

OnzeLabs IA

- Super. J'ai utilisé 5 USD pour tester le service.
- Le clonage de voix est génial.
- Échantillons audio ci-dessous :
- Anglais
- Japonais
- Espagnol

```
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("La variable d'environnement ELEVENLABS_API_KEY n'est pas définie.")

parser = argparse.ArgumentParser()
parser.add_argument("--file", type=str, help="Fichier Markdown à convertir en parole", required=False)
parser.add_argument("--text", type=str, help="Texte à convertir en parole", required=False)
parser.add_argument("--output", type=str, help="Nom du fichier de sortie", required=True)
parser.add_argument("--voice_id", type=str, default="21m00Tcm4TlvDq8iK2G8", help="ID de la voix à utiliser")

args = parser.parse_args()

if args.file:
    try:
        with open(args.file, 'r') as f:
            content = f.read()
            # Supprimer les informations frontales
            content = re.sub(r'---.*?---', '', content, flags=re.DOTALL)
            text = content.strip()
    except FileNotFoundError:
        print(f"Erreur : Fichier non trouvé : {args.file}")
```

```

    exit(1)

except Exception as e:
    print(f"Erreur lors de la lecture du fichier : {e}")
    exit(1)

elif args.text:
    text = args.text

else:
    print("Erreur : Il faut spécifier --file ou --text.")
    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 enregistré dans {args.output}")
else:
    print(f"Erreur : {response.status_code} - {response.text}")

```