**Министерство образования Тверской области**

**Государственное бюджетное профессиональное**

**образовательное учреждение**

**«Тверской промышленно-экономический колледж»**

 Специальность     09.02.07

Информационные системы и программирование

**Отчет по заданию учебной практики по УП.01**

Студентка группы 3-09ПС-2

Мазурская Елизавета Сергеевна \_\_\_\_\_\_\_\_\_

 Руководитель практики:

Антонов Вячеслав Андреевич \_\_\_\_\_\_\_\_

Оценка по УП.01

Г. Тверь 2020

ОГЛАВЛЕНИЕ

Глава 1. Проектирование программного обеспечения. Формирование требований к ПО

1.1 Описание предметной области организации……………………………3

1.2 Определение целей разработки…………………………………............7

1.3 Определение структуры организации (предприятия)………………....8

Глава 2. Проектирование программного обеспечения (Стр7-8)

2.1 Предварительное выявление требований к будущей системе

2.2 Построение моделей взаимодействия

Глава 3. Создание и отладка программного обеспечения (Стр10)

3.1.Установка и разворачивание локальных СУБД

3.2.Установка и разворачивание удаленных СУБД

3.3. Разработка серверной части БД.

3.4. Размещение проекта. Использование системы контроля версий

Глава 4. Внедрение программного обеспечения (Стр11-13)

4.1. Разработка клиентской части (модулей) и интерфейса приложения

4.2. Оформление документации и тестирование созданного приложения

Глава 1. Проектирование программного обеспечения. Формирование требований к ПО

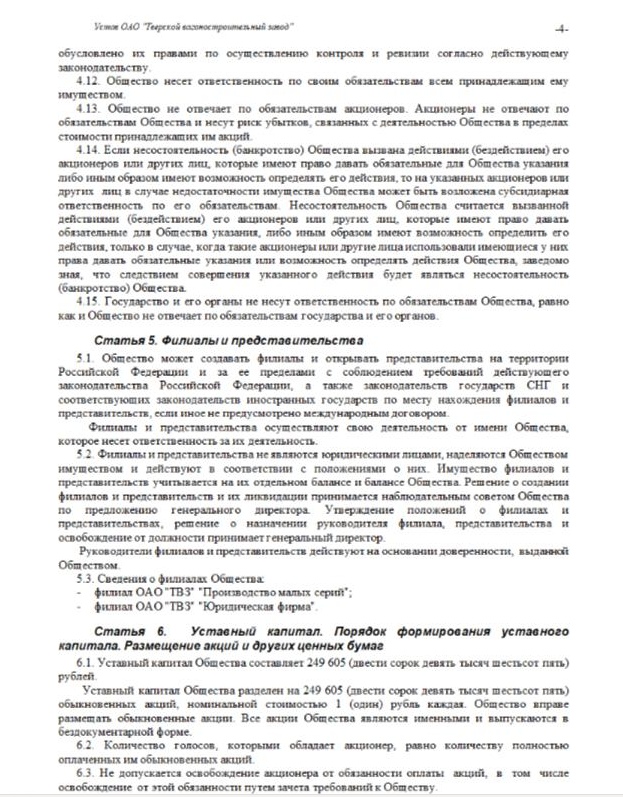
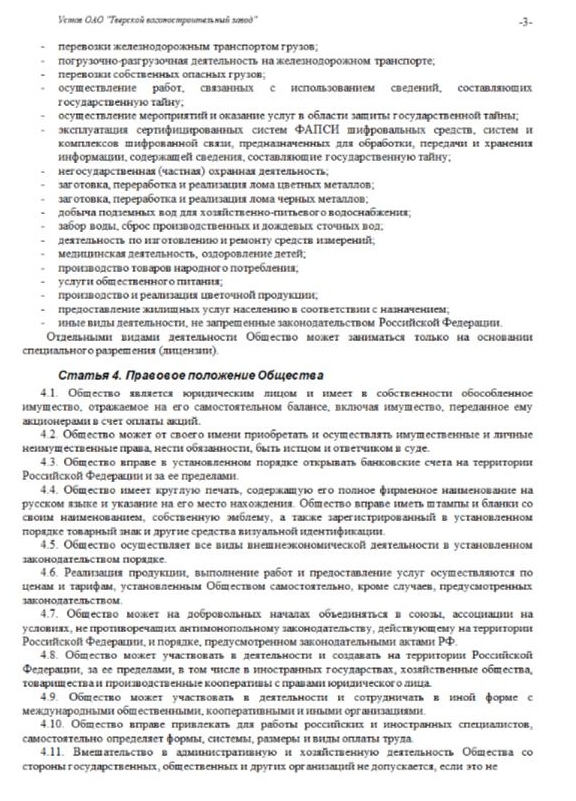
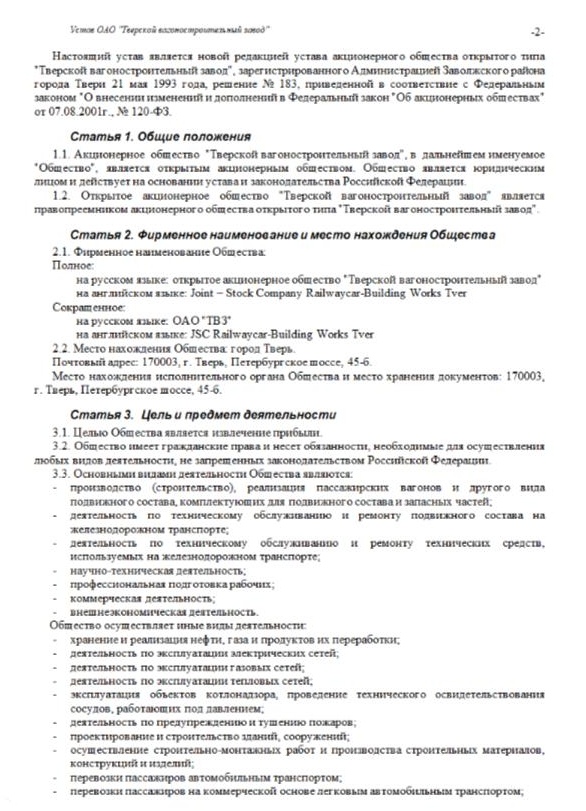
1.1 Описание предметной области

Прохождение практики осуществлялось в Твери на Тверском вагоностроительном заводе(Отдел системно-технического обслуживания телекоммуникаций и технических средств ).

