САНКТ-ПЕТЕРБУРГСКОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ

«ПОЛИТЕХНИЧЕСКИЙ КОЛЛЕДЖ ГОРОДСКОГО ХОЗЯЙСТВА»

09.02.03 Программирование в компьютерных системах

код специальности наименование специальности

**Отчет о прохождении учебной практики**

**УП 03.01 «Участие в интеграции программных модулей»**

**по профессиональному модулю ПМ 03**

**«Участие в интеграции программных модулей»**

**(МДК 01.02 «Прикладное программирование»)**

Группа\_\_ИП-17-4\_\_\_\_\_\_

Студент\_\_Замесов Л.С.\_\_\_\_\_\_\_\_

Ф.И.О

Дата проведения УП 03.01 с «14»ноября 2020 по «26»декабря 2020

Оценка за прохождение практики \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Руководитель практики Коноплев А.А.

Ф.И.О

Оглавление

[Задание 1 (Интеграция с сервером прогноза погоды) 2](#_Toc59734279)

[Задание 2 (Интеграция с новостным агрегатором) 7](#_Toc59734280)

[Задание 3 (Интеграция с Telegram) 15](#_Toc59734281)

# Задание 1 (Интеграция с сервером прогноза погоды)

Регистрируемся на сайте https://openweathermap.org/api , заходим в раздел Pricing и получаем API key

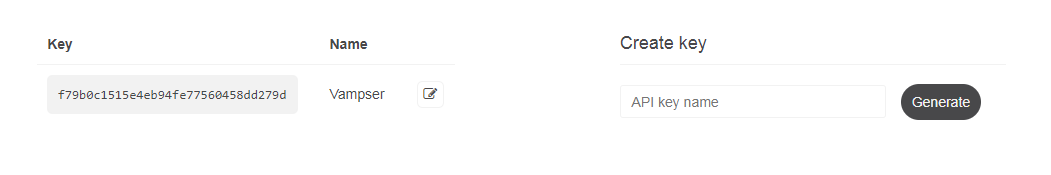


Рис.1 Получаем API Key на OpenWeather

Отправляем запрос к API в postman и получаем ответ в формате JSON:

{

    "coord": {

        "lon": 30.26,

        "lat": 59.89

    },

    "weather": [

        {

            "id": 804,

            "main": "Clouds",

            "description": "overcast clouds",

            "icon": "04n"

        }

    ],

    "base": "stations",

    "main": {

        "temp": -0.93,

        "feels\_like": -4.01,

        "temp\_min": -1.11,

        "temp\_max": -0.56,

        "pressure": 1019,

        "humidity": 86

    },

    "visibility": 10000,

    "wind": {

        "speed": 1,

        "deg": 0

    },

    "clouds": {

        "all": 90

    },

    "dt": 1608048753,

    "sys": {

        "type": 1,

        "id": 8926,

        "country": "RU",

        "sunrise": 1608015327,

        "sunset": 1608036788

    },

    "timezone": 10800,

    "id": 498817,

    "name": "Saint Petersburg",

    "cod": 200

}

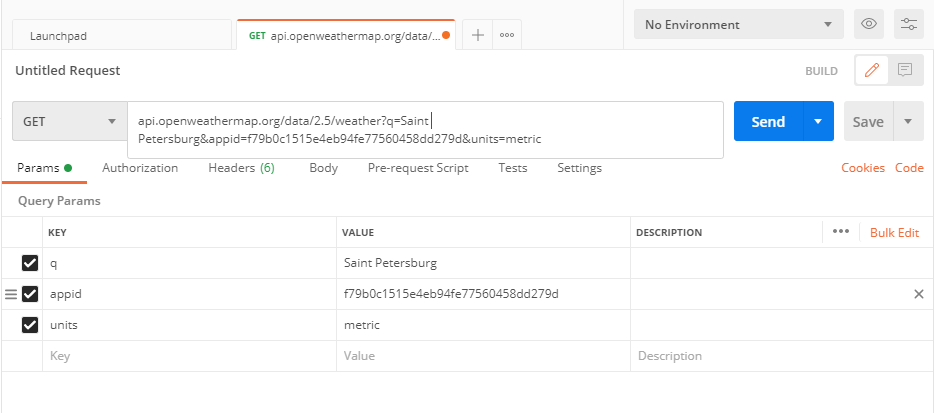


Рис.2 – Сформированный запрос

Программа обращающаяся в API

Блок схема:

