# Создание бота в мессенджере Telegram

Содержание

[Способ 1: Самостоятельно](https://lumpics.ru/how-to-make-bot-in-telegram/#_1)

[Шаг 1: Создание](https://lumpics.ru/how-to-make-bot-in-telegram/#_1-2)

[Шаг 2: Настройка](https://lumpics.ru/how-to-make-bot-in-telegram/#_2)

[Шаг 3: Разработка](https://lumpics.ru/how-to-make-bot-in-telegram/#_3)

[Шаг 4: Запуск и использование](https://lumpics.ru/how-to-make-bot-in-telegram/#_4)

[Шаг 5: Хостинг](https://lumpics.ru/how-to-make-bot-in-telegram/#_5)

[Способ 2: Специализированный сервис](https://lumpics.ru/how-to-make-bot-in-telegram/#_2-2)

[Вопросы и ответы](https://lumpics.ru/how-to-make-bot-in-telegram/#comments)

## Способ 1: Самостоятельно

Самостоятельное создание бота в [Telegram](https://lumpics.ru/telegram/) выполняется в несколько этапов и подразумевает комплексное использование как средств самого мессенджера, так и стороннего программного обеспечения.

### Шаг 1: Создание

Для того чтобы получить так называемую базовую основу для своего собственного бота, необходимо обратиться за помощью к специальному боту Телеграм.

[@BotFather](https://lumpics.ru/goto/aHR0cHM6Ly90Lm1lL0JvdEZhdGhlcg==)

1. Откройте мессенджер и либо воспользуйтесь представленной выше ссылкой, либо введите название бота в поисковую строку и откройте чат с ним. Нажмите на кнопку *«Запустить»*.
2. Отправьте указанную ниже команду:  
   /start
3. Затем либо введите и отправьте вручную, либо выберите в полученном сообщении следующее:  
   /newbot
4. Придумайте имя (название) для своего бота, введите и отправьте его. В дальнейшем это можно будет изменить.
5. Теперь придумайте имя пользователя для бота, которое одновременно будет и ссылкой на него. Этот адрес обязательно должен быть уникальным (свободным) и заканчиваться на *«\_bot»*.
6. В ответ вы получите сообщение, содержащее рабочую ссылку на пока еще пустого, не настроенного бота и токен, который будет использоваться для доступа к HTTP API. Обязательно сохраните его в безопасном месте, так как любой, кто получит доступ к этим данным, сможет управлять вашим ботом.

На этом непосредственное создание бота для Telegram можно считать завершенным, но на текущем этапе он будет представлять собой фактически пустой чат. Дальнейшая наша задача – его настройка средствами мессенджера и самостоятельная разработка.

### Шаг 2: Настройка

С помощью рассмотренного выше @BotFather можно фактически полностью настроить своего бота, сделать его оформление и интерфейс. Но главный нюанс здесь заключается в том, что данная процедура будет отличаться в каждом отдельном случае, так как определяется назначением и функциональностью создаваемого вами решения. Поэтому далее кратко рассмотрим лишь основные параметры.

Обратите внимание! Данный шаг инструкции подразумевает, что у вас уже есть как минимум основная идея для бота и общее представление о том, как он будет выглядеть и работать. О создании его «внутренней», программной части, будет рассказано в следующем шаге, а потому вы вполне можете начать с разработки и только после этого перейти к настройке.

1. Откройте чат с @BotFather, введите и отправьте указанную ниже команду:  
   /mybots
2. Выберите бота, которого требуется настроить.
3. Далее кликните по кнопке *«Edit Bot»*.
4. Поочередно нажимая на каждую (или только те, что сочтете нужными) кнопку, измените следующие параметры: 
   * *«Edit Name»* – имя бота (не @адрес и не ссылка, а то, что отображается в шапке чата);
   * *«Edit About»* – информация о боте;
   * *«Edit Description»* – описание возможностей;
   * *«Edit Description Picture»* – картинка для описания;
   * *«Edit Botpic»* – главное изображение, аватар;
   * *«Edit Commands»* – команды (кнопки), которые будут находиться в основном меню и выполнять заданные вами действия.

Для настройки каждого из доступных пунктов достаточно просто следовать подсказкам @BotFather. Даже если вы не знаете английский, понять их несложно, а в случае возникновения проблем всегда можно воспользоваться переводчиком.После того как вы внесете все необходимые изменения, кликните *«Back to Bot»*.

