

## **Szkolenia:**

<b>2020 - rok JavaScriptu (Angular, react, vue)</b>	<b>1</b>
<b>JavaScript: kurs podstawowy (dla programistów)</b>	<b>2</b>
<b>JavaScript: kurs zaawansowany</b>	<b>4</b>
<b>TypeScript</b>	<b>6</b>
<b>Angular: kurs podstawowy</b>	<b>8</b>
<b>Angular: kurs zaawansowany</b>	<b>10</b>
<b>Angular: kompedium wiedzy (TypeScript, RxJS)</b>	<b>12</b>
<b>React.js</b>	<b>14</b>
<b>Vue.js jako nowoczesne narzędzie do budowania aplikacji webowych</b>	<b>16</b>
<b>WebWorkers i Progressive Web Apps</b>	<b>17</b>
<b>Node.js: kurs podstawowy (dla programistów front-end)</b>	<b>18</b>
<b>Object-oriented programming w JavaScript</b>	<b>20</b>
<b>JavaScript: Programowanie funkcyjne i reaktywne</b>	<b>22</b>
<b>Architektura aplikacji frontendowych</b>	<b>24</b>
<b>ES6+</b>	<b>26</b>
<b>Testy automatyczne aplikacji webowych dla programistów</b>	<b>28</b>
<b>Wstęp do programowania z JavaScript</b>	<b>30</b>
<b>Wykorzystywanie kodu Java, Rust, C++ w przeglądarce - wprowadzenie do WebAssembly</b>	<b>31</b>
<b>JavaScript dla programistów Java</b>	<b>32</b>
<b>Progresywne przepisywanie aplikacji, zmniejszanie długu technologicznego i zasady clean code</b>	<b>33</b>
<b>GraphQL</b>	<b>34</b>
<b>REST - od nowicjusza do geniusza</b>	<b>35</b>

<b>regExp - wyrażenia regularne, zostań mistrzem!</b>	<b>36</b>
<b>Business Intelligence, wizualizacja big data w aplikacjach internetowych</b>	<b>37</b>
<b>Sieci neuronowe i uczenie maszynowe z tensorflow.js i brain.js</b>	<b>39</b>
<b>Prezentowanie treści technicznych</b>	<b>40</b>
<b>Komunikacja z developmentem (dla biznesu)</b>	<b>41</b>

# 2020 - rok JavaScriptu (Angular, react, vue)

## Opis szkolenia:

W czasie szkolenia uczestnicy będą mieli szansę dowiedzieć się jak działają obecnie najpopularniejsze biblioteki/frameworki na rynku aplikacji front-endowych. Skupimy się na ich wadach, zaletach, podobieństwach i różnicach tak aby uczestnicy mogli podejmować świadome decyzje dotyczące wyboru technologii do projektu.

## Profil uczestników:

Szkolenie adresowane jest do programistów znających język JavaScript

## Czas trwania:

3 dni

## Wiedza teoretyczna i praktyczna:

- Architektura komponentowa
- Narzędzia do zarządzania stanem aplikacji
- virtual DOM
- Komunikacja komponentów
- server-side rendering
- TypeScript vs JavaScript
- MVVM vs Flux
- Hooks vs Redux
- Reaktywne updaty

## Umiejętności:

Po ukończonym kursie uczestnicy będą znali zaawansowane mechanizmy frameworka. Będą wiedzieli jak optymalizować swoje aplikacje, jak pisać czysty, wydajny kod i jak tworzyć przejrzyste architektury angularowych aplikacji.

# JavaScript: kurs podstawowy (dla programistów)

## Opis szkolenia:

Celem szkolenia jest przekazanie wiedzy na temat mechanizmów działania języka JavaScript. Uczestnicy nauczą się budować aplikacje w oparciu o najnowsze standardy oraz poznają funkcyjne i obiektowe możliwości języka. Dostaną również szereg informacji na temat tego w jaki sposób pisać skalowalny kod i jak używać współczesnych wzorców projektowych takich jak MVC, MVVM, Flux. Dzięki przekazanej wiedzy uczestnicy będą rozumieć działanie języka co pozwoli na szybki dalszy rozwój oraz unikanie najczęstszych błędów przy tworzeniu aplikacji internetowych.

