, tcardid) && String.Equals(conid, tid))

                            {

                                continue;

                            }

                            if (String.Equals(textBox1.Text, tcardid))

                            {

                                k = false;

                                textBox1.Text = null;

                            }

                            if (String.Equals(textBox2.Text, tid))

                            {

                                k = false;

                                textBox2.Text = null;

                            }

                        }

                        conn.Close();

                        if (!k)

                        {

                            MessageBox.Show("Не могут повторяться: номера счетов");

                            return;

                        }

                        if (!found)

                        {

                            MessageBox.Show("Номера договора карты не существует");

                            textBox1.Text = null;

                            textBox2.Text = null;

                            return;

                        }

                    }

                }

                k = false;

                //проверка на существование паспорта сотруд

                using (SqlCommand StrQuer = new SqlCommand("SELECT emppas FROM emp " +

                    $"WHERE emppas = '{textBox7.Text}'", conn))

                {

                    conn.Open();

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    using (dr)

                    {

                        string empid = "-1";

                        while (dr.Read())

                        {

                            empid = dr.GetString(0); //пасп сотруд

                            //Если нашли запись - существует

                            if (String.Equals(textBox7.Text, empid))

                            {

                                k = true;

                            }

                        }

                        conn.Close();

                        if (!k)

                        {

                            MessageBox.Show("Сотрудника не существует");

                            return;

                        }

                    }

                }

                //существует ли счет

                bool t = true; //счет не существует?

                using (SqlCommand StrQuer = new SqlCommand("SELECT Id, clipas FROM acc " +

                   $"WHERE Id = '{textBox8.Text}'", conn))

                {

                    conn.Open();

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    using (dr)

                    {

                        k = true;

                        while (dr.Read())

                        {

                            string tid = "-1", tpas = "-1";

                            tid = dr.GetInt32(0).ToString(); //счет

                            tpas = dr.GetString(1); //паспорт кли

                            //Если нашли запись - существует

                            if (String.Equals(textBox8.Text, tid))

                            {

                                t = false;

                            }

                            if (!String.Equals(tpas, textBox6.Text))

                                k = false;

                        }

                        conn.Close();

                    }

                }

                if (!k)

                {

                    MessageBox.Show("Данный счет уже занят");

                    return;

                }

                //проверка на существование паспорта клиента

                using (SqlCommand StrQuer = new SqlCommand("SELECT clipas FROM cli " +

                   $"WHERE clipas = '{textBox6.Text}'", conn))

                {

                    string tcli = "-1";

                    conn.Open();

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    using (dr)

                    {

                        while (dr.Read())

                        {

                            string empid = "-1";

                            empid = dr.GetString(0); //пасп клиента

                            //Если нашли запись - существует

                            if (String.Equals(textBox6.Text, empid))

                            {

                                tcli = empid;

                                k = true;

                            }

                        }

                        conn.Close();

                        if (!k)

                        {

                            MessageBox.Show("Клиента не существует");

                            return;

                        }

                    }

                }

                //Добавляем счет, если нет

                if (t)

                {

                    conn.Open();

                    DateTime myDateTime1 = DateTime.Now;

                    DateTime myDateTime2 = DateTime.Now.AddYears(4);

                    string sqlFormattedDate1 = myDateTime1.ToString("yyyy-MM-dd HH:mm:ss.fff");

                    string sqlFormattedDate2 = myDateTime2.ToString("yyyy-MM-dd HH:mm:ss.fff");

                    using (SqlCommand StrQuer = new SqlCommand("INSERT INTO acc (Id, opens, closes, clipas) " +

                        $"VALUES ('{textBox8.Text}', '{sqlFormattedDate1}', '{sqlFormattedDate2}', '{textBox6.Text}');", conn))

                    {

                        SqlDataReader dr = StrQuer.ExecuteReader();

                        conn.Close();

                    }

                }

                DateTime myDateTime3 = Convert.ToDateTime(textBox3.Text);

                DateTime myDateTime5 = Convert.ToDateTime(textBox5.Text);

                string sqlFormattedDate3 = myDateTime3.ToString("yyyy-MM-dd HH:mm:ss.fff");

                string sqlFormattedDate5 = myDateTime5.ToString("yyyy-MM-dd HH:mm:ss.fff");

                //Апдейт

                conn.Open();

                int prcode = ((KeyValuePair<int, string>)comboBox1.SelectedItem).Key;

                using (SqlCommand StrQuer = new SqlCommand

                        (

                        $"UPDATE contcard SET Id = N'{textBox1.Text}', contid = N'{textBox2.Text}', closes = N'{sqlFormattedDate3}', prcard = N'{prcode}', " +

                        $"contdate = '{sqlFormattedDate5}', clipas = N'{textBox6.Text}', emppas = N'{textBox7.Text}', accid = N'{textBox8.Text}' " +

                        $" WHERE (contid = '{textBox2.Text}');", conn))

                {

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    conn.Close();

                }

                MessageBox.Show("Изменения сохранены");

                loadinfocard(textBox2.Text);

            }

            else //добавляем

            {

                textBox7.ReadOnly = true;

                bool k = false;

                //нужна проверка на наличие паспорта клиента в бд, если нет - ошибка

                using (SqlCommand StrQuer = new SqlCommand("SELECT clipas FROM cli " +

                  $"WHERE clipas = '{textBox6.Text}'", conn))

                {

                    string tcli = "-1";

                    conn.Open();

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    using (dr)

                    {

                        while (dr.Read())

                        {

                            string empid = "-1";

                            empid = dr.GetString(0); //пасп клиента

                            //Если нашли запись - существует

                            if (String.Equals(textBox6.Text, empid))

                            {

                                tcli = empid;

                                k = true;

                            }

                        }

                        conn.Close();

                        if (!k)

                        {

                            MessageBox.Show("Клиента не существует");

                            return;

                        }

                    }

                }

                //существует ли счет

                bool t = true; //счет не существует?

                using (SqlCommand StrQuer = new SqlCommand("SELECT Id, clipas FROM acc " +

                   $"WHERE Id = '{textBox8.Text}'", conn))

                {

                    conn.Open();

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    using (dr)

                    {

                        k = true;

                        while (dr.Read())

                        {

                            string tid = "-1", tpas = "-1";

                            tid = dr.GetInt32(0).ToString(); //счет

                            tpas = dr.GetString(1); //паспорт кли

                            //Если нашли запись - существует

                            if (String.Equals(textBox8.Text, tid))

                            {

                                t = false;

                            }

                            if (!String.Equals(tpas, textBox6.Text))

                                k = false;

                        }

                        conn.Close();

                    }

                }

                if (!k)

                {

                    MessageBox.Show("Данный счет уже занят");

                    return;

                }

                //Добавляем счет, если нет

                if (t)

                {

                    conn.Open();

                    DateTime myDateTime1 = DateTime.Now;

                    DateTime myDateTime2 = DateTime.Now.AddYears(4);

                    string sqlFormattedDate1 = myDateTime1.ToString("yyyy-MM-dd HH:mm:ss.fff");

                    string sqlFormattedDate2 = myDateTime2.ToString("yyyy-MM-dd HH:mm:ss.fff");

                    using (SqlCommand StrQuer = new SqlCommand("INSERT INTO acc (Id, opens, closes, clipas) " +

                        $"VALUES ('{textBox8.Text}', '{sqlFormattedDate1}', '{sqlFormattedDate2}', '{textBox6.Text}');", conn))

                    {

                        SqlDataReader dr = StrQuer.ExecuteReader();

                        conn.Close();

                    }

                }

                DateTime myDateTime3 = Convert.ToDateTime(textBox3.Text);

                DateTime myDateTime5 = Convert.ToDateTime(textBox5.Text);

                string sqlFormattedDate3 = myDateTime3.ToString("yyyy-MM-dd HH:mm:ss.fff");

                string sqlFormattedDate5 = myDateTime5.ToString("yyyy-MM-dd HH:mm:ss.fff");

                conn.Open();

                int prcode = ((KeyValuePair<int, string>)comboBox1.SelectedItem).Key;

                using (SqlCommand StrQuer = new SqlCommand

                        (

                        $"INSERT INTO contcard (Id, contid, closes, prcard, contdate, clipas,  emppas, accid) " +

                        $"VALUES ('{textBox1.Text}', '{textBox2.Text}', '{sqlFormattedDate3}', '{prcode}', '{sqlFormattedDate5}', '{textBox6.Text}', '{textBox7.Text}', '{textBox8.Text}'); " +

                        $" ", conn))

                {

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    conn.Close();

                }

                MessageBox.Show("Изменения сохранены");

                //В конце обновляем выбранный контракт

                loadinfocard(textBox2.Text);

            }

            ronly();

            edit.Enabled = true;

            create.Enabled = true;

            save.Visible = false;

            revert.Visible = false;

        }

        //отмена

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

        {

            if (Program.curcont != null)

                loadinfocard(Program.curcont);

            else

                wipe();

            save.Visible = false;

            revert.Visible = false;

            ronly();

            edit.Enabled = true;

            create.Enabled = true;

        }

        //создать

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

        {

            Program.curcont = null;

            wipe();

            write();

            edit.Enabled = false;

            save.Visible = true;

            revert.Visible = true;

            textBox7.Text = Program.emppas;

            textBox1.ReadOnly = true;

            textBox2.ReadOnly = true;

            textBox5.Text = DateTime.Now.ToShortDateString();

            var conn = new SqlConnection();

            conn.ConnectionString = Program.str;

            conn.Open();

            int maxloc = -1, maxc = -1;

            using (SqlCommand StrQuer = new SqlCommand($"SELECT MAX(contid) FROM contcard", conn))

            {

                SqlDataReader dr = StrQuer.ExecuteReader();

                using (dr)

                    while (dr.Read())

                    {

                        maxloc = dr.GetInt32(0);

                    }

            }

            using (SqlCommand StrQuer = new SqlCommand($"SELECT MAX(Id) FROM contract", conn))

            {

                SqlDataReader dr = StrQuer.ExecuteReader();

                using (dr)

                    while (dr.Read())

                    {

                        maxc = dr.GetInt32(0);

                    }

            }

            textBox1.Text = (maxloc + 1).ToString();

            textBox2.Text = (maxc + 1).ToString();

        }

    }

}

Adddepo.cs

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.Data.SqlClient;

namespace Bank

{

    public partial class adddepo : Form

    {

        public adddepo()

        {

            InitializeComponent();

        }

        Dictionary<int, string> test = new Dictionary<int, string>();

        Dictionary<bool, string> test1 = new Dictionary<bool, string>();

        private void addcard\_VisibleChanged(object sender, EventArgs e)

        {

            if (Program.isgovernor)

            {

                deletecon.Visible = true; //управляющий может удалять

            }

        }

        public void loadInfodepo (string contnum) //загрузка данных по ид договора

        {

            Program.curcont = contnum;

            var conn = new SqlConnection();

            conn.ConnectionString = Program.str;

            using (SqlCommand StrQuer = new SqlCommand("SELECT \* " +

                "FROM contdepo INNER JOIN prdepo ON contdepo.prdepo = prdepo.Id " +

                $"WHERE contdepo.contid = {contnum}", conn))

            {

                conn.Open();

