

# JavaScript Moderne

Découvrez les dernières évolutions du langage JavaScript

1 jour / 7h

## Objectifs pédagogiques

- Comprendre comment on écrit du JavaScript moderne
- Maîtriser les principales nouveautés apparues depuis ECMAScript 6 (Classes, arrow functions, déstructuration, yield, async / await, ...)
- Appliquer des patterns modernes pour des applications JavaScript (CQRS, Redux, MVC, ...)

## Public cible

- Développeurs

## Prérequis

- Avoir déjà pratiqué un langage "C style" orienté objet pendant au moins 1 an (C++, C#, Java, PHP, JavaScript, etc.)

## Programme

### Module 1 : Introduction

- Historique
- TC39 et ECMAScript

## **Module 2 : Closures, variables et hoisting**

- Variables var / const / let
- Closures
- Template string
- Déclaration des objets et des tableaux
- Déstructuration et spread operator

Ateliers:

- Déclaration de variables
- Déclaration d'objets
- Déstructuration

## **Module 3 : Property Descriptor & Object enhancements**

- Property Descriptor
- preventExtensions vs freeze vs seal
- Fonctions de manipulation des objets
- Manipulation du this
- Arrow function

## **Module 4 : Asynchronicité**

- Promise
- Async / Await
- Event loop

Ateliers:

- Promise

## **Module 5 : Classes**

- Prototype
- Classes

Ateliers:

- Classe

## **Module 6 : Itérateur et générateur**

- Itérateur et Itérable
- Set / Map

- Générateur

Ateliers:

- Générateur

## **Module 7 : Symboles**

- Symboles

## **Module 8 : Proxy**

- Proxy

Ateliers:

- Proxy

## **Module 9 : Les modules**

- AMD vs CommonJS vs UMD vs ESMODULE
- Module loader et bundler

Ateliers:

- ESMODULE

## **Module 10 : Programmation fonctionnelle**

- Rappel sur les fonctions
- Fonctions pures
- High Order Function
- map/filter/reduce

Ateliers:

- Map/filter/reduce
- programmation fonctionnelle et pipe

## **Module 11 : Patterns**

- CQRS
- MVC
- Redux

Ateliers:

- Redux

## **Module 12 : Introduction à TypeScript**

Note : module optionnel si cette formation est prise conjointement avec la formation TypeScript

- C'est quoi TypeScript ?
- Interfaces et fonctions
- Generics
- Surcharges
- Types Avancés (union, interception, guards)

Ateliers:

- TypeScript