Обратите внимание! Все доступные для настройки опции в дальнейшем можно изменить. Для этого можете использовать не только основное меню @BotFather, но и быстрые команды, основные из них представлены в таблице ниже.

| Команда | Описание |
| --- | --- |
| */setname* | Изменение отображаемого имени бота |
| */setdescription* | Добавление описания |
| */setuserpic* | Смена аватара |
| */setcommands* | Установка и редактирование перечня команд, на которые бот должен реагировать |
| */deletebot* | Удаление добавленного бота |

1. Помимо указанных выше команд, в меню главного бота Телеграм имеются и другие – с их помощью можно интегрировать в свой проект веб-приложения и игры, изменять многие другие параметры и т.д. (напротив каждого пункта имеется описание на английском).   
   Рассмотрение каждого из них отдельно существенно усложнит инструкцию, а для многих сделает ее излишне подробной, поэтому данную часть настройки по необходимости можете освоить самостоятельно. Мы же представим ниже еще несколько важных команд и перейдем к следующему шагу.

| Команда | Описание |
| --- | --- |
| */token* | Создание нового токена или замена старого, если оригинальный скомпрометирован |
| */setinline* | Включение inline-режима, при котором бот будет реагировать на соответствующие команды во всех чатах, куда он добавлен |
| */setinlinefeedback* | Ответ на ввод отображением заданных вариантов (показом сообщения-шаблона, изображения, кнопки и т. д.) |
| */setprivacy* | Активация приватного режима, при котором бот будет распознавать исключительно команды и адресованные ему сообщения |

### Шаг 3: Разработка

Наиболее важный и, наверное, сложный, если говорить о малоопытных пользователях, этап в создании собственного Telegram-бота – его непосредственная разработка. В примере далее нами будет использоваться Python.

1. Скачайте и установите на компьютер актуальную версию Python, если этого не было сделано ранее. Более детально об этом – в отдельных статьях на нашем сайте.
2. По необходимости установите комфортную для себя среду разработки – в нашем случае это PyCharm от компании JetBrains, но вполне подойдет и стандартная IDLE, устанавливаемая вместе с Python, и даже Notepad++.
3. Установите библиотеки, необходимые для создания и обеспечения работоспособности Телеграм-бота. Для этого:
   * Запустите *«Командную строку»* (консоль), *«PowerShell»* или *«Терминал»* – в нашем случае подойдет любое решение. 
   * Поочередно введите указанные ниже команды, не забывая нажимать клавишу *«Enter»* для выполнения каждой из них:  
     python -m pip install --upgrade pip  
     pip install pyTelegramBotAPI

Примечание: pyTelegramBotAPI – библиотека, необходимая для обеспечения работы Telegram-бота, но это не единственное такого рода решение. В качестве альтернативы можете использовать telegram-bot – соответствующая команда для установки указана ниже. По этой библиотеке можно найти довольно много инструкций в интернете, но в нашем случае она работала некорректно.  
pip install python-telegram-bot

1. Откройте редактор кода и сделайте следующее:
   * Прежде всего импортируйте в него установленную на предыдущем шаге библиотеку. В случае с pyTelegramBotAPI запрос должен выглядеть следующим образом:  
     import telebot
   * Затем укажите токен для бота, который был вами получен при его создании от @BotFather:  
     TOKEN = 'YOUR\_TELEGRAM\_BOT\_TOKEN'  
     (замените YOUR\_TELEGRAM\_BOT\_TOKEN на свой ключ)
   * Создайте экземпляр бота — объект ‘bot’, который будет использоваться для взаимодействия с Telegram API с помощью переданного токена:  
     bot = telebot.TeleBot(TOKEN)

Обратите внимание! Описанное выше, — это обязательная «основа» для будущего бота. Далее – пример простейшего кода команды запуска и ответа с пояснениями, в вашем случае это может и наверняка будет отличаться, так как зависит исключительно от поставленной задачи.

* + Создаем обработчик команды ‘/start’:  
    @bot.message\_handler(commands=['start'])  
    def send\_welcome(message):  
    bot.reply\_to(message, "Привет! Я простой бот, рад познакомиться!")

Пояснение: Это декоратор для функции send\_welcome, который указывает, что она должна выполняться в случае, если пользователь отправляет команду /start. Когда это происходит, бот отвечает приветственным сообщением с текстом *«Привет! Я простой бот, рад познакомиться!»*.

* + Затем – добавляем обработчик текстовых сообщений:  
    @bot.message\_handler(func=lambda message: True)  
    def echo\_message(message):  
    bot.reply\_to(message, message.text)

Пояснение: Этот обработчик, обозначенный декоратором echo\_message, будет вызываться для всех текстовых сообщений, которые отправляют пользователи. Когда бот получает текстовое сообщение, он просто повторяет это сообщение обратно отправителю с помощью функции *bot.reply\_to()*.

* + Завершающая команда – запуск бота:  
    if \_\_name\_\_ == "\_\_main\_\_":  
    bot.polling()

Пояснение: Эта команда запускает бота и начинает прослушивать входящие обновления от Telegram. Как только пользователь отправит сообщение, обработчики, определенные ранее, будут вызваны и выполнят необходимые действия.

1. Сохраните проект Телеграм-бота в удобном месте на диске ПК (проследите, чтобы в его названии и на пути к нему не содержалось кириллических символов), в формате .py (в нашем случае, так как использовался Python).