                SqlDataReader dr = StrQuer.ExecuteReader();

                using (dr) //для проверки на регистр в логине/пароле

                {

                    while (dr.Read())

                    {  //вставка полей в textbox

                        textBox1.Text = dr.GetInt32(0).ToString(); //ид вклада

                        textBox2.Text = dr.GetInt32(1).ToString(); //ид договора

                        textBox12.Text = dr.GetDecimal(2).ToString(); //сумма вклада

                        textBox11.Text = dr.GetDateTime(3).Date.ToString("d"); //нач

                        textBox3.Text = dr.GetDateTime(4).Date.ToString("d"); //заканчивается

                        comboBox1.SelectedIndex = comboBox1.FindStringExact(test[dr.GetInt32(5)]); //код программы вклада

                        textBox5.Text = dr.GetDateTime(6).Date.ToString("d"); //заключение

                        textBox6.Text = dr.GetString(7); //паспорт клиента

                        textBox7.Text = dr.GetString(8); //паспорт сотрудника

                        textBox8.Text = dr.GetInt32(9).ToString(); // номер счета

                        textBox9.Text = dr.GetDecimal(12).ToString(); // loan lim

                        textBox10.Text = dr.GetDecimal(13).ToString(); // loan max

                        textBox4.Text = dr.GetDouble(14).ToString(); //проц

                        comboBox2.SelectedIndex = comboBox2.FindStringExact(test1[dr.GetBoolean(15)]); //да нет

                        textBox13.Text = dr.GetDouble(16).ToString(); //проц при доср

                    }

                }

                conn.Close();

            }

        }

        private void addcard\_Load(object sender, EventArgs e) //загрузка словаря для комбобокса из ДБ

        {

            test1.Add(false, "Нет");

            test1.Add(true, "Да");

            comboBox2.DataSource = new BindingSource(test1, null);

            comboBox2.DisplayMember = "Value";

            comboBox2.ValueMember = "Key";

            var conn = new SqlConnection();

            conn.ConnectionString = Program.str;

            using (SqlCommand StrQuer = new SqlCommand("SELECT \* " +

                "FROM prdepo ", conn))

            {

                conn.Open();

                SqlDataReader dr = StrQuer.ExecuteReader();

                using (dr) //для проверки на регистр в логине/пароле

                {

                    while (dr.Read())

                    {  //вставка полей в textbox

                        int tmp;

                        string tmps;

                        tmp = dr.GetInt32(0); //ид программы

                        tmps = dr.GetString(1); //ид договора

                        test.Add(tmp, tmps);

                    }

                }

                conn.Close();

            }

            comboBox1.DataSource = new BindingSource(test, null);

            comboBox1.DisplayMember = "Value";

            comboBox1.ValueMember = "Key";

        }

        #region texbox

        //Для блока текстбоксов

        public void ronly()

        {

            textBox1.ReadOnly = true;

            textBox2.ReadOnly = true;

            textBox3.ReadOnly = true;

            comboBox1.Enabled = false;

            textBox5.ReadOnly = true;

            textBox6.ReadOnly = true;

            textBox7.ReadOnly = true;

            textBox8.ReadOnly = true;

            textBox11.ReadOnly = true;

            textBox12.ReadOnly = true;

        }

        public void write()

        {

            textBox1.ReadOnly = false;

            textBox2.ReadOnly = false;

            textBox3.ReadOnly = false;

            comboBox1.Enabled = true;

            textBox5.ReadOnly = false;

            textBox6.ReadOnly = false;

            textBox7.ReadOnly = false;

            textBox8.ReadOnly = false;

            textBox11.ReadOnly = false;

            textBox12.ReadOnly = false;

        }

        public void wipe()

        {

            textBox1.Text = null;

            textBox2.Text = null;

            textBox3.Text = null;

            textBox4.Text = null;

            textBox5.Text = null;

            textBox6.Text = null;

            textBox7.Text = null;

            textBox8.Text = null;

            textBox9.Text = null;

            textBox10.Text = null;

            textBox11.Text = null;

            textBox12.Text = null;

            textBox13.Text = null;

        }

        #endregion

        //удаление

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

        {

            DialogResult dialogResult = MessageBox.Show($"Удаление договора {Program.curcont}, уверены?", "Внимание", MessageBoxButtons.YesNo);

            if (dialogResult == DialogResult.Yes)

            {

                var conn = new SqlConnection();

                conn.ConnectionString = Program.str;

                using (SqlCommand StrQuer = new SqlCommand("BEGIN TRANSACTION " +

                    $"DELETE FROM contract WHERE Id = '{Program.curcont}'; " +

                    $"DELETE FROM contdepo WHERE contid = '{Program.curcont}';" +

                    $"COMMIT", conn))

                {

                    conn.Open();

                    SqlDataReader dr = StrQuer.ExecuteReader();

                }

                ronly();

                wipe();

                edit.Enabled = true;

                create.Enabled = true;

                save.Visible = false;

                revert.Visible = false;

                MessageBox.Show("Удаление завершено");

            }

            else if (dialogResult == DialogResult.No)

            {

                return;

            }

        }

        //Редактирование

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

        {

            create.Enabled = false;

            save.Visible = true;

            revert.Visible = true;

            write();

        }

        //сохранение

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

        {

            var conn = new SqlConnection();

            conn.ConnectionString = Program.str;

            if (edit.Enabled == true)  //редакт

            {

                //Получение ном дог и вклада текущ договора

                string cardid = "-1", conid = "-1";

                using (SqlCommand StrQuer = new SqlCommand("SELECT Id, contid " +

                    "FROM contdepo " +

                    $"WHERE contid = {Program.curcont}", conn))

                {

                    conn.Open();

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    using (dr)

                    {

                        while (dr.Read())

                        {  //вставка полей в textbox

                            cardid = dr.GetInt32(0).ToString(); //ид карты

                            conid = dr.GetInt32(1).ToString(); //ид договора

                        }

                    }

                    conn.Close();

                }

                //проверка текущего номера договора и карты на повторение (кроме себя самого)

                bool k = true; ;

                using (SqlCommand StrQuer = new SqlCommand("SELECT contdepo.Id, contract.Id " +

                    "FROM contdepo INNER JOIN contract ON contdepo.contid = contract.Id " +

                    $"WHERE contdepo.contid = '{textBox1.Text}' " +

                    $"OR contract.Id = '{textBox2.Text}'", conn))

                {

                    conn.Open();

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    using (dr)

                    {

                        bool found = false;

                        while (dr.Read())

                        {

                            found = true;

                            string tcardid = "-1", tid = "-1";

                            tcardid = dr.GetInt32(0).ToString(); //номер карты

                            tid = dr.GetInt32(1).ToString(); //номер договора

                            //Если нашли старую запись - скип

                            if (String.Equals(cardid, tcardid) && String.Equals(conid, tid))

                            {

                                continue;

                            }

                            if (String.Equals(textBox1.Text, tcardid))

                            {

                                k = false;

                                textBox1.Text = null;

                            }

                            if (String.Equals(textBox2.Text, tid))

                            {

                                k = false;

                                textBox2.Text = null;

                            }

                        }

                        conn.Close();

                        if (!k)

                        {

                            MessageBox.Show("Не могут повторяться: номера счетов");

                            return;

                        }

                        if (!found)

                        {

                            MessageBox.Show("Номера договора вклада не существует");

                            textBox1.Text = null;

                            textBox2.Text = null;

                            return;

                        }

                    }

                }

                k = false;

                //проверка на существование паспорта сотруд

                using (SqlCommand StrQuer = new SqlCommand("SELECT emppas FROM emp " +

                    $"WHERE emppas = '{textBox7.Text}'", conn))

                {

                    conn.Open();

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    using (dr)

                    {

                        string empid = "-1";

                        while (dr.Read())

                        {

                            empid = dr.GetString(0); //пасп сотруд

                            //Если нашли запись - существует

                            if (String.Equals(textBox7.Text, empid))

                            {

                                k = true;

                            }

                        }

                        conn.Close();

                        if (!k)

                        {

                            MessageBox.Show("Сотрудника не существует");

                            return;

                        }

                    }

                }

                //существует ли счет

                bool t = true; //счет не существует?

                using (SqlCommand StrQuer = new SqlCommand("SELECT Id, clipas FROM acc " +

                   $"WHERE Id = '{textBox8.Text}'", conn))

                {

                    conn.Open();

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    using (dr)

                    {

                        k = true;

                        while (dr.Read())

                        {

                            string tid = "-1", tpas = "-1";

                            tid = dr.GetInt32(0).ToString(); //счет

                            tpas = dr.GetString(1); //паспорт кли

                            //Если нашли запись - существует

                            if (String.Equals(textBox8.Text, tid))

                            {

                                t = false;

                            }

                            if (!String.Equals(tpas, textBox6.Text))

                                k = false;

                        }

                        conn.Close();

                    }

                }

                if (!k)

                {

                    MessageBox.Show("Данный счет уже занят");

                    return;

                }

                //проверка на существование паспорта клиента

                using (SqlCommand StrQuer = new SqlCommand("SELECT clipas FROM cli " +

                   $"WHERE clipas = '{textBox6.Text}'", conn))

                {

                    string tcli = "-1";

                    conn.Open();

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    using (dr)

                    {

                        while (dr.Read())

                        {

                            string empid = "-1";

                            empid = dr.GetString(0); //пасп клиента

                            //Если нашли запись - существует

                            if (String.Equals(textBox6.Text, empid))

                            {

                                tcli = empid;

                                k = true;

                            }

                        }

                        conn.Close();

                        if (!k)

                        {

                            MessageBox.Show("Клиента не существует");

                            return;

                        }

                    }

                }

                //Добавляем счет, если нет

                if (t)

                {

                    conn.Open();

                    DateTime myDateTime1 = DateTime.Now;

                    DateTime myDateTime2 = DateTime.Now.AddYears(4);

                    string sqlFormattedDate1 = myDateTime1.ToString("yyyy-MM-dd HH:mm:ss.fff");

                    string sqlFormattedDate2 = myDateTime2.ToString("yyyy-MM-dd HH:mm:ss.fff");

                    using (SqlCommand StrQuer = new SqlCommand("INSERT INTO acc (Id, opens, closes, clipas) " +

                        $"VALUES ('{textBox8.Text}', '{sqlFormattedDate1}', '{sqlFormattedDate2}', '{textBox6.Text}');", conn))

                    {

                        SqlDataReader dr = StrQuer.ExecuteReader();

                        conn.Close();

                    }

                }

                DateTime myDateTime11 = Convert.ToDateTime(textBox11.Text);

                DateTime myDateTime3 = Convert.ToDateTime(textBox3.Text);

                DateTime myDateTime5 = Convert.ToDateTime(textBox5.Text);

                string sqlFormattedDate3 = myDateTime3.ToString("yyyy-MM-dd HH:mm:ss.fff");

                string sqlFormattedDate5 = myDateTime5.ToString("yyyy-MM-dd HH:mm:ss.fff");

                string sqlFormattedDate11 = myDateTime11.ToString("yyyy-MM-dd HH:mm:ss.fff");

