

めげずにやってみよう！

video-use 導入ガイド

— Claude Code で動画編集 —

初めてでも大丈夫。わからなくなったら Claude Code に画面のスクショを貼って聞けばOK。落ち着いて一歩ずつ進めば、必ずできるようになります！

🕒 所要時間 30~60分

🍏 Mac ユーザー向け

🎓 初心者歓迎

👉 コピペで進む

📖 はじめに読んでほしい

わからなくても、焦らなくていいよ！

初めてターミナルを触る人にとって、このガイドは正直「むずかしそう」に見えるかもしれません。でも大丈夫。途中でつまづいたら Claude Code にそのまま聞けばいいんです。

エラーが出たら画面をスクショして Claude Code に貼るだけで、「ここをこうしてね」と日本語で教えてくれます。一人で抱え込まず、落ち着いて一歩ずつ進めていきましょう。

💡 **小さな一歩の積み重ねが大きな自由をつくる。**めげずにやってみよう！

この資料について

video-use は単体の CLI ツールというより、**Claude Code のスキル**として動作します。GitHub から取得して Python 環境を整え、**~/claude/skills** に登録。素材フォルダを Claude Code で開いて、日本語で編集指示を出せば動きます。

必要なもの



Mac



Homebrew



Git



Python 3.10+



Claude アカ
ント



API クレジット
& キー



Claude のサブスクと API は別枠です

video-use は Anthropic API を使うため、Console 側でクレジット購入と API キー発行が必須です。

📖 はじめての方へ | ミニ用語集

このガイドに出てくる用語を、初心者向けにざっくり一言で解説します。意味がわかれば作業中にビビりません。

ターミナル

Mac に最初から入っている黒い画面のアプリ。文字でPCに命令できます。

起動: Launchpad で「ターミナル」と検索

Homebrew (brew)

Mac 用のソフト管理ツール。App Store のターミナル版みたいなもの。

Python / pip

プログラミング言語 Python と、その追加パーツを入れる道具 pip。

仮想環境 (venv)

プロジェクト専用の隔離部屋。他のソフトと喧嘩しないように分けておく仕組み。

API / API キー

外部サービスを呼び出す窓口。キーは「会員証」で、利用は従量課金です。

シンボリックリンク

フォルダやファイルのショートカット。本体はそのままに「別名の入り口」を作る仕組み。



コマンドは「コピペ」でOK

一文字ずつ打つ必要はありません。灰色のコードブロックをコピーして、ターミナルに貼り付け→Enter で動きます。

10 ステップで **セットアップ**

1

事前確認

Python と Git が入っているか確認

ターミナルで次のコマンドを実行し、バージョンを確認します。

```
python3 --version  
git --version
```

BASH

⚠ Python 3.10 以上が必須

3.9 系では後段でエラーになります。古いバージョンの場合は次のステップで 3.10 を入れましょう。

2

リポジトリを取得

GitHub から clone

作業用フォルダを作り、公式リポジトリ **browser-use/video-use** を clone します。

```
mkdir -p ~/code  
cd ~/code  
git clone https://github.com/browser-use/video-use.git  
cd video-use
```

BASH

3

Python 3.10 と仮想環境

Homebrew で Python を入れて venv を作成

Homebrew で Python 3.10 を入れ、`video-use` 専用の仮想環境を作ります。

```
brew install python@3.10
cd ~/code/video-use
python3.10 -m venv .venv
source .venv/bin/activate
python --version
```

BASH



成功の目印

最後のコマンドで Python 3.10.x と表示されれば OK です。

4

video-use をインストール

pip install でセットアップ

リポジトリには `pyproject.toml` があるため、`pip install .` でインストールします。

```
cd ~/code/video-use
pip install .
```

BASH



成功の目印

Successfully installed ... video-use-0.1.0 のような表示が出ます。

5

API クレジットを用意

Anthropic Console で購入とキー発行

Anthropic Console では API 利用のためにクレジット購入が必要です。Claude のチャット契約とは **別課金** なので注意。

1. console.anthropic.com にログイン
2. クレジット購入画面で **まず USD 5 くらい** を選んで購入（お試しならこれで十分）
3. 「API Keys」画面で新しいキーを作成
4. 表示された `sk-ant-...` のキーをコピー



まずは \$5 で様子見でOK

いきなり大きい金額を入れる必要はありません。使ってみて足りなくなったら追加購入できます。



API キーは機密情報

SNS やチャットに貼らず、絶対に外部へ見せないでください。

6

API キーをセット

環境変数として export

仮想環境が有効なターミナルで、API キーを環境変数として設定します。

```
export ANTHROPIC_API_KEY="sk-ant-ここにAPIキーを貼る"
```

