

# ⚡ Automatise ce qui te fait perdre du temps — grâce à l'IA

## LIBÉREZ VOTRE CRÉATIVITÉ AVEC L'IA

Des outils créés grâce à l'IA pour transformer vos idées en projets — même sans coder

---

### 💡 Pourquoi ce guide ?

Ce guide est conçu pour tous les créateurs — artistes visuels, vidéastes, photographes, formateurs, indépendants — qui veulent concrétiser leurs idées avec l'IA.

Vous n'avez pas besoin d'être développeur. Il vous suffit d'une idée, d'un prompt, et d'une envie d'explorer.

L'objectif ?

Vous montrer comment transformer une simple idée en script fonctionnel, **sans aucune base en code**, simplement en apprenant à *parler à l'IA*.

---

### 🚀 Ce que vous allez débloquent avec ce guide

🎯 Une comparaison claire et objective des principales IA disponibles en Europe (GPT-4o, Gemini, Bing, Claude, Mistral, etc.)

💡 Un **tableau comparatif** clair pour choisir la bonne IA selon :

- la qualité du français,
- la souveraineté des données (RGPD),
- la capacité d'automatisation (API),
- les coûts et la facilité d'accès.

⚙️ Une **stratégie "stack hybride"** clé-en-main :

- Mix d'outils gratuits, souverains, et puissants,
- Exemples concrets d'automatisation via API ou outils no-code,
- Astuces pour les non-développeurs (avec ou sans code).

📁 Et surtout : un accès direct aux ressources de **Mon Studio IA**

→ scripts prêts à l'emploi, vidéos tutorielles, accompagnement pédagogique pour vous aider à tout mettre en œuvre.

### 💡 Ce que vous allez pouvoir faire

Grâce à ce pack, vous pouvez :

- Générer automatiquement des publications Instagram avec image, légende et hashtags
- Publier ces posts sans toucher à l'appli Instagram
- Programmer une publication quotidienne depuis un dossier de photos
- Créer automatiquement des YouTube Shorts à partir de vidéos longues
- Récupérer la liste des vidéos de votre chaîne youtube dans un fichier

Le tout **sans abonnement**, **sans interface compliquée**, et **sans coder**.

### 🔗 Scripts inclus dans le fichier `.zip`

Tous les fichiers sont prêts à l'emploi. Il suffit d'**adapter quelques chemins sur votre machine**, et vous êtes prêt à tester.

📁 **Script 1 :** `post_instagram_from_image.py`

**Fonction :**

Analyse une image avec GPT-4o et génère un post complet :

- 📄 Titre
- 💡 Légende poétique
- 🚩 Hashtags pertinents → Sans publier automatiquement.

Exemple d'utilisation :

```
python post_instagram_from_image.py chemin/vers/image.jpg
```

👤 VOIR LE TUTO COMPLET EN VIDÉO : <https://youtu.be/jb0CHGdmJkg>

🔊 Script 2 : post\_instagram\_from\_image\_with\_upload.py

Fonction :

Fait la même chose que le premier script, mais **publie automatiquement** sur Instagram via l'API Meta et un tunnel ngrok.

**Pré-requis :** Un fichier `.env` avec vos clés :

```
OPENAI_API_KEY=votre_clé_OpenAI
IG_USER_ID=...
IG_ACCESS_TOKEN=...
```

Exemple de commande :

```
python post_instagram_from_image_with_upload.py chemin/vers/image.jpg --upload
```

🔥 Le script démarre automatiquement ngrok pour rendre votre image accessible.

👤 VOIR LE TUTO COMPLET EN VIDÉO : <https://youtu.be/eQ27yNMSf7U>

📄 Script 3 : publier\_prochaine\_photo.py

Fonction :

- Va chercher la **première photo JPEG non publiée** dans un dossier
- Utilise le script précédent pour la publier
- Puis la **déplace** dans un sous-dossier `publiees`

À personnaliser dans le script :

```
DOSSIER_PHOTOS = pathlib.Path(r"VOTRE_CHEMIN_ABSOLU")
```

Structure recommandée :

```
Portfolio/
├─ IMG_001.jpg
├─ IMG_002.jpg
└─ publiees/
```

Commande à exécuter

```
python publier_prochaine_photo.py
```

👤 VOIR LE TUTO COMPLET EN VIDÉO : <https://youtu.be/eQ27yNMSf7U>

🎬 Script 4 : generate\_short.py

Fonction :

Génère automatiquement un **Short YouTube vertical 1080x1920** à partir :

- d'une vidéo longue
- d'un fichier `.yaml` contenant les textes, sous-titres, intro/outro

Exemple de commande :

```
python generate_short.py chemin/vers/video.mp4 chemin/vers/config.yaml
```

Le script crée un fichier `.mp4` prêt à publier sur YouTube Shorts.

