

# Инструкция по установке и подключению Telegram-бота AgileBot

Версия 1.x

# Содержание

Введение в установку и подключение	3
1. Создание учетной записи бота в Telegram	5
2. Установка программы Telegram-бота	7
3. Настройка программы Telegram-бота	10
4. Подключение и запуск программы	14
5. Что дальше. Админ-панель бота	16

## Введение в установку и подключение

AgileBot 1.x - это программа или веб-приложение, которая подключается к учетной записи бота в Telegram и **состоит из 2 составляющих**:

- основной программы бота (программа бота и база данных бота)
- дополнительной программы бота (админ-панель бота).

**Для установки, подключения и запуска программы AgileBot, необходимо:**

1. Проверить сервер на требования работы программы Telegram-бота
2. Создать учетную запись бота в Telegram

После подключения к Telegram, бот AgileBot начинает работать. А дальше необходимо создавать или обновлять шаги бота и шаги пользователя.

**Шаг бота** - сообщение от бота пользователю или в канал, либо любое другое действие, выполняемое программой бота на сервере.

**Шаг пользователя** - это сообщение от пользователя боту или нажатие кнопки.

### **Требования для установки, подключения и запуска бота**

1. Сервер должен быть веб-сервером.

**Веб-сервер** - сервер предназначенный для работы сайтов или веб-приложений.

Основная программа бота выполнена как веб-приложение, а база данных программы бота как база данных для СУБД MySQL.

2. К этому веб-серверу, подключена СУБД MySQL.

Например, основная СУБД для современных сайтов это MySQL.

3. Включен протокол HTTPS и есть SSL-сертификат для этого протокола.

Учетная запись бота в Telegram будет связываться с основной программой бота на веб-сервере по протоколу HTTPS.

Но для работы этого протокола, нужен официальный SSL-сертификат, полученный из центра сертификации.

Например: SSL-сертификат из центра GlobalSign.

Поэтому, чтобы программа Telegram-бота заработала, нужно чтобы был включен HTTPS-протокол и имелся SSL-сертификат!

**Примечания:**

1. Требования 2 и 3 заданы самим мессенджером Telegram.
2. В качестве примера в инструкции берется новая папка на сайте, например папка на сайте компании.
3. **Описание работы, возможностей Telegram-ботов, их разработку,** можно посмотреть на официальном сайте Telegram:  
<https://core.telegram.org/bots>  
и на популярном неофициальном сайте Telegram:  
<https://tlgrm.ru/docs/bots>  
<https://tlgrm.ru/docs/bots/api>
4. Подробно про SSL-сертификат и HTTPS-протокол можно прочитать:  
<https://help.reg.ru/support/ssl-sertifikaty/obshchaya-informatsiya-po-ssl>

# 1. Создание учетной записи бота в Telegram

Состоит из нескольких шагов, где 2 шага обязательные (1.3 и 1.4):

1.1. Открыть специальный бот в Telegram, адрес бота - @BotFather, в котором и будет создаваться (регистрироваться) учетная запись бота:

1.2. Ввести или выбрать из меню команду: "/newbot".

Далее бот спросит (2 обязательных шага):

1.3. Имя бота в Telegram (поле "Имя" или "First Name")

1.4. Адрес бота, это то название, которое пишется после знака @

1.5. После задания адреса бота (1.4), присылается сообщение - что бот создан и даются идентификационные данные учетной записи (аккаунта) бота в Telegram (токен-ключ).

Этот токен-ключ надо обязательно записать, то есть шаг 1.5 - это запись токена-ключа.

## **Токен-ключ нужен для подключения программы бота к его учетной записи в Telegram!**

Это как "логин и пароль", но только для подключения основной программы бота к его учетной записи в Telegram.

Имеет вид, пример: 110201543:AAHdqTcvCHlvGWJxfSeofSAs0K5PALDsaw

Основная программа бота - это **файлы** бота в формате файлов для сайта (.php, .js, .css и др.) и **база данных** бота в СУБД MySQL.

Основная программа подключается к учетной записи бота через главный файл основной программы бота: bot.php.

Подключение главного файла в разделе 4 инструкции.

Дополнительное в программе бота - это **админ-панель** бота.

Визуальный веб-интерфейс для работы с ботом: создание и редактирование шагов бота и пользователей. Админ-панель работает через браузер. Необязательные шаги, которые лучше сделать после создания аккаунта бота в Telegram:

1.6. Задать список команд:

1.6.1. Ввести команду: "/setcommands"

1.6.2. Написать список команд.