                //Апдейт

                conn.Open();

                int prcode = ((KeyValuePair<int, string>)comboBox1.SelectedItem).Key;

                using (SqlCommand StrQuer = new SqlCommand

                        (

                        $"UPDATE contdepo SET Id = N'{textBox1.Text}', contid = N'{textBox2.Text}', sum = N'{textBox12.Text.Replace(",0000","")}', " +

                        $"opens = N'{sqlFormattedDate11}', closes = N'{sqlFormattedDate3}', prdepo = N'{prcode}', " +

                        $"contdate = '{sqlFormattedDate5}', clipas = N'{textBox6.Text}', emppas = N'{textBox7.Text}', accid = N'{textBox8.Text}' " +

                        $" WHERE (contid = '{textBox2.Text}');", conn))

                {

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    conn.Close();

                }

                MessageBox.Show("Изменения сохранены");

                loadInfodepo(textBox2.Text);

            }

            else //добавляем

            {

                textBox7.ReadOnly = true;

                bool k = false;

                //нужна проверка на наличие паспорта клиента в бд, если нет - ошибка

                using (SqlCommand StrQuer = new SqlCommand("SELECT clipas FROM cli " +

                  $"WHERE clipas = '{textBox6.Text}'", conn))

                {

                    string tcli = "-1";

                    conn.Open();

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    using (dr)

                    {

                        while (dr.Read())

                        {

                            string empid = "-1";

                            empid = dr.GetString(0); //пасп клиента

                            //Если нашли запись - существует

                            if (String.Equals(textBox6.Text, empid))

                            {

                                tcli = empid;

                                k = true;

                            }

                        }

                        conn.Close();

                        if (!k)

                        {

                            MessageBox.Show("Клиента не существует");

                            return;

                        }

                    }

                }

                //существует ли счет

                bool t = true; //счет не существует?

                using (SqlCommand StrQuer = new SqlCommand("SELECT Id, clipas FROM acc " +

                   $"WHERE Id = '{textBox8.Text}'", conn))

                {

                    conn.Open();

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    using (dr)

                    {

                        k = true;

                        while (dr.Read())

                        {

                            string tid = "-1", tpas = "-1";

                            tid = dr.GetInt32(0).ToString(); //счет

                            tpas = dr.GetString(1); //паспорт кли

                            //Если нашли запись - существует

                            if (String.Equals(textBox8.Text, tid))

                            {

                                t = false;

                            }

                            if (!String.Equals(tpas, textBox6.Text))

                                k = false;

                        }

                        conn.Close();

                    }

                }

                if (!k)

                {

                    MessageBox.Show("Данный счет уже занят");

                    return;

                }

                //Добавляем счет, если нет

                if (t)

                {

                    conn.Open();

                    DateTime myDateTime1 = DateTime.Now;

                    DateTime myDateTime2 = DateTime.Now.AddYears(4);

                    string sqlFormattedDate1 = myDateTime1.ToString("yyyy-MM-dd HH:mm:ss.fff");

                    string sqlFormattedDate2 = myDateTime2.ToString("yyyy-MM-dd HH:mm:ss.fff");

                    using (SqlCommand StrQuer = new SqlCommand("INSERT INTO acc (Id, opens, closes, clipas) " +

                        $"VALUES ('{textBox8.Text}', '{sqlFormattedDate1}', '{sqlFormattedDate2}', '{textBox6.Text}');", conn))

                    {

                        SqlDataReader dr = StrQuer.ExecuteReader();

                        conn.Close();

                    }

                }

                DateTime myDateTime11 = Convert.ToDateTime(textBox11.Text);

                DateTime myDateTime3 = Convert.ToDateTime(textBox3.Text);

                DateTime myDateTime5 = Convert.ToDateTime(textBox5.Text);

                string sqlFormattedDate3 = myDateTime3.ToString("yyyy-MM-dd HH:mm:ss.fff");

                string sqlFormattedDate5 = myDateTime5.ToString("yyyy-MM-dd HH:mm:ss.fff");

                string sqlFormattedDate11 = myDateTime11.ToString("yyyy-MM-dd HH:mm:ss.fff");

                conn.Open();

                int prcode = ((KeyValuePair<int, string>)comboBox1.SelectedItem).Key;

                using (SqlCommand StrQuer = new SqlCommand

                        (

                        "BEGIN TRANSACTION " +

                        $"INSERT INTO contdepo (Id, contid, sum, opens, closes, prdepo, contdate, clipas,  emppas, accid) " +

                        $"VALUES ('{textBox1.Text}', '{textBox2.Text}', '{textBox12.Text.Replace(", 0000","")}', '{sqlFormattedDate11}', " +

                        $"'{sqlFormattedDate3}', '{prcode}', '{sqlFormattedDate5}', '{textBox6.Text}', '{textBox7.Text}', '{textBox8.Text}'); " +

                        $"INSERT INTO contract (Id) " +

                        $"VALUES ('{textBox2.Text}');" +

                        $" COMMIT", conn))

                {

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    conn.Close();

                }

                MessageBox.Show("Изменения сохранены");

                //В конце обновляем выбранный контракт

                loadInfodepo(textBox2.Text);

            }

            ronly();

            edit.Enabled = true;

            create.Enabled = true;

            save.Visible = false;

            revert.Visible = false;

        }

        //отмена

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

        {

            if (Program.curcont != null)

                loadInfodepo(Program.curcont);

            else

                wipe();

            save.Visible = false;

            revert.Visible = false;

            ronly();

            edit.Enabled = true;

            create.Enabled = true;

        }

        //создать

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

        {

            Program.curcont = null;

            wipe();

            write();

            edit.Enabled = false;

            save.Visible = true;

            revert.Visible = true;

            textBox7.Text = Program.emppas;

            textBox1.ReadOnly = true;

            textBox2.ReadOnly = true;

            textBox5.Text = DateTime.Now.ToShortDateString();

            //лочим кнопки договора 1 2

            //ищем макс знач в contract, depo

            //парсим макс+1 в текстбоксы

            var conn = new SqlConnection();

            conn.ConnectionString = Program.str;

            conn.Open();

            int maxloc = -1, maxc = -1;

            using (SqlCommand StrQuer = new SqlCommand($"SELECT MAX(contid) FROM contdepo", conn))

            {

                SqlDataReader dr = StrQuer.ExecuteReader();

                using (dr)

                    while (dr.Read())

                    {

                        maxloc = dr.GetInt32(0);

                    }

            }

            using (SqlCommand StrQuer = new SqlCommand($"SELECT MAX(Id) FROM contract", conn))

            {

                SqlDataReader dr = StrQuer.ExecuteReader();

                using (dr)

                    while (dr.Read())

                    {

                        maxc = dr.GetInt32(0);

                    }

            }

            textBox1.Text = (maxloc + 1).ToString();

            textBox2.Text = (maxc + 1).ToString();

        }

    }

}

Addloan.cs

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.Data.SqlClient;

namespace Bank

{

    public partial class addloan : Form

    {

        public addloan()

        {

            InitializeComponent();

        }

        Dictionary<int, string> test = new Dictionary<int, string>();

        private void addcard\_VisibleChanged(object sender, EventArgs e)

        {

            if (Program.isgovernor)

            {

                deletecon.Visible = true; //управляющий может удалять

            }

        }

        public void loadInfoloan(string contnum) //загрузка данных по ид договора

        {

            Program.curcont = contnum;

            var conn = new SqlConnection();

            conn.ConnectionString = Program.str;

            using (SqlCommand StrQuer = new SqlCommand("SELECT \* " +

                "FROM contloan INNER JOIN prloan ON contloan.prloan = prloan.Id " +

                $"WHERE contloan.contid = {contnum}", conn))

            {

                conn.Open();

                SqlDataReader dr = StrQuer.ExecuteReader();

                using (dr) //для проверки на регистр в логине/пароле

                {

                    while (dr.Read())

                    {  //вставка полей в textbox

                        textBox1.Text = dr.GetInt32(0).ToString(); //ид вклада

                        textBox2.Text = dr.GetInt32(1).ToString(); //ид договора

                        textBox12.Text = dr.GetDecimal(2).ToString(); //сумма вклада

                        textBox11.Text = dr.GetDateTime(3).Date.ToString("d"); //нач

                        textBox3.Text = dr.GetDateTime(4).Date.ToString("d"); //заканчивается

                        comboBox1.SelectedIndex = comboBox1.FindStringExact(test[dr.GetInt32(5)]); //код программы вклада

                        textBox5.Text = dr.GetDateTime(6).Date.ToString("d"); //заключение

                        textBox6.Text = dr.GetString(7); //паспорт клиента

                        textBox7.Text = dr.GetString(8); //паспорт сотрудника

                        textBox8.Text = dr.GetInt32(9).ToString(); // номер счета

                        textBox9.Text = dr.GetDecimal(12).ToString(); // loan lim

                        textBox10.Text = dr.GetDecimal(13).ToString(); // loan max

                        textBox4.Text = dr.GetDouble(14).ToString(); //проц

                    }

                }

                conn.Close();

            }

        }

        private void addcard\_Load(object sender, EventArgs e) //загрузка словаря для комбобокса из ДБ

        {

            var conn = new SqlConnection();

            conn.ConnectionString = Program.str;

            using (SqlCommand StrQuer = new SqlCommand("SELECT \* " +

                "FROM prloan ", conn))

            {

                conn.Open();

                SqlDataReader dr = StrQuer.ExecuteReader();

                using (dr) //для проверки на регистр в логине/пароле

                {

                    while (dr.Read())

                    {  //вставка полей в textbox

                        int tmp;

                        string tmps;

                        tmp = dr.GetInt32(0); //ид программы

                        tmps = dr.GetString(1); //ид договора

                        test.Add(tmp, tmps);

                    }

                }

                conn.Close();

            }

            comboBox1.DataSource = new BindingSource(test, null);

            comboBox1.DisplayMember = "Value";

            comboBox1.ValueMember = "Key";

        }

        #region texbox

        //Для блока текстбоксов

        public void ronly()

        {

            textBox1.ReadOnly = true;

            textBox2.ReadOnly = true;

            textBox3.ReadOnly = true;

            comboBox1.Enabled = false;

            textBox5.ReadOnly = true;

            textBox6.ReadOnly = true;

            textBox7.ReadOnly = true;

            textBox8.ReadOnly = true;

            textBox11.ReadOnly = true;

            textBox12.ReadOnly = true;

        }

        public void write()

        {

            textBox1.ReadOnly = false;

            textBox2.ReadOnly = false;

            textBox3.ReadOnly = false;

            comboBox1.Enabled = true;

            textBox5.ReadOnly = false;

            textBox6.ReadOnly = false;

            textBox7.ReadOnly = false;

            textBox8.ReadOnly = false;

            textBox11.ReadOnly = false;

            textBox12.ReadOnly = false;

        }

        public void wipe()

        {

            textBox1.Text = null;

            textBox2.Text = null;

            textBox3.Text = null;

            textBox4.Text = null;

            textBox5.Text = null;

            textBox6.Text = null;

            textBox7.Text = null;

            textBox8.Text = null;

            textBox9.Text = null;

            textBox10.Text = null;

            textBox11.Text = null;

            textBox12.Text = null;

