

Nouveau Sessions garanties disponibles

Formation OpenAI Codex : accélérer le développement logiciel avec l'IA

Réf. SF34543

Exploitez OpenAI Codex pour générer, refactorer et documenter votre code grâce au prompting, aux Tools personnalisés et aux agents spécialisés !

2 j Durée	1 600 € HT Inter / participant	4 000 € HT Intra (≤12 pers.)	1 Sessions à venir
--------------	-----------------------------------	---------------------------------	-----------------------

Objectifs pédagogiques

- ✓ Configurer OpenAI Codex dans son environnement de développement (CLI et IDE)
- ✓ Structurer des prompts techniques pour générer du code fiable et pertinent
- ✓ Utiliser les actions intégrées de Codex pour documenter, tester et refactorer du code
- ✓ Créer des Tools personnalisés via function calling pour automatiser des workflows
- ✓ Orchestrer des agents spécialisés pour des tâches de développement complexes
- ✓ Évaluer la qualité du code généré et appliquer les bonnes pratiques de sécurité

Public visé & prérequis

Public visé Développeurs

Prérequis- Maîtriser au moins un langage de programmation (Python, JavaScript, Java ou équivalent) - Connaître les bases de Git (commits, branches, pull requests) - Être à l'aise avec un IDE (VS Code, JetBrains ou équivalent)

Programme détaillé

Introduction à OpenAI Codex et à l'écosystème IA pour le code

- Positionnement de Codex dans l'écosystème : comparaison avec GitHub Copilot, Cursor et Claude Code pour comprendre les forces spécifiques de Codex
- Cas d'usage concrets : génération de code, refactoring, documentation automatique, tests et création d'outils internes
- Architecture agentique : comprendre le fonctionnement d'un assistant IA de codage et la notion d'agent exécutant des actions
- Modes d'automatisation : suggest, auto-edit et full-auto — savoir quand utiliser chaque niveau d'autonomie

Activités pratiques :

- Tour d'horizon des cas d'usage sur des exemples de projets réels
- Comparaison des résultats obtenus avec différents modes d'automatisation sur une même tâche

Installation, configuration et prise en main

- CLI Codex : installation, commandes de base et navigation dans un dépôt de code
- Extension IDE : mise en place dans VS Code (et JetBrains) avec configuration des raccourcis et préférences
- Accès API et gestion des clés : configuration sécurisée des credentials et gestion des modèles disponibles (o3, gpt-4.1, o3-code)
- Paramétrage du modèle : température, longueur de sortie, system prompt et leur impact sur les résultats
- Sandbox et sécurité : comprendre l'architecture de sandboxing et les permissions d'exécution

Activités pratiques :

- Installation complète de l'environnement (CLI + extension IDE) et première interaction avec Codex
- Configuration personnalisée des paramètres et test de l'impact sur la génération

Prompt engineering appliqué au code

- Structure d'un prompt technique : objectifs, contraintes, style de code et format de sortie attendu
- Context window et tokens : comprendre les limites et optimiser l'utilisation du contexte disponible
- Contexte persistant multi-fichiers : créer et gérer des filesets pour que Codex comprenne l'ensemble d'un projet
- Patterns de prompts réutilisables : constituer une bibliothèque de prompts pour les tâches récurrentes (génération, revue, debug)
- Langage naturel vs prompts structurés : choisir la bonne approche selon la complexité de la tâche

Activités pratiques :

- Rédaction de prompts progressivement complexes et comparaison de la qualité des résultats
- Construction d'un contexte multi-fichiers pour un projet existant et vérification de la compréhension par Codex

Actions intégrées : générer, documenter, tester et refactorer

- Actions de génération : Create File, Edit File et génération de blocs multi-fichiers
- Actions d'analyse : Review, Explain, Find Bugs — exploiter Codex pour la relecture de code
- Actions de qualité : Document, Generate README, génération de docstrings et guides d'usage
- Refactoring assisté : Plan, Refactor et suggestions contextualisées pour améliorer le code existant
- Génération de tests : création de tests unitaires et d'intégration à partir du code source
- Suivi et traçabilité : logs, historique des générations et diffs pour garder le contrôle

Activités pratiques :

- Générer un module complet avec ses tests et sa documentation en enchaînant les actions
- Refactorer un code legacy en utilisant les actions Review puis Refactor