Мои обязанности – разработка ПО ,проектирование баз данных, изучение веб-программирования.

Тверской вагоностроительный завод-высокотехнологичное производство ,единственное в России по созданию различных вагонов. Завод специализируется на выпуске одно и двухэтажных вагонов ,участвует в производстве вагонов метро низкопольных трамваев. Миссия ОАО “ТВЗ”- быть вагоностроительным предприятием мирового уровня.

Высокий уровень качества продукции Тверского вагоностроительного завода подтверждают многочисленные награды отраслевых ,всероссийских и международных конкурсов.



1.2 Определение целей разработки

Цель разработки:

создание программного обеспечения для Тверского вагоностроительного завода (Отдел системно-технического обслуживания телекоммуникаций и технических средств ).

закрепление навыков проектирования и разработки баз данных

участие в выработке требований к программному обеспечению

проектирование программного обеспечения с использованием программных пакетов

разработка технологической документации

закрепление навыков проектирования и разработки пользовательского интерфейса

Разработка алгоритмов решения и его реализация

Разработка кода программного продукта на основе готовой спецификации на уровне модуля

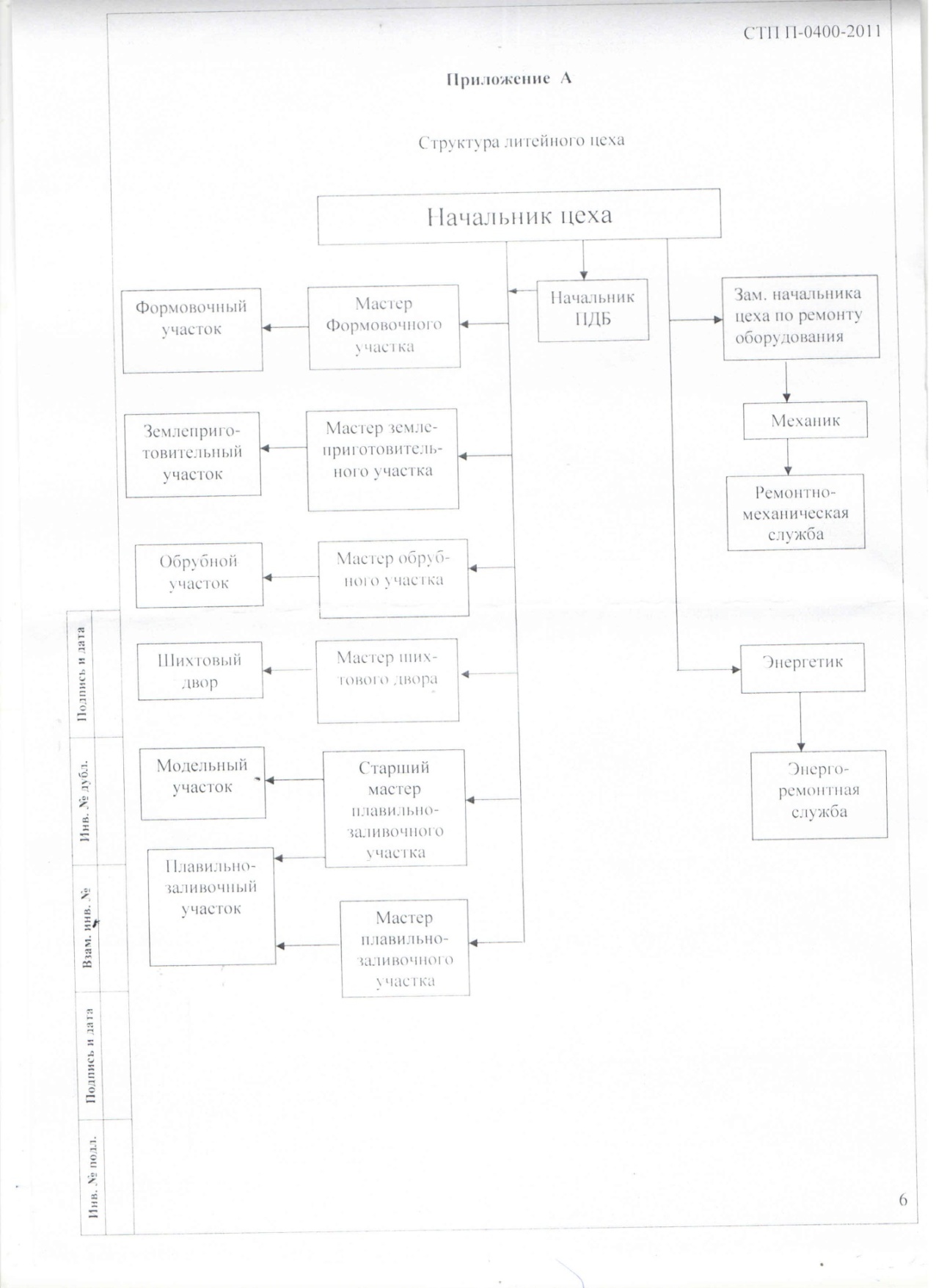
Использование инструментальных средств отладки программного продукта