        }

        #endregion

        //удаление

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

        {

            DialogResult dialogResult = MessageBox.Show($"Удаление договора {Program.curcont}, уверены?", "Внимание", MessageBoxButtons.YesNo);

            if (dialogResult == DialogResult.Yes)

            {

                var conn = new SqlConnection();

                conn.ConnectionString = Program.str;

                using (SqlCommand StrQuer = new SqlCommand("BEGIN TRANSACTION " +

                    $"DELETE FROM contract WHERE Id = '{Program.curcont}'; " +

                    $"DELETE FROM contloan WHERE contid = '{Program.curcont}';" +

                    $"COMMIT", conn))

                {

                    conn.Open();

                    SqlDataReader dr = StrQuer.ExecuteReader();

                }

                ronly();

                wipe();

                edit.Enabled = true;

                create.Enabled = true;

                save.Visible = false;

                revert.Visible = false;

                MessageBox.Show("Удаление завершено");

            }

            else if (dialogResult == DialogResult.No)

            {

                return;

            }

        }

        //Редактирование

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

        {

            create.Enabled = false;

            save.Visible = true;

            revert.Visible = true;

            write();

        }

        //сохранение

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

        {

            var conn = new SqlConnection();

            conn.ConnectionString = Program.str;

            if (edit.Enabled == true)  //редакт

            {

                //Получение ном дог и вклада текущ договора

                string cardid = "-1", conid = "-1";

                using (SqlCommand StrQuer = new SqlCommand("SELECT Id, contid " +

                    "FROM contloan " +

                    $"WHERE contid = {Program.curcont}", conn))

                {

                    conn.Open();

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    using (dr)

                    {

                        while (dr.Read())

                        {  //вставка полей в textbox

                            cardid = dr.GetInt32(0).ToString(); //ид карты

                            conid = dr.GetInt32(1).ToString(); //ид договора

                        }

                    }

                    conn.Close();

                }

                //проверка текущего номера договора и карты на повторение (кроме себя самого)

                bool k = true; ;

                using (SqlCommand StrQuer = new SqlCommand("SELECT contloan.Id, contract.Id " +

                    "FROM contloan INNER JOIN contract ON contloan.contid = contract.Id " +

                    $"WHERE contloan.contid = '{textBox1.Text}' " +

                    $"OR contract.Id = '{textBox2.Text}'", conn))

                {

                    conn.Open();

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    using (dr)

                    {

                        bool found = false;

                        while (dr.Read())

                        {

                            found = true;

                            string tcardid = "-1", tid = "-1";

                            tcardid = dr.GetInt32(0).ToString(); //номер карты

                            tid = dr.GetInt32(1).ToString(); //номер договора

                            //Если нашли старую запись - скип

                            if (String.Equals(cardid, tcardid) && String.Equals(conid, tid))

                            {

                                continue;

                            }

                            if (String.Equals(textBox1.Text, tcardid))

                            {

                                k = false;

                                textBox1.Text = null;

                            }

                            if (String.Equals(textBox2.Text, tid))

                            {

                                k = false;

                                textBox2.Text = null;

                            }

                        }

                        conn.Close();

                        if (!k)

                        {

                            MessageBox.Show("Не могут повторяться: номера счетов");

                            return;

                        }

                        if (!found)

                        {

                            MessageBox.Show("Номера договора вклада не существует");

                            textBox1.Text = null;

                            textBox2.Text = null;

                            return;

                        }

                    }

                }

                k = false;

                //проверка на существование паспорта сотруд

                using (SqlCommand StrQuer = new SqlCommand("SELECT emppas FROM emp " +

                    $"WHERE emppas = '{textBox7.Text}'", conn))

                {

                    conn.Open();

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    using (dr)

                    {

                        string empid = "-1";

                        while (dr.Read())

                        {

                            empid = dr.GetString(0); //пасп сотруд

                            //Если нашли запись - существует

                            if (String.Equals(textBox7.Text, empid))

                            {

                                k = true;

                            }

                        }

                        conn.Close();

                        if (!k)

                        {

                            MessageBox.Show("Сотрудника не существует");

                            return;

                        }

                    }

                }

                //существует ли счет

                bool t = true; //счет не существует?

                using (SqlCommand StrQuer = new SqlCommand("SELECT Id, clipas FROM acc " +

                   $"WHERE Id = '{textBox8.Text}'", conn))

                {

                    conn.Open();

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    using (dr)

                    {

                        k = true;

                        while (dr.Read())

                        {

                            string tid = "-1", tpas = "-1";

                            tid = dr.GetInt32(0).ToString(); //счет

                            tpas = dr.GetString(1); //паспорт кли

                            //Если нашли запись - существует

                            if (String.Equals(textBox8.Text, tid))

                            {

                                t = false;

                            }

                            if (!String.Equals(tpas, textBox6.Text))

                                k = false;

                        }

                        conn.Close();

                    }

                }

                if (!k)

                {

                    MessageBox.Show("Данный счет уже занят");

                    return;

                }

                //проверка на существование паспорта клиента

                using (SqlCommand StrQuer = new SqlCommand("SELECT clipas FROM cli " +

                   $"WHERE clipas = '{textBox6.Text}'", conn))

                {

                    string tcli = "-1";

                    conn.Open();

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    using (dr)

                    {

                        while (dr.Read())

                        {

                            string empid = "-1";

                            empid = dr.GetString(0); //пасп клиента

                            //Если нашли запись - существует

                            if (String.Equals(textBox6.Text, empid))

                            {

                                tcli = empid;

                                k = true;

                            }

                        }

                        conn.Close();

                        if (!k)

                        {

                            MessageBox.Show("Клиента не существует");

                            return;

                        }

                    }

                }

                //Добавляем счет, если нет

                if (t)

                {

                    conn.Open();

                    DateTime myDateTime1 = DateTime.Now;

                    DateTime myDateTime2 = DateTime.Now.AddYears(4);

                    string sqlFormattedDate1 = myDateTime1.ToString("yyyy-MM-dd HH:mm:ss.fff");

                    string sqlFormattedDate2 = myDateTime2.ToString("yyyy-MM-dd HH:mm:ss.fff");

                    using (SqlCommand StrQuer = new SqlCommand("INSERT INTO acc (Id, opens, closes, clipas) " +

                        $"VALUES ('{textBox8.Text}', '{sqlFormattedDate1}', '{sqlFormattedDate2}', '{textBox6.Text}');", conn))

                    {

                        SqlDataReader dr = StrQuer.ExecuteReader();

                        conn.Close();

                    }

                }

                DateTime myDateTime11 = Convert.ToDateTime(textBox11.Text);

                DateTime myDateTime3 = Convert.ToDateTime(textBox3.Text);

                DateTime myDateTime5 = Convert.ToDateTime(textBox5.Text);

                string sqlFormattedDate3 = myDateTime3.ToString("yyyy-MM-dd HH:mm:ss.fff");

                string sqlFormattedDate5 = myDateTime5.ToString("yyyy-MM-dd HH:mm:ss.fff");

                string sqlFormattedDate11 = myDateTime11.ToString("yyyy-MM-dd HH:mm:ss.fff");

                //Апдейт

                conn.Open();

                int prcode = ((KeyValuePair<int, string>)comboBox1.SelectedItem).Key;

                using (SqlCommand StrQuer = new SqlCommand

                        (

                        $"UPDATE contloan SET Id = N'{textBox1.Text}', contid = N'{textBox2.Text}', sum = N'{textBox12.Text.Replace(",0000", "")}', " +

                        $"opens = N'{sqlFormattedDate11}', closes = N'{sqlFormattedDate3}', prloan = N'{prcode}', " +

                        $"contdate = '{sqlFormattedDate5}', clipas = N'{textBox6.Text}', emppas = N'{textBox7.Text}', accid = N'{textBox8.Text}' " +

                        $" WHERE (contid = '{textBox2.Text}');", conn))

                {

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    conn.Close();

                }

                MessageBox.Show("Изменения сохранены");

                loadInfoloan(textBox2.Text);

            }

            else //добавляем

            {

                textBox7.ReadOnly = true;

                bool k = false;

                //нужна проверка на наличие паспорта клиента в бд, если нет - ошибка

                using (SqlCommand StrQuer = new SqlCommand("SELECT clipas FROM cli " +

                  $"WHERE clipas = '{textBox6.Text}'", conn))

                {

                    string tcli = "-1";

                    conn.Open();

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    using (dr)

                    {

                        while (dr.Read())

                        {

                            string empid = "-1";

                            empid = dr.GetString(0); //пасп клиента

                            //Если нашли запись - существует

                            if (String.Equals(textBox6.Text, empid))

                            {

                                tcli = empid;

                                k = true;

                            }

                        }

                        conn.Close();

                        if (!k)

                        {

                            MessageBox.Show("Клиента не существует");

                            return;

                        }

                    }

                }

                //существует ли счет

                bool t = true; //счет не существует?

                using (SqlCommand StrQuer = new SqlCommand("SELECT Id, clipas FROM acc " +

                   $"WHERE Id = '{textBox8.Text}'", conn))

                {

                    conn.Open();

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    using (dr)

                    {

                        k = true;

                        while (dr.Read())

                        {

                            string tid = "-1", tpas = "-1";

                            tid = dr.GetInt32(0).ToString(); //счет

                            tpas = dr.GetString(1); //паспорт кли

                            //Если нашли запись - существует

                            if (String.Equals(textBox8.Text, tid))

                            {

                                t = false;

                            }

                            if (!String.Equals(tpas, textBox6.Text))

                                k = false;

                        }

                        conn.Close();

                    }

                }

                if (!k)

                {

                    MessageBox.Show("Данный счет уже занят");

                    return;

                }

                //Добавляем счет, если нет

                if (t)

                {

                    conn.Open();

                    DateTime myDateTime1 = DateTime.Now;

                    DateTime myDateTime2 = DateTime.Now.AddYears(4);

                    string sqlFormattedDate1 = myDateTime1.ToString("yyyy-MM-dd HH:mm:ss.fff");

                    string sqlFormattedDate2 = myDateTime2.ToString("yyyy-MM-dd HH:mm:ss.fff");

                    using (SqlCommand StrQuer = new SqlCommand("INSERT INTO acc (Id, opens, closes, clipas) " +

                        $"VALUES ('{textBox8.Text}', '{sqlFormattedDate1}', '{sqlFormattedDate2}', '{textBox6.Text}');", conn))

                    {

                        SqlDataReader dr = StrQuer.ExecuteReader();

                        conn.Close();

                    }

                }

                DateTime myDateTime11 = Convert.ToDateTime(textBox11.Text);

                DateTime myDateTime3 = Convert.ToDateTime(textBox3.Text);

                DateTime myDateTime5 = Convert.ToDateTime(textBox5.Text);

                string sqlFormattedDate3 = myDateTime3.ToString("yyyy-MM-dd HH:mm:ss.fff");

                string sqlFormattedDate5 = myDateTime5.ToString("yyyy-MM-dd HH:mm:ss.fff");

                string sqlFormattedDate11 = myDateTime11.ToString("yyyy-MM-dd HH:mm:ss.fff");

                conn.Open();

                int prcode = ((KeyValuePair<int, string>)comboBox1.SelectedItem).Key;

                using (SqlCommand StrQuer = new SqlCommand

                        (

                        "BEGIN TRANSACTION " +

                        $"INSERT INTO contloan (Id, contid, sum, opens, closes, prloan, contdate, clipas,  emppas, accid) " +

