**Занятие № 23**

**Номер учебной группы:** П-16

**Фамилия, инициалы учащегося:** Хомич В.И.

**Дата выполнения работы:** 5.12.2022

**Тема работы:** «Отладка модулей программной системы»

**Цель работы:** научиться проверять правильность построения модульной структуры и выполнения программного агрегата, соответствующего данной модульной структуре.

**Ход работы**

**Задание 1**

Используя разработанную структуру программы, проверил правильность построения модульной структуры и выполнения их работы, исправил ошибки. Доработан модуль регистрации в программе. Ниже представлен код модуля регистрации.

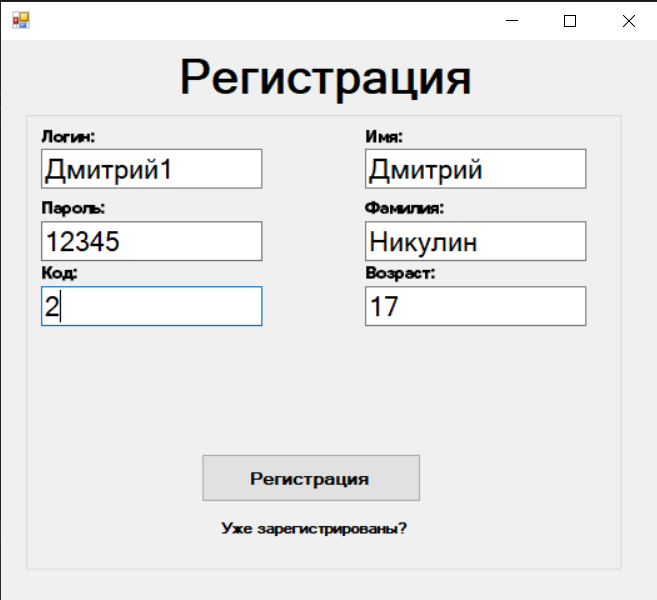


Рисунок 1 «Окно регистрации»

Листинг программных модулей

Листинг 1 - Модуль FormReg1.cs

{===================================================================================}

{Модуль Form1.cs}

{гр. П-15}

{Разработчик: Минич Д.А.}

{Модифицирован: 22 ноября 2021}

{-----------------------------------------------------------------------------------}

{\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*}

using MySql.Data.MySqlClient;

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 ProgramMed

{

public partial class FormReg1 : Form

{

public FormReg1()

{

InitializeComponent();

}

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

{

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

}

private void label1\_Click(object sender, EventArgs e)

{

Form ifrm = new FormLogin2();

ifrm.Left = this.Left; // задаём открываемой форме позицию слева равную позиции текущей формы

ifrm.Top = this.Top; // задаём открываемой форме позицию сверху равную позиции текущей формы

ifrm.Show(); // отображаем Form2

this.Hide(); // скрываем Form1 (this - текущая форма)

}

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

{

//Проверяю заполнили ли поля при регистрации

if (String.IsNullOrEmpty(LogtextBox1.Text))

{

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

return;

}

if (String.IsNullOrEmpty(PasstextBox2.Text))

{

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

return;

}

if (String.IsNullOrEmpty(NametextBox3.Text))

{

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

return;

}

if (String.IsNullOrEmpty(SurntextBox4.Text))

{

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

return;

}

if (String.IsNullOrEmpty(AgetextBox1.Text))

{

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

return;

}

//Проверяем есть ли такой пользователь, если есть то выходим и добавление не происходит

//То есть если поле возращает True то выходим и пользователь не регистрируется

if (isUserExists())

return;

//Добавление пользователя в базу.