Ввод города

Отправка запроса к API

Получение ответа

Вывод информации о погоде

Код c#:

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.Net.Http;

using Newtonsoft.Json;

using Погода.wthr;

namespace Погода

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private const string AppKey = "f79b0c1515e4eb94fe77560458dd279d";

private void listView1\_SelectedIndexChanged(object sender, EventArgs e)

{

}

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

{

var sity = Convert.ToString(listBox1.SelectedItem);

var client = new HttpClient();

var responce = client.GetAsync(new Uri($"http://api.openweathermap.org/data/2.5/weather?q={sity}&appid={AppKey}&units=metric&lang=ru")).GetAwaiter().GetResult();

var jsonResult = responce.Content.ReadAsStringAsync().GetAwaiter().GetResult();

var wthrInfo = JsonConvert.DeserializeObject<WeatherInfo>(jsonResult);

textBox1.Text = $"Город:{wthrInfo.Name},температура: { wthrInfo.Main.Temp}" + Environment.NewLine +$"Минимальная темп:{wthrInfo.Main.TempMin}, Максимальная темп: {wthrInfo.Main.TempMax}" + Environment.NewLine +

$"Ощущается как:{wthrInfo.Main.FeelsLike}";

}

private void listBox1\_SelectedIndexChanged(object sender, EventArgs e)

{

}

}

}

Классы:

1 класс

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using Newtonsoft.Json;

namespace Погода.wthr

{

public class WeatherInfo

{

[JsonProperty("main")]

public WeatherMain Main { get; set; }

[JsonProperty("id")]

public int Id { get; set; }

[JsonProperty("name")]

public string Name { get; set; }

}

}

2 класс

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using Newtonsoft.Json;

namespace Погода.wthr

{

public class WeatherMain

{

[JsonProperty("temp")]

public double Temp { get; set; }

[JsonProperty("feels\_like")]

public double FeelsLike { get; set; }

[JsonProperty("temp\_min")]

public double TempMin { get; set; }

[JsonProperty("temp\_max")]

public double TempMax { get; set; }

[JsonProperty("pressure")]

public int Pressure { get; set; }

[JsonProperty("humidity")]

public int Humidity { get; set; }

}

}

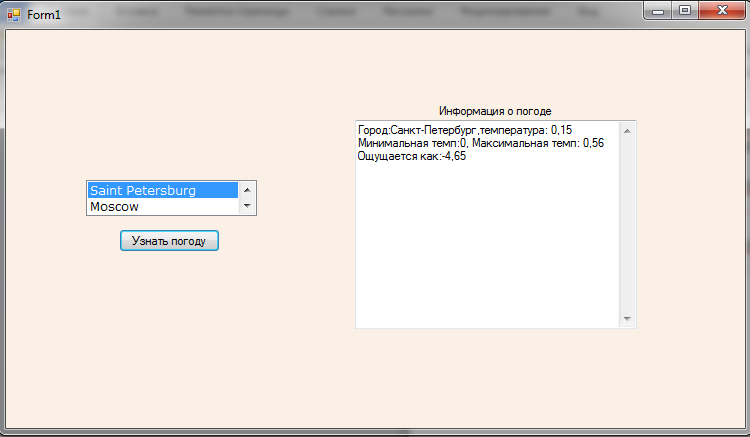


Рис.3 – Работа приложения

# Задание 2 (Интеграция с новостным агрегатором)

Заходим на сайт <https://apidoc.rwlabs.org/parameters>, переходим в раздел результаты и подбираем подходящий нам запрос к API

