**Setting up YouTube API Credentials**

To enable video uploads to YouTube, follow these steps:

1. **Create a YouTube Channel:**

* Go to [YouTube](https://www.youtube.com/) and create a channel.

1. **Set Up Google Cloud Project:**

* Go to [Google Cloud Console](https://console.cloud.google.com/apis/credentials).
* Select or create a new project.

1. **Configure OAuth Consent Screen:**

* Go to the "OAuth consent screen" section and select user type as "external".
* Provide app information including app name and user support email.
* In the app domain, use https://www.youtube.com/ for both application homepage and privacy policy.
* Add youtube.com to the authorized domains.
* Add an email address as developer contact information.

1. **Set Up OAuth Scopes:**

* Go to the "Scopes" section and add the following scopes:

1. <https://www.googleapis.com/auth/youtube>
2. <https://www.googleapis.com/auth/youtube.force-ssl>
3. <https://www.googleapis.com/auth/youtube.upload>
4. <https://www.googleapis.com/auth/youtube.readonly>

* In the "How the scopes will be used" field, specify that these scopes will be used to enable video uploads via scripts.
* Provide a demo video link (e.g., <https://www.youtube.com/watch?v=Pr0tC-6hh44).>

1. **Set Up OAuth 2.0 Client ID**

* Go to the "Credentials" page and create new credentials with "Application Type" as "Desktop".

1. **Prepare for Verification:**

* Go to the OAuth consent screen
* Go to the "Summary" section and select "Prepare for verification".
* Save and continue through all required sections.
* Submit for Verification:

1. **Download OAuth 2.0 Client ID:**

* Download the OAuth 2.0 Client ID you created.

1. **Place Credentials File:**

* Keep the downloaded file (OAuth 2.0 Client ID) in the folder containing all .py files.

1. **Update config.py:**

* In config.py, specify the filename of the downloaded OAuth 2.0 Client ID.

1. **Running for 1st Time**

* While running for the first time you will need to authenticate the google account.
* After signing in the account it will show it is harmful but go to advance option and continue.

**Configuration Setup**

This project uses a configuration file (config.py) for customization. Below are the key settings you can adjust:

**GCP Key**

* Update gcp\_key with your Google Cloud Platform service account credentials.

**OpenAI Key**

* Set openai\_api\_key to your OpenAI API key.

**Google Cloud Platform**

* **gcp\_bucket**: The name of your Google Cloud Storage bucket.
* **gcp\_path**: The directory path where media will be stored.
* **video\_filename**: The name of the video file.
* **video\_with\_music\_filename**: The name of the video file with added music.

**OpenAI Configuration**

* **openai\_engine:** The OpenAI engine to use.
* **openai\_temperature:** The temperature setting for text generation.
* **openai\_max\_tokens:** The maximum number of tokens for generated text.
* **title\_prompt:** The prompt used for generating title.
* **Description\_prompt:** The prompt for generating description.

**Youtube Key Configuration**

* **Yt\_key\_path:** Path of the youtube client secret file.

**Music Configuration**

* **Fps:**  Set the fps of the video
* **Music Folder Path:** The path of the folder containing audio files. One of them will be selected at random and trimmed according to the length of video and added to the video