DB db = new DB();

MySqlCommand command = new MySqlCommand("INSERT INTO `clients`( `Логин`,`Пароль`,`Имя`,`Фамилия`,`Возраст`,`Роль`) VALUES ( @login, @password, @name, @Surname, @Age, @Role)", db.getConnection());

command.Parameters.Add("@login", MySqlDbType.VarChar).Value = LogtextBox1.Text;

command.Parameters.Add("@password", MySqlDbType.VarChar).Value = PasstextBox2.Text;

command.Parameters.Add("@name", MySqlDbType.VarChar).Value = NametextBox3.Text;

command.Parameters.Add("@Surname", MySqlDbType.VarChar).Value = SurntextBox4.Text;

command.Parameters.Add("@Age", MySqlDbType.VarChar).Value = AgetextBox1.Text;

if (CodetextBox1.Text == "")

{

command.Parameters.Add("@Role", MySqlDbType.VarChar).Value = null;

}

else

command.Parameters.Add("@Role", MySqlDbType.VarChar).Value = CodetextBox1.Text;

//Открываем базу данных и если введённые данные в поля верны, проходит регистрация

db.openConnection();

if (command.ExecuteNonQuery() == 1)

MessageBox.Show("Регистрация прошла успешно");

else

MessageBox.Show("Что-то пошло не так, попробуй позже");

//Закрываем базу данных

db.closeConnection();

}

//Поле для проверки идентичных логинов

public Boolean isUserExists()

{

DB db = new DB();

DataTable table = new DataTable();

MySqlDataAdapter adapter = new MySqlDataAdapter();

MySqlCommand command = new MySqlCommand("Select \* FROM `clients` WHERE `Логин`=@LU " , db.getConnection());

command.Parameters.Add("@LU", MySqlDbType.VarChar).Value = LogtextBox1.Text;

//получаем данные и заносим в перемунную table, далее проверяем table, если там будет хоть 1 строка(запись) то выводим сообщение

//что такой логин уже есть и выходим, либо "Иначе, выходим"

adapter.SelectCommand = command;

adapter.Fill(table);

db.openConnection();

if (table.Rows.Count > 0)

{

MessageBox.Show("Такой логин уже есть!");

db.closeConnection();

return true;

}

else

return false;

}

private void label1\_MouseEnter(object sender, EventArgs e)

{

label1.ForeColor = Color.DeepSkyBlue; // смена цвета текста

}

private void label1\_MouseLeave(object sender, EventArgs e)

{

label1.ForeColor = Color.Black; // смена цвета текста

}

private void LogtextBox1\_KeyPress(object sender, KeyPressEventArgs e)

{

if (e.KeyChar == (int)Keys.Space)

e.KeyChar = '\0';

LogtextBox1.MaxLength = 15;

}

private void PasstextBox2\_KeyPress(object sender, KeyPressEventArgs e)

{

if (e.KeyChar == (int)Keys.Space)

e.KeyChar = '\0';

PasstextBox2.MaxLength = 15;

}

private void CodetextBox1\_KeyPress(object sender, KeyPressEventArgs e)

{

if (e.KeyChar == (int)Keys.Space)

e.KeyChar = '\0';

CodetextBox1.MaxLength = 4;

}

private void AgetextBox1\_KeyPress(object sender, KeyPressEventArgs e)

{

char number = e.KeyChar;

if (!Char.IsDigit(number) && number != 8) // цифры и клавиша BackSpace

{

e.Handled = true;

}

AgetextBox1.MaxLength = 2;

}

private void NametextBox3\_KeyPress(object sender, KeyPressEventArgs e)

{

if (!Char.IsDigit(e.KeyChar)) return;

else

e.Handled = true;

NametextBox3.MaxLength = 15;

}

private void SurntextBox4\_KeyPress(object sender, KeyPressEventArgs e)

{

if (!Char.IsDigit(e.KeyChar)) return;

else

e.Handled = true;

SurntextBox4.MaxLength = 15;

}

}

}

**Задание 2**

Ответьте на контрольные вопросы:

1. Структура программы это?

ОТВЕТ: Структура программ - это разметка рабочей области (области кода) с целью чёткого определения основных блоков программ и синтаксиса.

2. Перечислите правила кодирования модулей.

ОТВЕТ: Для кодирования программных модулей используются унифицированные (базовые) структуры, такие как: следование, развилка полная, развилка неполная, выбор, цикл-пока, цикл-до и цикл с параметром. Программные комплексы и программные модули, закодированные в соответствии с перечисленными правилами структурного программирования, называется структурированными.

3. Что означает модульная структура?

ОТВЕТ: Модульная структура программы представляет собой древовидную структуру, в узлах которой размещаются программные модули, а направленные дуги показывают статическую подчиненность модулей.