Formation Claude Code pour développer applications, jeux ou sites

Modèles de langage et Claude Code. Lire du code sans devenir développeur. Cadrer et piloter un projet. Nombreux exemples. Skills, références visuelles et fonctionnalités commandées. Architecturer une app. Atelier pratique de deux jours sur mesure.

DATES du 26/10/2026 au 30/10/2026 - du 07/12/2026 au 11/12/2026 -
DURÉE TOTALE 5J. (35H.)
MODALITES D'ORGANISATION Formation présentielle, en nos locaux, 13 rue Desargues, 75011 Paris.
HORAIRES 10h-13h puis 14h-18h
NOMBRE DE PARTICIPANTS 6 maximum
PARTICIPANTS Toute personne voulant découvrir et maîtriser Claude Code, pour créer des projets numériques : sites, applications, automatisations de tâches, plug ins & scripts... sans être développeur.
NIVEAU REQUIS Pratique des outils informatiques fluide. Pratique d'un outil IA conversationnel (chatGPT, Claude, Gemini...) Aucune connaissance en programmation n'est requise. Entretien pédagogique.
ACCESSIBILITÉ Nos formations sont accessibles et aménageables pour les personnes en situation de handicap. Fauteuils roulants autorisés en largeur maxi 70cm. Pour tout handicap, auditif, visuel ou autre, merci de nous contacter pour les aménagements possibles.
VALIDATION DES ACQUIS Contrôle continu tout au long de la formation, exercé par le formateur et supervisé par la responsable pédagogique. Délivrance d'un Certificat de Validation des Acquis de Formation.
FINANCEMENT En cas de financement « CPF », cette formation inclut la préparation et le paiement du test de certification : « Produire et réviser du contenu professionnel multimédia en utilisant les outils d'Intelligence Artificielle Générative (IAG) de façon responsable » Cette certification est enregistrée au Répertoire spécifique de France Compétences : RS6891 (Certificateur : CERTIFOPAC / 28.11.2024 – 28.11.2027). C'est l'organisme certificateur, Certifopac, qui évaluera les compétences acquises par les candidats. Afin de valider la certification, les apprenants doivent démontrer leur capacité à mettre en application les compétences du référentiel de certification, à savoir : Choix des outils IAG : capacité à sélectionner les technologies pertinentes en fonction des besoins Interaction avec l'IAG : aptitude à dialoguer efficacement avec l'IA pour obtenir des résultats optimaux Création de contenus : maîtrise des méthodes de production adaptées aux objectifs métier Optimisation des contenus : application des techniques d'amélioration des résultats générés par l'IA Considérations éthiques et réglementaires : prise en compte des biais, de la confidentialité et de la conformité légale pour concevoir sa solution métier Plus d'informations sur le référentiel de compétences ici : https://drive.google.com/file/d/1f3r0QGgvu28sZ-1x3Quu2SJbz4B3gA3-/view Le processus d'évaluation se déroule en 2 étapes : Une épreuve écrite : Rédaction d'un rapport, de 10 pages maximum, détaillant un cas pratique, réel ou fictif, en lien avec les outils d'IAG. Démonstration des compétences C1 à C4. Dépôt du rapport via WEDOF au moins 15 jours avant la soutenance Une épreuve orale : Présentation de 20 minutes en visioconférence devant un jury. Temps d'exposé : 10 minutes pour présenter les résultats obtenus. Questions du jury : 10 minutes pour valider la compétence C5 Le Bahut se charge de votre inscription à la certification auprès de Certifopac. Le coût d'un passage est inclus dans le tarif de la formation. Pour en savoir plus sur les modalités d'inscription, d'évaluation et de certification : https://certifopac.fr/certifiag
MOYENS TECHNIQUES 6 stagiaires maximum. 1 station par stagiaire, écran 27 pouces, Mac ou Windows. Comptes, licences et crédits Claude Max, chatGPT, Git & Git Hub, Visual Studio Code, Adobe Photoshop. Accès internet.
MOYENS ET MODALITES PEDAGOGIQUES Une salle de formation avec tableau blanc, un poste informatique pour chaque apprenant, un poste informatique vidéo-projeté pour les formateurs, avec accès internet et imprimante partagée. Alternance de séquences théoriques et d'exercices d'application réelle. Les exercices sont réalisés sous la supervision des formateurs, dans une logique d'apprentissage des compétences et d'autonomie progressive des apprenants. Des exercices plus longs et synthétiques permettent de reprendre un ensemble de compétences dans des cas pratiques types des métiers et compétences concernés, et

s'assurer de leur acquisition par les apprenants.
SUPPORT DE COURS Mémo écrit en interne, supports de cours online et pdf.
FORMATEURS Professionnel en activité, expert reconnu dans son domaine, animant régulièrement des formations.
LABELS QUALITÉS Centre certifié Qualiopi (certification nationale) / certifié ISQ-OPQF / référencé Data-Dock / centre certifié Adobe / Maxon / Blackmagic Design / centre de test Certiport

OBJECTIFS DE LA FORMATION

Ce stage de formation de 5 jours est une initiation à Claude Code, l'agent de développement d'Anthropic, et à cette manière nouvelle de concevoir des programmes, des applications ou des sites en dialoguant avec une intelligence artificielle plutôt qu'en écrivant soi-même du code.