🎥 VOIR LE TUTO COMPLET EN VIDÉO : <https://youtu.be/J2mgblhuCaM>

---

### 🧠 Script 5 : `liste_videos_youtube.py`

#### Fonction :

Ce script vous permet de **récupérer automatiquement la liste de vos vidéos YouTube (titres + liens)** à partir de votre propre chaîne. C'est un excellent point de départ pour **commencer à automatiser intelligemment la gestion de votre chaîne YouTube**, sans toucher une ligne de code.

Deux versions sont incluses dans le pack :

- 🖥 Le **script de base en ligne de commande**
- 🖱 Une **interface graphique (GUI)** très simple à utiliser, avec un bouton pour tout lancer

💡 Dans la vidéo associée, vous découvrirez également **comment transformer ce script en une application exécutable ( `.exe` ou `.app` ) prête à être partagée ou utilisée sans terminal**, à partir de vos propres paramètres. Cette partie n'est pas incluse dans le pack gratuit mais est expliquée pas à pas.

🎯 Ce script ouvre la voie à une série de vidéos "Mon Studio IA" dédiée à l'**automatisation YouTube pour créateurs visuels**, avec à venir :

- la publication automatique de vidéos
- la traduction multilingue de titres, descriptions et sous-titres
- la modification à la volée des métadonnées de vidéos
- et bien plus...

🎥 VOIR LE TUTO COMPLET EN VIDÉO → <https://youtu.be/xmplscCej6U>

---

### 🧠 Script 6 : `generate_carrousel.py`

#### Fonction :

Ce script vous permet de **créer un carrousel Instagram complet à partir d'un simple thème** (ex. : « oser se lancer », « profiter de l'instant présent », etc.), sans utiliser Canva et **sans coder**.

Il génère automatiquement :

- 🧠 le **contenu textuel des 6 slides** : accroche, intro, 3 citations inspirantes, conclusion
- 🎨 les **illustrations IA personnalisées**, avec ou sans template graphique
- ✍ l'**incrustation du texte** (avec gestion des emojis et de la lisibilité grâce à un fond flouté)
- 📌 une **légende Instagram avec hashtags** optimisés pour l'engagement

💡 Vous pouvez travailler de manière modulaire : créer uniquement le texte ( `--content` ), générer les images brutes ( `--image-only` ), incruster le texte ensuite ( `--incruster` ) et même ne régénérer qu'un seul slide ( `--index` ).

Parfait pour **ajuster le résultat final sans tout refaire**.

🎯 Ce script s'adresse aux **créateurs visuels** qui veulent publier des contenus impactants sur Instagram sans passer des heures sur des outils de design. Il peut être décliné en séries, storytelling, mini-formations ou capsules inspirationnelles.

🎥 VOIR LE TUTO COMPLET EN VIDÉO → <https://youtu.be/DGt48n6wpRc>

---

### 🌐 Script 7 : `traducteur_youtube.py`

#### Fonction :

Ce script vous permet de **traduire automatiquement une ou plusieurs vidéos YouTube dans 5 langues majeures** (anglais, espagnol, portugais brésilien, allemand, hindi) — **sans dépendre des traductions automatiques de YouTube**, souvent approximatives.

Il récupère pour chaque vidéo :

- 📄 le **titre et la description** dans leur version actuelle,

- 📄 les **sous-titres en français** (version manuelle ou automatique),
- 🧠 puis génère des **traductions contextualisées** grâce à l'IA, en respectant le ton et la culture de chaque langue.

