+33 (0)1 48 10 09 46
formation@ambient-it.net
140 Avenue Jean Lolive 93500 Pantin - France

Mis à jour le 19/05/2026

S'inscrire

Formation SGLang

3 jours (21 heures)

Présentation

SGLang est un framework orienté serving et programmation d'agents pour orchestrer des LLM avec des performances élevées. Il permet de construire des pipelines d'inférence fiables (chat, RAG, outils) tout en optimisant la latence et le débit.

Cette formation vise à rendre les participants autonomes sur la conception d'applications LLM en SGLang : structuration de prompts, gestion du contexte, appels d'outils, et mise en place de flux multi-étapes. L'approche met l'accent sur la reproductibilité, l'observabilité et les bonnes pratiques de déploiement.

La pédagogie est centrée sur des ateliers et démos : création d'un service d'inférence, implémentation d'un mini-RAG, ajout de garde-fous, puis tests de charge. Livrables : scripts SGLang, configuration de serving, jeux de tests, et checklist d'industrialisation.

Comme toutes nos formations, celle-ci vous présentera **la dernière version stable** de la technologie et ses nouveautés.

Objectifs

- Installer et configurer un environnement SGLang opérationnel.
- Écrire des programmes SGLang pour chat, RAG et workflows multi-étapes.
- Intégrer des outils (functions) et gérer les erreurs/retours structurés.
- Optimiser latence et débit via paramètres de serving et bonnes pratiques.
- Tester, tracer et packager une API prête pour la production.

Public visé

- Développeurs Python

- Ingénieurs ML/LLM
- Data engineers
- Architectes/Tech leads

Pré-requis

- Maîtrise de Python (fonctions, modules, environnements)
- Notions d’API HTTP et formats JSON
- Bases sur les LLM (prompting, tokens, contexte)
- Connaissances Git et ligne de commande

Pré-requis techniques

- Machine avec 16 Go RAM minimum (32 Go recommandé)
- Linux ou macOS ; Windows possible via WSL2
- Python 3.10+, pip/venv, éditeur de code (VS Code ou équivalent)
- Accès à une GPU NVIDIA (CUDA) recommandé pour l’inférence locale, sinon exécution CPU/serveur distant

Programme de notre formation SGLang

[Jour 1 - Matin]

Prise en main de SGLang et environnement d’exécution

- Positionner SGLang : objectifs (serving LLM, agents, workflows) et cas d’usage en production
- Installer et valider l’environnement : Python, dépendances, GPU/CPU, variables et configuration de base
- Comprendre le modèle d’exécution : scripts SGLang, runtime, gestion des sessions et du contexte
- Écrire ses premiers prompts structurés : rôles, gabarits, paramètres et sorties typées
- Atelier pratique : Exécuter un premier script SGLang et obtenir une sortie JSON exploitable.

[Jour 1 - Après-midi]

Contrôle des sorties et composition de prompts

- Structurer des réponses fiables : contraintes de format, champs obligatoires, validation côté application
- Composer des étapes : enchaînement de sous-tâches, réutilisation de variables et fonctions utilitaires
- Réduction des hallucinations : consignes, garde-fous, stratégies de reformulation et vérifications
- Gestion du contexte : découpage, résumés, mémoire courte vs longue et limites de tokens
- Atelier pratique : Construire un pipeline “analyse > extraction > restitution” avec sortie strictement structurée.

[Jour 2 - Matin]

Appels outillés (tools) et intégration applicative

- Définir des tools : schémas d’entrées/sorties, sérialisation et gestion des erreurs
- Orchestrer des appels externes : APIs REST, bases de données, fonctions métier et services internes
- Stratégies de sélection d’outil : routage, règles, priorités et fallback
- Observabilité : logs, traces, métriques et corrélation requête/réponse
- Atelier pratique : Ajouter un tool “recherche produit” et produire une réponse finale justifiée et traçable.

[Jour 2 - Après-midi]

Performance et serving : latence, débit et coûts

- Comprendre les leviers de performance : batching, parallélisme, cache, streaming et gestion de la concurrence
- Paramétrer l’inférence : température, top-p, max tokens, stop sequences et impact sur qualité/latence
- Optimiser les prompts : réduction de tokens, gabarits réutilisables et prompts “diff”
- Gestion des ressources : GPU memory, limites, timeouts et protection contre les requêtes coûteuses
- Atelier pratique : Mesurer latence et throughput, puis appliquer 3 optimisations et comparer les résultats.

[Jour 3 - Matin]

Qualité, tests et sécurité des applications LLM

- Mettre en place une stratégie de tests : jeux de cas, assertions, tests de non-régression et golden outputs
- Évaluer la qualité : critères, scoring, vérification de format et contrôles sémantiques
- Sécuriser les entrées : prompt injection, données sensibles, filtrage et politiques de contenu
- Gérer les échecs : retries, backoff, timeouts, dégradation contrôlée et messages utilisateurs
- Atelier pratique : Créer une suite de tests automatisés et durcir un workflow contre l’injection.

[Jour 3 - Après-midi]

Mise en production : packaging, déploiement et exploitation

- Industrialiser un projet : structure, configuration par environnement et gestion des secrets
- Exposer un service : endpoints, contrats d'API, quotas, authentification et journalisation
- Superviser en production : tableaux de bord, alerting, suivi des coûts et analyse des erreurs
- Plan de maintenance : versioning des prompts, migrations, A/B tests et rollbacks
- Atelier pratique : Livrer un mini-service SGLang prêt prod (API + monitoring + runbook).

Sociétés concernées

Cette formation s'adresse à la fois aux particuliers ainsi qu'aux entreprises, petites ou grandes, souhaitant former ses équipes à une nouvelle technologie informatique avancée ou bien à acquérir des connaissances métiers spécifiques ou des méthodes modernes.

Positionnement à l'entrée en formation

Le positionnement à l'entrée en formation respecte les critères qualité Qualiopi. Dès son inscription définitive, l'apprenant reçoit un questionnaire d'auto-évaluation nous permettant d'apprécier son niveau estimé sur différents types de technologies, ses attentes et objectifs personnels quant à la formation à venir, dans les limites imposées par le format sélectionné. Ce questionnaire nous permet également d'anticiper certaines difficultés de connexion ou de sécurité interne en entreprise (intraentreprise ou classe virtuelle) qui pourraient être problématiques pour le suivi et le bon déroulement de la session de formation.

Méthodes pédagogiques

Stage Pratique : 60% Pratique, 40% Théorie. Support de la formation distribué au format numérique à tous les participants.

Organisation

Le cours alterne les apports théoriques du formateur soutenus par des exemples et des séances de réflexions, et de travail en groupe.

Validation

À la fin de la session, un questionnaire à choix multiples permet de vérifier l'acquisition correcte des compétences.

Sanction

Une attestation sera remise à chaque stagiaire qui aura suivi la totalité de la formation.