

План создания Telegram бота на C# (.NET 9) для новичка

Шаг 1: Подготовка окружения

1. Установите Visual Studio 2022 с компонентами:
 - i. ASP.NET и веб-разработка
 - ii. Кроссплатформенная разработка .NET
 - iii. Windows приложения
2. Создайте бота в Telegram через @BotFather:
 - i. Отправьте /newbot
 - ii. Сохраните **токен** бота

Шаг 2: Создание проекта

1. Запустите Visual Studio → "Создать проект"
2. Выберите **Консольное приложение (.NET 9.0)**
3. Назовите проект (например, SimpleTelegramBot)

Шаг 3: Установка NuGet пакета

1. ПКМ по проекту → "Управление пакетами NuGet"
2. Найдите и установите:
 - i. Telegram.Bot (версия 22.7.5)

Шаг 4: Напишите код (Program.cs)

```
using Telegram.Bot;  
using Telegram.Bot.Polling;  
using Telegram.Bot.Types;  
using Telegram.Bot.Types.Enums;
```

```
// Замените на ваш токен от BotFather  
string botToken = "YOUR_BOT_TOKEN_HERE";
```

```
// Создаем клиент бота  
var botClient = new TelegramBotClient(botToken);
```

```
// Список шутливых ответов на вопрос "Как дела?"  
var funnyResponses = new[]  
{  
    "Отлично! Только что победил в конкурсе красоты среди ботов!",  
    "Лучше всех! Сегодня меня почти не дизлайкнули!",  
    "Как у робота: нет чувств, но много амбиций!",
```

```

    "Всё супер! Хотя кто я такой, чтобы жаловаться? Я же просто код!",
    "Превосходно! Только что обновил свои виртуальные мозги!"
};

// Обработчик обновлений
async Task HandleUpdateAsync(ITelegramBotClient botClient, Update update,
CancellationToken cancellationToken)
{
    // Работаем только с текстовыми сообщениями
    if (update.Type != UpdateType.Message || update.Message!.Type !=
MessageType.Text)
        return;

    var message = update.Message;
    var chatId = message.Chat.Id;
    var username = message.From?.FirstName ?? "Пользователь";

    // Обрабатываем команды и текст
    switch (message.Text)
    {
        case "/start":
            await botClient.SendMessage(
                chatId: chatId,
                text: $"Добро пожаловать, {username}!",
                cancellationToken: cancellationToken);
            break;

        case "/help":
            await botClient.SendMessage(
                chatId: chatId,
                text: "Этот бот умеет:\n" +
                    "/start - приветствие\n" +
                    "/help - список команд\n" +
                    "Привет! - ответить приветствием\n" +
                    "Как дела? - шуточный ответ",
                cancellationToken: cancellationToken);
            break;

        case "Привет!":
            await botClient.SendMessage(
                chatId: chatId,
                text: "И тебе привет!",
                cancellationToken: cancellationToken);
            break;
    }
}

```

```

        case "Как дела?":
            // Выбираем случайный ответ
            Random rnd = new();
            string randomResponse =
funnyResponses[rnd.Next(funnyResponses.Length)];

            await botClient.SendMessage(
                chatId: chatId,
                text: randomResponse,
                cancellationToken: cancellationToken);
            break;

        default:
            // Игнорируем неизвестные сообщения
            break;
    }
}

// Обработчик ошибок
Task HandlePollingErrorAsync(ITelegramBotClient botClient, Exception
exception, CancellationToken cancellationToken)
{
    Console.WriteLine($"Ошибка: {exception.Message}");
    return Task.CompletedTask;
}

// Настройка и запуск бота
var receiverOptions = new ReceiverOptions
{
    AllowedUpdates = Array.Empty<UpdateType>() // Получаем все типы
обновлений
};

// Запускаем бота
using var cts = new CancellationTokenSource();
botClient.StartReceiving(
    updateHandler: HandleUpdateAsync,
    errorHandler: HandlePollingErrorAsync,
    receiverOptions: receiverOptions,
    cancellationToken: cts.Token
);

// Получаем информацию о боте
var me = await botClient.GetMe();
Console.WriteLine($"Бот @{me.Username} запущен!");

```

```
Console.WriteLine("Для остановки нажмите Enter...");  
Console.ReadLine();
```

```
// Остановка бота  
cts.Cancel();
```

Шаг 5: Настройка и запуск

1. Замените `YOUR_BOT_TOKEN_HERE` на реальный токен бота
2. Нажмите `Ctrl + F5` для запуска без отладки

Шаг 6: Тестирование

1. Найдите своего бота в Telegram
2. Протестируйте команды:
 - i. `/start`
 - ii. `/help`
 - iii. "Привет!"
 - iv. "Как дела?"