Выполнение отладки и тестирования программы на уровне модуля

Осуществление рефакторинга и оптимизации программного кода

Использование системы контроля версий

1.3 Определение структуры организации (предприятия)

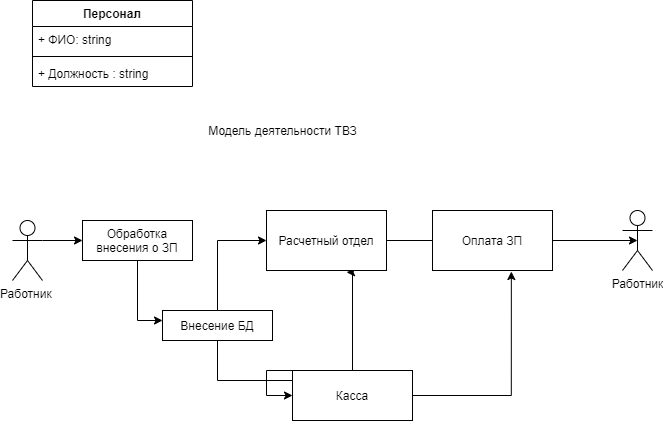


Глава 2. Проектирование программного обеспечения

2.1 Предварительное выявление требований к будущей системе

1. Внесение данных о зп;
2. Внесение данных о больничных листах;

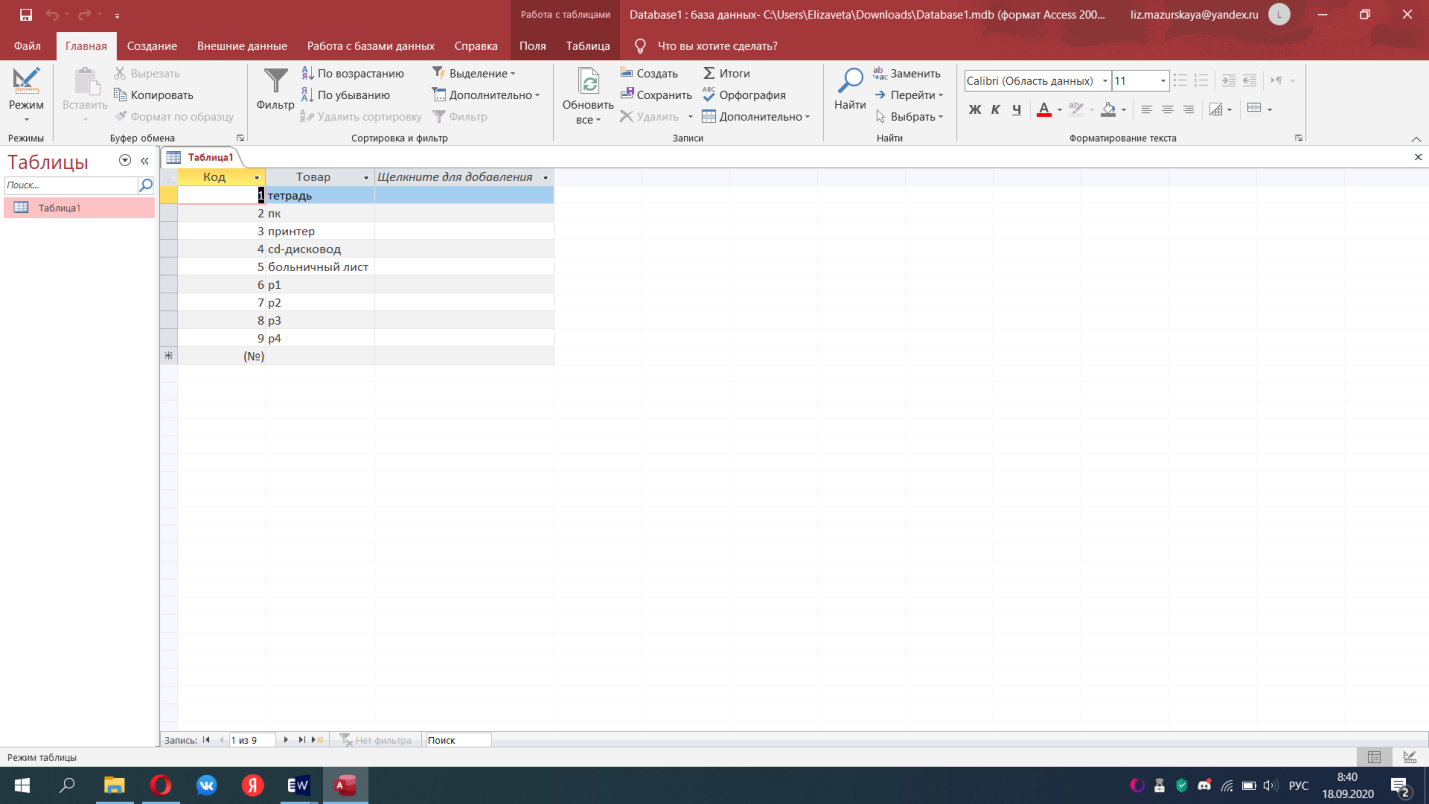
2.2 Построение моделей взаимодействия

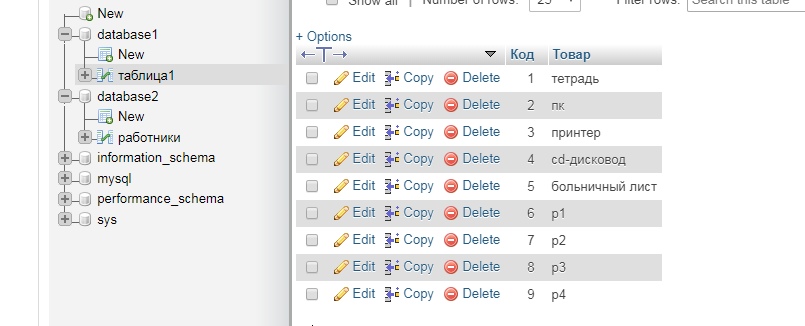


Глава 3. Создание и отладка программного обеспечения

3.1 Установка и разворачивание локальных\удалённых СУБД и разработка серверной части БД