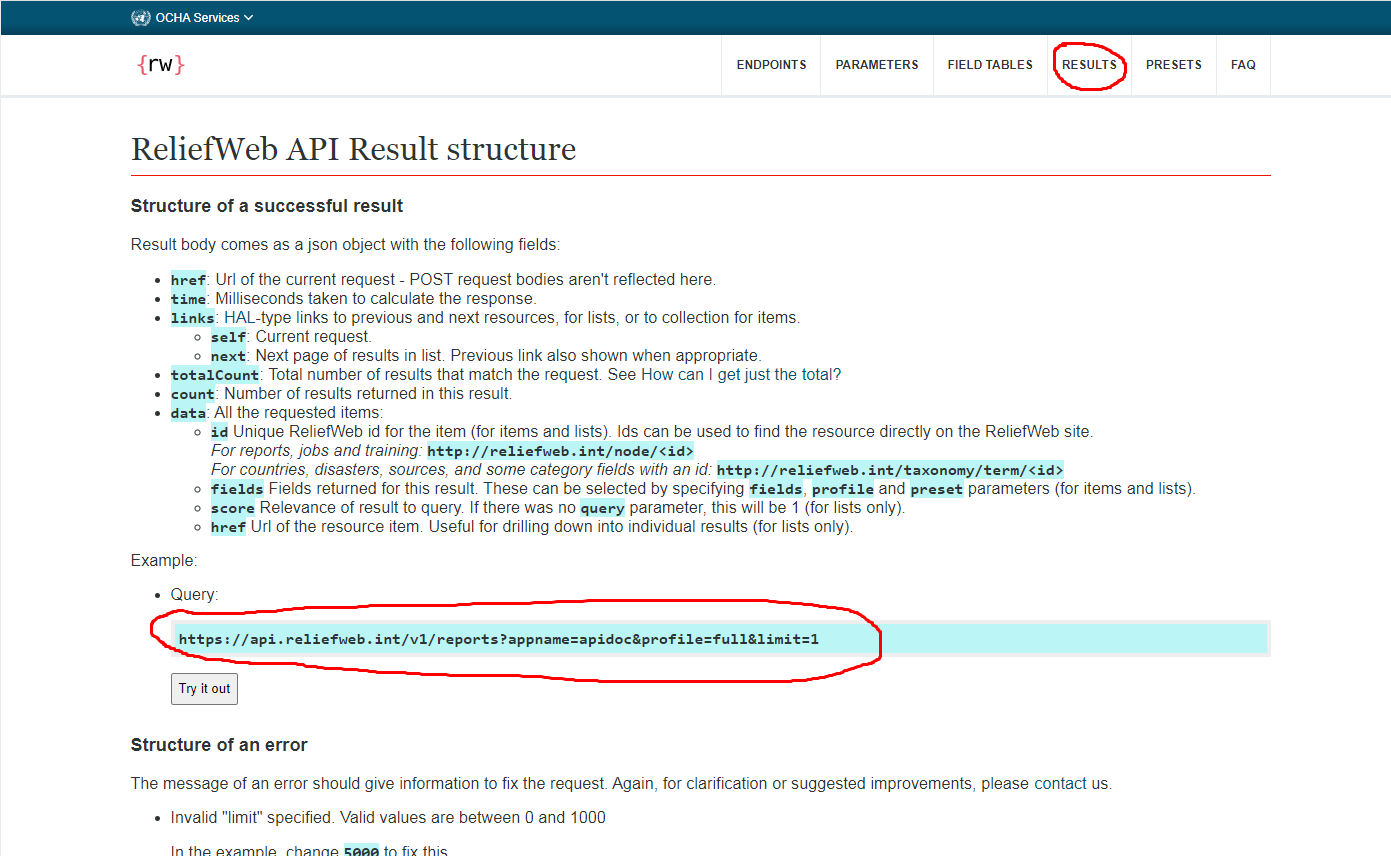


Рис.3 – Параметры запроса

Результат в формате JSON для данного запроса в Postman:

{

    "time": 30,

    "href": "https://api.reliefweb.int/v1/reports?appname=apidoc&profile=full&limit=1",

    "links": {

        "self": {

            "href": "https://api.reliefweb.int/v1/reports?appname=apidoc&profile=full&limit=1"

        },

        "next": {

            "href": "https://api.reliefweb.int/v1/reports?appname=apidoc&profile=full&limit=1&offset=1"

        }

    },

    "took": 21,

    "totalCount": 867011,

    "count": 1,

    "data": [

        {

            "id": "3182459",

            "score": 1,

            "fields": {

                "date": {

                    "changed": "2019-06-21T19:15:13+00:00",

                    "created": "2019-06-21T15:56:52+00:00",

                    "original": "2019-06-20T00:00:00+00:00"

                },

                "country": [

                    {

                        "href": "https://api.reliefweb.int/v1/countries/216",

                        "name": "Somalia",

                        "location": {

                            "lon": 47.33,

                            "lat": 5.79

                        },

                        "id": 216,

                        "shortname": "Somalia",

                        "iso3": "som",

                        "primary": **true**

                    }

                ],

                "primary\_country": {

                    "href": "https://api.reliefweb.int/v1/countries/216",

                    "name": "Somalia",

                    "location": {

                        "lon": 47.33,

                        "lat": 5.79

                    },

                    "id": 216,

                    "shortname": "Somalia",

                    "iso3": "som"

                },

                "format": [

                    {

                        "name": "News and Press Release",

                        "id": 8

                    }

                ],

                "language": [

                    {

                        "code": "en",

                        "name": "English",

                        "id": 267

                    }

                ],

                "source": [

                    {

                        "href": "https://api.reliefweb.int/v1/sources/12714",

                        "longname": "United Nations Assistance Mission in Somalia",

                        "name": "UN Assistance Mission in Somalia",

                        "id": 12714,

                        "type": {

                            "name": "International Organization",

                            "id": 272

                        },

                        "shortname": "UNSOM",

                        "homepage": "https://unsom.unmissions.org/"

                    }

                ],

                "title": "More than 100 former Al-Shabaab combatants graduate from Baidoa Rehabilitation Centre",

                "body": "Baidoa, 20 June 2019 - One hundred twenty-six former Al-Shabaab combatants today graduated from the Baidoa Rehabilitation Centre in Somalia’s South West State after completing an intensive rehabilitation programme.\n\nThe programme graduates, comprising youths from various parts of the country, received vocational training and psychological support from the centre. They thanked the programme managers, saying they were ready to join their fellow citizens in contributing positively towards Somalia’s stabilisation process.\n\nPresent at the ceremony were officials from the South West administration, the International Organisation for Migration (IOM) and the United Nations Assistance Mission in Somalia (UNSOM) and representatives of various UN agencies.\n\nSpeaking at the graduation ceremony, South West State Disarmament, Demobilisation and Reintegration (DDR) director, Ilyas Ali Nur, said the programme has proven to be a success, noting that hundreds of youths who defected from Al-Shabaab have been trained and reintegrated into the community.\n\n“On behalf of the South West regional government, I wish to announce that we are very happy with the success of this program. This graduation is for 126 ex-combatants, who were previously misled, but now are fully rehabilitated,” said Mr. Nur.\n \nThe training and reintegration of former combatants is an integral part of the National Programme for the Treatment and Handling of Disengaged Combatants in Somalia, led by the Office of the Defectors Rehabilitation Programme, under the federal Ministry of Internal Security.\n \nThe programme contributes to the long-term stabilisation of Somalia by promoting defections of Al-Shabaab combatants and reintegration of disengaged combatants into their communities.\n\nMr. Nur said the South West administration had played a major role in the success of the programme by providing security and necessary facilities, and monitoring progress of the defectors.\n \n“We also promise that we will receive and welcome with open hands as many combatants as possible,” he added.\n\nHis remarks were echoed by Aden Hussein, IOM National Project Officer, Disarmament Demobilisation and Reintegration Programme, who noted that the rehabilitation programme will encourage more youths in Al-Shabaab to lay down arms and contribute to the country’s peace process.\n\n“The ex-combatants undertook various vocational training lessons, among them, electricity, masonry, tailoring as well as welding,” said Mr. Hussein.\n\nHe called for more support for the programme, saying Al-Shabaab cannot only be fought using arms, but also through training and the reintegration of ex-combatants. “This [DDR programme] is the best method, and hopefully the graduates will integrate back into the community and be constructive in their communities,” Mr. Aden observed.\n\nUNSOM, IOM, the German government and other development partners have supported the establishment of similar rehabilitation and transition centres in Mogadishu, Belet Weyne and Kismayo.\n\nThe centres offer vocational training and psychological counselling to prepare former combatants for their reintegration into local communities.",