Tools et function calling : étendre Codex avec des outils personnalisés

- Concept de Tool dans l'écosystème OpenAI : comprendre le function calling et son rôle dans l'automatisation
- Création d'un Tool personnalisé : déclaration du schéma JSON, définition des paramètres et contraintes
- Cas d'usage métier : accès base de données, appels à une API interne, génération de migrations, manipulation de données
- Orchestration Codex + Tools : chaîner les appels entre Codex et les outils pour des workflows complets
- Sécurisation : validation des entrées/sorties et gestion des erreurs dans les Tools

Activités pratiques :

- Créer un Tool personnalisé qui interroge une API REST et intégrer ses résultats dans la génération de code
- Automatiser une tâche de build ou de migration en combinant Codex et un Tool dédié

Agents spécialisés et orchestration multi-agents

- Agents assistants vs agents exécutants : comprendre les différents rôles et niveaux d'autonomie
- Conception d'agents spécialisés : reviewer, coder, architect — définir le périmètre et les instructions de chaque agent
- Orchestration multi-agents : faire collaborer plusieurs agents sur un même projet avec des responsabilités distinctes
- Exécution sécurisée : sandbox, validation des résultats et réduction des hallucinations
- Limites actuelles et bonnes pratiques : savoir quand l'approche multi-agents apporte de la valeur et quand rester simple

Activités pratiques :

- Mettre en place un duo d'agents (architecte + développeur) pour implémenter une nouvelle fonctionnalité
- Créer un agent reviewer qui analyse automatiquement une pull request et produit des commentaires

Bonnes pratiques, sécurité et projet fil rouge

- Qualité du code généré : stratégies de validation, tests systématiques et critères d'acceptation
- Sécurité et secrets : gestion des clés API, protection des données sensibles dans les prompts et les Tools
- Workflows d'approbation : établir des règles d'équipe sur l'utilisation de Codex (revue humaine, niveaux d'autonomie)
- Pièges courants : hallucinations, sur-confiance dans le code généré, dépendance excessive à l'outil
- Inputs multimodaux : utiliser des screenshots et diagrammes comme entrées pour la génération de code
- Projet fil rouge : consolidation de toutes les compétences dans un mini-projet combinant prompting, actions, Tools et agents

Activités pratiques :

- Réalisation du projet fil rouge : développement d'une fonctionnalité complète en exploitant l'ensemble des techniques vues (génération, tools, agents, tests)

– Mise en place d'un workflow d'équipe avec règles d'approbation et bonnes pratiques partagées

Prochaines sessions

DATE	LIEU	STATUT
14/12/2026	à distance	Sur confirmation

Pourquoi choisir Sparks Formation ?

20 ans d'expertise IT, des formateurs experts et des sessions garanties contre l'annulation.

20 ans
d'expertise

1 068+
formations

11 178
apprenants formés/an

+67
Net Promoter Score

Avis des participants

Note moyenne des participants de la filière Développement et Ingénierie de Logiciels : 4,9/5 — 165 avis

Selim B. — Très bonne formation, très instructif même après quelques années de Python dans les bras. Une très petite équipe de stagiaires qui facilite les échanges. Je redoutais le mode...

Alexis C. — Anthony explique bien les concepts attendus et prend le temps de répondre aux questions et aux différents problèmes rencontrés. Le support présenté créé par lui-même est différent des...

Votre formateur

Camilo R. — Formateur

Camilo intervient notamment sur l'IA appliquée aux métiers et sa gouvernance, ainsi que sur l'ingénierie des LLM et l'IA générative. Son activité de formation a débuté en 2024. La conception...

Ils nous font confiance

Cette année notre filière Développement et Ingénierie de Logiciels a accompagné plus de 459 entreprises dont :

SOPRA STERIA RENAULT SAS ORANGE CAMPUS TECH ORANGE INNOVATION SCHOOL STMICROELECTRONICS ARPEGE GRANT THORNTON FRANCE ORANGE BUSINESS RTE SII ILE DE FRANCE +449 autres

Financement & certifications

Cette formation est éligible aux financements OPCO et FNE-Formation. Nos équipes vous accompagnent dans le montage de votre dossier.



Réservez votre place dès maintenant

20 ans d'expertise · 1 000+ formations · NPS +67 · réponse sous 24h ouvrées

0 805 950 800

demande@sparks-formation.com