Введите следующий список команд, которые работают в конструкторе бота, написаны в формате записи команд:

start - старт бота

menu - меню бота

help - список команд

Формат записи одной команды:

- это **одна строка**,
- название команды пишется без слэша,
- после записи строки, запятая не ставится (после краткого описания команды),
- формат записи в одной строке: название команды - краткое описание команды

После правильного ввода появится сообщение со словом "Success" (Успешно).

1.7. Задать краткое описания бота:

1.7.1. Ввести команду: "/setdescription"

1.7.2. Написать краткое описание бота - виде одной строки.

Например, оно будет показываться с названием бота (1.3) при открытии ссылки (URL-адреса) на бота в браузере, ссылка вида:

t.me/<адрес бота>, где <адрес бота> (1.4) в ссылке пишется без знака @.

Подробнее про создание учетной записи на неофициальном сайте:

<https://tigrm.ru/docs/bots#botfather>

## 2. Установка программы Telegram-бота

Для установки программы и подключения Telegram-бота, необходимо::

### 1. Комплект продукта (zip-архив), а точнее:

- основная программа бота, это файлы в папке `"/files"`,
- база данных бота, это файл `"basebot.sql"`.

### 2. Доступы (они доступны через админ-панель хостинга):

- к файлам и папкам на веб-сервере, где будет создаваться новая папка бота,
- через файловый менеджер (например программа "Проводник" в Windows), который предоставляется хостингом.  
А также можно через программу файловый менеджер на компьютере по FTP-протоколу (то что называют FTP-доступом)
- к СУБД MySQL через программу "phpMyAdmin".

После установки (раздел 2), настройки (раздел 3) и подключения Telegram-бота (раздел 4), бот работает и им можно будет пользоваться!

### Шаги по установке программы Telegram-бота:

#### 2.1. Установить основную программу бота на веб-сервер.

Установка основной программы - это просто копирование файлов.

2.1.1. Создать новую папку в корне сайта или другой вложенной папки на сайте,

2.1.2. Скопировать файлы из папки `"/files"` в созданную новую папку.

#### 2.2. Установка базы данных бота.

Установка базы данных - это установка дампа базы данных в виде одного файла в формате sql (`basebot.sql`).

2.2.1. Открыть панель хостинга, на котором находится сайт(открывается в браузере, например название "ISPmanager").

2.2.2. Создать новую базу данных: названия пишутся на английском языке.

2.2.2.1. Задать имя базы данных, имя любое,

2.2.2.2. Выбрать кодировку "utf8"!

Внимательно, выберите эту кодировку. Могут быть разные варианты с названием utf8, например "utf8mb4", надо выбрать просто "utf8"!

2.2.2.3. Задать имя пользователя, имя может быть любым.

2.2.2.4. Пароль пользователя любой (рекомендуется делать сложный от 9 символов). После выполнения шагов 2.2.2.1-2.2.2.4, создается пустая база данных.

2.2.3. Далее открыть веб-приложение phpMyAdmin.

Делается это через панель управления хостингом - найти пункт в меню панели управления на веб-приложении phpMyAdmin или например такой пункт "Web-интерфейс БД" или если есть ссылка (адрес) на phpMyAdmin например в письме при регистрации хостинга.

Скопировать адрес (ссылку) в адресную строку браузера и открыть. Веб-приложение работает как админ-панель для работы с СУБД MySQL.

Админ-панель и панель управления - это все веб-приложения (веб-программы).

Веб означает веб-формат предоставляемых данных - основная программа для получения данных в формате веб - браузер, например Chrome, Firefox, Yandex Browser, Safari, Edge.

2.2.4. Найти созданную базу данных. Для этого в веб-приложении в левой панели щелкнуть по названию базы данных.

В левой панели отображается список баз данных, которые есть у пользователя.

2.2.5. Открыть справа вкладку "Импорт",

2.2.5.1. Далее нажать кнопку "Выберите файл" и в появившемся окне выбрать файл дампа базы данных бота "*basebot.sql*" в папке на компьютере.



2.2.5.2. Проверить или выбрать кодировку "utf8". Остальные настройки импорта не трогаем.

2.2.5.3. Нажать кнопку "Вперед". После импорта дампа базы данных должна отобразиться фраза об успешном выполнении установки с зеленой подсветкой.

Зеленая подсветка как раз отражает успешное выполнение. Команда импорта выполняется как запросы к СУБД, множество запросов, которые после выполнения создают таблицы в этой базе данных.

Всё база данных бота установлена.

Чтобы посмотреть щелкните слева снова по названию базы данных, откроются

названия таблиц, первые 3:

"StepsBot" - таблица шагов бота,

"StepsChat" - таблица шагов пользователя,

"TableData" - таблица данных бота - данные полученные от пользователей.

*Примечание:* создавать и редактировать шаги бота или пользователя также можно здесь! Админ-панель бота - это тоже работа с таблицами, но в простом и в более удобном виде для обычного пользователя и плюс есть справка как работать.