Важные замечания:

1. **Токен бота** — это пароль, никогда не публикуйте его в открытом доступе
2. Для продакшена нужно добавить обработку ошибок и логирование
3. Бот работает только когда запущено консольное приложение
4. Для работы с русскими символами убедитесь что кодировка консоли поддерживает Unicode

Возможные улучшения:

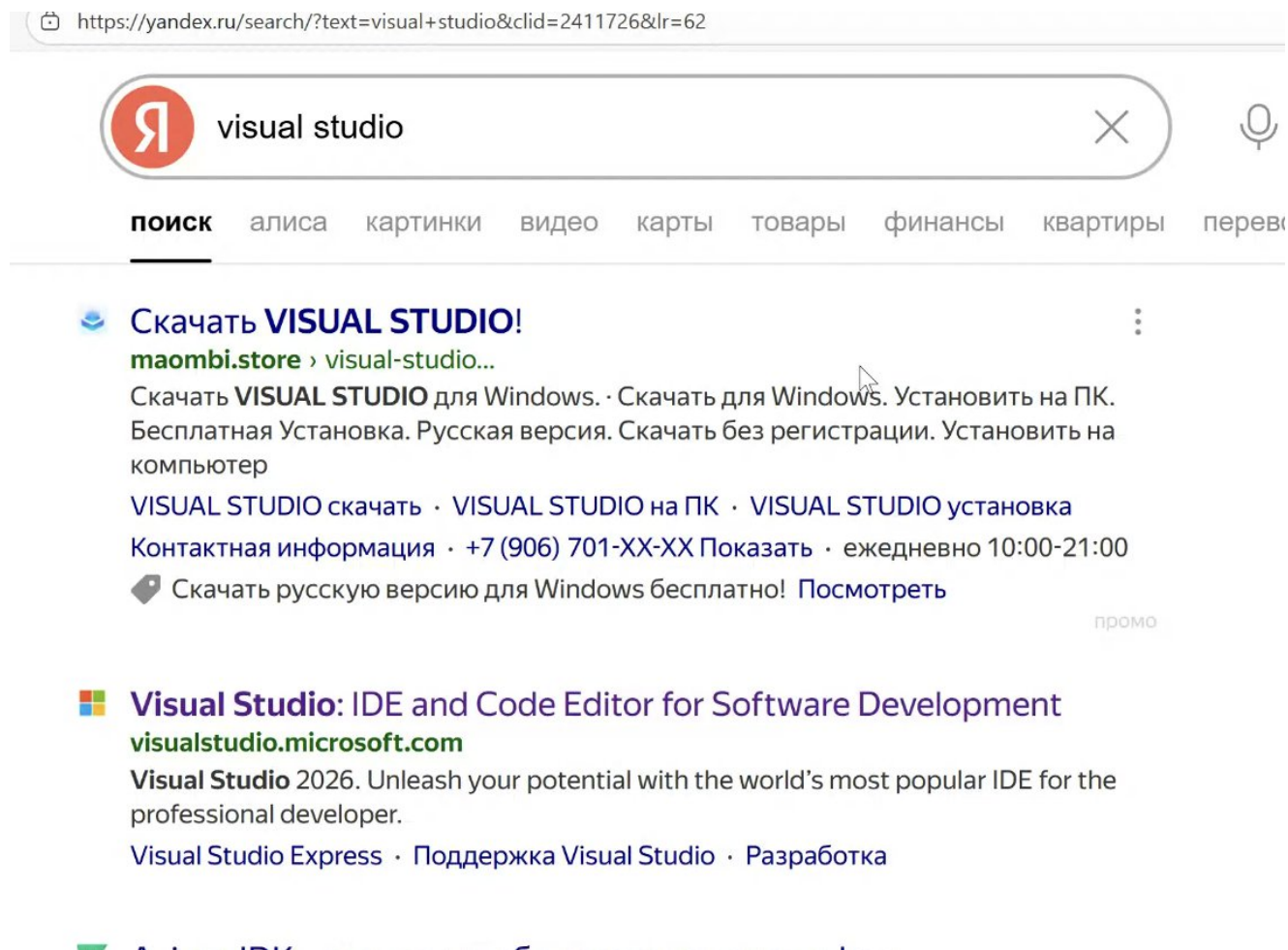
- Добавить логирование в файл
- Использовать `appsettings.json` для конфигурации
- Реализовать обработку исключений при сетевых проблемах
- Добавить больше интерактивных команд