                "url": "https://reliefweb.int/node/3182459",

                "url\_alias": "https://reliefweb.int/report/somalia/more-100-former-al-shabaab-combatants-graduate-baidoa-rehabilitation-centre",

                "theme": [

                    {

                        "name": "Peacekeeping and Peacebuilding",

                        "id": 4599

                    }

                ],

                "id": 3182459,

                "status": "published",

                "body-html": "<p>Baidoa, 20 June 2019 - One hundred twenty-six former Al-Shabaab combatants today graduated from the Baidoa Rehabilitation Centre in Somalia’s South West State after completing an intensive rehabilitation programme.</p>\n\n<p>The programme graduates, comprising youths from various parts of the country, received vocational training and psychological support from the centre. They thanked the programme managers, saying they were ready to join their fellow citizens in contributing positively towards Somalia’s stabilisation process.</p>\n\n<p>Present at the ceremony were officials from the South West administration, the International Organisation for Migration (IOM) and the United Nations Assistance Mission in Somalia (UNSOM) and representatives of various UN agencies.</p>\n\n<p>Speaking at the graduation ceremony, South West State Disarmament, Demobilisation and Reintegration (DDR) director, Ilyas Ali Nur, said the programme has proven to be a success, noting that hundreds of youths who defected from Al-Shabaab have been trained and reintegrated into the community.</p>\n\n<p>“On behalf of the South West regional government, I wish to announce that we are very happy with the success of this program. This graduation is for 126 ex-combatants, who were previously misled, but now are fully rehabilitated,” said Mr. Nur.</p>\n\n<p>The training and reintegration of former combatants is an integral part of the National Programme for the Treatment and Handling of Disengaged Combatants in Somalia, led by the Office of the Defectors Rehabilitation Programme, under the federal Ministry of Internal Security.</p>\n\n<p>The programme contributes to the long-term stabilisation of Somalia by promoting defections of Al-Shabaab combatants and reintegration of disengaged combatants into their communities.</p>\n\n<p>Mr. Nur said the South West administration had played a major role in the success of the programme by providing security and necessary facilities, and monitoring progress of the defectors.</p>\n\n<p>“We also promise that we will receive and welcome with open hands as many combatants as possible,” he added.</p>\n\n<p>His remarks were echoed by Aden Hussein, IOM National Project Officer, Disarmament Demobilisation and Reintegration Programme, who noted that the rehabilitation programme will encourage more youths in Al-Shabaab to lay down arms and contribute to the country’s peace process.</p>\n\n<p>“The ex-combatants undertook various vocational training lessons, among them, electricity, masonry, tailoring as well as welding,” said Mr. Hussein.</p>\n\n<p>He called for more support for the programme, saying Al-Shabaab cannot only be fought using arms, but also through training and the reintegration of ex-combatants. “This [DDR programme] is the best method, and hopefully the graduates will integrate back into the community and be constructive in their communities,” Mr. Aden observed.</p>\n\n<p>UNSOM, IOM, the German government and other development partners have supported the establishment of similar rehabilitation and transition centres in Mogadishu, Belet Weyne and Kismayo.</p>\n\n<p>The centres offer vocational training and psychological counselling to prepare former combatants for their reintegration into local communities.</p>\n"