## Profil uczestników:

Szkolenie przeznaczone jest dla osób, znających dowolny język programowania. Dla tych, którzy dopiero zaczynają swoją przygodę z programowaniem sporządzony został kurs - Wstęp do programowania z JavaScript

## Czas trwania:

3 dni

## Wiedza teoretyczna i praktyczna:

- Standardy ECMAScript
- Typy zmiennych
- Zasięg zmiennych
- Hoisting
- Słabe typowanie i automatyczne rzutowanie
- Sterowanie przepływem programu
- Kontrolowanie czasu
- Mechanizmy funkcyjne i obiektowe języka JavaScript
- Kontekst (this)
- Łańcuchy prototypów
- Asynchroniczność (Callback, Promise, async/await)

- ECMAScript Modules
- Kolekcje
- Zdarzenia w JavaScript (Event loop)
- Komunikacja z Document Object Model (DOM)
- Walidacja formularzy
- Komunikacja z serwerem - protokół REST
- Architektury aplikacji frontendowych (MV\*, flux)
- Debugowanie, użycie DevTools
- Obsługa wyjątków

### Umiejętności:

Po zakończonym szkoleniu uczestnicy będą znać podstawy tworzenia aplikacji internetowych. Będą znali różne podejścia programistyczne i będą wiedzieć w jaki sposób będą mogli w prosty sposób rozwijać się dalej w wybranym kierunku.

# JavaScript: kurs zaawansowany

## Opis szkolenia:

Celem szkolenia jest przekazanie informacji na temat zaawansowanych mechanizmów działania języka JavaScript. Uczestnicy dowiedzą się w jaki sposób mogą tworzyć Web Components bez wsparcia żadnego frameworka, poznają sposób działania nowoczesnych featurów języka takich jak np. generatory, iterables, proxies, pogłębią informacje na temat mechanizmów takich jak prototypy, zmiana kontekstu jak również dostaną szereg wiadomości na temat asynchroniczności, funkcyjnego podejścia i silników przeglądarek. Nabyta wiedza pozwoli pisać bardziej ekspresywny, czysty i wydajny kod.

## Profil uczestników:

Szkolenie adresowane jest do programistów, którzy znają podstawy języka JavaScript - posiadają wiedzę na poziomie ukończonego szkolenia JavaScript kurs podstawowy (dla programistów).

## Czas trwania:

3 dni

## Wiedza teoretyczna i praktyczna:

- Zaawansowane funkcyjne mechanizmy języka
- higher-order functions
- currying
- partially applied functions
- transductions
- Zaawansowane definiowane propertiesów
- Gettery
- Settery
- Propertiesy niemutowalne
- Prototypy w szczegółach
- Mechanizmy ES6 w szczegółach
- Destructuring i deep destructuring
- Rest/spread operator
- Symbols

- Generators
- Iterables
- Async/await
- ES6 Promises
- String templates tags
- Proxy
- Reflect
- Web components
- Słniki JavaScript oraz alokowanie pamięci
- Wzorce architektoniczne
- MV\* (MVC, MVVM, MVP)
- MVI
- Flux (Redux, MobX)

#### Umiejętności:

Po ukończonym szkoleniu uczestnicy będą znać szczegółowo zasady działania prototypów oraz innych nisko poziomowych mechanizmów w JavaScriptcie, umieć wykorzystać najnowocześniejsze featury języka oraz rozumieć nowoczesne wzorce architektoniczne.

# TypeScript

## Opis szkolenia:

Celem szkolenia jest przekazanie wiedzy na temat języka TypeScript. Uczestnicy dowiedzą się w jaki sposób mogą polepszyć jakość kodu dodając do niego typy, również w postaci skomplikowanych struktur. Nauczą się również korzystać featurów, które udostępnia TypeScript - nauczą się np. w jaki sposób pisać dekoratory klas, metod oraz pól. Dzięki temu szkoleniu Dzięki temu szkoleniu uczestnicy będą wiedzieć w jaki sposób zaadoptować techniki znane z klasycznych języków programowania takich jak Java do pisania aplikacji frontendowych.

## Profil uczestników:

Szkolenie adresowane jest do programistów, którzy znają podstawy języka JavaScript - posiadają wiedzę na poziomie ukończonego szkolenia JavaScript kurs podstawowy (dla programistów).

## Czas trwania:

1 dzień

## Wiedza teoretyczna i praktyczna:

- Silne typowanie i słabe typowanie
- Typy podstawowe
- Typy złożone
- Typowanie funkcji
- Interfejsy
- Klasy
- Klasy abstrakcyjne
- Dziedziczenie
- Enkapsulacja
- Dekoratory
- Generic types
- Intersection types
- Overloading
- Enum



- Literal types
- Type guards
- Non-nullable-types
- Record
- Partial / Required
- Pick / Omit
- Union
- Discriminated Union
- keyof and Lookup Types
- Mapped types
- Conditional types

#### Umiejętności:

Po zakończonym szkoleniu uczestnicy będą swobodnie posługiwać się językiem TypeScript, będą wiedzieli w jaki sposób można pisać aplikacje frontendowe używając praktyk znanych z klasycznych języków obiektowych takich jak Java.

