

ElevenLabs AI

- Great. I used \$5 USD to test the service.
- Voice cloning is great.
- Audio samples below:
- English
- Japanese
- Spanish

```
import os
import requests
from dotenv import load_dotenv
import argparse
import re

load_dotenv()

api_key = os.environ.get("ELEVENLABS_API_KEY")
if not api_key:
    raise ValueError("ELEVENLABS_API_KEY environment variable not set.")

parser = argparse.ArgumentParser()
parser.add_argument("--file", type=str, help="Markdown file to convert to speech", required=False)
parser.add_argument("--text", type=str, help="Text to convert to speech", required=False)
parser.add_argument("--output", type=str, help="Output file name", required=True)
parser.add_argument("--voice_id", type=str, default="21m00Tcm4TlvDq8iK2G8", help="Voice ID to use")

args = parser.parse_args()

if args.file:
    try:
        with open(args.file, 'r') as f:
            content = f.read()
            # Remove front matter
            content = re.sub(r'---.*?---', '', content, flags=re.DOTALL)
            text = content.strip()
    except FileNotFoundError:
        print(f"Error: File not found: {args.file}")


```

```

    exit(1)

except Exception as e:
    print(f"Error reading file: {e}")
    exit(1)

elif args.text:
    text = args.text

else:
    print("Error: Either --file or --text must be specified.")
    exit(1)

url = f"https://api.elevenlabs.io/v1/text-to-speech/{args.voice_id}"

headers = {
    "Accept": "audio/mpeg",
    "Content-Type": "application/json",
    "xi-api-key": api_key
}

data = {
    "text": text,
    "model_id": "eleven_flash_v2_5",
    "voice_settings": {
        "stability": 0.5,
        "similarity_boost": 0.5
    }
}

response = requests.post(url, json=data, headers=headers)

if response.status_code == 200:
    with open(args.output, 'wb') as f:
        f.write(response.content)
    print(f"Audio saved to {args.output}")
else:
    print(f"Error: {response.status_code} - {response.text}")

```