Ensuite, le script :

- 📁 sauvegarde **chaque version traduite dans un sous-dossier dédié**,
- 🌐 **met à jour les localisations** (titre et description) directement sur YouTube,
- 🗣️ et **upload les sous-titres multilingues** via l'API, bloc par bloc, pour garder la synchronisation parfaite.

💡 Vous pouvez choisir ce que vous voulez traduire ( `--title` , `--description` , `--subtitles` , ou `--all` ) et même activer uniquement l'upload si les fichiers existent déjà ( `--upload-only` ).

🎯 Ce script est pensé pour les **créateurs qui veulent toucher une audience internationale** sans sacrifier la qualité éditoriale, en gardant le contrôle total sur **chaque mot publié**.

📺 VOIR LE TUTO COMPLET EN VIDÉO → <https://youtu.be/PYvXz9kxeZM>

🧩 **Script 8 :** `generate_story.py` et `generate_story_ffmpeg.py`

Fonction :

Ces deux scripts transforment automatiquement vos images en **story Instagram verticale 1080x1920**, avec :

- 🧠 génération automatique du **texte** (titre + citation ou teaser)
- 🖋️ **incrustation typographique** avec contour, position, style
- 🎵 **ajout automatique d'une musique** en fond
- 🎞️ **animation de type Ken Burns** pour donner vie aux photos

Deux versions sont disponibles :

- `generate_story.py` → **version autonome** (utilise MoviePy, plus simple à modifier)
- `generate_story_ffmpeg.py` → **version optimisée FFmpeg** (export plus rapide et fluide, idéale pour production)

💡 Les deux scripts utilisent un fichier de style `.yaml` pour ajuster :

- la **police** ( `font` )
- la **taille** ( `font_size` )
- les **couleurs** ( `color` , `stroke_color` )
- les **animations** ( `fade_duration` , `ken_burns` , `zoom_speed` , etc.)

Exemple de commande (version MoviePy) :

```
python generate_story.py --images images/ --mode teaser --style style.yaml
```

Exemple de commande (version FFmpeg) :

```
python generate_story_ffmpeg.py --images images/ --musique musique/ --style style_teaser_ffmpeg.yaml
```

📺 VOIR LE TUTO COMPLET EN VIDÉO → <https://youtu.be/mELio3h69G0>

🧠 **Script 9 :** `transcrire_video_youtube.py`

Fonction :

Ce script te permet de **transcrire automatiquement une vidéo YouTube (ou tout fichier vidéo local)** en texte brut, sous-titres `.srt` ou article de blog — selon les options choisies.

Il s'appuie sur **Whisper (modèle large-v3 optimisé)**, tournant **en local**, grâce à la librairie **faster-whisper** (basée sur CTranslate2).

Aucune donnée n'est envoyée à un service cloud : tu restes maître de ton contenu.

Trois sorties possibles :

- `.txt` : la transcription brute, parfaite pour retravailler un script, extraire des idées ou générer un article.
- `.srt` : les sous-titres formatés automatiquement, à uploader dans YouTube Studio.
- `_corrige.srt` : la version **corrigée avec GPT-4o-mini**, ponctuation et orthographe propres (option payante).
- `_article.txt` : un article de blog rédigé à partir de la transcription (option payante).

Commandes typiques :

```
python transcrire_video_youtube.py ma_video.mp4 python transcrire_video_youtube.py ma_video.mp4 --srt python transcrire_video_youtube.py ma_video.mp4 --srt --correction python transcrire_video_youtube.py ma_video.mp4 --article python transcrire_video_youtube.py ma_video.mp4 --srt --correction --article
```

💡 **Le script détecte automatiquement ta configuration :**

- Sur PC avec carte NVIDIA, il passe en mode CUDA (GPU).

- Sur Mac, il tourne en mode CPU (compatible M1/M2/M3).
- Tu peux relancer plusieurs fois sans tout refaire : chaque fichier est généré une seule fois.

🔗 Ce script est idéal pour :

- Créer des **sous-titres professionnels** (corrigés) pour YouTube.
- Réutiliser une vidéo pour **un article de blog, une newsletter, ou une fiche pédagogique**.
- Accélérer ton **workflow de publication multiformat**, sans dépendre d’outils commerciaux.