Для начала я в Microsoft Access заполнила тестовыми данными БД:



Дальше я конвертировала Бд

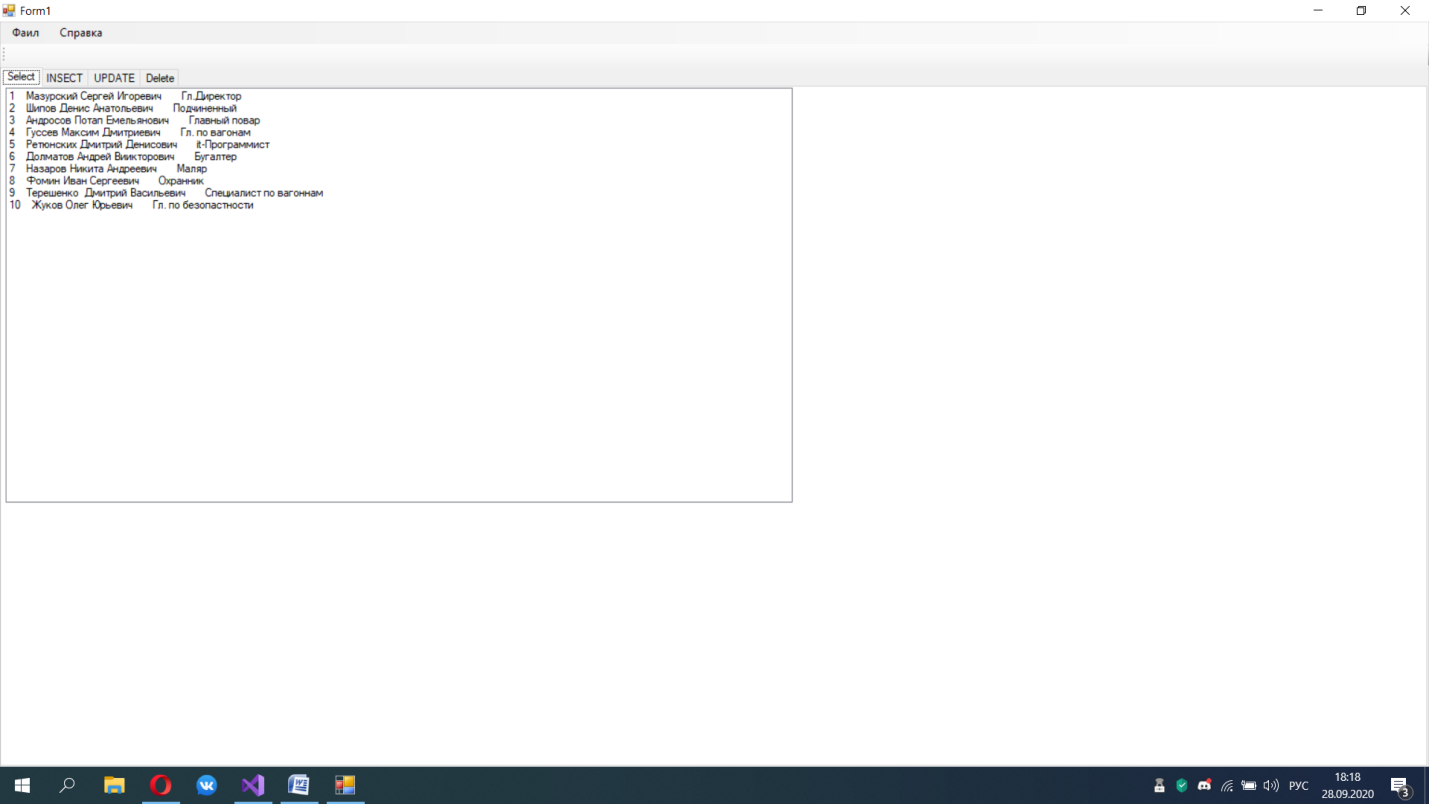
4.1. Разработка клиентской части (модулей) и интерфейса приложения

Разработанное приложение обладает следующими функциями:

1. Чтение информации из БД

2. Создание новых записей

3. Редактирование, в том числе удаление, существующих записей



using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Data.SqlClient;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace WorkingWithDB

{

public partial class Form1 : Form

{

SqlConnection sqlConnection;

public Form1()

{

InitializeComponent();

}

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

{

string connectionString = @"Data Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\Elizaveta\source\repos\WindowsFormsApp1\WindowsFormsApp1\Database1.mdf;Integrated Security=True";

sqlConnection = new SqlConnection(connectionString);

await sqlConnection.OpenAsync();

SqlDataReader sqlReader = null;

SqlCommand command = new SqlCommand("SELECT \* FROM [FSC]", sqlConnection);

try

{

sqlReader = await command.ExecuteReaderAsync();

while (await sqlReader.ReadAsync())

{

listBox1.Items.Add(Convert.ToString(sqlReader["Id"]) + " " + Convert.ToString(sqlReader["FSC] +" " + Convert.ToString(sqlReader["Position"];)

