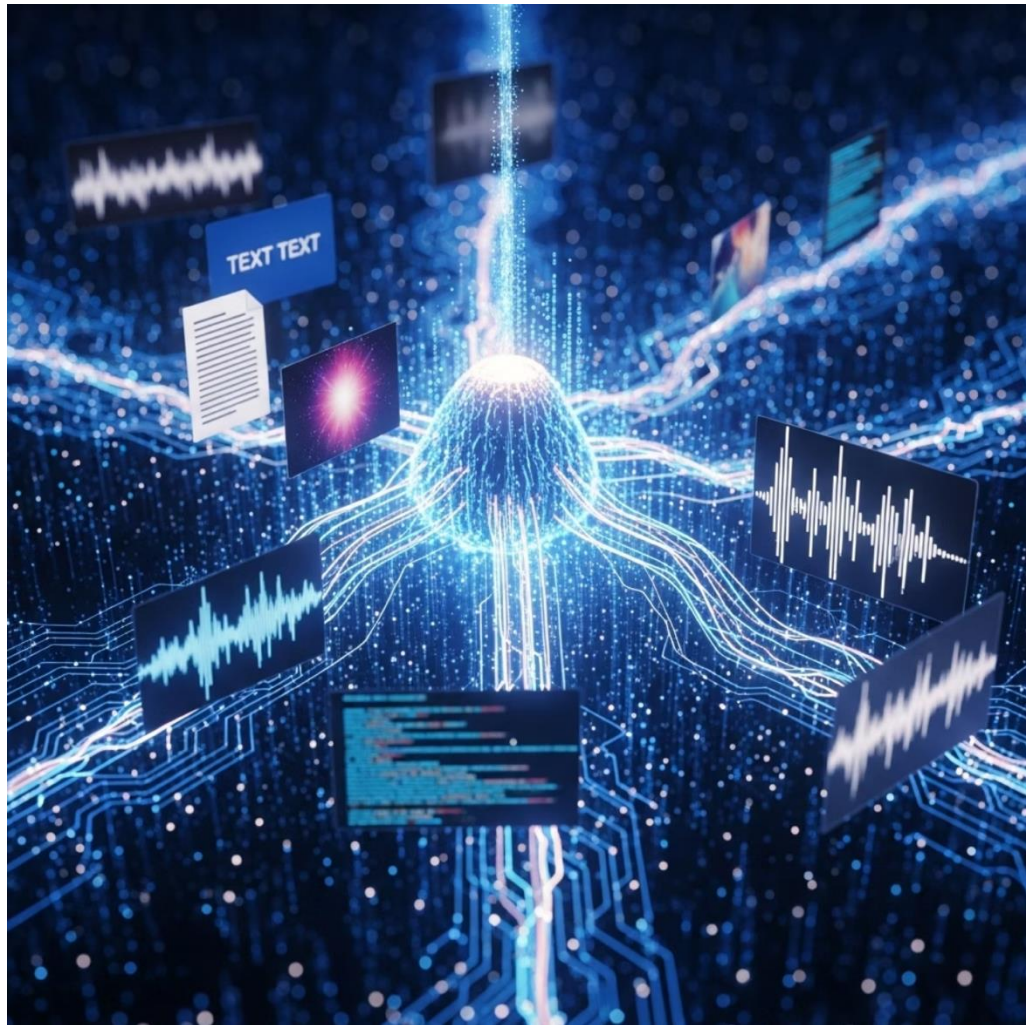


Challenge GenAI Express: Démarrer avec l'IA Générative en 5 Jours

Un Challenge intensif pour les professionnels souhaitant créer rapidement leur premier assistant IA fonctionnel et applicable en entreprise.



Qu'est-ce que l'IA Générative?



L'IA générative est capable de **créer du contenu nouveau** à partir des données sur lesquelles elle a été entraînée:

- **Texte:** ChatGPT (OpenAI)
- **Image:** MidJourney, DALL-E
- **Audio:** ElevenLabs
- **Code:** GitHub Copilot

Elle analyse des motifs dans d'immenses quantités de données pour générer du contenu original et pertinent.

Les 4 Approches de l'IA Générative



Prompt Engineering

L'art de poser les bonnes questions à l'IA pour obtenir des réponses optimales.