📺 VOIR LE TUTO COMPLET EN VIDÉO → [https://youtu.be/xpPNe\[\]\(\)ZeMTTY](https://youtu.be/xpPNe[]()ZeMTTY)

🔧 **Script 10 :** `modifier_description_video_youtube.py`

Fonction :

Ce script te permet de **modifier en masse les descriptions de plusieurs vidéos YouTube**, en ajoutant un préfixe, un suffixe, ou en remplaçant des éléments de texte (y compris avec des expressions régulières).

Tu peux cibler la **langue par défaut** ou une **langue de localisation spécifique** ( `--language en` , `--language es` , etc.).  
Idéal pour ajouter un lien, corriger une erreur ou uniformiser la description de toutes tes vidéos en une seule commande.

Options disponibles :

- `--prefix` "Texte à ajouter au début"
- `--suffix` "Texte à ajouter à la fin"
- `--replacements` "ancien=nouveau" `"/regex/=remplacement"`
- `--language en` (par défaut `fr`)

💡 Le script détecte automatiquement les langues disponibles pour chaque vidéo et ajuste la bonne zone ( `snippet` ou `localizations` ) selon le cas.

Fichier CSV attendu :

```
titre,url
Ma première vidéo,https://youtu.be/abc123xyz
Une autre vidéo,https://www.youtube.com/watch?v=def456uvw
```

Exemple de commande :

```
python modifier_description_video_youtube.py \
--input liste_video.csv \
--prefix "📺 Télécharge ton bonus IA : https://monstudioia.fr\n\n" \
--replacements '/\..fr(?:!\/en)/=\.fr/en/' \
--language en
```

📺 VOIR LE TUTO COMPLET EN VIDÉO → [https://youtu.be/9KI\\_YZMfT20](https://youtu.be/9KI_YZMfT20)

🔒 Où configurer vos paramètres (chemins, clés API...)

Voici les points à adapter manuellement :

Script	Élément à modifier	Exemple
<code>post_instagram_from_image.py</code>	Chemin vers l’image	<code>chemin/vers/image.jpg</code>
<code>post_instagram_from_image_with_upload.py</code>	Chemin vers l’image + fichier <code>.env</code>	<code>OPENAI_API_KEY=...</code> dans <code>.env</code>
<code>publier_prochaine_photo.py</code>	Chemin vers le dossier de photos	<code>F:/Portfolio/Artdecoandco/1024/</code>
<code>generate_short.py</code>	Chemin vidéo + fichier YAML config	<code>video.mp4</code> , <code>config.yaml</code>
<code>liste_videos_youtube.py</code> <code>gui_liste_videos_youtube.py</code>	<code>.env</code> contenant :  <code>YOUTUBE_API_KEY</code> , <code>CHANNEL_ID</code>	Exemple :  <code>YOUTUBE_API_KEY="AIza..."</code> <code>CHANNEL_ID="UCabc123xyz..."</code>
<code>generate_carrousel.py</code>	<code>.env</code> contenant <code>OPENAI_API_KEY</code>	<code>OPENAI_API_KEY=sk-xxx...</code> + chemins qénérés automatiquement dans <code>carrousel/</code>
<code>traducteur_youtube.py</code>	<code>.env</code> contenant <code>OPENAI_API_KEY</code> + <code>client_secret.json</code>	- <code>.env</code> : <code>OPENAI_API_KEY=sk-...</code> - <code>client_secret.json</code> (clé OAuth2 YouTube)

<code>generate_story.py</code> <code>generate_story_ffmpeg.py</code>	Dossier images, <code>.yaml</code> de style, mode de texte IA	<code>--images images/ --style style.yaml --mode teaser</code> <code>- .env : OPENAI_API_KEY=sk-...</code>
---	---	---

## Tous les fichiers sont fournis dans le `.zip`

- Les scripts `.py`
- Un modèle de fichier `.env`
- Un exemple de fichier `.yaml` pour les Shorts

## INFO

Ces scripts sont conçus pour être **modulaires et combinables**.