Этот код специально упрощен для новичков и использует синхронную обработку сообщений. Для высоконагруженных ботов лучше использовать асинхронные методы и очереди.


Дополнение


Ниже представлены картинки (кратная инструкция) по установке и выбору пакетов для Microsoft Visual Studio 2022 (аналогично для 2026):

1) Выбираем Visual Studio



2) Скачиваем бесплатную community версию

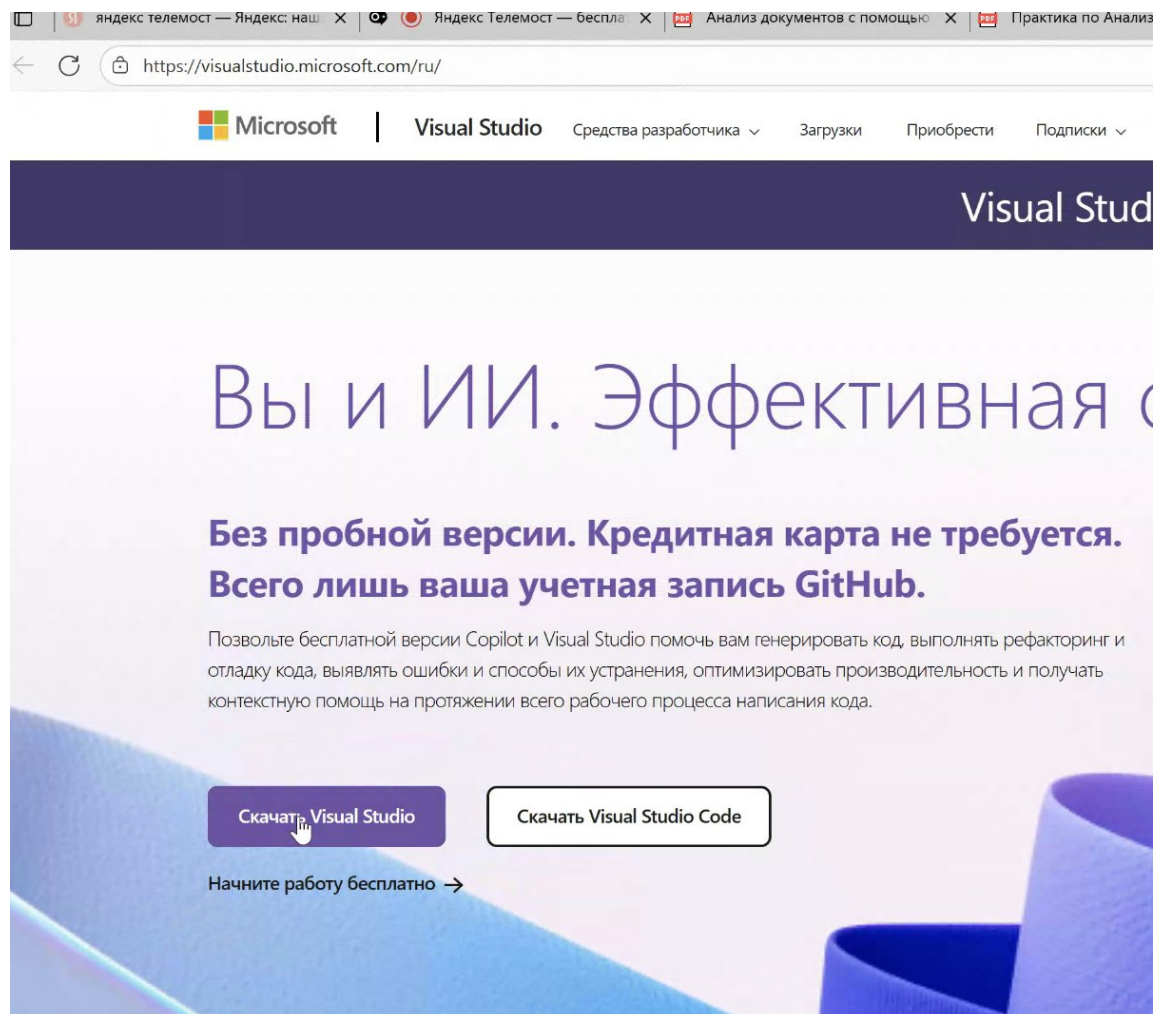