}

}

catch (Exception ex)

{

MessageBox.Show(ex.Message.ToString(), ex.Source.ToString(), MessageBoxButtons.OK, MessageBoxIcon.Error);

}

finally

{

if (sqlReader != null)

sqlReader.Close();

}

}

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

{

if (sqlConnection != null && sqlConnection.State != ConnectionState.Closed)

sqlConnection.Close();

}

private void Form1\_FormClosing(object sender, FormClosingEventArgs e)

{

if (sqlConnection != null && sqlConnection.State != ConnectionState.Closed)

sqlConnection.Close();

}

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

{

if (label7.Visible)

label7.Visible = false;

if (!string.IsNullOrEmpty(textBox1.Text) && !string.IsNullOrWhiteSpace(textBox1.Text) &&

!string.IsNullOrEmpty(textBox2.Text) && !string.IsNullOrWhiteSpace(textBox2.Text))

{

SqlCommand command = new SqlCommand("INSERT INTO [id] (FSC, position)VALUES(@Fcs, @position)", sqlConnection);

command.Parameters.AddWithValue("FCs", textBox1.Text);

command.Parameters.AddWithValue("Position", textBox2.Text);

await command.ExecuteNonQueryAsync();

}

else

{

label7.Visible = true;

label7.Text = "Поля 'ФИО' и 'Должность ' должны быть заполнены!";

}

}

private async void обновитьToolStripMenuItem\_Click(object sender, EventArgs e)

{

listBox1.Items.Clear();

SqlDataReader sqlReader = null;

SqlCommand command = new SqlCommand("SELECT \* FROM [FSc]", sqlConnection);

try

{

sqlReader = await command.ExecuteReaderAsync();

while (await sqlReader.ReadAsync())

{

listBox1.Items.Add(Convert.ToString(sqlReader["Id"]) + " " + Convert.ToString(sqlReader["FCs"]) + " " + Convert.ToString(sqlReader["Position"]));

}

}

catch (Exception ex)

{

MessageBox.Show(ex.Message.ToString(), ex.Source.ToString(), MessageBoxButtons.OK, MessageBoxIcon.Error);

}

finally

{

if (sqlReader != null)

sqlReader.Close();

}

}

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

{

if (label8.Visible)

label8.Visible = false;

if (!string.IsNullOrEmpty(textBox3.Text) && !string.IsNullOrWhiteSpace(textBox3.Text) &&

!string.IsNullOrEmpty(textBox4.Text) && !string.IsNullOrWhiteSpace(textBox4.Text) &&

!string.IsNullOrEmpty(textBox5.Text) && !string.IsNullOrWhiteSpace(textBox5.Text))

{

SqlCommand command = new SqlCommand("UPDATE [id] SET [Fcs]=@Fcs, [position]=@Position WHERE [Id]=@Id", sqlConnection);

command.Parameters.AddWithValue("Id", textBox5.Text);

command.Parameters.AddWithValue("FCs", textBox4.Text);

command.Parameters.AddWithValue("Price", textBox3.Text);

await command.ExecuteNonQueryAsync();

}

else if (!string.IsNullOrEmpty(textBox4.Text) && !string.IsNullOrWhiteSpace(textBox4.Text))

{

label8.Visible = true;

label8.Text = "Id должнен быть заполнен!";

}

else

{

label8.Visible = true;

label8.Text = "Поля 'Id', 'ФИО' и 'Должность' должны быть заполнены!";

}

}

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

{

if (label8.Visible)

label8.Visible = false;

if (!string.IsNullOrEmpty(textBox6.Text) && !string.IsNullOrWhiteSpace(textBox6.Text))

{

SqlCommand command = new SqlCommand("DELETE FROM [FSC] WHERE [Id]=@Id", sqlConnection);

command.Parameters.AddWithValue("Id", textBox6.Text);

await command.ExecuteNonQueryAsync();

}

else

{

label8.Visible = true;

label8.Text = "Id должнен быть заполнен!";