                        $"VALUES ('{textBox1.Text}', '{textBox2.Text}', '{textBox12.Text.Replace(", 0000", "")}', '{sqlFormattedDate11}', " +

                        $"'{sqlFormattedDate3}', '{prcode}', '{sqlFormattedDate5}', '{textBox6.Text}', '{textBox7.Text}', '{textBox8.Text}'); " +

                        $"INSERT INTO contract (Id) " +

                        $"VALUES ('{textBox2.Text}');" +

                        $" COMMIT", conn))

                {

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    conn.Close();

                }

                MessageBox.Show("Изменения сохранены");

                //В конце обновляем выбранный контракт

                loadInfoloan(textBox2.Text);

            }

            ronly();

            edit.Enabled = true;

            create.Enabled = true;

            save.Visible = false;

            revert.Visible = false;

        }

        //отмена

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

        {

            if (Program.curcont != null)

                loadInfoloan(Program.curcont);

            else

                wipe();

            save.Visible = false;

            revert.Visible = false;

            ronly();

            edit.Enabled = true;

            create.Enabled = true;

        }

        //создать

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

        {

            Program.curcont = null;

            wipe();

            write();

            edit.Enabled = false;

            save.Visible = true;

            revert.Visible = true;

            textBox7.Text = Program.emppas;

            textBox1.ReadOnly = true;

            textBox2.ReadOnly = true;

            textBox5.Text = DateTime.Now.ToShortDateString();

            //лочим кнопки договора 1 2

            //ищем макс знач в contract, depo

            //парсим макс+1 в текстбоксы

            var conn = new SqlConnection();

            conn.ConnectionString = Program.str;

            conn.Open();

            int maxloc = -1, maxc = -1;

            using (SqlCommand StrQuer = new SqlCommand($"SELECT MAX(contid) FROM contdepo", conn))

            {

                SqlDataReader dr = StrQuer.ExecuteReader();

                using (dr)

                    while (dr.Read())

                    {

                        maxloc = dr.GetInt32(0);

                    }

            }

            using (SqlCommand StrQuer = new SqlCommand($"SELECT MAX(Id) FROM contract", conn))

            {

                SqlDataReader dr = StrQuer.ExecuteReader();

                using (dr)

                    while (dr.Read())

                    {

                        maxc = dr.GetInt32(0);

                    }

            }

            textBox1.Text = (maxloc + 1).ToString();

            textBox2.Text = (maxc + 1).ToString();

        }

    }

}

Clieinfo.cs

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.Data.SqlClient;

namespace Bank

{

    public partial class clieinfo : Form

    {

        public clieinfo()

        {

            InitializeComponent();

        }

        Dictionary<int, string> test2 = new Dictionary<int, string>();

        Dictionary<int, string> test3 = new Dictionary<int, string>();

        public void cliinfo\_Load(object sender, EventArgs e)

        {

            // Загрузка возможных счетов ид закр

            var conn = new SqlConnection();

            conn.ConnectionString = Program.str;

            // Город

            using (SqlCommand StrQuer = new SqlCommand("SELECT \* FROM city", conn))

            {

                conn.Open();

                SqlDataReader dr = StrQuer.ExecuteReader();

                using (dr)

                {

                    while (dr.Read())

                    {  //вставка полей в textbox

                        int tmp;

                        string tmps;

                        tmp = dr.GetInt32(0); //ид города

                        tmps = dr.GetString(1); //название

                        test3.Add(tmp, tmps);

                    }

                }

                conn.Close();

                comboBox3.DataSource = new BindingSource(test3, null); // города

                comboBox3.DisplayMember = "Value";

                comboBox3.ValueMember = "Key";

                //((KeyValuePair<int, string>)comboBox1.SelectedItem).Key.ToString(); //вот так получать 0 1 из комбобокса

            }

        }

        public void clidata(string ppas) //парс даты по паспорту

        {

            var conn = new SqlConnection();

            conn.ConnectionString = Program.str;

            test2.Clear();

            using (SqlCommand StrQuer = new SqlCommand("SELECT \* " +

          $"FROM acc WHERE clipas = '{Program.curcpas}'", conn))

            {

                conn.Open();

                SqlDataReader dr = StrQuer.ExecuteReader();

                using (dr)

                {

                    while (dr.Read())

                    {  //вставка полей в textbox

                        int tmp;

                        string tmps;

                        tmp = dr.GetInt32(0); //ид

                        tmps = dr.GetDateTime(2).ToString(); //закрывается

                        test2.Add(tmp, tmps);

                    }

                }

                conn.Close();

            }

            comboBox2.DataSource = new BindingSource(test2, null); //счета

            comboBox2.DisplayMember = "Key";

            comboBox2.ValueMember = "Key";

            using (SqlCommand StrQuer = new SqlCommand("SELECT \* " +

                $"FROM cli WHERE clipas = '{ppas}' ", conn)) //берем всю запись

            {

                conn.Open();

                SqlDataReader dr = StrQuer.ExecuteReader();

                using (dr) //для проверки на регистр в логине/пароле

                {

                    while (dr.Read())

                    {  //вставка полей в textbox

                        textBox1.Text = dr.GetString(0); //пасс

                        textBox2.Text = dr.GetString(1); //ФИО

                        textBox3.Text = dr.GetString(2);

                        textBox4.Text = dr.GetString(3);

                        textBox5.Text = dr.GetDateTime(4).Date.ToString("d"); //др

                        textBox6.Text = dr.GetDecimal(5).ToString(); //доход

                        textBox10.Text = dr.GetString(6); //инн

                        textBox9.Text = dr.GetString(7); //снилс

                        comboBox3.SelectedIndex = comboBox3.FindStringExact(test3[dr.GetInt32(8)]);  //город

                        textBox7.Text = dr.GetString(9); //адрес

                        textBox13.Text = dr.GetString(10); //тек адрес

                        textBox12.Text = dr.GetString(11); //тел личный

                        textBox11.Text = dr.GetString(12); //тел рабоч

                        textBox17.Text = dr.GetString(13); //емейл

                        comboBox2.SelectedIndex = comboBox2.FindStringExact(test2[dr.GetInt32(14)]);  //ид счета

                        textBox15.Text = dr.GetDateTime(15).Date.ToString("d"); //updated

                    }

                }

                conn.Close();

            }

        }

        #region texbox

        //Для блока текстбоксов

        public void ronly()

        {

            textBox1.ReadOnly = true;

            textBox2.ReadOnly = true;

            textBox3.ReadOnly = true;

            textBox4.ReadOnly = true;

            textBox5.ReadOnly = true;

            comboBox2.Enabled = false;

            textBox6.ReadOnly = true;

            textBox7.ReadOnly = true;

            comboBox2.Enabled = false;

            textBox9.ReadOnly = true;

            textBox10.ReadOnly = true;

            textBox11.ReadOnly = true;

            textBox12.ReadOnly = true;

            textBox13.ReadOnly = true;

            textBox15.ReadOnly = true;

            textBox17.ReadOnly = true;

        }

        public void write()

        {

            textBox1.ReadOnly = false;

            textBox2.ReadOnly = false;

            textBox3.ReadOnly = false;

            textBox4.ReadOnly = false;

            textBox5.ReadOnly = false;

            comboBox2.Enabled = true;

            textBox6.ReadOnly = false;

            textBox7.ReadOnly = false;

            comboBox3.Enabled = true;

            textBox9.ReadOnly = false;

            textBox10.ReadOnly = false;

            textBox11.ReadOnly = false;

            textBox12.ReadOnly = false;

            textBox13.ReadOnly = false;

            textBox15.ReadOnly = false;

            textBox17.ReadOnly = false;

        }

        public void wipe()

        {

            textBox1.Text = null;

            textBox2.Text = null;

            textBox3.Text = null;

            textBox4.Text = null;

            textBox5.Text = null;

            textBox6.Text = null;

            textBox7.Text = null;

            textBox9.Text = null;

            textBox10.Text = null;

            textBox11.Text = null;

            textBox12.Text = null;

            textBox13.Text = null;

            textBox15.Text = null;

            textBox17.Text = null;

        }

        //Включение отключение кнопок

        public void changeBut(byte i)

        {

            if (i == 1) //вкл

            {

                write();

                button1.Visible = true;

                button6.Visible = true;

            }

            else

            {

                button1.Visible = false;

                button6.Visible = false;

                button4.Enabled = true;

                button5.Enabled = true;

            }

        }

        #endregion

        private void clieinfo\_VisibleChanged(object sender, EventArgs e)

        {

        }

        //редактирование

        public void button4\_Click\_1(object sender, EventArgs e)

        {

            Program.curepas = textBox1.Text;

            write();

            button5.Enabled = false;

            changeBut(1);

        }

        //добавление

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

        {

            wipe();

            button4.Enabled = false;

            changeBut(1);

            var conn = new SqlConnection();

            conn.ConnectionString = Program.str;

            //comboBox2.Enabled = false;

            int maxid = 0;

            conn.Open();

            using (SqlCommand StrQuer = new SqlCommand($"SELECT MAX(Id) FROM acc", conn)) //добавляем счет сразу

            {

                SqlDataReader dr = StrQuer.ExecuteReader();

                using (dr)

                    while (dr.Read())

                    {

                        maxid = 1 + dr.GetInt32(0);

                    }

            }

            textBox8.Text = maxid.ToString();

            textBox15.Text = DateTime.Now.ToShortDateString();

            textBox15.ReadOnly = true;

            textBox8.Visible = true;

            comboBox2.Enabled = false;

            comboBox2.Visible = false;

            button2.PerformClick();

        }

        // Отмена

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

        {

            textBox8.Visible = false;

            changeBut(0);

            if (Program.curcpas == null) //при добавлении нового сотрудника - зануляем

                wipe();

            else

            {

                clidata(Program.curcpas);

                button2.PerformClick();

            }

            ronly();

        }

        // Сохранение

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

        {

            var conn = new SqlConnection();

            conn.ConnectionString = Program.str;

            if (button4.Enabled == true)  //редакт

            {

                bool k = true;

                //Данные текущего клиента

                string clipas = "0", inn = "0", snils = "0";

                using (SqlCommand StrQuer = new SqlCommand($"SELECT clipas, inn, snils " +

                    $"FROM cli WHERE clipas = N'{Program.curcpas}'", conn))

                {

                    conn.Open();

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    using (dr)

                    {

                        while (dr.Read())

                        {

                            clipas = dr.GetString(0); //пасс

                            inn = dr.GetString(1); //инн

                            snils = dr.GetString(2); //снилс

                        }

                    }

                    conn.Close();

                }

                //Проверка данных текущего клиента с тем, что найдено в дб

                using (SqlCommand StrQuer = new SqlCommand($"SELECT clipas, inn, snils " +

                $"FROM cli WHERE clipas LIKE '{textBox1.Text}%' OR inn LIKE '{textBox10.Text}%' " +

                $"OR snils LIKE '{textBox9.Text}%' ", conn))

                {

                    conn.Open();

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    using (dr)