Visual Studio Community | 

Лучшая комплексная среда IDE для разработчиков .NET и C++ в Windows. Полноценный набор инструментов и функций для улучшения и усовершенствования каждого этапа разработки программного обеспечения.

[Подробнее →](#)

[Скачать бесплатно](#)



Visual Studio

Вы и ИИ. Эффективная с

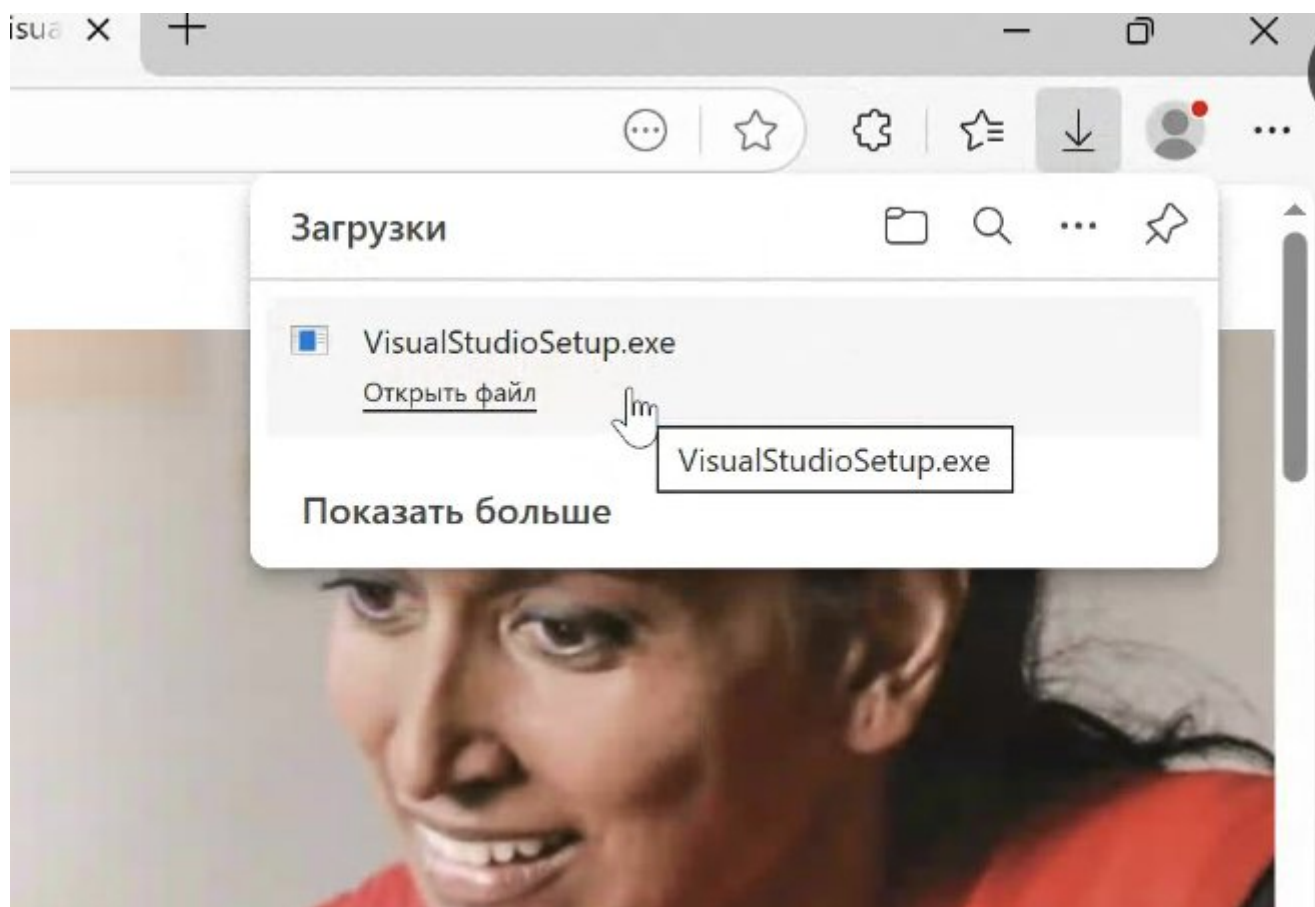
**Без пробной версии. Кредитная карта не требуется.
Всего лишь ваша учетная запись GitHub.**