}

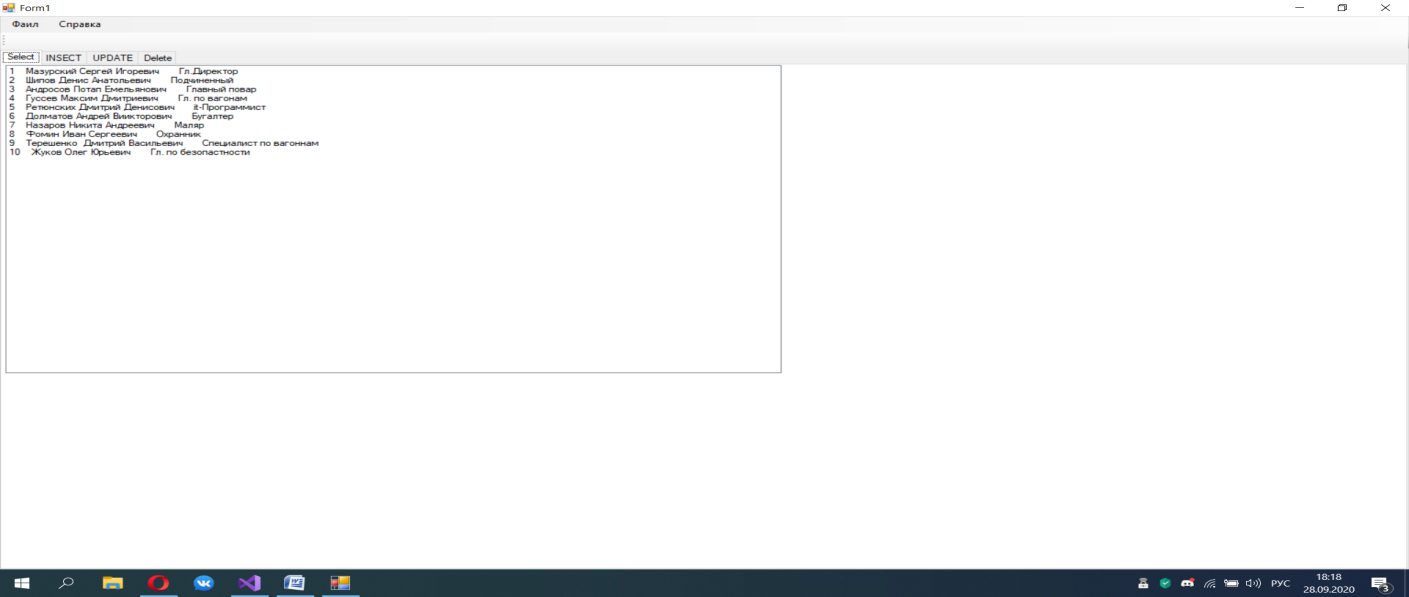
}

}

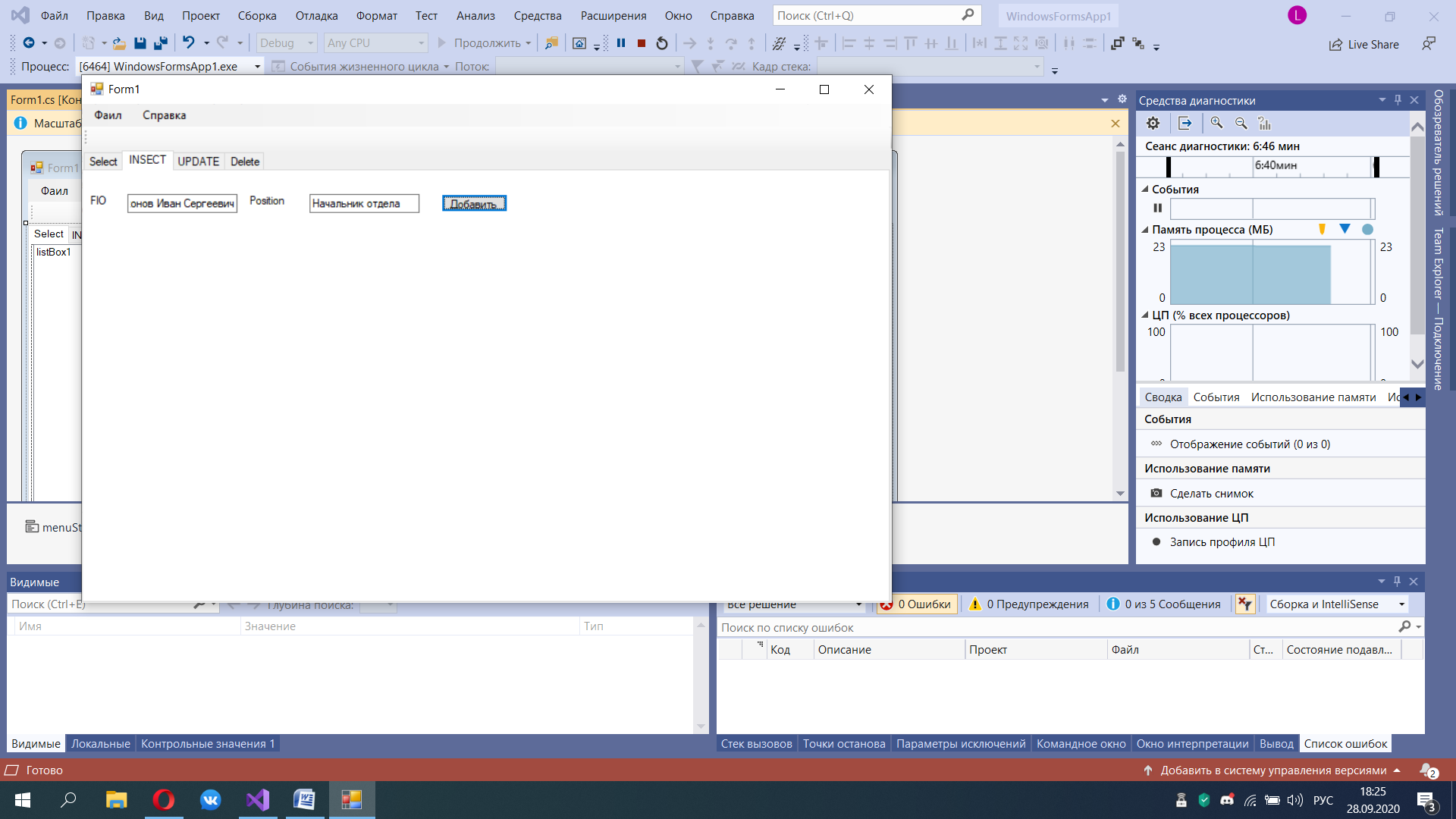
}