            },

            "href": "https://api.reliefweb.int/v1/reports/3182459"

        }

    ]

}

Получаем в JSON формате и видим что в body принялась вся статья. Дальше делаем приложение удалив параметр limit.

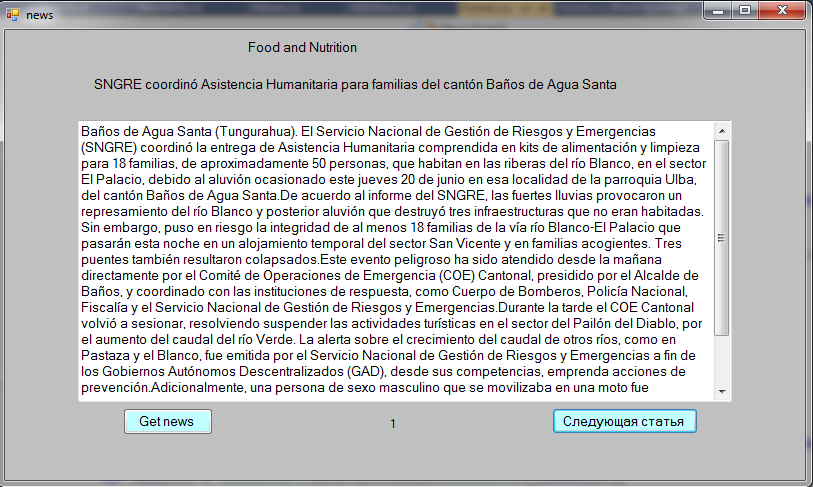


Рис.4 – Готовое приложение

На форму выводится информация из поля body в textbox, поля theme и title в label.

Блок схема:

Нажатие на кнопку Следующая статья

Нажатие на кнопку Get news

Отправка соответствующего запроса к API

Вывод полученной новости

Код программы:

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.Net.Http;

using Newtonsoft.Json;

using News.drugoe;

using System.Collections;

namespace News

{

public partial class Form1 : Form

{

private int M = 0;

public Form1()

{

InitializeComponent();

}

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

{

var client = new HttpClient();

var responce = await client.GetAsync(new Uri($"https://api.reliefweb.int/v1/reports?appname=apidoc&profile=full"));

var jsonResult = await responce.Content.ReadAsStringAsync();

var newsInfo = JsonConvert.DeserializeObject<NewsInfo>(jsonResult);

//textBox3.Text = $"\t Краткое содержание:" + Environment.NewLine +$"{newsInfo.Results[0].SummaryShort}";

label1.Text = $"{newsInfo.Data[M].AllIsNews.Theme[0].Name}";

textBox1.Text = $"{newsInfo.Data[M].AllIsNews.Body}";

label2.Text = Convert.ToString(M);

}

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

{

var client = new HttpClient();

var responce = await client.GetAsync(new Uri($"https://api.reliefweb.int/v1/reports?appname=apidoc&profile=full"));

var jsonResult = await responce.Content.ReadAsStringAsync();

var newsInfo = JsonConvert.DeserializeObject<NewsInfo>(jsonResult);

M += 1;

label1.Text = $"{newsInfo.Data[M].AllIsNews.Theme[0].Name}";

textBox1.Text = $"{newsInfo.Data[M].AllIsNews.Body}";

label2.Text = Convert.ToString(M);