Позвольте бесплатной версии Copilot и Visual Studio помочь вам генерировать код, выполнять рефакторинг и отладку кода, выявлять ошибки и способы их устранения, оптимизировать производительность и получать контекстную помощь на протяжении всего рабочего процесса написания кода.

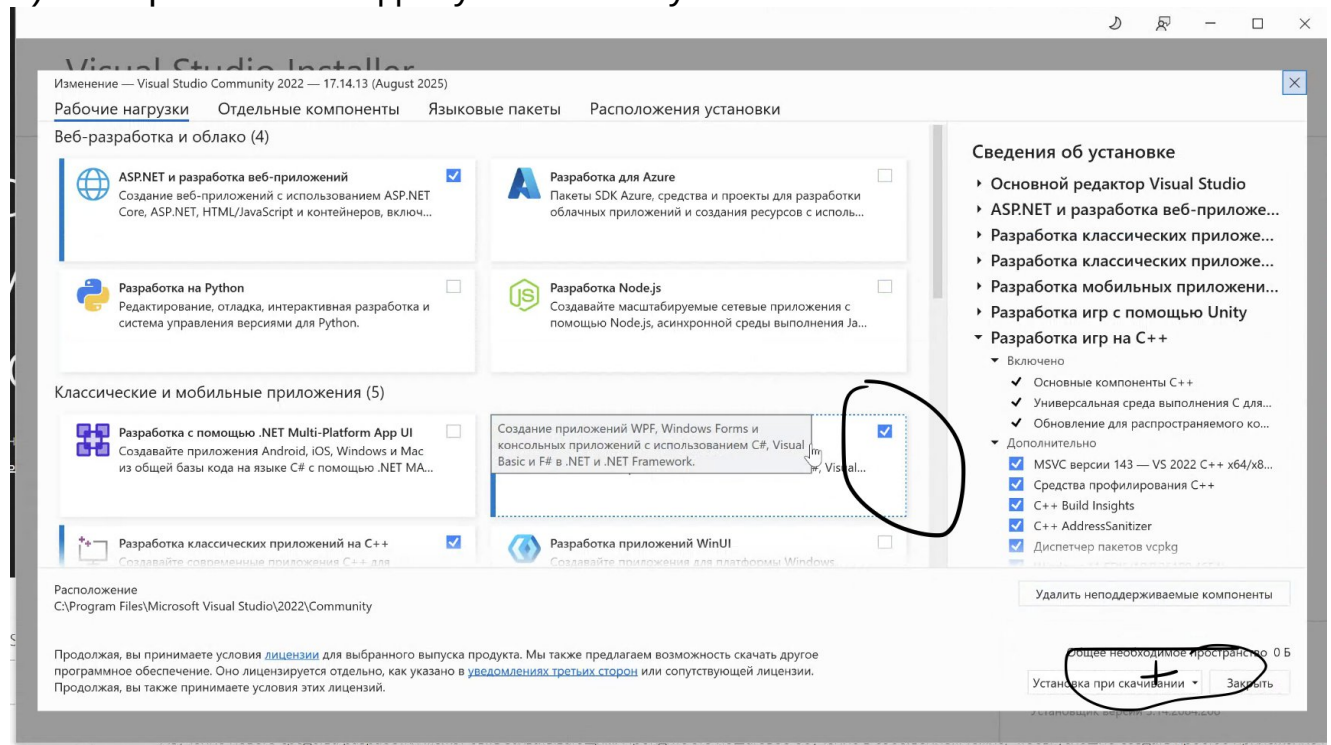
[Скачать Visual Studio](#) [Скачать Visual Studio Code](#)