                    {

                        while (dr.Read())

                        {

                            string tpas = "0", tinn = "0", tsnils = "0";

                            tpas = dr.GetString(0); //пасс

                            tinn = dr.GetString(1); //инн

                            tsnils = dr.GetString(2); //снилс

                            //Если нашли старую запись - скип

                            if (String.Equals(clipas, tpas) && String.Equals(inn, tinn) && String.Equals(snils, tsnils))

                            {

                                continue;

                            }

                            if (String.Equals(textBox1.Text, tpas))

                            {

                                k = false;

                                textBox1.Text = null;

                            }

                            if (String.Equals(textBox10.Text, tinn))

                            {

                                k = false;

                                textBox10.Text = null;

                            }

                            if (String.Equals(textBox9.Text, tsnils))

                            {

                                k = false;

                                textBox9.Text = null;

                            }

                        }

                        conn.Close();

                        if (!k)

                        {

                            MessageBox.Show("Не могут повторяться: паспорт, ИНН, СНИЛС");

                            return;

                        }

                    }

                }

                int citycode = ((KeyValuePair<int, string>)comboBox3.SelectedItem).Key;

                string acccode = ((KeyValuePair<int, string>)comboBox2.SelectedItem).Key.ToString();

                conn.Open();

                DateTime myDateTime1 = Convert.ToDateTime(textBox5.Text);

                string sqlFormattedDate1 = myDateTime1.ToString("yyyy-MM-dd HH:mm:ss.fff");

                DateTime myDateTime2 = DateTime.Now;

                string sqlFormattedDate2 = myDateTime2.ToString("yyyy-MM-dd HH:mm:ss.fff");

                using (SqlCommand StrQuer = new SqlCommand

                        ("BEGIN TRANSACTION " +

                        $"UPDATE contcard SET contcard.clipas = '{textBox1.Text}' WHERE(((contcard.clipas) = '{Program.curcpas}'));" +

                        $"UPDATE contdepo SET contdepo.clipas = '{textBox1.Text}' WHERE(((contdepo.clipas) = '{Program.curcpas}'));" +

                        $"UPDATE contloan SET contloan.clipas = '{textBox1.Text}' WHERE(((contloan.clipas) = '{Program.curcpas}')); " +

                        $"UPDATE cli SET cli.clipas = N'{textBox1.Text}', cli.fam = N'{textBox2.Text}', cli.name = N'{textBox3.Text}', " +

                        $"cli.sname = N'{textBox4.Text}', cli.birth = '{sqlFormattedDate1}', " +

                        $"cli.income = N'{textBox6.Text.Replace(",0000", "")}', cli.inn = '{textBox10.Text}', cli.snils = '{textBox9.Text}', cli.city = '{citycode}', " +

                        $"cli.adr = N'{textBox7.Text}', cli.cadr = N'{textBox13.Text}', cli.phone = N'{textBox12.Text}', " +

                        $"cli.jphone = N'{textBox11.Text}', cli.email = N'{textBox17.Text}', cli.updated = '{sqlFormattedDate2}', cli.accnum = '{acccode}'" +

                        $" WHERE (cli.clipas = '{Program.curcpas}'); " +

                        $" COMMIT", conn))

                {

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    conn.Close();

                }

                MessageBox.Show("Изменения сохранены");

            }

            else //INSERT

            {

                //проверка на повторение emppas, inn, snils

                if (checkcdata())

                {

                    //Получаем код города

                    int citycode = ((KeyValuePair<int, string>)comboBox3.SelectedItem).Key;

                    string acccode = textBox8.Text; //код аккаунта

                    DateTime myDateTime1 = DateTime.Now;

                    DateTime myDateTime2 = DateTime.Now.AddYears(4);

                    string sqlFormattedDate1 = myDateTime1.ToString("yyyy-MM-dd HH:mm:ss.fff");

                    string sqlFormattedDate2 = myDateTime2.ToString("yyyy-MM-dd HH:mm:ss.fff");

                    DateTime myDateTime3 = Convert.ToDateTime(textBox5.Text);

                    string sqlFormattedDate3 = myDateTime3.ToString("yyyy-MM-dd HH:mm:ss.fff");

                    conn.Open();

                    using (SqlCommand StrQuer = new SqlCommand("BEGIN TRANSACTION " +

                        "INSERT INTO cli (clipas, fam, name, sname, birth, income, inn, snils, city," +

                        " adr, cadr, phone, jphone, email, accnum, updated) " +

                        $"VALUES ('{textBox1.Text}', N'{textBox2.Text}', N'{textBox3.Text}', N'{textBox3.Text}', '{sqlFormattedDate3}', '{textBox6.Text}', " +

                        $"'{textBox10.Text}', '{textBox9.Text}', '{citycode}', N'{textBox7.Text}', N'{textBox13.Text}', '{textBox12.Text}', '{textBox11.Text}', " +

                        $"'{textBox17.Text}', '{acccode}', '{sqlFormattedDate1}'); " +

                        $"INSERT INTO acc (Id, opens, closes, clipas) " +

                        $"VALUES ('{acccode}', '{sqlFormattedDate1}', '{sqlFormattedDate2}', '{textBox1.Text}'); " +

                        $"COMMIT", conn))

                    {

                        SqlDataReader dr = StrQuer.ExecuteReader();

                        conn.Close();

                    }

                    MessageBox.Show("Изменения сохранены");

                    test2.Clear();

                    int tmp = Int32.Parse(acccode);

                    test2.Add(tmp, myDateTime2.ToShortDateString());

                }

            }

            textBox8.Visible = false;

            changeBut(0);

            ronly();

            Program.curcpas = textBox1.Text;

            clidata(Program.curcpas);

            button2.PerformClick();

            comboBox2.Visible = true;

        }

        public bool checkcdata()  //проверка на повторение clipas, inn, snils

        {

            bool k = true;

            var conn = new SqlConnection();

            conn.ConnectionString = Program.str;

            using (SqlCommand StrQuer = new SqlCommand($"SELECT clipas, inn, snils " +

                $"FROM cli WHERE clipas LIKE '{textBox1.Text}%' AND inn LIKE '{textBox10.Text}%' " +

                $"AND snils LIKE '{textBox9.Text}%'", conn))

            {

                conn.Open();

                SqlDataReader dr = StrQuer.ExecuteReader();

                using (dr)

                {

                    while (dr.Read())

                    {

                        string tpas, tinn, tsnils;

                        string res = "Не могут повторяться:\n";

                        tpas = dr.GetString(0); //пасс

                        tinn = dr.GetString(1); //инн

                        tsnils = dr.GetString(2); //снилс

                        if (String.Equals(textBox1.Text, tpas))

                        {

                            k = false;

                            res = res + "паспорт\n";

                            textBox1.Text = null;

                        }

                        if (String.Equals(textBox10.Text, tinn))

                        {

                            k = false;

                            res = res + "ИНН\n";

                            textBox10.Text = null;

                        }

                        if (String.Equals(textBox9.Text, tsnils))

                        {

                            k = false;

                            res = res + "СНИЛС\n";

                            textBox9.Text = null;

                        }

                        if (!k)

                            MessageBox.Show(res);

                    }

                }

                conn.Close();

            }

            return k;

        }

        //Обновление информации о человеке чей паспорт в textbox1

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

        {

            dataGridView1.Rows.Clear();

            var conn = new SqlConnection();

            conn.ConnectionString = Program.str;

            List<string[]> data = new List<string[]>();

            conn.Open();

            using (SqlCommand StrQuer = new SqlCommand($"SELECT contdepo.contid, contdate FROM contdepo WHERE(((contdepo.clipas) = '{textBox1.Text}'));", conn))

            {

                SqlDataReader dr = StrQuer.ExecuteReader();

                using (dr)

                {

                    while (dr.Read())

                    {

                        data.Add(new string[2]);

                        data[data.Count - 1][0] = dr[0].ToString();

                        data[data.Count - 1][1] = dr[1].ToString().Replace(" 0:00:00", "");

                    }

                }

            }

            using (SqlCommand StrQuer = new SqlCommand($"SELECT contcard.contid, contdate FROM contcard WHERE(((contcard.clipas) = '{textBox1.Text}'));", conn))

            {

                SqlDataReader dr = StrQuer.ExecuteReader();

                using (dr)

                {

                    while (dr.Read())

                    {

                        data.Add(new string[2]);

                        data[data.Count - 1][0] = dr[0].ToString();

                        data[data.Count - 1][1] = dr[1].ToString().Replace(" 0:00:00", "");

                    }

                }

            }

            using (SqlCommand StrQuer = new SqlCommand($"SELECT contloan.contid, contdate FROM contloan WHERE(((contloan.clipas) = '{textBox1.Text}'));", conn))

            {

                SqlDataReader dr = StrQuer.ExecuteReader();

                using (dr)

                {

                    while (dr.Read())

                    {

                        data.Add(new string[2]);

                        data[data.Count - 1][0] = dr[0].ToString();

                        data[data.Count - 1][1] = dr[1].ToString().Replace(" 0:00:00", "");

                    }

                    foreach (string[] s in data)

                        dataGridView1.Rows.Add(s);

                }

            }

            conn.Close();

            dataGridView1.Sort(dataGridView1.Columns[1], ListSortDirection.Ascending);

        }

        private void dataGridView1\_CellClick(object sender, DataGridViewCellEventArgs e) //получаем id из клетки и переходим в форму договора

        {

            try //при даблклике по названию колонки

            {

                var conn = new SqlConnection();

                conn.ConnectionString = Program.str;

                var item = dataGridView1.Rows[e.RowIndex].Cells[0].Value;

                int loan, depo, card;

                conn.Open();

                using (SqlCommand StrQuer = new SqlCommand($"SELECT contid FROM contcard WHERE contid=N'{item}'", conn))

                {

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    using (dr)

                        while (dr.Read())

                        {

                            card = dr.GetInt32(0);

                            addcard tmp = new addcard();

                            tmp.Show();

                            tmp.loadinfocard(card.ToString());

                            conn.Close();

                            return;

                        }

                }

                using (SqlCommand StrQuer = new SqlCommand($"SELECT contid FROM contdepo WHERE contid=N'{item}'", conn))

                {

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    using (dr)

                        while (dr.Read())

                        {

                            depo = dr.GetInt32(0);

                            adddepo tmp = new adddepo();

                            tmp.Show();

                            tmp.loadInfodepo(depo.ToString());

                            conn.Close();

                            return;

                        }

                }

                using (SqlCommand StrQuer = new SqlCommand($"SELECT contid FROM contloan WHERE contid=N'{item}'", conn))

                {

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    using (dr)

                        while (dr.Read())

                        {

                            loan = dr.GetInt32(0);

                            addloan tmp = new addloan();

                            tmp.Show();

                            tmp.loadInfoloan(loan.ToString());

                            conn.Close();

                            return;

                        }

                }

            }

            catch (Exception)

            { }

        }

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

        {

            addcard tmp = new addcard();

            tmp.Show();

            tmp.create.PerformClick();

            tmp.textBox6.Text = textBox1.Text;

        }

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

        {

            adddepo tmp = new adddepo();

            tmp.Show();

            tmp.create.PerformClick();

            tmp.textBox6.Text = textBox1.Text;