# Angular: kurs podstawowy

## Opis szkolenia:

Celem szkolenia jest nauczenie uczestników w jaki sposób budować aplikacje w oparciu o framework Angular. Uczestnicy poznają mechanizmy frameworka co pozwoli im na samodzielną pracę w tym środowisku. Dodatkowo poznają struktury architektoniczne, które mogą zostać zaimplementowane w angularowych aplikacjach.

## Profil uczestników:

Szkolenie adresowane jest do programistów znających język JavaScript.

## Czas trwania:

3 dni

## Wiedza teoretyczna i praktyczna:

- Component-based architecture
- Składowe języka TypeScript
- Angular-cli
- Komponenty
- Shadow DOM
- Dekoratory komponentów
- Dyrektywy
- Dyrektywy wbudowane (NgIf, NgSwitch, NgStyle, NgClass, NgFor, NgNonBindable)
- Modele
- Pipes
- Tworzenie własnych dyrektyw i pipes
- Cykl życia komponentu
- Dependency Injection
- Podstawy Change Detection i zone.js
- Formularze (Template driven oraz Model driven)
- Event Emitters
- Podstawy pracy z Observables i Rx.js

- Komunikacja z serwerem
- Routing
- Tworzenie aplikacji typu Single Page Application
- Testowanie komponentów (unit-tests)
- Wstęp do wzorców architektonicznych
- MV\* (MVC, MVVM, MVP)
- MVI
- Flux (ngrx)

#### Umiejętności:

Po ukończonym szkoleniu uczestnicy będą potrafili tworzyć aplikacje używając frameworka Angular. Będą posiadali informacje na temat możliwych architektur aplikacji oraz głównych mechanizmów działania frameworka.

# Angular: kurs zaawansowany

## Opis szkolenia:

Celem szkolenia jest przekazanie informacji na temat zaawansowanych mechanizmów działania frameworka Angular. Uczestnicy dowiedzą się w jaki sposób tworzyć m.in. własne dyrektywy, dynamiczne komponenty, komponenty reaktywne, zagnieżdżony routing, dostaną szereg informacji na temat performance'u i możliwych memory leak'ów oraz dowiedzą się w jaki sposób mogą użyć Angulara razem z nowoczesnymi podejściami do zarządzania stanem takimi jak Flux czy Model-View-Intent. Uczestnicy poznają także w szczególności zasadę pracy z Observables i nauczą się operować na strumieniach danych.

## Profil uczestników:

Szkolenie adresowane jest do programistów znających podstawy frameworka Angular.

## Czas trwania:

3 dni

## Wiedza teoretyczna i praktyczna:

- Lazy loading
- Dynamiczne tworzenie komponentów
- Tworzenie bardziej skomplikowanych dyrektyw
- Element Host
- @ViewChild and @ViewChildren
- Hot reload
- Routing zagnieżdżony
- Routing guards
- Animacje
- Dzielenie angularowej aplikacji na moduły
- Incremental DOM
- Reaktywne komponenty
- Strategie change detection w szczególności
- Performance

- AiT i JiT
- End-to-end tests
- Angular Material Design
- Implementacja wzorców architektonicznych w angularze
- MV\* (MVC, MVVM, MVP)
- Model-View-Intent

### Umiejętności:

Po ukończonym kursie uczestnicy będą znali zaawansowane mechanizmy frameworka. Będą wiedzieli jak optymalizować swoje aplikacje, jak pisać czysty, wydajny kod i jak tworzyć przejrzyste architektury angularowych aplikacji.

# Angular: kompedium wiedzy (TypeScript, RxJS)

## Opis szkolenia:

Celem szkolenia jest nauczenie uczestników w jaki sposób budować aplikacje w oparciu o framework Angular. Uczestnicy poznają mechanizmy frameworka co pozwoli im na samodzielną pracę w tym środowisku. Dodatkowo poznają struktury architektoniczne, które mogą zostać zaimplementowane w angularowych aplikacjach.

## Profil uczestników:

Szkolenie adresowane jest do programistów znających język JavaScript, chcących się nauczyć Reacta od podstaw.