}

}

}

Код класса NewsInfo:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using Newtonsoft.Json;

namespace News.drugoe

{

public class NewsInfo

{

[JsonProperty("time")]

public int Time { get; set; }

[JsonProperty("data")]

public List<Datum> Data { get; set; }

}

}

Код класса Datum:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using Newtonsoft.Json;

namespace News.drugoe

{

public class Datum

{

[JsonProperty("fields")]

public AllIsNews AllIsNews { get; set; }

}

}

Код класса AllIsNews:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using Newtonsoft.Json;

namespace News.drugoe

{

public class AllIsNews

{

[JsonProperty("title")]

public string Title { get; set; }

[JsonProperty("body")]

public string Body { get; set; }

[JsonProperty("theme")]

public List<Theme> Theme { get; set; }

}

}

Код класса Theme:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using Newtonsoft.Json;

namespace News.drugoe

{

public class Theme

{

[JsonProperty("name")]

public string Name { get; set; }

}

}

# Задание 3 (Интеграция с Telegram)

Для интеграции с Telegram для начала создаем собственного бота. Для этого обратимся к BotFather



Рис.5 – BotFather

Выполним команду /start и в полученном списке команд выбрать команду /newbot

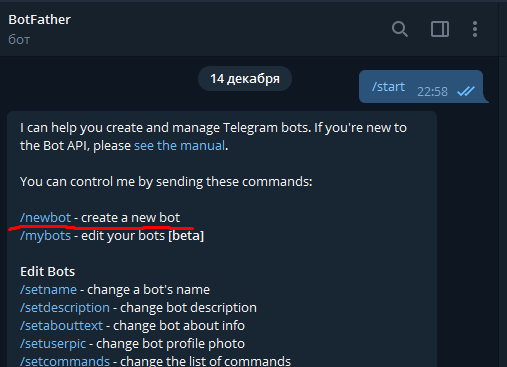


Рис.6 – Команда newbot

BotFather попросит придумать ему имя и юзернейм(условие оно должно оканчиваться на bot например TetrisBot или tetris\_bot)

После этого BotFather отправит вам токен для интеграции

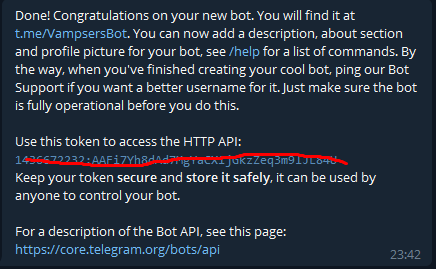


Рис.7 – Токен

С помощью токена мы будем отправлять запросы к API telegram в следующем виде: [https://api.telegram.org/bot<token>/НАЗВАНИЕ\_МЕТОДА](https://api.telegram.org/bot%3ctoken%3e/НАЗВАНИЕ_МЕТОДА)

После создания бота и получения токена приступаем к созданию приложения для обработки запросов к боту:

Создаем консольное приложение в VisualStudio, для получения сообщения от пользователя на понадобится метод getUpdates с полями из JSON text и chat\_id. Для отправки информации о погоде воспользуемся методом sendMessage, где в параметр chat\_id запишем id пользователя, а в text информацию о погоде, которую получим с помощью интеграции с новостным API, как в задании 2.

Блок схема:

Подключение к телеграм боту

Отправка сообщения телеграм боту

Получения сообщения из телеграма в программу

Обработка сообщения

Вывод ответа

Запускаем консольное приложение и проверяем работоспособность бота:

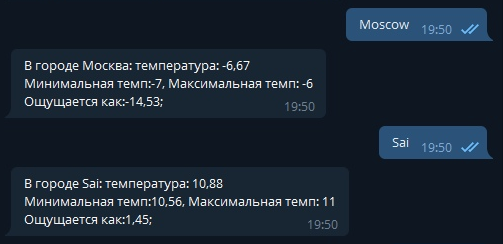


Рис.8 – Ответ от бота

Проверяем на другом пользователе:

