

Pôle Sciences, Technologies et Numérique
Centre des Académies et des Technologies (CAT)



**certificat en langage javascript niveau
débutant (JSE – certified Entry-Level
javascript programmer)**

présentation de la certification

Le certificat JSE™ (Certified Entry-Level JavaScript Programmer) est une certification professionnelle de niveau débutant, délivrée par le OpenEDG JavaScript Institute, en partenariat avec Pearson VUE.

Cette certification valide la maîtrise des bases du langage JavaScript, notamment la syntaxe fondamentale, les types de données, les structures de contrôle et les fonctions simples. Elle atteste de la capacité du candidat à comprendre, écrire et exécuter des scripts JavaScript élémentaires en appliquant une logique algorithmique de base. Reconnue à l'international, la certification JSE™ constitue une première étape structurante dans le parcours de formation en JavaScript et prépare à l'obtention de la certification JSA™ – Certified Associate JavaScript Programmer.

Inscrivez- vous

<https://forms.gle/JDTpeq2NF6SnXXfYA>

objectifs pédagogiques

À la fin de cette certification, l'apprenant doit être capable de :

- Comprendre les concepts fondamentaux de la programmation et du langage JavaScript.
- Utiliser correctement les variables, types de données et opérateurs pour créer des scripts.
- Contrôler le flux d'un programme à l'aide de structures conditionnelles et de boucles.
- Déclarer et appeler des fonctions, y compris des fonctions fléchées et récursives.
- Déboguer et gérer les erreurs dans un programme JavaScript.



Pourquoi choisir ce certificat ?

- Certification internationale reconnue délivrée par le OpenEDG JS Institute
- Excellente porte d'entrée vers la programmation JavaScript
- Alignée sur les standards professionnels du développement web
- Permet d'acquérir des bases solides et structurées en programmation
- Prépare efficacement aux certifications JavaScript de niveau supérieur (JSA™)
- Valorise le profil académique et professionnel des apprenants
- Compatible avec une approche pédagogique progressive (débutant - avancé)

Public cible

Ce certificat s'adresse principalement à :

- Débutants en programmation informatique
- Étudiants en informatique, réseaux, multimédia ou génie logiciel
- Apprenants souhaitant découvrir le développement web
- Professionnels désirant acquérir des bases en JavaScript
- Enseignants et formateurs en technologies numériques
- Toute personne souhaitant préparer une carrière dans le développement web

Durée et modalités

- **Durée** : 40 jours
- **Format** : Autoformation guidée 100% en ligne
- **Supports** : Cours interactifs, exercices pratiques, compilations réelles, tests d'évaluation
- **Évaluation** : Evaluation progressive durant chaque module + examen de certification à la fin
- **Badge numérique** délivré par CISCO à la fin de chaque module
- **Attestation de maîtrise** délivrée par le CAT à la fin de la formation
- **Certified Associate Web Developer** délivré par JS Institute après passage de la certification chez l'un des certificateurs agréés (Test Now ou Pearson Vue)
- **Coût** : Des coûts de formation et de délivrance de l'attestation pourraient être demandés par le CAT



CONTENU DE LA FORMATION

Module 1 – Introduction à JavaScript et programmation

Objectifs pédagogiques

- Présenter le langage JavaScript dans le contexte de l'informatique.
- Expliquer les concepts programmatiques de base.
- Initier à l'environnement d'exécution JS (navigateur, console)

Contenus abordés

- Concepts de base de la programmation.
- Fonctionnement de JavaScript et différences avec d'autres langages.
- Mise en place d'un environnement de développement.
- Exécution de scripts dans un navigateur ou via console.

Module 2 – Variables, types de données et conversions

Objectifs pédagogiques

- Savoir déclarer et utiliser des variables.
- Maîtriser les principaux types de données de JavaScript.
- Comprendre et appliquer le type casting.

Contenus abordés

- Variables : déclaration, initialisation et modification.
- Types primitifs : string, number, boolean, null, undefined, bigint.
- Types complexes : Array et Object.
- Conversion de types et opérations sur les données.

Module 4 – Opérateurs et interaction utilisateur

Objectifs pédagogiques

- Utiliser les opérateurs pour effectuer des calculs et des comparaisons.
- Interagir avec l'utilisateur via dialogues de navigateur.

Contenus abordés

- Opérateurs arithmétiques, logiques, de comparaison et d'affectation.
- Opérateur conditionnel et opérateurs spéciaux (typeof, instanceof, delete).
- Interaction : alert(), prompt(), confirm().



Module 4 – Contrôle de flux : conditions et boucles

Objectifs pédagogiques

- Implémenter des structures de décision et de boucle.
- Contrôler l'exécution répétée d'instructions

Contenus abordés

- Structures conditionnelles if, else, switch.
- Boucles for, while, do...while.
- Instructions de contrôle break et continue.
- Itération sur objets et tableaux (for...in, for...of)

Module 5 – Fonctions

Objectifs pédagogiques

- Déclarer et appeler des fonctions.
- Comprendre l'utilisation des fonctions comme objets de première classe.
- Utiliser des fonctions fléchées et des fonctions récursives.

Contenus abordés

- Déclaration et invocation de fonctions.
- Passage d'arguments et retour de valeurs.
- Variables locales, scope et shadowing.
- Fonctions fléchées (=>) et fonctions expressions.
- Concepts de callback, setTimeout et setInterval.

Module 6 – Erreurs, exceptions et débogage

Objectifs pédagogiques





- Comprendre les types d'erreurs en JS.
- Gérer les exceptions avec try...catch...finally.
- Diagnostiquer et corriger les bugs.

Contenus abordés

- Types d'erreurs : syntaxique, sémantique et logique.
- Exception handling : try, catch, throw, finally.
- Utilisation de l'outil de débogage pour analyse pas à pas.
- Mesure du temps d'exécution et inspection des variables





 Cité du savoir - Diamniadio
Avenue Bourguiba, rue n°13, Immeuble Adja Rokhaya
 +221 30 108 41 53
 BP 15126 Dakar - Fann
 www.unchk.sn



Foo nekk foofu la