        }

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

        {

            addloan tmp = new addloan();

            tmp.Show();

            tmp.create.PerformClick();

            tmp.textBox6.Text = textBox1.Text;

        }

    }

}

Empinfo.cs

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.Windows.Input;

using System.Data.SqlClient;

namespace Bank

{

    public partial class empinfo : Form

    {

        public empinfo()

        {

            InitializeComponent();

        }

        Dictionary<int, string> test2 = new Dictionary<int, string>();

        Dictionary<bool, string> test = new Dictionary<bool, string>();

        Dictionary<int, string> test3 = new Dictionary<int, string>();

        #region texbox

        //Для блока текстбоксов

        public void ronly()

        {

            textBox1.ReadOnly = true;

            textBox2.ReadOnly = true;

            textBox3.ReadOnly = true;

            textBox4.ReadOnly = true;

            textBox5.ReadOnly = true;

            comboBox2.Enabled = false;

            comboBox1.Enabled = false;

            textBox7.ReadOnly = true;

            comboBox2.Enabled = false;

            textBox9.ReadOnly = true;

            textBox10.ReadOnly = true;

            textBox11.ReadOnly = true;

            textBox12.ReadOnly = true;

            textBox13.ReadOnly = true;

            textBox14.ReadOnly = true;

            textBox15.ReadOnly = true;

            textBox16.ReadOnly = true;

            textBox17.ReadOnly = true;

        }

        public void write()

        {

            textBox1.ReadOnly = false;

            textBox2.ReadOnly = false;

            textBox3.ReadOnly = false;

            textBox4.ReadOnly = false;

            textBox5.ReadOnly = false;

            comboBox2.Enabled = true;

            comboBox1.Enabled = true;

            textBox7.ReadOnly = false;

            comboBox3.Enabled = true;

            textBox9.ReadOnly = false;

            textBox10.ReadOnly = false;

            textBox11.ReadOnly = false;

            textBox12.ReadOnly = false;

            textBox13.ReadOnly = false;

            textBox14.ReadOnly = false;

            textBox15.ReadOnly = false;

            textBox16.ReadOnly = false;

           textBox17.ReadOnly = false;

        }

        public void wipe()

        {

            textBox1.Text = null;

            textBox2.Text = null;

            textBox3.Text = null;

            textBox4.Text = null;

            textBox5.Text = null;

            textBox7.Text = null;

            textBox9.Text = null;

            textBox10.Text = null;

            textBox11.Text = null;

            textBox12.Text = null;

            textBox13.Text = null;

            textBox14.Text = null;

            textBox15.Text = null;

            textBox16.Text = null;

            textBox17.Text = null;

        }

        //Включение отключение кнопок

        public void changeBut(byte i)

        {

            if (i == 1) //вкл

            {

                write();

                button1.Visible = true;

                button6.Visible = true;

            }

            else

            {

                button1.Visible = false;

                button6.Visible = false;

                button4.Enabled = true;

                button5.Enabled = true;

            }

        }

        #endregion

        private void empinfo\_VisibleChanged(object sender, EventArgs e)

        {

            if (Program.isgovernor)

            {

                label21.Visible = true;

                label17.Visible = true;

                textBox14.Visible = true; //упр показывается логин и пароль

                textBox16.Visible = true;

                button4.Visible = true;

                button5.Visible = true;

                label22.Visible = true;

                comboBox1.Visible = true;

            }

        }

        //редактирование

        public void button4\_Click\_1(object sender, EventArgs e)

        {

            Program.curepas = textBox1.Text;

            write();

            button5.Enabled = false;

            changeBut(1);

        }

        //добавление

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

        {

            wipe();

            button4.Enabled = false;

            changeBut(1);

            textBox15.Text = DateTime.Now.ToShortDateString();

        }

        // Отмена

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

        {

            changeBut(0);

            if (Program.curepas == null) //при добавлении нового сотрудника - зануляем

                wipe();

            else

                empdata(Program.curepas);

            ronly();

        }

private void dataGridView1\_CellClick(object sender, DataGridViewCellEventArgs e) //получаем id из клетки и переходим в форму договора

        {

            try

            {

                var conn = new SqlConnection();

                conn.ConnectionString = Program.str;

                var item = dataGridView1.Rows[e.RowIndex].Cells[0].Value;

                int loan, depo, card;

                conn.Open();

                using (SqlCommand StrQuer = new SqlCommand($"SELECT contid FROM contcard WHERE contid=N'{item}'", conn))

                {

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    using (dr)

                        while (dr.Read())

                        {

                            card = dr.GetInt32(0);

                            addcard tmp = new addcard();

                            tmp.Show();

                            tmp.loadinfocard(card.ToString());

                            conn.Close();

                            return;

                        }

                }

                using (SqlCommand StrQuer = new SqlCommand($"SELECT contid FROM contdepo WHERE contid=N'{item}'", conn))

                {

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    using (dr)

                        while (dr.Read())

                        {

                            depo = dr.GetInt32(0);

                            adddepo tmp = new adddepo();

                            tmp.Show();

                            tmp.loadInfodepo(depo.ToString());

                            conn.Close();

                            return;

                        }

                }

                using (SqlCommand StrQuer = new SqlCommand($"SELECT contid FROM contloan WHERE contid=N'{item}'", conn))

                {

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    using (dr)

                        while (dr.Read())

                        {

                            loan = dr.GetInt32(0);

                            addloan tmp = new addloan();

                            tmp.Show();

                            tmp.loadInfoloan(loan.ToString());

                            conn.Close();

                            return;

                        }

                }

            }

            catch (Exception )

            { }

        }

        // Сохранение

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

        {

            if (button4.Enabled == true)  //редакт

            {

                bool k = true;

                var conn = new SqlConnection();

                conn.ConnectionString = Program.str;

                //Данные текущего пользователя

                string emppas="0", inn="0", snils="0", login="0", password="0", logid="-1";

                using (SqlCommand StrQuer = new SqlCommand($"SELECT emppas, inn, snils, Login.login, Login.password, Login.Id " +

                    $"FROM emp INNER JOIN Login ON emp.logid = Login.Id WHERE emppas = N'{Program.curepas}'", conn))

                {

                    conn.Open();

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    using (dr)

                    {

                        while (dr.Read())

                        {

                            emppas = dr.GetString(0); //пасс

                            inn = dr.GetString(1); //инн

                            snils = dr.GetString(2); //снилс

                            login = dr.GetString(3); // login

                            password = dr.GetString(4); //pass

                            logid = dr.GetInt32(5).ToString();

                        }

                    }

                    conn.Close();

                }

                //Проверка данных текущего пользователя с тем, что найдено в дб

                    using (SqlCommand StrQuer = new SqlCommand($"SELECT emppas, inn, snils, Login.login, Login.password " +

                    $"FROM emp INNER JOIN Login ON emp.logid = Login.ID WHERE emppas LIKE '{textBox1.Text}%' OR inn LIKE '{textBox10.Text}%' " +

                    $"OR snils LIKE '{textBox9.Text}%' OR Login.login LIKE '{textBox16.Text}%' OR Login.password LIKE '{textBox14.Text}%'", conn))

                {

                    conn.Open();

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    using (dr)

                    {

                        while (dr.Read())

                        {

                            string tpas="0", tinn="0", tsnils="0", tlogin="0", tpassword="0";

                            tpas = dr.GetString(0); //пасс

                            tinn = dr.GetString(1); //инн

                            tsnils = dr.GetString(2); //снилс

                            tlogin = dr.GetString(3);

                            tpassword = dr.GetString(4); //pass

                            //Если нашли старую запись - скип

                            if (String.Equals(emppas, tpas) && String.Equals(inn, tinn) && String.Equals(snils, tsnils) && (String.Equals(password, tpassword) && String.Equals(login, tlogin)))

                            {

                                continue;

                            }

                            if (String.Equals(textBox1.Text, tpas))

                            {

                                k = false;

                                textBox1.Text = null;

                            }

                            if (String.Equals(textBox10.Text, tinn))

                            {

                                k = false;

                                textBox10.Text = null;

                            }

                            if (String.Equals(textBox9.Text, tsnils))

                            {

                                k = false;

                                textBox9.Text = null;

                            }

                            if ((String.Equals(textBox16.Text, tpassword) && String.Equals(textBox14.Text, tlogin)))

                            {

                                k = false;

                                textBox14.Text = null;

                                textBox16.Text = null;

                            }

                        }

                        conn.Close();

                        if (!k)

                        {

                            MessageBox.Show("Не могут повторяться: паспорт, ИНН, СНИЛС, имя пользователи или пароль");

                            return;

                        }

                    }

                }

                int citycode = ((KeyValuePair<int, string>)comboBox3.SelectedItem).Key;

                string jobcode = ((KeyValuePair<int, string>)comboBox2.SelectedItem).Key.ToString();

                string actcode = ((KeyValuePair<bool, string>)comboBox1.SelectedItem).Key.ToString();

                conn.Open();

                DateTime myDateTime1 = Convert.ToDateTime(textBox5.Text);

                string sqlFormattedDate1 = myDateTime1.ToString("yyyy-MM-dd HH:mm:ss.fff");

                DateTime myDateTime2 = DateTime.Now;

                string sqlFormattedDate2 = myDateTime2.ToString("yyyy-MM-dd HH:mm:ss.fff");

                using (SqlCommand StrQuer = new SqlCommand

                        ("BEGIN TRANSACTION " +

                        $"UPDATE contcard SET contcard.emppas = '{textBox1.Text}' WHERE(((contcard.emppas) = '{Program.curepas}'));" +

                        $"UPDATE contdepo SET contdepo.emppas = '{textBox1.Text}' WHERE(((contdepo.emppas) = '{Program.curepas}'));" +

                        $"UPDATE contloan SET contloan.emppas = '{textBox1.Text}' WHERE(((contloan.emppas) = '{Program.curepas}')); " +

                        $"UPDATE emp SET emp.emppas = N'{textBox1.Text}', emp.fam = N'{textBox2.Text}', emp.name = N'{textBox3.Text}', emp.sname = N'{textBox4.Text}', emp.birth = '{sqlFormattedDate1}', " +

                        $"emp.job = N'{jobcode}', emp.inn = '{textBox10.Text}', emp.snils = '{textBox9.Text}', emp.city = '{citycode}', emp.adr = N'{textBox7.Text}', emp.cadr = N'{textBox13.Text}', emp.phone = N'{textBox12.Text}', " +

                        $"emp.jphone = N'{textBox11.Text}', emp.email = N'{textBox17.Text}', emp.updated = '{sqlFormattedDate2}' " +

                        $" WHERE (emp.emppas = '{Program.curepas}'); " +

                        $"UPDATE Login SET Login.login = N'{textBox16.Text}', Login.password = N'{textBox14.Text}', Login.active = N'{actcode}' " +

                        $" WHERE (Login.Id = '{logid}');" +

                        $" COMMIT", conn))

                {

                    SqlDataReader dr = StrQuer.ExecuteReader();

                    conn.Close();

                }

                Program.curepas = textBox1.Text;

                empdata(Program.curepas);

                button2.PerformClick();

                MessageBox.Show("Изменения сохранены");