### 3. Настройка программы Telegram-бота

После копирования файлов на сервер и установки (импорта) дампа базы данных, надо:

- **обязательно** сделать настройки в файле настроек бота "*configbot.php*";
- не обязательно в файле "*varsbot.php*".

3.1. Открыть файл "*configbot.php*" через файловый менеджер в панели хостинга или перед копированием в папку бота на сервере, открыть файл на компьютере.

3.2. И заполнить значения переменных (переменная термин кода программы).

#### Формат заполнения:

- текстовые значения, пишутся в кавычках (это все значения, кроме одного),
- числовые значения, пишутся числом, без кавычек! (в файле одно такое значение - в переменной "***\$Chat\_Id\_AdminBot***")
- после написания значения обязательно стоит точка с запятой ";" (в файле они поставлены, просто проверьте наличие).

3.2.1. В переменную ***\$BotData*** после знака "=" и в кавычках написать - **токен (ключ)**, который получили после создания учетной записи бота в виде: 110201543:AAHdqTcvCH1vGWJxfSeofSAs0K5PALDsaw.

Далее проверяем стоит ли точка запятой ";" и в последующих ниже переменных также проверять ";" после написания значения.

3.2.2. В переменную ***\$database*** написать - **название созданной базы данных** (пункт 2.2.2.1).

3.2.3. В переменную ***\$user*** - **имя пользователя** базы данных (2.2.2.3).

3.2.4. В переменную ***\$password*** - **пароль пользователя** базы данных (2.2.2.4).

3.2.5. В переменную ***\$LoginForm*** - написать любое **имя пользователя** админ-панели бота, по умолчанию написано значение - "Adminbot".

3.2.6. В переменную ***\$PasswordForm*** - **пароль пользователя** админ-панели бота, задайте обязательно сложный (пароль от 12 символов, буквы в нижнем и верхнем регистрах, а также и цифры).

Далее переменные, заполнение которых, не обязательно:

3.2.7. В переменную ***\$PathBot*** - **адрес до папки бота**, начинается с ,  
заполнение необязательное, но если нужно отправлять изображения тогда обязательно.

Это для типа шага бота: "SendImageWithButtons".

3.2.8. В переменную ***\$PathLog*** - **путь до папки логов**.

Задано значение по умолчанию, поэтому можно не заполнять.

3.2.9. В переменную ***\$PathFile*** - **путь до папки файлов**.

В этой папке будут файлы, которые нужны для записи данных от пользователей или любой заданной информации менеджером бота при выполнении типа шага бота: "SaveFile".

Задано значение по умолчанию, поэтому можно не заполнять.

3.2.10 В выражении в скобках, где строка:

"define("ReportError", true)" - задать разрешение на вывод сообщений об ошибках в чат бота в Телеграме при выполнении там шагов бота:

**true** - разрешить вывод,

**false** - не разрешить вывод.

Указываются значения без кавычек!

По умолчанию, значение true.

**Примечание:** этот вывод (репорт) применяется при тестировании (отладки) шагов бота, после того как их (шаги бота) создали или редактировали (обновляли) в админ-панели бота.

Если точнее, то эти текстовые сообщения нужны для выявления ошибок в задании параметров шагов бота (которые задаются в поле "Дополнение" на субвкладках "Создание шага" и "Редактирование шага" во вкладке "Шаги бота").

Если шаг выполняется верно, **то сообщение об ошибки не выводится!**

Поэтому можно оставить **постоянно включенным этот параметр**.

3.2.11. В переменную ***\$Chat\_Id\_AdminBot*** - **Id** главного менеджера бота.

Является не обязательным. Сюда вводится Id пользователя Telegram, который будет главным менеджером ботом или просто менеджером бота. Менеджер бота - пользователь Telegram, который будет добавлять и редактировать шаги или просто менеджер, которому будут приходиться данные от пользователей.

Для заполнения этой переменной нужно выполнить следующие шаги:

3.2.11.1. В строке где переменная убрать два слэша - "//".

Это означает раскомментировать строку на языке PHP.

3.2.11.2. Написать значение в переменную **`$Chat_Id_AdminBot - Id пользователя Telegram`**, значение числовое, без кавычек (около 10 цифр).

#### **Примечание:**

Числовые значения в PHP задаются без кавычек!

#### **Как узнать Id пользователя Telegram:**

- через специального бота в Telegram: `@getmyid_bot`
- либо в "Таблице шагов" в столбце "Id пользователя", тогда этому пользователю надо сделать один шаг - написать сообщение боту или нажать на кнопку в чате, и этот шаг запишется в таблицу шагов, а также в лог-файл бота.