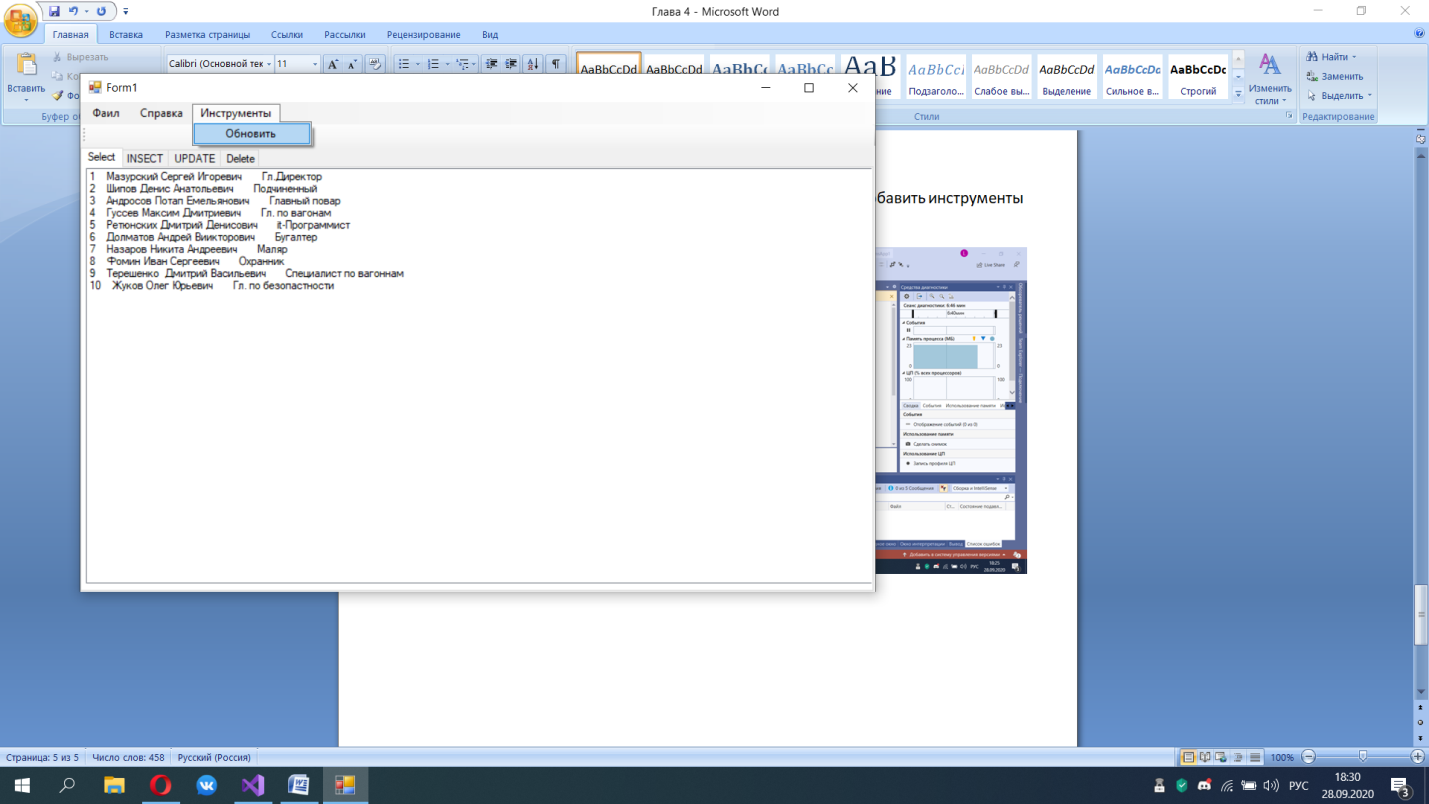
**4.2**. Оформление документации и тестирование созданного приложения