Tu peux :

- Automatiser la publication chaque jour avec `cron` ou le planificateur Windows
- Personnaliser le prompt GPT pour changer le ton de tes posts
- Modifier la fréquence des publications selon tes besoins

## Pour aller plus loin:

Amuse toi à modifier la partie du code du script où tu précises le prompt à envoyer à OpenAI afin de personnaliser sa réponse

## Comment exécuter votre premier script IA

1. Installez Python et suivez les recommandations de ma vidéo ci-dessous (**n'oubliez pas de bien cocher la case "Add Python to PATH" pendant l'installation**).

<https://youtu.be/dvN1qnyjnpw>

1. **Créez un fichier script**

Ouvrez le Bloc-notes, copiez-collez le script fourni (ou celui du PDF),  
puis enregistrez sous le nom `mon_script.py`.  
(avec bloc-notes ou n'importe quel éditeur de texte, mais installer Visual Code ça peut être sympa aussi)  
Stockez ce script quelque part sur votre ordinateur, notez le chemin pour y accéder

1. **Ouvrir un terminal ou un invite de commande:**

Sur Windows : touche Windows → taper "cmd"  
Sur Mac : Finder → Applications → Terminal  
Sur Linux : Ctrl + Alt + T

2. **Lancer le script**

Dans le terminal, allez dans le bon dossier :

```
cd chemin/vers/votre/fichier
Puis tapez :

python mon_script.py
```

3. **Et si j'ai une erreur ?**

Pas de panique ! Voici les cas les plus fréquents :

Python n'est pas reconnu → relancez le terminal

Module manquant (comme `openai`) ?

```
pip install openai
```

🔗 Commande à intégrer dans votre environnement Python, comme expliqué pendant la vidéo

## 1. 📁 Fichier requirements.txt

Un fichier `requirements.txt` est désormais fourni dans le pack : il liste **toutes** les dépendances Python nécessaires (PyYAML, etc.).

Pour installer en une seule commande :

```
pip install -r requirements.txt
```

## 🔑 Créer un compte OpenAI et obtenir votre clé API

Pour que l'IA puisse générer du texte ou exécuter des requêtes à votre place, il vous faut une **clé API OpenAI**. Voici comment l'obtenir, gratuitement ou avec un budget maîtrisé.

### 1. Créez un compte sur <https://platform.openai.com>

- Cliquez sur "Sign Up"
- Vous pouvez utiliser votre e-mail, Google, ou Microsoft
- Une fois connecté, vous êtes sur le *dashboard OpenAI*

### 2. Allez dans la section API Keys

Depuis le menu en haut à droite (profil), cliquez sur :

| "View API Keys" → puis "Create new secret key"

- Donnez un nom à votre clé (ex. : `formation_IA_script`)
- Copiez la clé générée immédiatement (elle ne s'affichera qu'une seule fois)

Exemple de clé :

```
sk-abcdef1234567890zyx...
```

| ⚠️ *Ne partagez jamais votre clé publiquement.*

## 🔑 Comment utiliser votre clé OpenAI dans le script

Pour que le script fonctionne, vous devez lui donner votre clé OpenAI.

C'est ce qui permet au script d'accéder à l'intelligence artificielle.

La bonne nouvelle ?

**Avec la version actuelle du script, vous n'avez rien besoin d'écrire dans le code.**

Vous devez juste **faire en sorte que la clé soit disponible dans votre environnement.**

Voici comment faire :

### 🔧 Méthode rapide : taper une commande avant de lancer le script

Quand vous ouvrez votre terminal (ou invite de commande), tapez :

- Sur **Windows (PowerShell)** :

```
$Env:OPENAI_API_KEY="votre_clé_ici"
```



- Sur **Mac ou Linux** :

```
export OPENAI_API_KEY="votre_clé_ici"
```

Puis lancez tout de suite le script normalement :

```
python post_instagram_from_image.py chemin/vers/photo.jpg
```

✅ Le script détectera tout seul votre clé grâce au SDK OpenAI.

### 🔧 Méthode tranquille : enregistrer la clé dans un fichier `.env`

Si vous ne voulez pas retaper votre clé à chaque fois :

1. Ouvrez **Bloc-notes** (ou tout éditeur de texte).
2. Écrivez dedans :

```
OPENAI_API_KEY=votre_clé_ici
```

3. Enregistrez ce fichier sous le nom `**.env**` dans **le même dossier que votre script**.

Et c'est tout.

