

Telegram Bot

Cloud Functions

Ссылка для вызова:

<https://functions.yandexcloud.net/d4erdi2u7dd75398752e>

```
import json
import requests
import telebot
import os
```

```
API_TOKEN = os.environ.get('TELEGRAM_BOT_TOKEN')
if not API_TOKEN:
    raise ValueError("Telegram bot token not provided in the environment variables.")
```

```
bot = telebot.TeleBot(API_TOKEN)
```

```
LIBRETRANSLATE_URL = "https://translate.flossboxin.org.in/translate"
```

```
TARGET_LANGUAGE = "en"
```

```
def translate_text(text, target_language):
    """Переводит текст на указанный язык с помощью LibreTranslate."""
    try:
        response = requests.post(
            LIBRETRANSLATE_URL,
            json={
                "q": text,
                "source": "auto", # Автоматическое определение языка
                "target": target_language,
                "format": "text" # Указываем, что формат текста - обычный текст
            }
        )
        response.raise_for_status() # Поднимает исключение для HTTP ошибок
        return response.json()["translatedText"]
    except requests.exceptions.RequestException as e:
        return f"Ошибка перевода: {e}"
```

```
@bot.message_handler(commands=['start', 'help'])
```

```
def handle_start_help(message):
    """Обрабатывает команды /start и /help."""
    if message.text == "/start":
        bot.reply_to(message, "Привет! Я бот-переводчик. Просто отправь мне текст, и я переведу его на английский.")
    elif message.text == "/help":
        bot.reply_to(message, "Я перевожу любой текст на английский язык. Используйте команду /start для начала работы.")
```

```
@bot.message_handler(func=lambda message: True)
```

```
def translate_message(message):
    """Переводит текст, отправленный пользователем."""
    translated_text = translate_text(message.text, TARGET_LANGUAGE)
```

```
bot.reply_to(message, translated_text)
```

```
def handler(event, context):
    """Обработчик Cloud Functions."""
    try:
        json_string = event['body']
        update = telebot.types.Update.de_json(json_string)
        bot.process_new_updates([update])
        return {
            'statusCode': 200,
            'body': 'OK',
        }
    except Exception as e:
        return {
            'statusCode': 500,
            'body': f"Error processing request: {e}",
        }
```

API-шлюз

Имя tarassiky

Идентификатор d5d1c8h3l0skl7vd75nn

Статус Active

Дата создания 20.03.2025, в 12:38

Служебный домен <https://d5d1c8h3l0skl7vd75nn.4b4k4pg5.apigw.yandexcloud.net>

Служебный домен для WebSocket <wss://d5d1c8h3l0skl7vd75nn.4b4k4pg5.apigw.yandexcloud.net>

Таймаут обработки запроса 5 минут

Спецификация

```
openapi: 3.0.0
info:
  title: Sample API
  version: 1.0.0
servers:
  - url: https://d5d1c8h3l0skl7vd75nn.4b4k4pg5.apigw.yandexcloud.net
paths:
  /:
    get:
      x-yc-apigateway-integration:
        type: dummy
        content:
          '*': Hello, world!
        http_code: 200
        http_headers:
          Content-Type: text/plain
    post:
      x-yc-apigateway-integration:
        type: cloud_functions
        function_id: d4erdi2u7dd75398752e
        operationId: tgbot
```

Hook

@tarassiky_bot