A part créer des sites web, quelles applications peut-on imaginer ? Tout outil utile.. par exemple : catalogues, portfolios en ligne, calendriers, outils de comptabilité, jeux vidéo, inventaires, outils de gestion, appli de suivi de versions, carnet de repérages, simulateur de tarifs & devis, suivi des notes de frais, calculateurs, plugs in et scripts pour VFX et 3D…

L'histoire de la programmation est celle d'un empilement de couches successives : on a d'abord codé en binaire, puis en assembleur, puis dans des langages de bas niveau comme le C, puis dans des langages de plus haut niveau comme le C# ou le Swift. À chaque étape, une couche supplémentaire est venue masquer la précédente et rendre l'écriture plus lisible, plus proche de la pensée humaine. Claude Code et les outils comparables constituent peut-être une nouvelle marche de cet escalier, où l'on formule en langage naturel ce que l'on souhaite obtenir. Cette formation propose simplement d'aller voir ce que cela change concrètement.

Aucun prérequis en programmation n'est nécessaire : la formation est conçue pour un public non-développeur, curieux d'explorer ces outils. Les stagiaires n'apprennent pas à écrire du code, mais à le **lire, le comprendre et le vérifier** — assez pour rester maîtres de ce qu'ils produisent et pour repérer quand le résultat s'écarte de ce qui était demandé. Cette capacité de contrôle est le cœur de la formation.

La progression de la formation est entièrement fondée sur la pratique et commence en douceur.
Les premiers exercices consistent à faire réaliser de petits scripts pour prendre en main l'outil et obtenir des résultats immédiats, avant de monter progressivement en ambition jusqu'à la conception d'une application. En chemin, la formation aborde les **skills** -ces méthodes réutilisables que l'on encode une fois et que l'IA applique ensuite systématiquement-, la création de fichiers de cadrage et de suivi de projet, l'usage de Git et GitHub, ainsi que la façon de nourrir Claude de références visuelles pour traduire une intention esthétique en résultat concret.

Les deux dernières journées sont consacrées à la réalisation d'un projet plus ambitieux, que chaque stagiaire peut choisir en fonction de ses propres besoins, avec l'accompagnement du formateur.

OBJECTIFS DE LA FORMATION :

- Comprendre fonctionnement et limites d'un modèle de langage
- Gérer fenêtre de contexte et structurer une demande efficace selon différentes méthodes de prompt engineering
- Identifier les briques fondamentales d'un programme, expliquer ce que fait un extrait de code, se repérer dans l'arborescence d'un projet et comprendre l'architecture d'une application : interface, serveur et base de données
- Distinguer langage, bibliothèque, framework et outil
- Planifier un projet avant de le produire, en utilisant le mode plan
- Rédiger et faire vivre un fichier de conventions propre à un projet
- Configurer les permissions et modes de travail de Claude Code selon le niveau de risque d'une tâche
- Appliquer le cycle de vérification : lire une modification, tester, repérer une dérive et recadrer l'IA
- Utiliser des skills et comprendre ce qu'ils apportent à la régularité d'un travail
- Nourrir l'IA de références visuelles pour traduire une intention esthétique en résultat
- Créer et tenir à jour des documents de cadrage et de suivi de projet
- Versionner un projet avec Git et GitHub : commit, historique, retour en arrière

- Utiliser Claude Code dans un environnement de développement (Visual Studio Code)
- Concevoir et réaliser un projet numérique personnel avec l'accompagnement du formateur
- E?valuer Claude Code et ses re?sultats en termes de biais, e?thique, impact e?cologique et conformite? re?glementaire

INSCRIPTION PAR LE CPF ? inscrivez-vous directement sur la plateforme MonCompteFormation !!!!

TEST DE CERTIFICATION inclus uniquement en cas de financement CPF.
Pour tout autre moyen de financement (France Travail, Opco, fonds propres…), la formation n'incluera pas le passage du test de certification.

FORMATEUR PRINCIPAL Vincent Coste, expert certifié en IAG, monteur et truquiste, expert des workflows numériques, développeur d'applications.

CONTENU COMPLET DE LA FORMATION

BASES DES MODÈLES DE LANGAGE ET PRISE EN MAIN

- (Toute) petite histoire de la programmation : du binaire aux langages de haut niveau, jusqu'au langage naturel.
  Principes et limites d'un modèle de langage : ce qu'il sait faire, ce qu'il ne sait pas faire.
- La fenêtre de contexte : ce que l'IA garde en mémoire, comment la gérer, quand repartir de zéro.
- Les différentes interfaces de Claude et le choix du modèle selon la tâche.
- Méthodes de prompt engineering : la structure CCOF (Contexte, Contraintes, Objectif, Format), le reverse questioning, l'usage d'exemples, le découpage d'une tâche complexe.
- Installation et prise en main de Claude Code.