### Шаг 4: Запуск и использование

Для того чтобы бот работал и был доступен к использованию в Telegram, его необходимо запустить. Сделать это можно с помощью любого варианта системной консоли, будь то *«Командная строка»*, *«PowerShell»* или *«Терминал»*.

1. Перейдите в папку с проектом вашего бота, зажмите клавишу *«Shift»* на клавиатуре, не отпуская ее, кликните правой кнопкой мышки (ПКМ) в пустой области и, далее, в зависимости от версии используемой операционной системы (в нашем примере – Windows 11) и/или ваших собственных пожеланий, выберите предпочтительный вариант консоли в контекстном меню.
2. Введите команду следующего вида и нажмите *«Enter»*:  
   python your\_telegram\_bot.py  
   ‘your\_telegram\_bot’ – название вашего бота, присвоенное ему при сохранении на последнем шаге предыдущей инструкции.

Примечание: Вместо выполнения двух предыдущих шагов вы можете самостоятельно запустить консоль, перейти из нее в папку с проектом и выполнить его запуск. Для этого поочередно введите и выполните следующие команды:  
cd путь\_к\_папке\_с\_проектом  
python your\_telegram\_bot.py

1. Перейдите в Телеграм и проверьте работоспособность своего проекта.
2. Бот будет работать и даже станет доступным для других пользователей мессенджера, но только тогда, когда он запущен в консоли.   
   Для остановки достаточно воспользоваться клавишами *«Ctrl+C»* или просто закрыть окно, для повторного запуска – выполнить действия из двух первых пунктов текущей инструкции.

### Шаг 5: Хостинг

Для того чтобы Telegram-бот работал постоянно, даже когда ваш компьютер выключен и/или проект не запущен в консоли, его необходимо разместить на хостинге. Это не самая сложная задача, но и откровенно простой ее назвать нельзя. Причем важно отметить, что сделать это бесплатно на сегодняшний день не получится, по крайней мере, если не рассматривать ограниченные по времени и/или функциональности решения. А с учетом того, что использование и настройка в каждом отдельном случае существенно отличается (это может делаться на веб-сайте, в десктопной программе или консоли), предоставить универсальную инструкцию не получиться.

Единственное, что мы можем порекомендовать – ввести в поисковую систему запрос приблизительно указанного ниже вида и последовательно ознакомиться со всей доступной информацией по теме. Наверняка вы сможете отыскать подходящее предложение.

хостинг телеграм бот

## Способ 2: Специализированный сервис

Альтернативой решению от Telegram и последующей самостоятельной разработке будет один из сторонних сервисов, который позволяет как создавать ботов, так и добавлять к ним дополнительную функциональность, причём последнее не требует навыков программирования. Одним из самых удобных решений такого рода является проект Manybot, которым мы и воспользуемся.

[Перейти на главную страницу Manybot](https://lumpics.ru/goto/aHR0cHM6Ly9tYW55Ym90LmlvL3J1)

1. Воспользуйтесь ссылкой выше, а после загрузки страницы нажмите на кнопку *«Создать бота»*.
2. Далее кликните *«Открыть Manybot в Telegram»*.  
   *Важно!* Этот сервис работает только с клиентским приложением, веб-версия не поддерживается!  
   Следуйте инструкциям используемого браузера для открытия приложения.
3. В появившемся чате нажмите на кнопку *«Запустить»*.
4. Выберите предпочтительный язык для «общения».
5. Далее отобразится приветственное сообщение с первоначальной инструкцией. Для создания нового бота необходимо нажать на соответствующую кнопку либо в поле ввода прописать команду */addbot*.
6. Для того чтобы создать нового бота, придётся воспользоваться упомянутым выше @botfather: повторите действия шагов 1-6 предыдущего способа. Затем скопируйте полученный токен, вставьте в форму сообщения и нажмите на кнопку отправки.
7. На этом шаге можно добавить описание бота – одно-два предложения о его возможностях и предназначении, которое необходимо отправить как сообщение. Если это вам не нужно, нажмите *«Пропустить»* или используйте команду /skip.
8. Основные действия по созданию бота выполнены, остаётся настроить его под конкретные нужды. Например, автоматика должна собирать информацию из RSS-потока на вашем сайте и постить его в имеющийся канал. Посредством Manybot это выполняется следующим образом: перейдите к своему боту (можно использовать ссылку, которая присутствует в основном диалоге Manybot) и нажмите *«Запустить»*.
9. Сразу появится сообщение с некоторыми инструкциями и доступом к параметрам. Для нашей цели нужны *«Настройки»*, поэтому нажмите на соответствующую кнопку.
10. Здесь выберите *«Автопостинг»*.
11. Воспользуйтесь опцией *«RSS лента»*.   
    Далее скопируйте соответствующую ссылку с вашего сайта, вставьте в поле ввода сообщения и отправьте боту.
12. Готово – теперь бот, подключенный к каналу, будет автоматически постить туда RSS-сообщения.

Функциональность ботов, созданных Manybot, достаточно обширная, поэтому применений можно найти почти под любые требования.

