

Développeur Web HTML 5, CSS 3 et JavaScript

JavaScript - Nouveautés apportées par les versions ECMAScript 6 à 9

2 jours (14h00) | ★★★★★ 4,6/5 | JVS-EC | Code Certif Info : 94827 | Certification M2i
Langages de Programmation (non incluse) | Évaluation qualitative de fin de stage | Formation
délivrée en présentiel ou distanciel ⁽¹⁾

Formations Informatique > Langages et développement > Développeur Web HTML 5, CSS 3 et JavaScript



À l'issue de ce stage vous serez capable de :

- Connaître et utiliser les apports de la norme ES2015, ou ECMAScript 6
- Connaître ES2016 (ECMAScript 7), ES2017 et ES2018 (ECMAScript 8 et 9)
- Exploiter ces particularités dans les nouveaux projets.

Niveau requis

Avoir une bonne connaissance de JavaScript et des fondamentaux de l'objet.

Public concerné

Développeurs ayant à réaliser des applications Web responsives.

Cette formation :

- Est animée par un consultant-formateur dont les compétences techniques, professionnelles et pédagogiques ont été validées par des diplômes et/ou testées et approuvées par l'éditeur et/ou par M2i Formation
- Bénéficie d'un suivi de son exécution par une feuille de présence émarginée par demi-journée par les stagiaires et le formateur.

(1) Modalité et moyens pédagogique :

Formation délivrée en présentiel ou distanciel * (e-learning, classe virtuelle, présentiel à distance). Le formateur alterne entre méthodes ** démonstrative, interrogative et active (via des travaux pratiques et/ou des mises en situation). La validation des acquis peut se faire via des études de cas, des quiz et/ou une certification.

Les moyens pédagogiques mis en oeuvre (variables suivant les formations) sont : ordinateurs Mac ou PC (sauf pour les cours de l'offre Management), connexion internet fibre, tableau blanc ou paperboard, vidéoprojecteur ou écran tactile interactif (pour le distanciel). Environnements de formation installés sur les postes de travail ou en ligne. Supports de cours et exercices.

* Nous consulter pour la faisabilité en distanciel. ** Ratio variable selon le cours suivi.

Programme

Jour 1

Introduction

- Rappels sur les aspects avancés de JavaScript
- Synthèse des apports de ES2015 et ES2016
- Compatibilité actuelle des browsers
- Tour d'horizon des outils de développement et d'intégration actuelle
- Compilateurs disponibles

Apports de ES2015

- Mot-clé "let"
- Assignation des variables
- Constantes
- Modification des API
- Utilisation de la nouvelle syntaxe des "arrow functions"
- Assignations destructurées
- Formatage des chaînes de caractères
- Object API, les nouvelles méthodes

Exemples de travaux pratiques (à titre indicatif)

- Utilisation des nouvelles syntaxes de ES2015
- Utilisation de "let", fonctions arrow et formatage

Programmation objet en ES2015

- Classe et héritage
- Méthodes statiques
- Création de proxy
- Nouveaux types
 - Set
 - Map
- Nouveaux objets héritables

Exemples de travaux pratiques (à titre indicatif)

- Création d'un projet à base de classes
- Utilisation de nouveaux types

Modularisation en ES2015

- Modularisation avec AMD et CommonJS
- Modularisation avec ES2015
- Différences entre les trois approches de modularisation
- Façons pour l'utiliser
- Gestion des dépendances
- "Dynamique loading"

Exemples de travaux pratiques (à titre indicatif)

- Modularisation du projet préalablement conçu avec AMD puis avec ES2015
- Gestion de dépendances

Jour 2

Itérateurs et générateurs

- Création d'un itérateur
- Toutes les nouvelles boucles "For"
- Création d'un générateur
- Exploitation d'un générateur

Asynchronisme avec JavaScript

- Présentation des "promises"
- Création et utilisation des "promises"

Exemple de travaux pratiques (à titre indicatif)

- *Création d'un projet en utilisant des "promises" et "promises" chaînées*

Déployer une application JavaScript à partir de ES2015

- JavaScript et TypeScript
- Transpileurs
- Package managers
- Traceurs
- Produit "Babel"
- Nécessité de packager son code
- Gestion des packages avec npm
- Outils de Lint et de test

Exemple de travaux pratiques (à titre indicatif)

- *Utilisation du transpileur Babel pour exécuter JavaScript sur d'anciens browsers*

Apports de ES2016 (ECMAScript 7)

- La fonction Array.prototype.include()
- L'opérateur Exponentiation

Apports de ES2017 (ECMAScript 8)

- Async functions
- Shared memory et les atomics

Apports de ES2018 (ECMAScript 9)

- Les itérations asynchrones
- Les propriétés REST / Spread
- Nouvelles expressions régulières
- La fonctionnalité Promise.prototype.finally()

Exemple de travaux pratiques (à titre indicatif)

- *Ecriture de sources JavaScript permettant de mettre en oeuvre les apports des versions 2016, 2017 et 2018 de JavaScript*

Certification (en option)

- Prévoir l'achat de la certification en supplément
- L'examen (en français) sera passé le dernier jour, à l'issue de la formation et s'effectuera en ligne
- Il s'agit d'un QCM dont la durée moyenne est d'1h30 et dont le score obtenu attestera d'un niveau de compétence

Modalités d'évaluation des acquis

L'évaluation des acquis se fait :

- En cours de formation, par des études de cas ou des travaux pratiques
- Et, en fin de formation, par un questionnaire d'auto-évaluation ou une certification (M2i ou éditeur)

Compétences visées

- Développer des applications ou logiciels conformément au cahier des charges
- Intégrer un produit ou un programme à partir des outils, méthodes ou langages
- Etre autonome dans l'exécution des tâches
- Préparer les plans de tests d'une application
- Conduire les tests unitaires
- Optimiser les tests de performance d'une application
- Assurer la veille technique et concurrentielle.