Atelier pratique : déléguer à Claude Code la création de premiers scripts utiles — un compte à rebours sonore, puis un outil de classement automatique de fichiers, transformé en application double-cliquable.

LIRE ET COMPRENDRE DU CODE

Il ne s'agit pas d'apprendre à coder, mais d'acquérir de quoi lire, vérifier et se repérer — comme savoir lire une carte sans construire la route.

- Les briques fondamentales d'un programme : variable, type, fonction, condition, boucle, transformation de valeurs.
- Le point d'entrée d'un programme : par où tout commence.
- Se repérer dans l'arborescence d'un projet : fichiers de configuration, dossiers de dépendances.
- Distinguer langage, bibliothèque, framework et outil.
- Le réflexe d'autonomie : faire expliquer par Claude tout code ou concept inconnu.

Atelier pratique : ouvrir les scripts produits la veille et y identifier ensemble chaque brique, puis interroger Claude sur les passages obscurs.

CADRER ET PILOTER UN PREMIER PROJET

Le passage du script isolé au projet structuré : dépasser l'effet « waouh » de la génération automatique pour acquérir une méthode de travail.

- Préparer ses matériaux avant de construire : standardisation et conventions de nommage.
- Le mode plan : faire analyser et proposer sans rien modifier, puis valider avant de produire.
- Le fichier de conventions CLAUDE.md : capturer ce qui vaut pour tout le projet, et le distinguer de la demande ponctuelle.
- Les permissions et modes de travail : régler ce que Claude a le droit de faire selon le niveau de risque.
- Le cycle de vérification : lire une modification, tester, repérer une dérive, recadrer.
- La gestion du contexte en session longue : compacter, repartir de zéro, travailler en sessions parallèles.

Atelier pratique : construire un projet complet à partir de contenus fournis, en dialoguant et en itérant, jusqu'à la mise en ligne.

SKILLS, RÉFÉRENCES VISUELLES ET COMMANDE CLIENT

Les skills : encoder une fois une méthode, une charte ou un ensemble de règles, et les faire appliquer systématiquement par l'IA. Nourrir Claude de références : captures d'écran, gabarits, chartes graphiques, extraits de code — et savoir nommer ce que l'on en retient. Traduire une intention en instructions exploitables. Répondre à un cahier des charges externe et vérifier la conformité du résultat livré. Documents de cadrage et de suivi : direction artistique, architecture, jalons d'avancement. Faire commenter et tester le code produit. Git et GitHub : commit, historique, retour en arrière, publication et mise à jour automatique.

Atelier pratique : répondre à une commande client fictive — un festival de cinéma — en intégrant ses exigences fonctionnelles et graphiques, puis livrer et publier le résultat.

ARCHITECTURE D'UNE APPLICATION

Ce que change le passage d'un projet statique à une véritable application : des données qui persistent et une interface qui réagit.

Les trois couches : interface, serveur et base de données, et la frontière entre elles. Les notions d'API et de requête : comment ces couches se parlent. La persistance des données : ce qui disparaît, ce qui survit. Les quatre opérations fondamentales sur des données : créer, lire, modifier, supprimer. Situer une bibliothèque comme React dans l'écosystème.

Atelier pratique : construire une application de gestion de fiches reliée à une base de données locale, couche par couche.

BIAIS, DROITS, ÉTHIQUE, IMPACT, CONFORMITÉ RÉGLEMENTAIRE

- Questions des droits : propriété du code produit avec une IA, licences des bibliothèques et gabarits utilisés, frontière entre inspiration et

copie.
- Problématiques des biais des IA génératives.
- Éthique et éco-responsabilité : enjeux sociaux et écologiques, empreinte carbone.
  Comprendre l'IA Act.
- Problématiques de confidentialité.
- La vérification du résultat comme responsabilité professionnelle : ce que l'on signe reste de sa responsabilité.
- Quelles modalités techniques et quels outils protègent les sources, les données et les droits ?

ATELIER PRATIQUE PERSONNEL

Utilisation de Claude Code dans un environnement de développement (Visual Studio Code).

Projet final personnalisé : chaque stagiaire mène de bout en bout un projet plus ambitieux, choisi selon ses besoins.

Durant deux jours, le formateur accompagne chaque stagiaire sur projet librement choisi :

- Cadrage de l'idée
- planification
- itérations
- vérification
- livraison

PERSONNALISATION DE LA FORMATION AUX BESOINS DE CHACUN

Cette formation prend en compte les métiers, les expériences et les besoins de chaque stagiaire, afin d'adapter les exemples, les exercices et les projets proposés aux différentes personnes.

CERTIFICATION CERTIFIAG

Pour les stagiaires participant à la formation par le biais CPF et avec la certification CertifIAG, des moments spécifiques permettront de comprendre les compétences à rpouver. Les exercices pratiques, ateliers personnels notamment, seront l'occasion de flécher les compétences à valider et permettront de préparer au mieux le passage de la certification.