            }

            else //INSERT

            {

                var conn = new SqlConnection();

                conn.ConnectionString = Program.str;

                //проверка на повторение emppas, inn, snils, login.password

                if (checkedata())

                {

                    //Получаем код города

                    int citycode = ((KeyValuePair<int, string>)comboBox3.SelectedItem).Key;

                    int maxid = 0;

                    string jobcode = ((KeyValuePair<int, string>)comboBox2.SelectedItem).Key.ToString();

                    string actcode = ((KeyValuePair<bool, string>)comboBox1.SelectedItem).Key.ToString();

                    conn.Open();

                    using (SqlCommand StrQuer = new SqlCommand($"SELECT MAX(Id) FROM Login", conn))

                    {

                        SqlDataReader dr = StrQuer.ExecuteReader();

                        using (dr)

                            while (dr.Read())

                            {

                                maxid = dr.GetInt32(0);

                            }

                    }

                    DateTime myDateTime3 = Convert.ToDateTime(textBox5.Text);

                    string sqlFormattedDate3 = myDateTime3.ToString("yyyy-MM-dd HH:mm:ss.fff");

                    DateTime myDateTime1 = DateTime.Now;

                    string sqlFormattedDate1 = myDateTime1.ToString("yyyy-MM-dd HH:mm:ss.fff");

                    using (SqlCommand StrQuer = new SqlCommand("BEGIN TRANSACTION " +

                        "INSERT INTO emp (emppas, fam, emp.name, sname, birth, job, inn, snils, emp.city," +

                        " adr, cadr, phone, jphone, email, updated, logid) " +

                        $"VALUES ('{textBox1.Text}', N'{textBox2.Text}', N'{textBox3.Text}', N'{textBox3.Text}', '{sqlFormattedDate3}', '{jobcode}', " +

                        $"'{textBox10.Text}', '{textBox9.Text}', '{citycode}', N'{textBox7.Text}', N'{textBox13.Text}', '{textBox12.Text}', '{textBox11.Text}', " +

                        $"'{textBox17.Text}', '{sqlFormattedDate1}', '{maxid + 1}'); " +

                        $"INSERT INTO Login (login, password, active) " +

                        $"VALUES ('{textBox16.Text}', '{textBox14.Text}', '{actcode}'); " +

                        $"COMMIT", conn))

                    {

                        SqlDataReader dr = StrQuer.ExecuteReader();

                        conn.Close();

                    }

                    Program.curepas = textBox1.Text;

                    empdata(Program.curepas);

                    button2.PerformClick();

                    MessageBox.Show("Изменения сохранены");

                }

            }

            changeBut(0);

            ronly();

        }

        public void empinfo\_Load(object sender, EventArgs e)

        {

            //Показ Да Нет для активности аккаунта

            test.Add(false, "Нет");

            test.Add(true, "Да");

            comboBox1.DataSource = new BindingSource(test, null);

            comboBox1.DisplayMember = "Value";

            comboBox1.ValueMember = "Key";

            // Загрузка возможных должностей

            var conn = new SqlConnection();

            conn.ConnectionString = Program.str;

            using (SqlCommand StrQuer = new SqlCommand("SELECT \* " +

                "FROM job ", conn))

            {

                conn.Open();

                SqlDataReader dr = StrQuer.ExecuteReader();

                using (dr) //для проверки на регистр в логине/пароле

                {

                    while (dr.Read())

                    {  //вставка полей в textbox

                        int tmp;

                        string tmps;

                        tmp = dr.GetInt32(0); //ид должности

                        tmps = dr.GetString(1); //название

                        test2.Add(tmp, tmps);

                    }

                }

                conn.Close();

            }

            using (SqlCommand StrQuer = new SqlCommand("SELECT \* FROM city", conn))

            {

                conn.Open();

                SqlDataReader dr = StrQuer.ExecuteReader();

                using (dr) //для проверки на регистр в логине/пароле

                {

                    while (dr.Read())

                    {  //вставка полей в textbox

                        int tmp;

                        string tmps;

                        tmp = dr.GetInt32(0); //ид города

                        tmps = dr.GetString(1); //название

                        test3.Add(tmp, tmps);

                    }

                }

                conn.Close();

                comboBox3.DataSource = new BindingSource(test3, null);

                comboBox3.DisplayMember = "Value";

                comboBox3.ValueMember = "Key";

                //test2.Add(0, "Менеджер");

                //test2.Add(1, "Управляющий");

                comboBox2.DataSource = new BindingSource(test2, null);

                comboBox2.DisplayMember = "Value";

                comboBox2.ValueMember = "Key";

                //((KeyValuePair<int, string>)comboBox1.SelectedItem).Key.ToString(); //вот так получать 0 1 из комбобокса

            }

        }

        public bool checkedata()  //проверка на повторение emppas, inn, snils, login.password

        {

            bool k = true;

            var conn = new SqlConnection();

            conn.ConnectionString = Program.str;

            using (SqlCommand StrQuer = new SqlCommand($"SELECT emppas, inn, snils, Login.login, Login.password " +

                $"FROM emp INNER JOIN Login ON emp.logid = Login.ID WHERE emppas LIKE '{textBox1.Text}%' AND inn LIKE '{textBox10.Text}%' " +

                $"AND snils LIKE '{textBox9.Text}%' AND Login.login LIKE '{textBox16.Text}%' AND Login.password LIKE '{textBox14.Text}%'", conn))

            {

                //StrQuer.Parameters.AddWithValue("@emppas", textBox1.Text);

                conn.Open();

                SqlDataReader dr = StrQuer.ExecuteReader();

                using (dr)

                {

                    while (dr.Read())

                    {

                        string tpas, tinn, tsnils, tlogin, tpassword;

                        string res = "Не могут повторяться:\n";

                        tpas = dr.GetString(0); //пасс

                        tinn = dr.GetString(1); //инн

                        tsnils = dr.GetString(2); //снилс

                        tlogin = dr.GetString(3);

                        tpassword = dr.GetString(4); //pass

                        if (String.Equals(textBox1.Text, tpas))

                        {

                            k = false;

                            res = res + "паспорт\n";

                            textBox1.Text = null;

                        }

                        if (String.Equals(textBox10.Text, tinn))

                        {

                            k = false;

                            res = res + "ИНН\n";

                            textBox10.Text = null;

                        }

                        if (String.Equals(textBox9.Text, tsnils))

                        {

                            k = false;

                            res = res + "СНИЛС\n";

                            textBox9.Text = null;

                        }

                        if (String.Equals(textBox14.Text, tpassword) && String.Equals(textBox16.Text, tlogin))

                        {

                            k = false;

                            res = res + "имя пользователи или пароль";

                            textBox14.Text = null;

                            textBox16.Text = null;

                        }

                        if (!k)

                            MessageBox.Show(res);

                    }

                }

                conn.Close();

            }

            return k;

        }

        public void empdata(string ppas) //парс даты по паспорту

        {

            var conn = new SqlConnection();

            conn.ConnectionString = Program.str;

            using (SqlCommand StrQuer = new SqlCommand("SELECT emppas, fam, emp.name, sname, birth, job, inn, snils, city.Id," +

                " adr, cadr, phone, jphone, email, updated, Login.login, Login.password, Login.active  " +

                "FROM emp INNER JOIN city ON emp.city = city.Id INNER JOIN Login ON emp.logid = Login.ID " +

                "WHERE emppas=@emppas", conn)) //берем всю запись

            {

                StrQuer.Parameters.AddWithValue("@emppas", ppas);

                conn.Open();

                SqlDataReader dr = StrQuer.ExecuteReader();

                using (dr) //для проверки на регистр в логине/пароле

                {

                    while (dr.Read())

                    {  //вставка полей в textbox

                        textBox1.Text = dr.GetString(0); //пасс

                        textBox2.Text = dr.GetString(1); //ФИО

                        textBox3.Text = dr.GetString(2);

                        textBox4.Text = dr.GetString(3);

                        textBox5.Text = dr.GetDateTime(4).Date.ToString("d");

                        comboBox2.SelectedIndex = comboBox2.FindStringExact(test2[dr.GetInt32(5)]);  //ид должности

                        textBox10.Text = dr.GetString(6); //инн

                        textBox9.Text = dr.GetString(7); //снилс

                        comboBox3.SelectedIndex = comboBox3.FindStringExact(test3[dr.GetInt32(8)]);  //город

                        textBox7.Text = dr.GetString(9); //адрес

                        textBox13.Text = dr.GetString(10); //тек адрес

                        textBox12.Text = dr.GetString(11); //тел личный

                        textBox11.Text = dr.GetString(12); //тел рабоч

                        textBox17.Text = dr.GetString(13); //емейл

                        textBox15.Text = dr.GetDateTime(14).Date.ToString("d"); //updated

                        textBox16.Text = dr.GetString(15); //login

                        textBox14.Text = dr.GetString(16); //pass

                        comboBox1.SelectedIndex = comboBox1.FindStringExact(test[dr.GetBoolean(17)]);  //активен

                    }

                }

                conn.Close();

            }

        }

        //Обновление информации о человеке чей паспорт в textbox1

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

        {

            dataGridView1.Rows.Clear();

            var conn = new SqlConnection();

            conn.ConnectionString = Program.str;

            List<string[]> data = new List<string[]>();

            conn.Open();

            using (SqlCommand StrQuer = new SqlCommand($"SELECT contdepo.contid, contdate FROM contdepo WHERE(((contdepo.emppas) = '{Program.curepas}'));", conn))

            {

                SqlDataReader dr = StrQuer.ExecuteReader();

                using (dr)

                {

                    while (dr.Read())

                    {

                        data.Add(new string[2]);

                        data[data.Count - 1][0] = dr[0].ToString();

                        data[data.Count - 1][1] = dr[1].ToString().Replace(" 0:00:00", "");

                    }

                }

            }

            using (SqlCommand StrQuer = new SqlCommand($"SELECT contcard.contid, contdate FROM contcard WHERE(((contcard.emppas) = '{Program.curepas}'));", conn))

            {

                SqlDataReader dr = StrQuer.ExecuteReader();

                using (dr)

                {

                    while (dr.Read())

                    {

                        data.Add(new string[2]);

                        data[data.Count - 1][0] = dr[0].ToString();

                        data[data.Count - 1][1] = dr[1].ToString().Replace(" 0:00:00", "");

                    }

                }

            }

            using (SqlCommand StrQuer = new SqlCommand($"SELECT contloan.contid, contdate FROM contloan WHERE(((contloan.emppas) = '{Program.curepas}'));", conn))

            {

                SqlDataReader dr = StrQuer.ExecuteReader();

                using (dr)

                {

                    while (dr.Read())

                    {

                        data.Add(new string[2]);

                        data[data.Count - 1][0] = dr[0].ToString();

                        data[data.Count - 1][1] = dr[1].ToString().Replace(" 0:00:00", "");

                    }

                    foreach (string[] s in data)

                        dataGridView1.Rows.Add(s);

                }

            }

            conn.Close();

            dataGridView1.Sort(dataGridView1.Columns[1], ListSortDirection.Ascending);

        }

    }

}

Проект Visual Studio:

<https://github.com/makspepe/HSE2021_2Course_CoursePaper>