Все шаги пользователей и бота записываются в "Таблицу шагов" во вкладке "Информация" в адм и файл логов бота.

- посмотреть в лог-файле бота: найти массив с названием "from" и в нем найти параметр "id". Массив "from" находится в большой текстовой строке в лог-файле.

#### **Примечание:**

В этой строке данные, которые передаются из Telegram в программу бота: каждое сообщение от пользователя или действие, передается из Telegram в программу в виде большой текстовой строки (в виде несколько массивов данных, один из них массив "from" - данные от пользователя, который написал сообщение или например нажал на кнопку).

В терминах Telegram оно называется как "Update" (Обновления).

Этот термин для разработчиков Telegram-ботов.

- посмотреть в лог-файле: найти параметр с названием "Chat Id", который расположен ниже за большой текстовой строкой.

3.2.11. Закрывать файл.

Файл *varsbot.php* содержит текстовые сообщения ошибок при неверном вводе менеджером бота параметров типа шага бота (параметры, которые задаются в поле “Дополнения”).

Эти сообщения выводятся в чате бота с пользователем, если менеджер бота не правильно записал параметр, например неправильное имя параметра или значение параметра.

То есть как сказано было выше применяются при тестировании шагов бота и пользователя - когда шаги бота и пользователя через админ-панели были введены, и затем тестируются на правильную работу.

**Примечание:**

Особенность этих сообщений об ошибках, в том что они выводиться в напрямую чат при выполнении шага бота, что позволит сразу отследить неверный заданный шаг бота.

Если шаг задан верно, то сообщения не будут появляться.

## 4. Подключение и запуск программы

Запуск Telegram-бота заключается просто в подключение главного файла бота (*bot.php*) на веб-сервере к учетной записи бота в Telegram:

4.1. Открыть браузер,

4.2. Написать:

<https://api.telegram.org/bot<token>/setWebHook?url=<addressbota>>

Подробнее на странице документации:

<https://tigrm.ru/docs/bots/api#setwebhook>

<token> - токен(ключ) учетной записи бота (пункт 1.5),

<addressbota> - адрес до главного файла бота, это в папки бота файл "bot.php".

Пример: токен возьмем, который как пример, указан выше:

110201543:AAHdqTcvCH1vGWJxfSeofSAs0K5PALDsaw.

Отдельная папка бота на сервере пусть будет иметь название "/bot", а название сайта компании "company.ru", тогда полный URL-адрес: (открытие этого URL-адреса в браузере - это как выполнение запроса на подключение):

<https://api.telegram.org/bot110201543:AAHdqTcvCH1vGWJxfSeofSAs0K5PALDsaw/setWebHook?url=https://company.ru/bot/bot.php>

Пишем этот URL-адрес (2 выше строки это 1 строка) в браузере и открываем. На самом деле при открытии этого URL-адреса, на сайте Telegram выполняется скрипт, который подключает главный файл основной программы бота с сервером мессенджера Telegram!

**При успешном подключении**, бот (программа бота) **начинает работать!**

И теперь бот может принимать и отправлять сообщения через свою учетную запись в Telegram!

После открытия этого адреса в браузере (выполнения скрипта) откроется страница, в которой в одну строку будет написано сообщение на английском языке - (на самом деле это JSON массив), что WebHook установлен:

```
{"ok":true,"result":true,"description":"Webhook was set"}
```

Это сообщение надо обязательно проверить! - Так оно не означает что бот работает и файл правильно подключен!

### ***Для проверки подключения:***

1. Надо перейти в чат бота в Телеграме и написать или выбрать команду: “/start”.
2. Если появиться сообщение от бота, начинающиеся со слов: “Добрый день! Это бот ..” - значит подключение произведено!

Если сообщение в Телеграме не появилось, то нужно проверить, основное:

- **правильно написан** в URL-запросе **полный путь до главного файла** и сам главный файл (**название файла**) в параметре url=,
- **установлен (импортирован) дамп базы данных** в СУБД MySQL (то есть сделаны и полностью шаги, которые в пункте 2.2 этого руководства).

## 5. Что дальше. Админ-панель бота.

После установки и подключения бота к Телеграму, дальше создание или обновление (редактирование) шагов бота и шагов пользователя и через админ-панель бота в браузере.

Админ-панель доступна по адресу:

`<https://><название сайта>/<путь до главной папки бота>/login.php`

или

`<https://><название сайта>/<путь до главной папки бота>/adminpanel.php`

Например: если главная папка бота на сайте `"/bot"` и эта папка размещается на сайте компании `"https://company.ru/"`, то адрес то админ-панели будет следующий:

`https://company.ru/bot/login.php`

или

`https://company.ru/bot/adminpanel.php`