

IA: évolution, compréhension, application, programmation

Module 1: Introduction à l'Intelligence Artificielle

- Survol et présentation avec Q/R (1h30)
- Définition et évolution de l'IA :
 - Terminologie et évolution
 - Des premiers algorithmes aux réseaux de neurones
 - Apprentissage profond et IA générative
 - IA faible, générale et supérieure
 - Exemples d'applications dans divers domaines
- Facteurs qui soutiennent l'IA :
 - Structures algorithmiques et heuristiques
 - Aspects matériels (CPU, GPU, TPU)
 - Langages (R, Python, C++, Rust, Mojo)
 - Logiciels et cadres
- Aperçu du Machine Learning :
 - Principales catégories de modèles
 - Apprentissage supervisé et non supervisé
 - Applications et domaines d'utilisation
- Aperçu du Deep Learning :
 - Différences avec le Machine Learning
 - Concepts et modèles
 - Applications et domaines d'utilisation
- Particularités du traitement du langage naturel
- Aperçu de l'IA générative :
 - Modèles de langage, image et multimodaux
 - Applications et produits sur le marché
 - Exemple : Google Gemini
- Problèmes et dangers :
 - Éthique et biais
 - Vie privée et impact environnemental
 - Risque de dystopies

Module 2: Démonstration de l'Intelligence Artificielle

- Survol et présentation avec Q/R (1h30)
- DevOps et pipelines IA :
 - Prétraitement des données
 - Entraînement et optimisation
 - Déploiement
- Machine Learning en pratique :
 - Démonstration avec Scikit-Learn
 - Approche conviviale avec PyCaret

- Note: Codes sources avec Google Colab
- Deep Learning en pratique :
 - Démonstration avec Numpy
 - Utilisation de PyTorch
 - Note: Codes sources avec Google Colab
- IA générative :
 - Exploration des grands modèles de langage et multimodaux
 - Affinage et agents autonomes
 - Note: Pratique dans le module suivant avec Google Gemini

Module 3: Utilisation de Google Gemini

- Expérimentation avec Google Colab (3h30)
- Avec le chatbox :
 - Formulation des invites simples
 - Conseils pour les invites Python et autres
- Avec l'API Python :
 - Conception d'invites avancées
 - Optimisation de code et tests unitaires
- Avec le Studio :
 - Fine tuning et Retrieval-Augmented Generation
 - Construction d'un agent autonome
 - Projets expérimentaux avec Python et autres

Suite à cette formation avec OpenAI

[Formation OpenAI pour les développeurs](#)