

# Modern JavaScript

Ducin IT  
Consulting

## ES2015 - ES2020

Ducin IT Consulting - Program szkolenia

Czas trwania: 2-3 dni

Formuła: 20% teoria, 80% praktyka

Szkolenie przeznaczone jest dla programistów mających doświadczenie z klasycznym JavaScript (ES5), chcących zacząć skutecznie używać nowych standardów języka, jak również dla osób rozpoczynających pracę w JavaScriptcie. Każda z części szkolenia jest najpierw omawiana teoretycznie, a następnie uczestnicy rozwiązują liczne ćwiczenia w stylu Test-Driven Development.

Ćwiczenia te badają zrozumienie danego zagadnienia oraz umiejętność samodzielnego rozwiązywania problemów. Kładą nacisk na dobre praktyki programistyczne. Są także „naszpikowane” klasycznymi pułapkami, w jakie programiści często wpadają – pod okiem trenera zdobywają „doświadczenie w boju”, mające na celu uchronienie przed popełnianiem błędów w projektach.

## Kluczowe Aspekty:

- Praktyczne zastosowanie ES6+ w codziennej pracy
- Porównanie mechanizmów ES6+ z odpowiednikami ES5 oraz innymi językami programowania
- Nauka w oparciu o TDD

## Program szkolenia:

### 1. Entry - Evolution of JS

- 1.1. browsers, compatibility
- 1.2. polyfills, transpiling
- 1.3. runtime environments
- 1.4. tc39, proposals system

### 2. Basics & Syntax

- 2.1. var, let & const keywords
- 2.2. Symbols
- 2.3. Destructuring Objects & Arrays
- 2.4. Spread and Rest Operators
- 2.5. Default Parameters
- 2.6. Template Literals
- 2.7. Tagged template literals

### 3. ES Updates

- 3.1. Nullish Coalescing
- 3.2. Optional Chaining
- 3.3. Class Private Members
- 3.4. Class Static Members
- 3.5. Array: flat, flatmap
- 3.6. Object: keys, values, entries, fromEntries

### 4. Arrow functions

- 4.1. Lexical Scope
- 4.2. ES5 functions vs ES6 arrow functions

- 4.3. Managing function context (this)
- 4.4. Advanced usage, common mistakes
- 4.5. Functional programming with Arrow Functions

## 5. Classes

- 5.1. Prototypal inheritance in ES6 classes
- 5.2. Class Properties
- 5.3. Static members, Private members
- 5.4. ES5 constructors, ES6 classes, TypeScript classes – comparison
- 5.5. OOP and FP – two approaches in JavaScript

## 6. Iteration Protocol

- 6.1. Iterators
- 6.2. for..of loops
- 6.3. Generators
- 6.4. Custom Iterators
- 6.5. Async Iterators

## 7. Modules

- 7.1. Imports, exports
- 7.2. ESM, CommonJS, AMD - comparison
- 7.3. Dynamic Imports

## 8. Promises

- 8.1. Asynchronicity in JS (event loop, message queue, ...)
- 8.2. Basic promise usage
- 8.3. Parallel and Sequential processing
- 8.4. Aggregate operations
- 8.5. Best practices and anti-patterns
- 8.6. ES6 promises vs jQuery, bluebird

## 9. Async Functions

9.1. Coroutines

9.2. `coa`, `bluebird.coroutine`

9.3. Async Await

9.4. Top-level await

## 10. Automation

10.1. Transpiling with Babel

10.2. Unit Testing with Karma and/or Jest