В данном скриншоте можно увидеть данные из БД

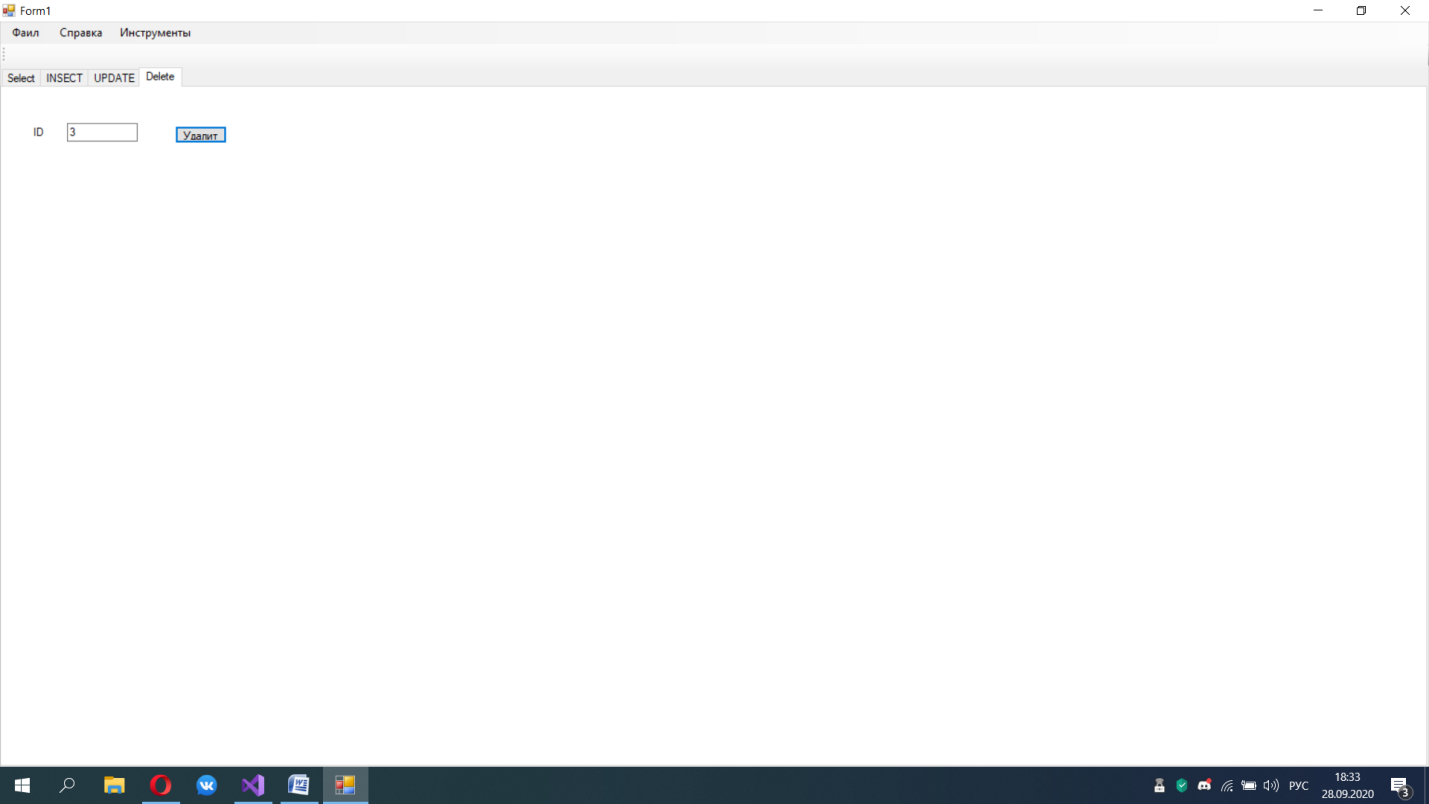


Здесь добавляю новые Должности в БД, в ФИО нажимаю добавить инструменты нажимаю обновить и добавляется Работник.

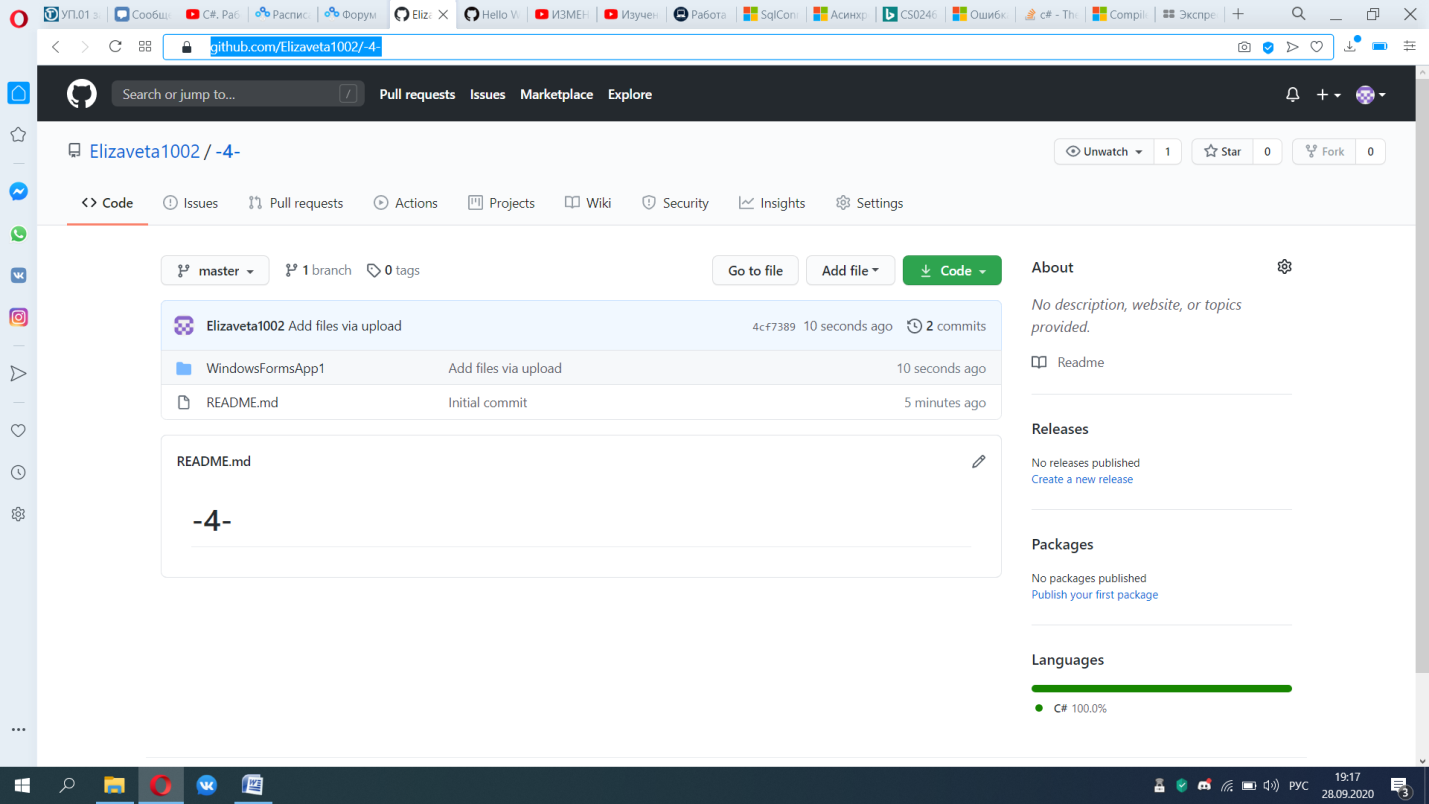




Здесь удаляю то что не надо, пишу id Работника и нажимаю удалить, дальше в инструментах нажимаю обновить и id удаляется.



4.3. Размещение проекта. Использование системы контроля версий.



<https://github.com/Elizaveta1002/-4->