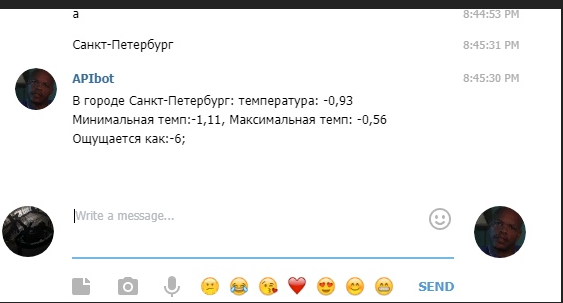


Рис.9 – Ответ от бота другому пользователю

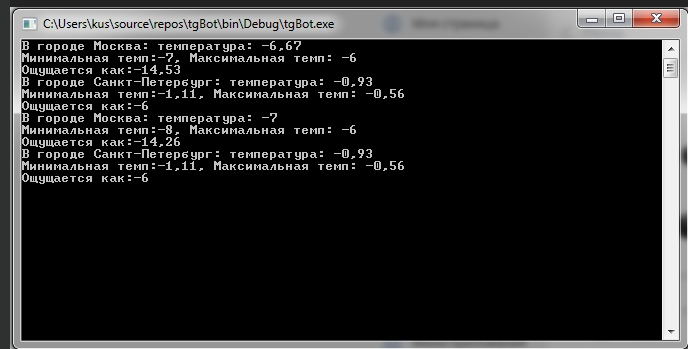


Рис.10 – Результат работы программы в консольном приложении

Код программы:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using Newtonsoft.Json;

using tgBot.tgClass;

using tgBot.weather;

using System.Net.Http;

using System.Net;

namespace tgBot

{

class Program

{

static void Main(string[] args)

{

var token = "1436672232:AAFi7Yh8dAd7MgYaCX1jGkzZeq3m9lJL84U";

var AppKey = "2f8579b26c23fa5a12ef2c67ced3778e";

var client = new HttpClient();

var client2 = new HttpClient();

var sity2 = "";

while (true)

{

ServicePointManager.SecurityProtocol = SecurityProtocolType.Tls12 | SecurityProtocolType.Tls11 | SecurityProtocolType.Tls;

var url = client2.GetAsync(new Uri($"https://api.telegram.org/bot{token}/getUpdates")).GetAwaiter().GetResult();

var tgJson = url.Content.ReadAsStringAsync().GetAwaiter().GetResult();

var tgInfo = JsonConvert.DeserializeObject<RootMessage>(tgJson);

var sity = tgInfo.Result.Last().Message.Text;

var id = tgInfo.Result.Last().Message.Chat.Id;

if (sity != sity2)

{

var urlW = client.GetAsync(new Uri($"http://api.openweathermap.org/data/2.5/weather?q={sity}&appid={AppKey}&units=metric&lang=ru")).GetAwaiter().GetResult();

var jsonResult = urlW.Content.ReadAsStringAsync().GetAwaiter().GetResult();

var wthrInfo = JsonConvert.DeserializeObject<weatherInfo>(jsonResult);

if (wthrInfo.Name == null)

{ continue; }

else

{

var pogoda = $"В городе {wthrInfo.Name}: температура: { wthrInfo.Main.Temp}" + Environment.NewLine + $"Минимальная темп:{wthrInfo.Main.TempMin}, Максимальная темп: {wthrInfo.Main.TempMax}" + Environment.NewLine +

$"Ощущается как:{wthrInfo.Main.FeelsLike}";

Console.WriteLine(pogoda);

//sendMessage

client2.GetAsync(new Uri($"https://api.telegram.org/bot{token}/sendMessage?chat\_id={id}&text={pogoda};")).GetAwaiter().GetResult();

sity2 = sity;

}

}

}

}

}

}

Используемые классы:

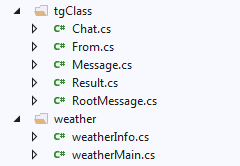


Рис.11 - Классы