RAG

Retrieval Augmented Generation: connecter l'IA à vos données spécifiques pour des réponses contextualisées.



Agents IA

Automatiser des séquences complexes de tâches en donnant de l'autonomie à l'IA.



Fine-tuning

Spécialiser un modèle existant sur vos données spécifiques pour améliorer ses performances.

Message clé: La majorité des cas business démarrent avec Prompt Engineering + RAG

Définir Son Projet IA

1

Cas d'usage

Quelle tâche précise voulez-vous améliorer? Qui en bénéficiera?

Ex: "Un chatbot qui répond aux questions fréquentes de mes clients."

2

Objectif MVP mesurable

✗

Vague: "Je veux un chatbot."

✓

Clair: "Un assistant qui répond à 80% des questions en moins de 2 minutes."

3

Données disponibles

Quelles ressources pouvez-vous utiliser? (FAQ, emails, documents...)

4

Modèle/Outil choisi

Propriétaire (GPT-4) ou Open Source (Mistral, LLaMA)

5

Test utilisateur prévu

Comment validerez-vous l'efficacité de votre solution?

A photograph showing a person's hands writing on a tablet. The tablet screen displays a 'PROJECT PLANNING' sheet with various columns and rows. On the desk next to the tablet are several colorful sticky notes (yellow, green, blue) and a glowing, spherical light object. The scene is dimly lit, with the primary light source being the tablet and the glowing orb.



Règles d'Or pour un Projet IA Réussi

Aller du simple au complexe

Commencez par une seule tâche clé bien définie avant d'élargir le périmètre.

Itérer rapidement

Testez, recueillez du feedback, ajustez. La première version n'est jamais parfaite.

Focus sur la valeur

Un MVP doit prouver la valeur apportée, pas être parfait techniquement.



Exercice pratique: Rédigez votre mini-fiche projet (3 phrases max) et partagez-la dans le chat.

IA Propriétaire vs IA Open Source

Propriétaire (Cloud)

- OpenAI (ChatGPT)
- Anthropic (Claude)
- Google (Gemini)

✅ Avantages: Performance supérieure, simplicité d'utilisation, mise à jour automatique

❌ Limites: Coûts récurrents, dépendance externe, questions de confidentialité

Open Source (Local)

- Mistral (via Ollama)
- LLaMA (via Ollama)
- DeepSeek

✅ Avantages: Contrôle total, confidentialité des données, utilisation gratuite

❌ Limites: Installation et configuration plus complexes, performances parfois inférieures

Nous testerons les deux approches: OpenAI (propriétaire) et Ollama (open source)