BASH



有効範囲は現在のターミナルのみ

永続化したい場合は `~/.zshrc` などに追記します。

7

Claude Code にスキル登録

シンボリックリンクを作成

公式 README 推奨の方法で、`video-use` をスキルディレクトリへリンクします。

```
cd ~/code/video-use  
mkdir -p ~/.claude/skills  
ln -s "$(pwd)" ~/.claude/skills/video-use
```

BASH

これで Claude Code から `video-use` スキルを参照できます。

8

素材フォルダを用意

編集したい動画を入れる

好きな場所にフォルダを作り、編集したい mp4 ファイルなどを入れます。

```
mkdir -p ~/Desktop/video-use-test
```

BASH

このフォルダに素材動画をドラッグ&ドロップで入れるだけで OK です。

9

Claude Code で指示を出す

日本語で編集をお願いします

Claude Code で素材フォルダを開いたら、日本語でこう伝えるだけ。

```
video-use スキルを使って、このフォルダ内の動画を  
1本の完成動画に編集してください。  
不要な間や言い淀みはカットして、テンポよく。  
まず編集方針を日本語で説明してから作業を始めてください。  
完成動画は edit/final.mp4 に出力してください。
```

PROMPT



出力先は自動で作られます

素材フォルダ内に edit/final.mp4 が生成されます。事前に output フォルダを作る必要はありません。

ナレーション生成など、音声合成を使う場合に必要

編集指示を出したあと、`video-use` は **ナレーション生成や音声合成のために ElevenLabs の API を使う** ことがあります。その場合、Claude Code から ElevenLabs API キーの設定を求められます。



ElevenLabs とは？

超リアルな音声を生成できる AI サービス。動画に自動でナレーションを乗せる用途で使われます。Anthropic と同じく

別課金の API

です。

手順

1. elevenlabs.io にアクセスしてアカウント作成
2. 右上プロフィール → 「API Keys」 からキーを発行
3. 無料枠もありますが、継続利用は有料プラン or クレジット購入が必要
4. 発行されたキーをコピーし、ターミナルで環境変数に設定

```
export ELEVENLABS_API_KEY="ここにElevenLabsのAPIキーを貼る"
```

BASH



両方セットしておくとお楽

Anthropic と ElevenLabs のキーをまとめて `~/.zshrc` に書いておけば、ターミナルを開き直すたびに `export` する手間が省けます。

```
export ANTHROPIC_API_KEY="sk-ant-..."  
export ELEVENLABS_API_KEY="..."
```

~/ .ZSHRC に追記する例

⚠️ 音声を使わない編集だけなら不要

カット編集や繋ぎ合わせのみの場合は ElevenLabs の設定は求められません。必要になったタイミングで設定すればOKです。

よくある つまづき

Q Python 3.9 しか入っていない

`requires a different Python: 3.9.6 not in '>=3.10'` のエラーが出ます。手順 3 に戻り、Homebrew で `python@3.10` を入れて仮想環境を作り直してください。

Q requirements.txt が見つからない

このリポジトリは `pyproject.toml` ベースです。 `pip install -r requirements.txt` ではなく、 `pip install .` を使います。

Q video-use コマンドが見つからない

本質的な問題ではありません。 `video-use` は単体 CLI ではなく、**Claude Code のスキルとして呼び出す**のが公式の使い方です。手順 7 のシンボリックリンクが正しく張られていれば OK です。

Q Claude の有料プランを契約しているのに API が使えない

Claude の通常利用と Anthropic API は**完全に別課金**です。Console でクレジットを購入しないと API キーの発行・利用ができません。

最短チェックリスト

- 1 GitHub から `video-use` を clone
- 2 Python 3.10 の仮想環境を作成
- 3 `pip install .` でインストール
- 4 Anthropic Console でクレジット購入 + API キー発行
- 5 `export ANTHROPIC_API_KEY="..."` を実行
- 6 `ln -s "$(pwd)" ~/.claude/skills/video-use` でスキル登録
- 7 素材動画入りフォルダを Claude Code で開く
- 8 「video-use スキルで編集して」と日本語で指示
- 9 音声生成を使う場合は ElevenLabs API キーも `export`

大丈夫、一歩ずつで必ずできる！

エラーが出ても、詰まっても、それは **失敗ではなく「進んでいる証拠」** です。わからなくなったら **Claude Code に画面のスクショとエラー文を貼り付けて質問** してみてください。日本語でやさしく教えてくれます。

今日の小さな一歩が、明日の自由な動画制作につながる。

焦らず、楽しみながら、やってみよう！

video-use 導入ガイド / Mac・初心者向け・written by ちゃびのすけ