

Mis à jour le 09/06/2026

S'inscrire

# Formation Cloudflare Workers

3 jours (21 heures)

## Présentation

Cloudflare Workers est une plateforme serverless permettant de développer, déployer et faire évoluer des applications directement sur le réseau mondial de Cloudflare. Elle permet de créer des APIs, fonctions edge, middlewares, traitements asynchrones et applications fullstack sans gérer d'infrastructure serveur.

Notre formation Cloudflare Workers vous permettra de maîtriser le développement d'applications serverless edge avec JavaScript, TypeScript et l'outil CLI Wrangler.

Vous apprendrez à créer des Workers, structurer des APIs HTTP, gérer les environnements, configurer les bindings, manipuler les secrets et connecter vos fonctions à des services de stockage comme KV, D1 et R2.

À l'issue de la formation, vous serez en mesure de concevoir des services serverless performants, de gérer de l'état avec Durable Objects, de créer des traitements asynchrones avec Queues, d'automatiser vos déploiements et de superviser vos applications en production.

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

## Objectifs

- Comprendre l'architecture et les usages de Cloudflare Workers
- Créer, tester et déployer des Workers avec Wrangler
- Développer des APIs serverless avec JavaScript ou TypeScript
- Connecter un Worker à KV, D1, R2, Durable Objects et Queues
- Sécuriser les secrets, bindings, routes et accès applicatifs
- Automatiser le déploiement, observer et optimiser des Workers en production

## Public visé

- Développeurs backend, fullstack JavaScript ou TypeScript
- Développeurs web souhaitant créer des APIs serverless
- Architectes applicatifs et cloud
- Ingénieurs DevOps ou cloud travaillant avec des équipes applicatives

## Pré-requis

- Connaissances de base en JavaScript ou TypeScript
- Notions d'API HTTP et de développement web
- Une première expérience avec Git ou un terminal est recommandée

## Pré-requis techniques

- Disposer d'un ordinateur avec Linux, macOS ou Windows avec WSL2
- Installer Node.js, npm et un éditeur de code comme Visual Studio Code
- Disposer d'un compte Cloudflare pour les ateliers de déploiement
- Prévoir une connexion Internet stable

## Programme de notre formation Cloudflare Workers

[Jour 1 - Matin]

### Comprendre Cloudflare Workers et le serverless edge

- Comprendre le rôle de Cloudflare Workers dans une architecture serverless et edge computing
- Identifier les cas d'usage : APIs, middlewares, webhooks, redirections, cache, routage, authentification et applications fullstack
- Différencier les architectures serverless traditionnelles et le modèle basé sur les isolates
- Comprendre les bénéfices : faible latence, montée en charge automatique, déploiement global et absence de gestion serveur
- Découvrir l'écosystème Workers : Wrangler, bindings, KV, D1, R2, Durable Objects, Queues et Cron Triggers
- Atelier pratique : créer un premier Worker, le lancer en local et déployer une fonction simple sur Cloudflare

[Jour 1 - Après-midi]

### Développement local avec Wrangler et TypeScript

- Installer et configurer Wrangler, l'outil CLI officiel pour Cloudflare Workers
- Créer un projet Worker avec JavaScript ou TypeScript

- Comprendre la structure d'un projet : endpoint, handler fetch, configuration et variables d'environnement
- Utiliser le serveur de développement local, les logs et les commandes de déploiement
- Gérer les environnements de développement, staging et production
- Atelier pratique : développer une API HTTP TypeScript avec routes, paramètres, réponses JSON et gestion d'erreurs

## Routage, requêtes HTTP et APIs serverless

- Manipuler les objets Request, Response, headers, méthodes HTTP et status codes
- Créer des routes pour exposer une API serverless légère et maintenable
- Gérer les paramètres d'URL, query strings, body JSON et validations simples
- Mettre en place une gestion d'erreurs cohérente pour les appels API
- Comprendre les contraintes d'exécution liées au runtime Workers
- Structurer un Worker pour séparer routes, services, types et configuration

[Jour 2 - Matin]

## Stockage avec KV, D1 et R2

- Comprendre les différents services de stockage intégrés à Cloudflare Workers
- Utiliser Workers KV pour stocker des données clé-valeur à lecture rapide
- Créer et interroger une base D1 pour des usages SQL serverless
- Manipuler R2 pour stocker des fichiers, médias ou objets applicatifs
- Choisir le bon stockage selon les besoins : cache, configuration, données relationnelles ou fichiers
- Atelier pratique : connecter un Worker à KV et D1 pour créer une mini-API de gestion de contenus

[Jour 2 - Après-midi]

## Bindings, secrets et intégration avec services externes

- Comprendre le rôle des bindings pour connecter un Worker à des ressources Cloudflare
- Configurer les variables d'environnement, secrets et paramètres par environnement
- Appeler des APIs externes depuis un Worker et gérer les erreurs réseau
- Mettre en place une stratégie de configuration propre pour les projets d'équipe
- Sécuriser les clés API, tokens et secrets applicatifs
- Atelier pratique : intégrer une API externe et sécuriser sa clé avec les secrets Wrangler

## Durable Objects, état partagé et coordination

- Comprendre les limites d'un Worker stateless et les usages de Durable Objects
- Créer un Durable Object pour gérer un état coordonné ou une logique temps réel
- Identifier les cas d'usage : sessions, compteurs, verrouillage, coordination, rooms et workflows interactifs
- Comprendre la persistance, l'adressage et la cohérence d'un Durable Object
- Évaluer les choix entre KV, D1, R2 et Durable Objects selon le besoin métier

- Atelier pratique : créer un compteur ou gestionnaire de session avec Durable Objects

## [Jour 3 - Matin]

### Queues, tâches asynchrones et automatisation

- Comprendre les usages de Cloudflare Queues pour découpler les traitements applicatifs
- Créer un producteur et un consommateur de messages dans Workers
- Gérer les traitements asynchrones, retries, erreurs et messages en échec
- Utiliser les Cron Triggers pour planifier des tâches récurrentes
- Identifier les patterns : webhooks, traitement différé, synchronisation, notifications et jobs périodiques
- Atelier pratique : créer un workflow asynchrone avec un Worker producteur, une Queue et un consumer

## [Jour 3 - Après-midi]

### Déploiement, CI/CD et observabilité

- Déployer un Worker avec Wrangler et gérer les environnements
- Comprendre les versions, rollbacks, déploiements progressifs et bonnes pratiques de release
- Intégrer Cloudflare Workers dans une chaîne CI/CD avec GitHub Actions ou GitLab CI
- Exploiter les logs, métriques, traces et outils d'observabilité Cloudflare
- Diagnostiquer les erreurs de production, limites runtime, latence et problèmes de configuration
- Atelier pratique : automatiser le déploiement d'un Worker et analyser ses logs après exécution

### Sécurité, performance et architecture de production

- Appliquer les bonnes pratiques de sécurité : secrets, headers, CORS, validation d'entrées et limitation d'abus
- Mettre en place des stratégies de cache, routage, rate limiting et protection applicative
- Optimiser les performances : latence, temps CPU, appels externes, stockage et architecture des bindings
- Comprendre les coûts, quotas, limites et choix d'architecture à l'échelle
- Concevoir une architecture serverless edge adaptée à un cas d'usage métier
- Atelier pratique : finaliser une API serverless sécurisée, observable et prête pour un déploiement production

## 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.