

Szkolenie: Compendium CE  
**JavaScript - architektura aplikacji webowych - zaawansowane programowanie**


FORMA SZKOLENIA	MATERIAŁY SZKOLENIOWE	CENA	CZAS TRWANIA
Stacjonarne	Tradycyjne	2800 PLN NETTO*	3 dni
Stacjonarne	Cyfrowe	2800 PLN NETTO*	3 dni
Stacjonarne	Tablet CTAB	3400 PLN NETTO*	3 dni
Metoda dlearning	Tradycyjne	2800 PLN NETTO*	3 dni
Metoda dlearning	Cyfrowe	2800 PLN NETTO*	3 dni
Metoda dlearning	Tablet CTAB	2800 PLN NETTO*	3 dni

\* (+VAT zgodnie z obowiązującą stawką w dniu wystawienia faktury)

## LOKALIZACJE

Kraków - ul. Tatarska 5, II piętro, godz. 9:00 - 16:00

Warszawa - ul. Bielska 17, godz. 9:00 - 16:00

## DOSTĘPNE TERMINY

2019-09-16 | 3 dni | Warszawa

## Cel szkolenia:

Celem szkolenia jest omówienie zaawansowanych koncepcji programistycznych w języku **JavaScript**, stosowanych przy tworzeniu **aplikacji webowych** (Single Page App).

Omówione zostaną paradygmaty programowania obiektowego i funkcyjnego, przydatne w tworzeniu aplikacji wzorce projektowe, wzorce i biblioteki MV\* tworzone w JavaScript, a także "best practices" i "pułapki" związane z programowaniem w JavaScript.

Ponadto przedstawiony będzie nowoczesny "warsztat" programisty **JavaScript**, z uwzględnieniem zaawansowanych narzędzi developerskich dostępnych w przeglądarkach, testowaniem poprawności kodu oraz narzędziami typu "build tool".

## Plan szkolenia:

- Programowanie obiektowe w JS
  - Przypomnienie wiadomości
  - Obiekty

- Własności
- Metody
- Konstruktory
- Dziedziczenie
- Polimorfizm
- Zakresy (scope) zmiennych i funkcji
- Programowanie funkcyjne (funkcjonalne) w JS
  - Podstawowe funkcje:
    - each
    - map
    - fold (reduce)
    - partial (curry)
    - compose
    - filter
    - every / some
  - Tworzenie programów za pomocą dostępnej biblioteki funkcji
- Wzorce projektowe używane w JS
  - Czym są wzorce projektowe?
  - Elementy wzorca
  - Wzorce:
    - Moduł (module)
    - Obserwator (observer)
    - PubSub (publisher / subscriber)
    - Konstruktor (constructor)
    - Mediator
    - Fasada (facade)
    - Polecenie (command)
    - Mixin
    - Obietnice (promises / deferreds)
- Wzorce MV\*
  - Wprowadzenie - Smalltalk MVC
  - Wzorzec MVC
    - Model
    - View
    - Controller
  - Wzorzec MVP

- Model
- View
- Presenter
- Wzorzec MVVM
  - Model
  - View
  - ViewModel
- "Stabilne" biblioteki MV\*
  - Backbone
  - Knockout.js
- Modularyzacja aplikacji JS
  - Standard AMD
    - require.js
  - Standard CommonJS
  - Przyszłość: ECMAScript Harmony
- Narzędzia programistyczne
  - Firebug
  - Chrome Developer Tools
    - Konsola (console)
    - Podgląd DOM (elements)
    - Zasoby (resources)
    - Sieć (network)
    - Źródła (sources)
    - Linia czasu (timeline)
    - Profilowanie (profiler)
    - Audyty (audits)
  - js hint / js lint
- Testowanie kodu
  - testy jednostkowe qUnit
    - asercje
    - testy
    - moduły
- Profilowanie i optymalizacja kodu
  - silnik V8
  - garbage collector
  - dereferencja

- korzystanie z closures
- hoisting
- unikanie reflow
- walka z jank
- Zaawansowana architektura aplikacji JS
  - czym są “duże aplikacje”?
  - założenia architektoniczne
  - problemy “dużych aplikacji”
  - decoupling
  - fasadowanie bibliotek
  - mediator

## Wymagania:

- znajomość **JavaScript** na poziomie co najmniej dobrym
- dobra znajomość **HTML**
- podstawowa znajomość **CSS**
- dobra znajomość podstawowych technik programowania (**zmienne, typy, pętle, kontrola przepływu, funkcje**)

## Rekomendowane szkolenia:

- [CIW: Site Development Associate](#)
- [CIW: JavaScript Specialist](#)

## Poziom trudności



## Certyfikaty:

Uczestnicy szkolenia otrzymują **certyfikat** sygnowany przez **Compendium Centrum Edukacyjne**.

## Prowadzący:

Wykładowca Compendium Centrum Edukacyjnego.