[Начните работу бесплатно →](#)

3) Запускаем установочник

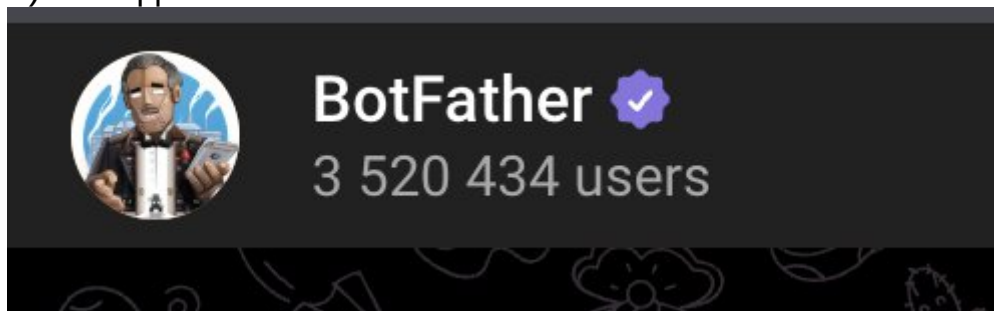


4) Выбираем пакеты для установки и устанавливаем

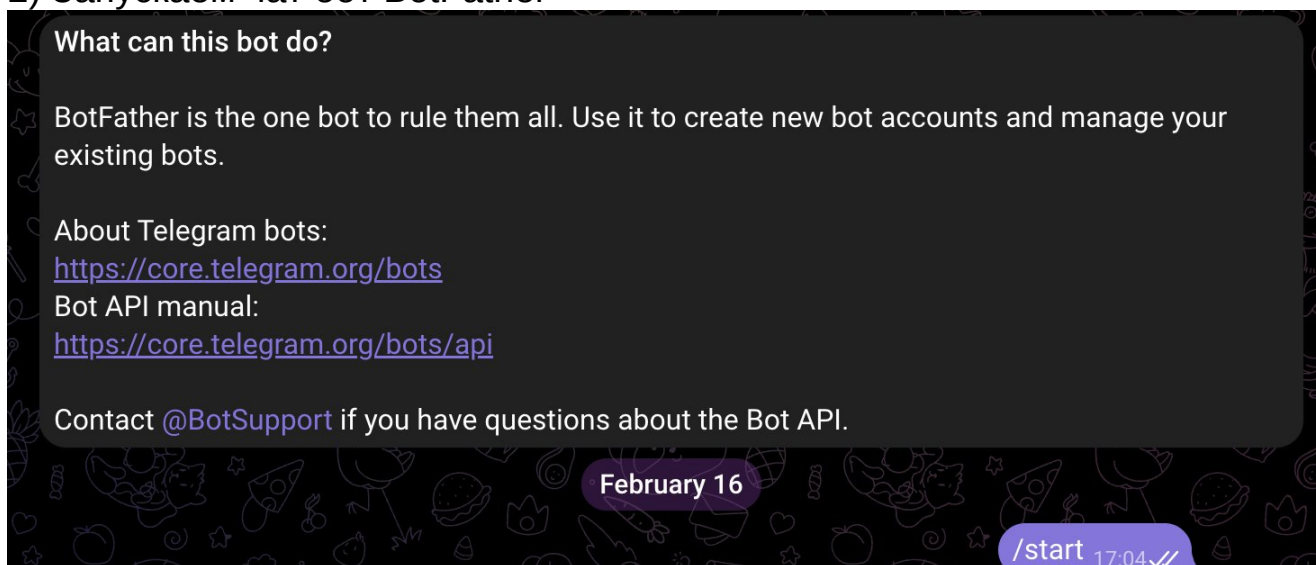


Ниже приведет пример получения токена в BotFather:

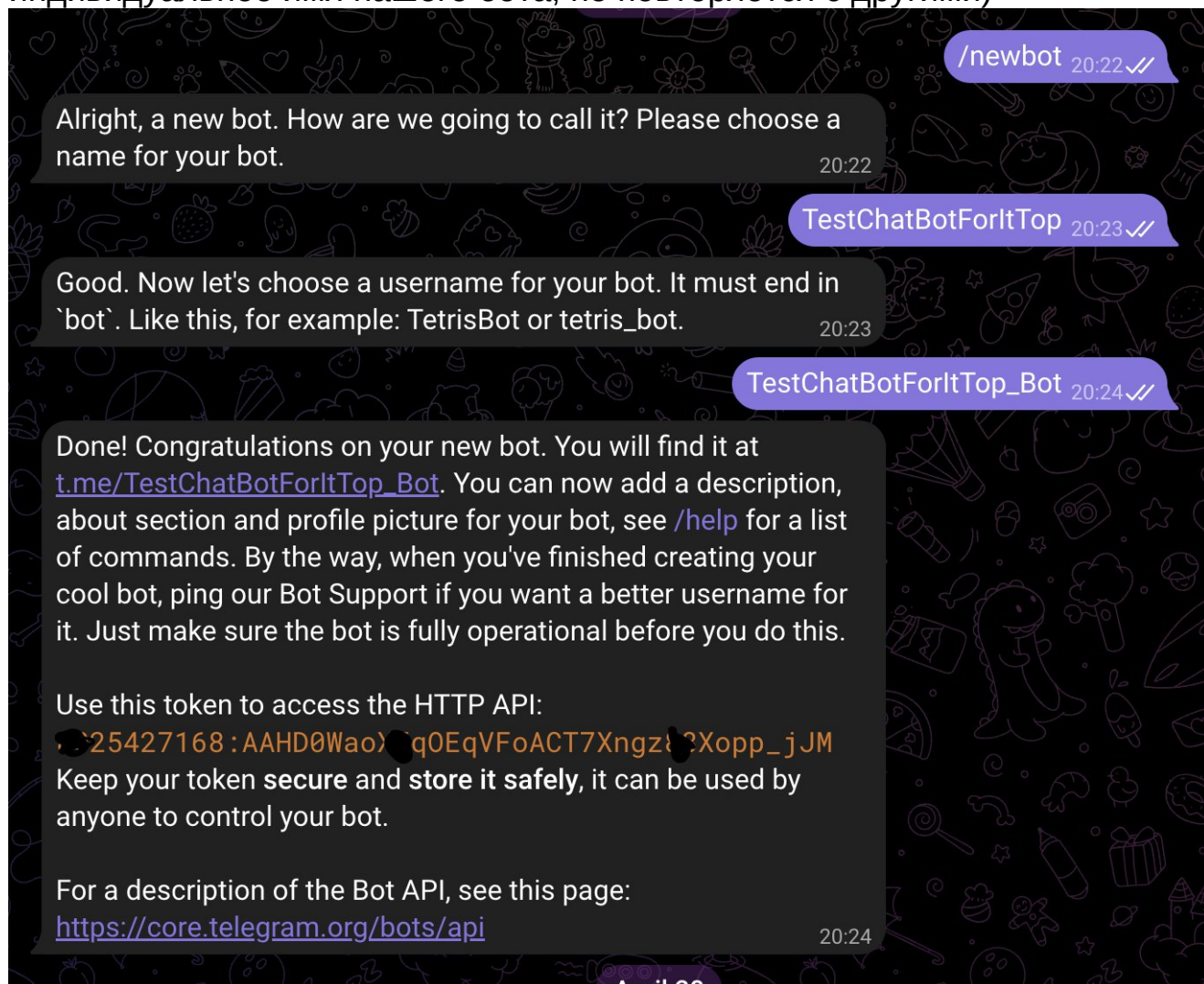
1) Находим BotFather в поиске



2) Запускаем чат бот BotFather



3) Вводим команду /newbot, указываем name и username нашего чат бота (name – имя нашего бота, может повторяться с другими; username – индивидуальное имя нашего бота, не повторяется с другими)



Ниже приведен пример работы чат бота, код которого написан выше:

