

PROGRAMME DE FORMATION

AGENTS IA SOUVERAINS

FICHE SYNTHÉTIQUE DE LA FORMATION

Intitulé	Agents IA souverains — outils, données et workflows sous contrôle
Référence	IA-AGENTS-SOUV-2J
Durée	2 jours (14 heures)
Modalités	Présentiel ou distanciel synchrone (classe virtuelle)
Tarif	1 400 € HT / participant (inter) — intra sur devis
Délais d'accès	Inscription possible jusqu'à 14 jours avant le début de la session
Lieu	Paris 17e ou distanciel via Zoom
Effectif	4 participants maximum par session
Contact	contact@pythonia.fr – 06 66 06 38 59
Accessibilité PSH	Formation accessible aux PSH. Contactez notre référent handicap pour toute adaptation.
Dernière mise à jour	Octobre 2026

1. Public visé

Cette formation s'adresse aux développeurs, data scientists, architectes, équipes DSI et responsables de l'automatisation qui veulent construire des agents IA capables d'agir sur leurs systèmes (outils, bases de données, workflows) tout en gardant la maîtrise de l'infrastructure, des données et des actions réalisées.

2. Prérequis

- Maîtriser les bases de Python (fonctions, classes simples, modules, appels d'API)
- Avoir déjà utilisé un assistant IA générative et connaître la notion de prompt
- Avoir suivi « Installer et exploiter un LLM en local » ou disposer de connaissances équivalentes (recommandé)
- Poste avec 16 Go de RAM minimum, 30 Go d'espace disque libre, droits administrateur et Docker installé

Un test de positionnement est proposé avant l'entrée en formation afin de vérifier l'adéquation du parcours avec le profil du participant et la compatibilité de son poste.

3. Objectifs pédagogiques

À l'issue de cette formation, le stagiaire sera capable de :

- Expliquer ce qu'est un agent IA, ses composants (modèle, outils, mémoire, boucle de décision) et ses limites
- Doter un LLM local d'outils grâce à l'appel de fonctions (function calling) et au protocole MCP
- Construire un agent qui interroge des données internes (SQL, documents, API) de façon sûre
- Orchestrer des workflows multi-étapes et multi-agents avec LangGraph en local
- Mettre en place des mécanismes de contrôle : validation humaine, limites d'action, journalisation, évaluation

4. Programme détaillé

Fil rouge : les participants construisent un agent métier qui interroge une base de données interne, consulte des documents et exécute un workflow avec validation humaine, entièrement sur une infrastructure maîtrisée.

JOUR 1 — Comprendre et construire un agent outillé (7h)

Matin (3h30) — Du LLM à l'agent

Qu'est-ce qu'un agent IA ?

- Assistant, chaîne, agent : différences et cas d'usage
- La boucle de raisonnement : observer, planifier, agir (ReAct)
- Risques propres aux agents : actions non voulues, boucles, coûts, fuites de données

Agents et souveraineté

- Choisir un modèle local performant en appel d'outils (Qwen, Mistral, Llama)
- Hébergement interne vs cloud européen, RGPD, AI Act
- Panorama des frameworks : LangGraph, smolagents, PydanticAI, CrewAI

Appel de fonctions

- Décrire un outil : nom, description, schéma des paramètres (JSON Schema)
- Function calling avec Ollama et l'API compatible OpenAI
- Sorties structurées et validation avec Pydantic

Travaux pratiques :

- TP1 : un agent local doté de trois outils (calcul, recherche documentaire, date) écrit sans framework

Après-midi (3h30) — Outils et accès aux données

Le protocole MCP

- Principe du Model Context Protocol : serveurs, outils, ressources
- Utiliser des serveurs MCP existants (fichiers, base de données)
- Écrire son propre serveur MCP en Python

Interroger des données internes

- Agent SQL : génération de requêtes en lecture seule et contrôle du schéma
- Combiner l'agent avec un RAG documentaire
- Appel d'API internes et gestion des erreurs

Travaux pratiques :

- TP2 : écrire un serveur MCP exposant une base de données métier
- TP3 : agent qui répond aux questions en combinant SQL et documents

✓ **Évaluation formative : quiz de 10 questions et revue collective des TP en fin de journée**

JOUR 2 — Orchestration, contrôle et mise en production (7h)

Matin (3h30) — Orchestrer des workflows

Workflows avec LangGraph

- Graphes d'états : nœuds, transitions, conditions
- Mémoire de conversation et état persistant
- Reprise après erreur et points de sauvegarde

Systèmes multi-agents

- Superviseur et agents spécialisés
- Répartition des tâches et transmission du contexte
- Quand un seul agent suffit : éviter la complexité inutile

Travaux pratiques :

- TP4 : workflow de traitement de demandes (analyse, recherche, rédaction de réponse) orchestré avec LangGraph

Après-midi (3h30) — Contrôle, sécurité et déploiement

Garder le contrôle

- Validation humaine avant les actions sensibles (human-in-the-loop)
- Liste blanche d'outils, droits minimaux, limites de coûts et d'itérations
- Injections de prompt via les données et les outils : prévention
- Journalisation complète des décisions et des actions

Évaluer et déployer

- Évaluer un agent : taux de réussite des tâches, choix des outils, traces
- Exposer l'agent via une API et une interface
- Conteneurisation, supervision et feuille de route de déploiement

Travaux pratiques :

- TP5 : ajouter une validation humaine et des garde-fous au workflow
- TP6 : évaluer l'agent sur une série de tâches de référence

✓ **Évaluation sommative : QCM final de 20 questions et démonstration de l'agent — critère de réussite : 60 %**

5. Méthodes et moyens pédagogiques

Méthodes mobilisées

- Apports théoriques : exposés interactifs avec diaporama et démonstrations en direct
- Mise en pratique : TP progressifs construisant un agent complet sur deux jours
- Pédagogie active : cas d'usage apportés par les participants
- Alternance théorie/pratique : ratio 30 % théorie / 70 % pratique

Moyens techniques

- Salle de formation équipée : vidéoprojecteur, paperboard, connexion Wi-Fi haut débit
- Poste informatique par stagiaire (ou poste personnel répondant aux prérequis)
- En distanciel : Zoom avec partage d'écran, tableau blanc collaboratif
- Supports de cours, notebooks et code source complet remis aux stagiaires
- Base de données et documents d'exemple fournis

6. Modalités d'évaluation

Positionnement à l'entrée

Avant l'entrée en formation, chaque participant complète un test de positionnement évaluant son niveau en Python, ses connaissances en IA générative et la configuration de son poste.

Évaluations en cours de formation

- Évaluation formative : quiz de 10 questions en fin de jour 1
- Évaluation sommative : QCM final de 20 questions et démonstration de l'agent en fin de jour 2

- Critère de réussite : obtenir 60 % au QCM final
- Attestation de fin de formation : remise à l'issue, mentionnant objectifs, durée et résultats

Satisfaction

- Questionnaire de satisfaction en fin de formation (formateur, contenu, logistique, atteinte des objectifs)

7. Accessibilité aux personnes en situation de handicap

Pythonia s'engage à rendre ses formations accessibles aux personnes en situation de handicap. Notre référent handicap est disponible pour étudier toute demande d'adaptation :

- Adaptation des supports pédagogiques (taille de police, contraste, format audio)
- Aménagement des conditions de passage (temps majoré, pauses adaptées)
- Orientation vers nos partenaires spécialisés : Agefiph, Cap Emploi, MDPH

Contact référent handicap : contact@pythonia.fr – 06 66 06 38 59

8. Engagement et suivi

- Suivi de l'assiduité : feuille d'émargement signée par demi-journée
- Prévention des abandons : suivi individualisé, adaptation du rythme, relance en cas d'absence
- Réclamations : toute réclamation peut être adressée par email à contact@pythonia.fr. Elle sera traitée sous 48h ouvrées.

PythonIA — Centre de Formation Python & Intelligence Artificielle

255 boulevard Pereire, 75017 Paris | contact@pythonia.fr | 06 66 06 38 59