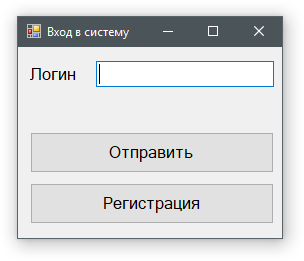
**Лабораторная работа №2. Аутентификация.Слово-вызов.**

**Десятов Александр**

**09-641**

****

using System;

using System.Windows.Forms;

using MySql.Data.MySqlClient;

namespace Клиентское

{

public partial class FormMain : Form

{

public FormMain()

{

InitializeComponent();

pF = new PublicFunctions();

}

MySqlConnection con;

PublicFunctions pF;// Для того, чтобы функции можно выполнять в разных формах

private void FormMain\_Shown(object sender, EventArgs e)

{

// Подключаемся к БД при открытии

con = new MySqlConnection("server=localhost;port=3306;database=usersandpasswords;user=root; charset=utf8");

con.Open();

}

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

{

// Отключение от БД при закрытии формы

con.Close();

}

// Клик по кнопке "Регистрация"

private void btn\_Registration\_Click(object sender, EventArgs e)

{

// В форму регистрации передаем логин, введенный в главной форме (чтоб быстрее зарегистрироваться)

FormRegistration fR = new FormRegistration(con, tB\_login.Text);

// При загрузки формы регистрации, спрятать главное окно

fR.Load += (s, args) => Hide();

// При закрытии формы регистрации, перейти обратно к главному окну (можно будет потом другую форму)

fR.FormClosed += (s, args) => Show();

// Показать форму регистрации

fR.Show();

}

// Клик по кнопке "Отправить"

private void btn\_Send\_Click(object sender, EventArgs e)

{

// Если такой логин не существует

if (!pF.CheckLoginExisting(con, tB\_login.Text))

{

label\_Message.Text = "Логин не существует";

return;

}

label\_Message.Text = "";

// Вызов функции (на сервере) (принимает лоигн), которая записывает "слово соли" в БД и его возвращает

string sold\_word = WriteAndReturnWord(tB\_login.Text);

// В форму аутентификации передаем логин, введенный в главной форме

FormAuthentication fA = new FormAuthentication(con, tB\_login.Text, sold\_word);

// При загрузки формы аутентификации, спрятать главное окно

fA.Load += (s, args) => Hide();

// При закрытии формы аутентификации, перейти обратно к главному окну

fA.FormClosed += (s, args) => Show();

// Показать форму аутентификации

fA.Show();

}

// Вызов функции (на сервере) (принимает лоигн), которая записывает "слово" в БД и его возвращает

private string WriteAndReturnWord(string login)

{

label\_Message.Text = "";

string par\_answer = "answer";

MySqlCommand command = new MySqlCommand("SELECT WriteAndReturnWord(@parInput) AS @parAnswer", con);

command.Parameters.AddWithValue("@parInput", login); // Передаем логин

command.Parameters.AddWithValue("@parAnswer", par\_answer);

MySqlDataReader reader = command.ExecuteReader();

reader.Read();

string word = (string)reader[par\_answer];

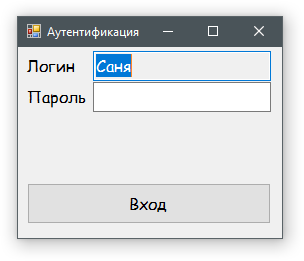
reader.Close();

return word;

}

}

}

****

using System;

using System.Windows.Forms;

using MySql.Data.MySqlClient;

namespace Клиентское

{

public partial class FormAuthentication : System.Windows.Forms.Form

{

public FormAuthentication(MySqlConnection MScon, string login, string sold\_word\_C)

{

InitializeComponent();

pF = new PublicFunctions();

// В форму регистрации передаем логин, введенный в форме аутентификации

sold\_word = sold\_word\_C;

tB\_login.Text = login;

con = MScon;

}

MySqlConnection con;

PublicFunctions pF;// Для того, чтобы функции можно выполнять в разных формах

string sold\_word;

// Клик по кнопке Вход

private void btn\_Login\_Click(object sender, EventArgs e)

{

//MD5(MD5(пароль)+слово) отправляем на сервер, как входной параметр для функции Разрешения Входа

string md5\_hash = pF.StringToMD5(pF.StringToMD5(tB\_password.Text) + sold\_word);

// Если возврат труе, то вход успешен, иначе пароль введен не верно.

if (CompareWithPasswordAndWord(tB\_login.Text, md5\_hash))

{

MessageBox.Show("Успешный вход");

label\_Message.Text = "";

}

else

{

label\_Message.Text = "Неверный пароль";

}

}

// Сравнить (на сервере) пароль с солью у клиента и в БД

private bool CompareWithPasswordAndWord(string login, string hashPasswordAndWord)

{

string par\_answer = "answer";

MySqlCommand command = new MySqlCommand("SELECT CompareWithPasswAndWord(@loginInput, @hashInput) AS @parAnswer", con);

command.Parameters.AddWithValue("@loginInput", login); // Передаем логин

command.Parameters.AddWithValue("@hashInput", hashPasswordAndWord); // Передаем хэш

command.Parameters.AddWithValue("@parAnswer", par\_answer);

MySqlDataReader reader = command.ExecuteReader();

reader.Read();

bool res = (bool)reader[par\_answer];

reader.Close();

return res;

}

}

}

**Серверная часть**

CREATE DEFINER = 'root'@'localhost'

FUNCTION usersandpasswords.CompareWithPasswAndWord(login\_input VARCHAR(255), hashPasswordAndWord\_input VARCHAR(255))

RETURNS tinyint(1)

READS SQL DATA

BEGIN

DECLARE hashPasswordAndWord\_DB, password\_DB, word\_DB VARCHAR(255);

SELECT

password, word

INTO

password\_DB, word\_DB

FROM users WHERE login = login\_input;

SET hashPasswordAndWord\_DB = MD5(CONCAT(password\_DB, word\_DB));

RETURN hashPasswordAndWord\_DB LIKE BINARY hashPasswordAndWord\_input;

END

CREATE DEFINER = 'root'@'localhost'

FUNCTION usersandpasswords.IsExistLogin(login\_input VARCHAR(255))

RETURNS tinyint(1)

READS SQL DATA

BEGIN

RETURN EXISTS (SELECT \*

FROM users

WHERE login = login\_input);

END

CREATE DEFINER = 'root'@'localhost'

FUNCTION usersandpasswords.WriteAndReturnWord(login\_from\_client VARCHAR(255))

RETURNS varchar(255) CHARSET utf8

DETERMINISTIC

MODIFIES SQL DATA

BEGIN

DECLARE new\_word VARCHAR(255);

# UNIX\_TIMESTAMP() - секунд с 1970

SET new\_word = CONVERT(UNIX\_TIMESTAMP(), CHAR);

# Обновляем в таблице

UPDATE users SET word = new\_word WHERE login = login\_from\_client;

RETURN new\_word;

END