Pas besoin d'ajouter `load_dotenv()` dans votre script : le SDK OpenAI est assez intelligent pour lire automatiquement votre fichier `.env` s'il existe.

### 🎯 Résumé simple

Si vous voulez...	Faites...
Tester vite fait une fois	Tapez votre clé dans le terminal avant de lancer le script.
Ne plus jamais y penser	Créez un fichier <code>.env</code> dans le dossier du script.

### ⚠️ Rappel important :

Votre clé OpenAI est **personnelle**.  
Ne la partagez jamais, ne la publiez pas en ligne, et évitez de l'écrire directement dans votre code.

✅ **C'est bon, vous êtes prêt à faire coder l'IA pour vous !**

### 🧩 Comparatif des IAs

Modèle	Français	Performance	RGPD/Souveraineté	Créativité	Prix	Automatisable	Notes clés
GPT-4o (ChatGPT)	★★★★★	★★★★★	★☆☆☆☆	★★★★★	20 €/mois ou API au token	Oui (API OpenAI)	Polyvalent, créatif, mais hébergé aux USA
Gemini 2.5 (Google)	★★★★☆	★★★★☆	★★★★☆	★★★★☆	22 €/mois Google One Premium	Partiellement (API encore limitée)	Très bon dans l'écosystème Google
Claude 4 (Anthropic)	★★★★☆	★★★★☆	★★★★☆	★★★★☆	15 €/mois (Opus)	Oui (API Bedrock, Vertex AI)	Structuré, fiable, bon pour les textes longs



Microsoft Copilot (Bing)	★★★★☆	★★★★☆	☆☆☆☆☆	★★★★☆	Gratuit / 22 €/mois Pro	Non	Utile pour veille, non scriptable
DeepSeek V2	★★★★☆	★★★★☆	☆☆☆☆☆	★★★★☆	Gratuit ou API low-cost	Oui (Hugging Face / DeepSeek API)	Techniquement fort, mais censure + Chine
Le Chat (Mistral Large)	★★★★★	★★★★☆	★★★★☆	★★★★☆	Gratuit ou 14,99 €/mois Pro	Oui (API Mistral)	Rapide, souverain partiellement (Azure)
Mistral 7B / Mixtral (local)	★★★★☆	★★★★☆	★★★★★	★★★★☆	Gratuit (self-host ou HF)	Oui	Idéal pour auto-hébergement, hautement souverain

## 🧩 Créez votre stack IA hybride : performante, souveraine, sur mesure

Pourquoi choisir **une seule IA**, quand on peut les combiner intelligemment ?

Avec une **stack hybride**, vous utilisez :

Besoin	Solution recommandée
Rédaction en français	GPT-4o ou Mistral Large via Le Chat
Publication automatisée	OpenAI API (texte + Whisper)
Souveraineté des données	BLOOM ou Mistral 7B en local
Gratuité / Navigation web	Bing Chat (GPT-4 gratuit)
Intégration bureau	Gemini 2.5 + Docs/Sheets

### Exemple de workflow automatisé :

- 📷 Vous déposez une photo dans un dossier →
- 🧠 Le script la fait analyser par GPT-4 Vision →
- ✍️ Il génère un titre, une légende, des hashtags →
- 📱 Il programme la publication sur Instagram.

Tout ça, **sans coder**, grâce aux scripts fournis et à notre accompagnement.

## 🙌 Vous n’êtes pas développeur ? Tant mieux.

Chez mon **studio IA**, ma mission, c’est de faire de l’IA **un outil créatif accessible**.

🎯 Vous pouvez tout tester avec :

- 1 clé API,
- 1 interface simple (Python ou no-code),
- Et mes explications pas-à-pas.

🔒 Vous ne serez jamais dépendant d’une seule plateforme.

## 🚀 Et après?

🔥 Et si vous voulez aller plus loin, la chaîne **Mon Studio IA** vous accompagne chaque semaine avec des vidéos concrètes, pédagogiques, et inspirantes.

Abonnez-vous, testez les scripts, posez vos questions — et surtout, créez à votre manière.

👉 [YouTube – Mon Studio IA](#)

👉 Bonus disponibles à vie dans votre boîte mail.