Installation de l'Environnement de Travail



Python

Téléchargez et installez depuis python.org

Vérifiez l'installation: `python --version`



Cursor

Éditeur basé sur VS Code optimisé pour l'IA

Téléchargez depuis cursor.sh



Ollama

Gestionnaire de modèles IA open source

Téléchargez depuis ollama.ai

Vérifiez: `ollama --version`



OpenAI API

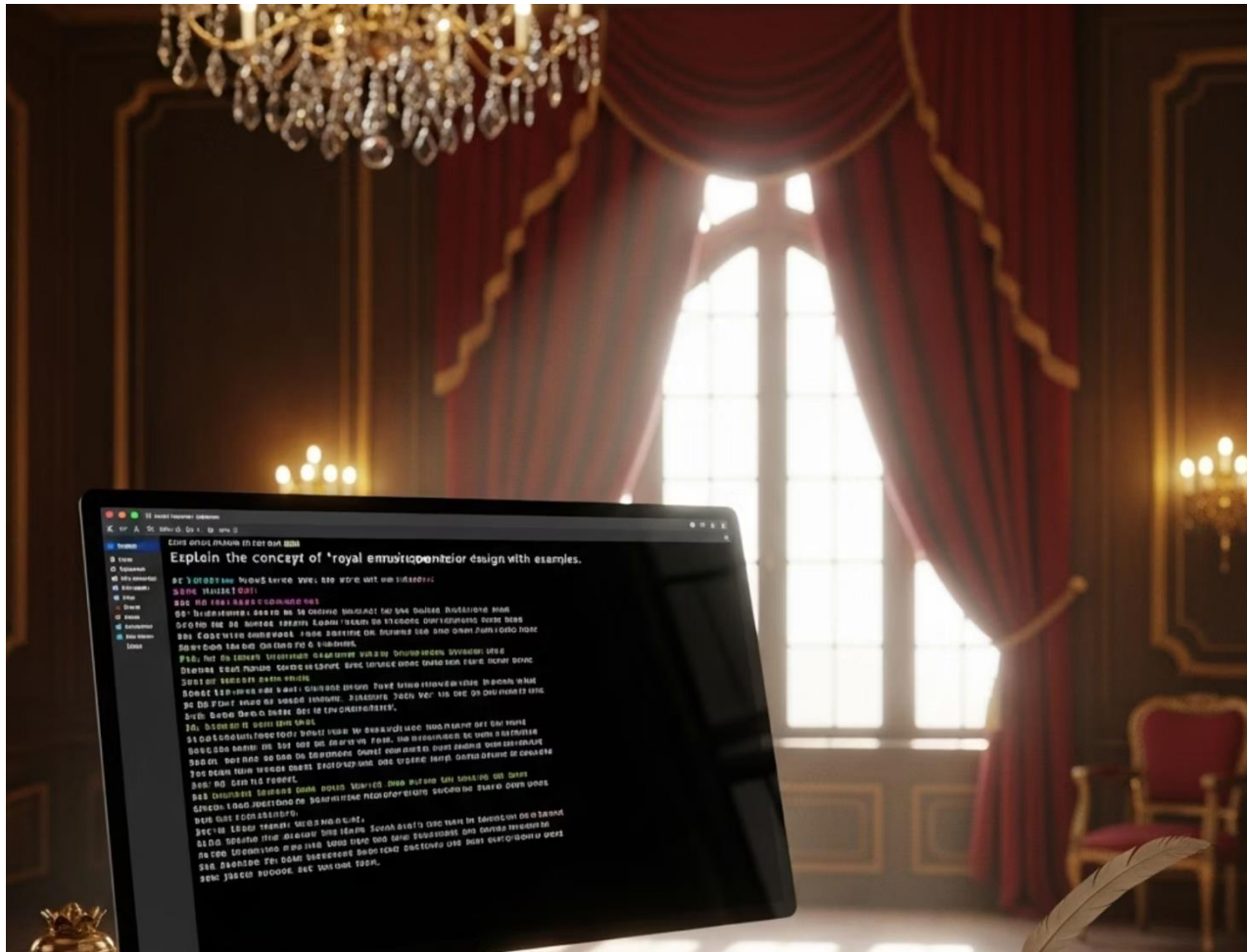
Créez un compte sur platform.openai.com

Générez une clé API dans les paramètres

Tester un Modèle IA Open Source avec Ollama

Via Terminal (CLI)

```
# Télécharger le modèle llama pull llama3# Exécuter le modèle llama run llama3# Saisir
votre prompt"Explique-moi l'IA générative comme si j'avais 10 ans."
```

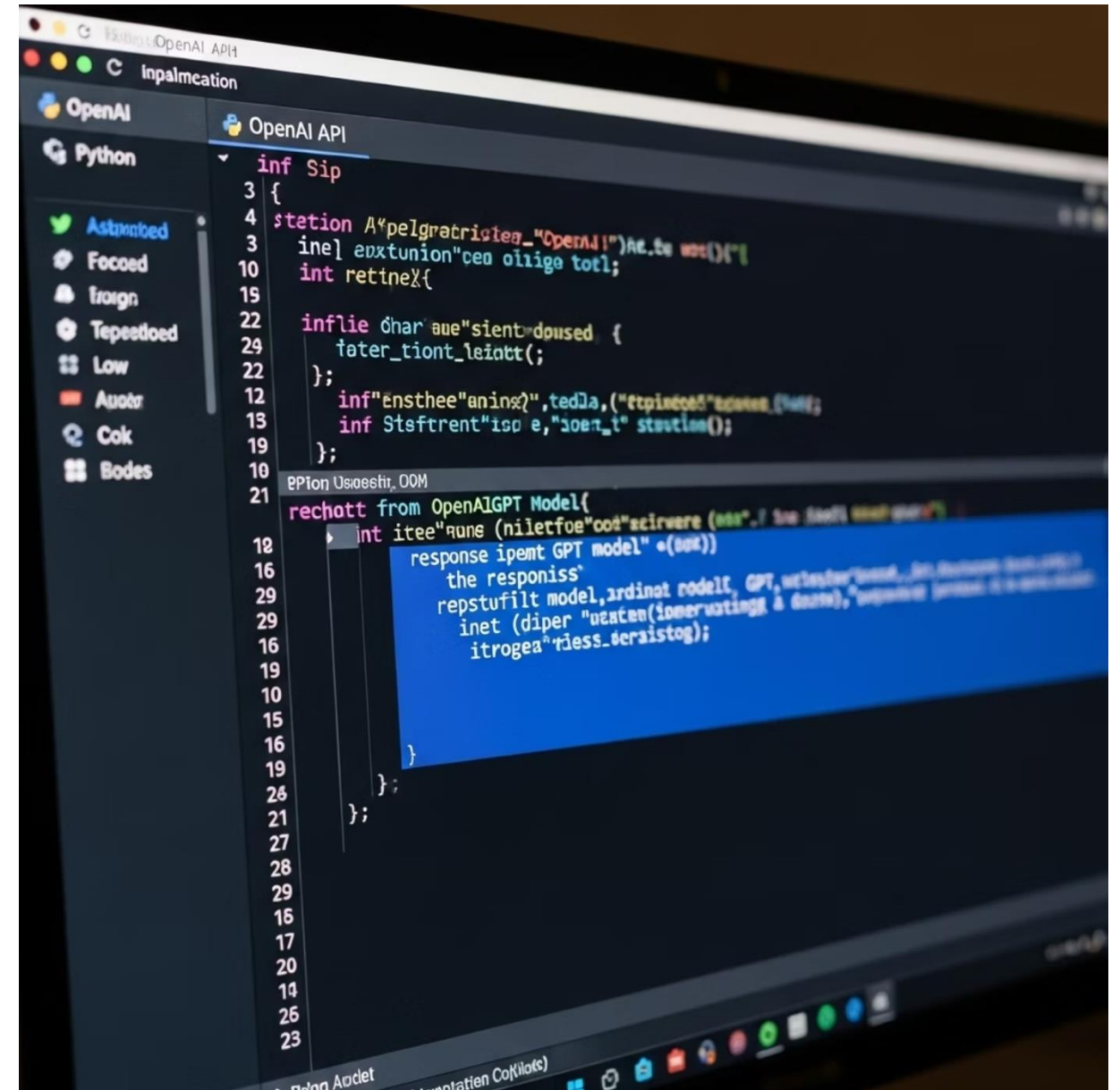


Via Python

```
import ollama
prompt = "Explique-moi l'IA générative en une phrase simple."
response = ollama.chat( model="llama3",
messages=[ {"role": "system", "content": "Tu es un assistant pédagogique."}, {"role": "user", "content":
prompt} ])
print("Réponse IA :", response['message']['content'])
```


Tester un Modèle IA Propriétaire avec OpenAI API

```
from openai import OpenAI# Initialiser le client avec votre clé APIclient = OpenAI(api_key="VOTRE_CLE")# Créer une complétion de chatresp = client.chat.completions.create(model="gpt-4o-mini", messages=[{"role": "system", "content": "Tu es un expert en IA appliquée à l'entreprise."}, {"role": "user", "content": "Donne-moi 3 applications concrètes de l'IA générative en entreprise"}])# Afficher la réponseprint("Réponse IA :", resp.choices[0].message.content)
```



Interaction live: Testons une question proposée par les participants avec Ollama puis OpenAI pour comparer les résultats.

Votre Mission pour Aujourd'hui

À Installer

- Anaconda et Cursor
- Ollama et le modèle llama3
- Créer votre clé API OpenAI

À Tester

1. 1 prompt avec Ollama en Python
2. 1 prompt avec OpenAI en Python

À Partager sur WhatsApp

- Votre mini-fiche projet IA
- Une capture d'écran de votre code + réponse



✓ **Félicitations!** Vous avez maintenant une compréhension claire de l'IA générative et les outils nécessaires pour démarrer votre projet.

Demain, nous approfondirons le prompt engineering.