Київський національний університет імені Тараса Шевченка

Факультет інформаційних технологій

Кафедра програмних систем і технологій

ЗВІТ

з лабораторної роботи № 6

Тема: “Мережеве програмування”

Дисципліна «Об’єктно-орієнтоване конструювання програм»

Підготував:

студент гр. ІПЗ-23(1)

Гоша Давід Олександрович

Дата : 12.04.22

Перевірила:

доц. Зубик Л.В.

# Київ – 2022

**Тема: Мережеве програмування**

**ЗАВДАННЯ**

Розробити просту програму «електронна відомість». Реалізувати можливість за запитом клієнта-викладача: додати ПІП студента та його оцінку в базу, змінити оцінку вказаного студента, видалити студента, повернути список усіх студентів і їх оцінок.

**Варіант 4**

**Код**

using System;

using System.Threading;

using System.Net.Sockets;

using System.Text;

using Newtonsoft.Json;

namespace ClientObject

{

class Program

{

static string userName;

private const string host = "127.0.0.1";

private const int port = 8888;

static TcpClient client;

static NetworkStream stream;

public class Package

{

public string To { get; set; }

public string From { get; set; }

public string Data { get; set; }

}

static void Main(string[] args)

{

Console.Write("Введите свое имя: ");

userName = Console.ReadLine();

client = new TcpClient();

try

{

client.Connect(host, port); //подключение клиента

stream = client.GetStream(); // получаем поток

//string message = userName;

//byte[] data = Encoding.Unicode.GetBytes(message);

//stream.Write(data, 0, data.Length);

// запускаем новый поток для получения данных

Thread receiveThread = new Thread(new ThreadStart(ReceiveMessage));

receiveThread.Start(); //старт потока

Console.WriteLine("Добро пожаловать, {0}", userName);

SendMessage(stream);

}

catch (Exception ex)

{

Console.WriteLine(ex.Message);

}

finally

{

Disconnect();

}

}

// отправка сообщений

static void SendMessage(NetworkStream stream)

{

string message = "Unknown error";

while (true)

{

Console.WriteLine("Введите /команду адресс и сообщение: ");

message = Console.ReadLine();

string[] splited = message.Split(" ");

switch (splited[0])

{

case "/exit":

Disconnect();

break;

case "/PrintStudents":

SendToContact(userName, splited[0], stream);

break;

case "/AddStudent":

AddStudent(splited[1], splited, stream);

break;

case "/NewMark":

NewMark(splited, stream);

break;

case "/DelStudent":

DelStudent(splited, stream);

break;

default:

SendToContact(userName, splited[0], stream);

break;

}

}

}

// получение сообщений

static void ReceiveMessage()

{

while (true)

{

try

{

byte[] data = new byte[64]; // буфер для получаемых данных

StringBuilder builder = new StringBuilder();

int bytes = 0;

do

{

bytes = stream.Read(data, 0, data.Length);

builder.Append(Encoding.Unicode.GetString(data, 0, bytes));

}

while (stream.DataAvailable);

string message = builder.ToString();

Console.WriteLine(message);//вывод сообщения

}

catch

{

Console.WriteLine("Подключение прервано!"); //соединение было прервано

Console.ReadLine();

Disconnect();

}

}

}

public static void SendToContact(string address, string message, NetworkStream stream)

{

Package new\_pack = new Package();

{

new\_pack.From = userName;

new\_pack.Data = message;

}

new\_pack.To = address;

Send(new\_pack, stream);

}

public static void AddStudent(string address, string[] message, NetworkStream stream)

{

string msg = "";

for (int i = 0; i < message.Length; i++)

{

msg += message[i] + " ";

}

Package new\_pack = new Package();

{

new\_pack.From = userName;

new\_pack.Data = msg;

}

new\_pack.To = address;

Send(new\_pack, stream);

}

public static void NewMark(string[] message, NetworkStream stream)

{

string msg = "";

for (int i = 0; i < message.Length; i++)

{

msg += message[i] + " ";

}

Package new\_pack = new Package();

{

new\_pack.From = userName;

new\_pack.Data = msg;

}

new\_pack.To = "Server";

Send(new\_pack, stream);

}

public static void DelStudent(string[] message, NetworkStream stream)

{

string msg = "";

for (int i = 0; i < message.Length; i++)

{

msg += message[i] + " ";

}

Package new\_pack = new Package();

{

new\_pack.From = userName;

new\_pack.Data = msg;

}

new\_pack.To = "Server";

Send(new\_pack, stream);

}

public static void Send(Package package, NetworkStream stream)

{

var json = JsonConvert.SerializeObject(package);

byte[] data = Encoding.UTF8.GetBytes(json);

stream.Write(data, 0, data.Length);

}

static void Disconnect()

{

if (stream != null)

stream.Close();//отключение потока

if (client != null)

client.Close();//отключение клиента

Environment.Exit(0); //завершение процесса

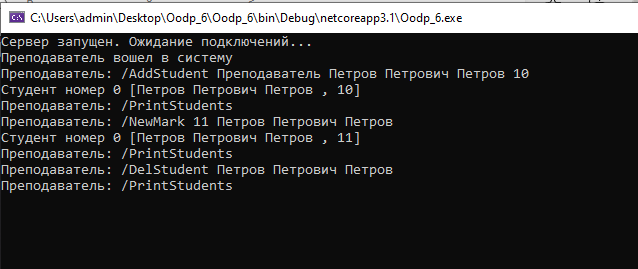
}

}

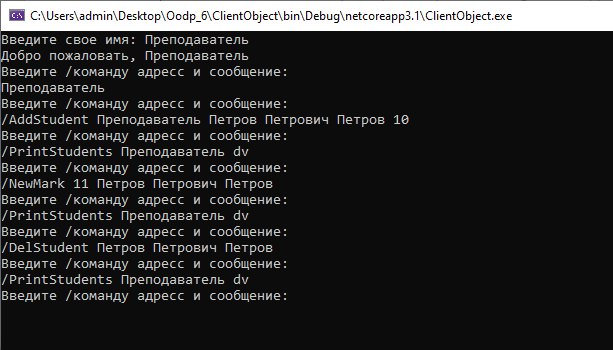
}

## Скриншот Виконання

Сервер



Клієнт



**Висновки:**

У лабораторній роботі номер 6 був розроблеий простий застосунок «електронна відомість». Реалізувані можливість за запитом клієнта-викладача:

* додати ПІП студента та його оцінку в базу,
* змінити оцінку вказаного студента,
* видалити студента,
* повернути список усіх студентів і їх оцінок.