



## FORMATIONS DIPLÔMANTES AUX MÉTIERS DU WEB

### DIPLÔME FRONT-END DEVELOPER

---

## Certification WebProgramer | JavaScript ECMA 6

### 1. Structure du langage JS, Variables, Fonctions, Structures de contrôle, Collections.

- Concepts de programmation
  - Programme, instructions, séquences, alternatives, boucles, fonctions et procédures
  - Algorithmes
- Modélisation d'un programme
- Informations manipulées:
  - Types, Variables, Constantes, Paramètres, Visibilité et portée
- Introduction à JavaScript
  - Environnement de développement JS : éditeur, navigateur, débogueur
  - Bases du langage:
    - Variables
    - Constantes
    - Fonctions
    - Chaines de caractère, concaténation et reconnaissance de variables
    - Séquences, commentaires
  - Fonctionnalités centrales et natives de JavaScript:
    - Instructions, Opérateurs, Structures de contrôle
- Méthodes de travail
  - Structure des fichiers
  - Accès au document HTML depuis le JavaScript : lire et écrire dans l'interface
  - Lancement d'un programme sur un événement
- Variables: Noms, déclaration, visibilité (portée)
- Structure et Types de données dans les variables
- Fonctions
- Contrôle du flux d'instructions et gestion des erreurs
- Boucles et itération
- Opérateurs et Expressions
- Nombres et dates
- Les Collections (Arrays)
  - Collections indexées
  - Collections avec clés
- Formats de texte
- Les expressions rationnelles

#### Pré-requis

bonnes connaissances des langages HTML5/CSS3

#### Organisation

Durée : 30 jours

Horaire : 14.00 -18.00

Inscriptions : Entretien, CV



## Certification WebProgramer | JavaScript ECMA 6 - 2

### 2. Document, Événements, Animations, jQueryCore, Prog. Asynchrone (AJAX)

- Introduction au DOM
  - Concept
  - Conception de l'arborescence : les nœuds
  - Types de nœuds
  - Accès aux éléments et attributs du DOM
  - Propriétés de base d'un nœud
  - Attributs et propriétés personnalisées d'un nœud dans le DOM
  - Synchronisation des attributs et des propriétés
- Modification du contenu dans un document
- Accès aux feuilles de style (CSS) par le DOM
- Interagir avec l'interface utilisateur : fenêtre, navigateur et interface utilisateur
- Événements JavaScript
  - Programmation événementielle
  - Le Type Event : paramètres, propriétés
  - PreventDefault, stopPropagation
  - Programmation générique
  - Les événements
  - Types d'événements
  - Phase de capture et de propagation
- Animation JS, gestion du temps
  - setTimeout, setInterval
  - Manipulation du curseur, position
  - Animations JavaScript
  - Processus parallèles
  - Drag and drop
- Canvas
  - Dessiner par programmation
  - Canvas et animations dans Canvas
  - Traitement des pixels
  - Accès aux données du Canvas
- jQuery Core
  - L'objet jQuery
  - Manipulation d'objets et du DOM
  - Effets visuels
  - Les selecteurs
  - Manipulation de chaînes de caractères
  - Gestion des événements
- AJAX, programmation asynchrone
  - Le modèle AJAX
  - L'objet XMLHttpRequest et l'objet JSON (JavaScript Object Notation)
  - Les "promise" en JavaScript
- Le stockage local des données



## Certification WebProgramer | JavaScript ECMA 6 - 3

### 3. Modèle Objet JS ES6, Modules, Node.js

- Le modèle objet en JavaScript ES6
  - Le concept d'objet en JavaScript
  - Énumération des propriétés d'un objet
  - Créer de nouveaux objets
  - Déf. des Méthodes associées aux objets
  - Définir des accesseurs et des mutateurs (getters et setters)
  - Supprimer des propriétés
  - Les objets et les propriétés
  - Personnalisation des Objets
  - L'héritage des propriétés
  - Définir des méthodes
  - Comparer des objets
- Les Classes ES6
  - Langages de prototypes / Langages de classes
  - Fonctionnement et cycle de vie d'une Classe
  - Instanciation d'une Classe. Ajouter ou retirer des propriétés
  - Création d'une Classe de démarrage d'une application multipage
  - La création de la hiérarchie
  - Les constructeurs
  - L'héritage de propriétés : les subtilités
  - Manipuler une Classe à partir d'une autre Classe
- Les Modules
  - Définition du système des modules
  - Usage des fonctions comme des espaces de nom
  - Gestion des dépendances automatisées
  - Gestion des importations et d'exportations des modules
- Organisation et gestion du code
  - Conception de la structure de l'application
  - Organisation des modules
  - Gestion des Ressources diverses
- Introduction à Node.js
- Présentation de npm
- TaskRunner : automatisation des tâches (GULP)
- Détecter et interpréter les erreurs en provenance de Node.js.
- Compilateurs et Transpilateurs
  - Babel.js: méthodes d'utilisation, installation, pilotage depuis Gulp
  - Minification des fichiers (JS, CSS, etc) avec Gulp
  - Gestion des fichiers static et des ressources
  - Livereload & Browsersync.



## Certification WebProgramer | JavaScript ECMA 6 - 4

### 4. BBDD NoSQL (Firebase), DevOps et gestion de projet JS

- Back-end: Firebase
  - Base de Données Non-relationnelle, Firebase et modèle de BBDD NoSQL
  - Installation simple de Firebase dans un projet JS via
  - Firebase: coté serveur (BBDD) et coté-client (données)
  - Connexion local ou serveur et structure d'organisation des données
  - Synchronisation d'une collection de données objet
  - Sauvegarder des données dans une collection
  - Gérer les listes de données avec `firebase.database.ref()`
  - Les types d'événements Firebase Listener RealTime Database
  - Requêtes, filtrage et tri et requêtes spécifiques
  - Système d'authentification de l'utilisateur avec Firebase Authentication
  - Divers méthodes d'authentification disponibles
  - Fonctionnement de `firebase.auth()` et cycle de vie
  - Récupérer les informations de l'utilisateur connecté avec « `.currentUser` »
  - Gérer les listes de données avec `firebase.database.ref()`
  - Création de collection de données `firebase.database()` liée avec `firebase.auth()`.
  - Surveiller l'état d'authentification de l'utilisateur avec « `onAuthStateChanged()` »
  - Gestion des droits et de la sécurité
- Mise en pratique de Firebase dans un environnement de production
- Deployer une application JS (mise en production)
  - Environnement d'exécution.
  - Configuration des accès à une BDD distant du type Firebase.
  - Config. des adresses d'accès (URL base) des ressources et de l'application
  - Sécurisation basic de son code JS avec la minimisation.
  - Cacher les sources du code ES6.
  - Maintenance et mise à jour d'une application JS.
- Outils de développeur
  - CLI (Terminal)
  - Git et GitHub
- Méthodes de Gestion d'un Projet JS
  - Définition des objectifs
  - Lister fonctionnalités de l'app.
  - Mockup interface HTML/CSS
  - Modélisation de l'architecture
  - Créer un Prototype
- Initialisation d'un Projet JS
  - Analyse du Projet
  - Interface HTML/CSS
  - Méthode de travail
  - Développement Front et Back-End