## Czas trwania:

5 dni

## Wiedza teoretyczna i praktyczna:

- Component-based architecture
- Składowe języka TypeScript
- Angular-cli
- Komponenty
- Shadow DOM
- Dekoratory komponentów
- Dyrektywy
- Dyrektywy wbudowane (NgIf, NgSwitch, NgStyle, NgClass, NgFor, NgNonBindable)
- Modele
- Pipes
- Tworzenie własnych dyrektyw i pipes
- Cykl życia komponentu
- Dependency Injection
- Podstawy Change Detection i zone.js

- Formularze (Template driven oraz Model driven)
- Event Emitters
- Podstawy pracy z Observables i Rx.js
- Komunikacja z serwerem
- Routing
- Tworzenie aplikacji typu Single Page Application
- Testowanie komponentów (unit-tests)
- Wstęp do wzorców architektonicznych
- MV\* (MVC, MVVM, MVP)
- MVI
- Flux (ngrx)
- Lazy loading
- Dynamiczne tworzenie komponentów
- Tworzenie bardziej skomplikowanych dyrektyw
- Element Host
- @ViewChild and @ViewChildren
- Hot reload
- Routing zagnieżdżony
- Routing guards
- Animacje
- Dzielenie angularowej aplikacji na moduły
- Incremental DOM
- Reaktywne komponenty
- Strategie change detection w szczegółach
- Performance
- AOT i JIT
- End-to-end tests
- Angular Material Design
- Implementacja wzorców architektonicznych w angularze
- MV\* (MVC, MVVM, MVP)
- Model-View-Intent

### Umiejętności:

Po ukończonym kursie uczestnicy będą znali zaawansowane mechanizmy frameworka. Będą wiedzieli jak optymalizować swoje aplikacje, jak pisać czysty, wydajny kod i jak tworzyć przejrzyste architektury angularowych aplikacji.

# React.js

## Opis szkolenia:

Celem szkolenia jest nauczanie od podstaw biblioteki React.js oraz praktycznego jej wykorzystania do tworzenia nowoczesnych aplikacji internetowych. Uczestnicy dowiedzą się w jaki sposób dzielić aplikacje na komponenty oraz na czym polegają współczesne architektury do zarządzania stanem aplikacji - takie jak React hooks, Redux czy MobX i czym różnią się od klasycznych podejść typu Model-View-Controller. Po ukończonym szkoleniu uczestnicy będą w stanie samodzielnie pisać aplikacje wykorzystujące bibliotekę React.

## Profil uczestników:

Szkolenie adresowane jest do programistów znających język JavaScript, chcących się nauczyć Reacta od podstaw.

## Czas trwania:

2 dni

## Wiedza teoretyczna i praktyczna:

- Component-based architecture
- Podstawy ES6
- Podstawy funkcyjnego programowania w JS
- VirtualDOM
- JSX
- Komponenty
- Stylowanie komponentów
- React hooks
- Cykl życia komponentu
- Komponenty stanowe i bezstanowe
- Jednokierunkowy przepływ danych
- Renderowanie warunkowe
- Obsługa zdarzeń
- Formularze
- Komunikacja z serwerem



- Routing
- Architektura typu Flux (Redux, mobX)
- Niemutowalne struktury danych
- Server-side rendering
- React saga
- Wstęp do GraphQL

### Umiejętności:

Po ukończonym szkoleniu uczestnicy będą znać zasady działania biblioteki React i będą wiedzieć jak użyć tej biblioteki razem z architekturami Fluxowymi takimi jak Redux oraz MobX. oraz z React hooks Zdobędą informacje na temat dobrych praktyk, podejścia funkcyjnego oraz tworzenia przejrzystych struktur aplikacji reactowych.

# Vue.js jako nowoczesne narzędzie do budowania aplikacji webowych

## Opis szkolenia:

W czasie szkolenia uczestnicy zdobędą szereg informacji na temat biblioteki Vue oraz nauczą się w jaki sposób poprawnie zarządzać stanem aplikacji używając takich narzędzi jak Vuex i Rxjs

## Profil uczestników:

Szkolenie adresowane jest dla osób, które znają podstawy języka JavaScript

## Czas trwania:

3 dni

## Wiedza teoretyczna i praktyczna:

- Architektura komponentowa
- Interpolacja zmiennych
- Obsługa formularzy
- computed props
- watchers
- Vuex i zarządzanie stanem aplikacji
- Cykl życia komponentu
- Kompozycja komponentów
- Jednokierunkowy przepływ danych
- Routing
- Komunikacja z serwerem
- Routing

## Umiejętności:

undefined

# WebWorkers i Progressive Web Apps

## Opis szkolenia:

Celem szkolenia jest przekazać wiedzę na temat technologii związanych z progressive web applications. Uczestnicy dowiedzą się w jaki sposób budować skalowalne aplikacje, które z poziomu użytkownika będą wyglądały jak aplikacje natywne

## Profil uczestników:

Szkolenie adresowane jest dla osób, które znają JS w stopniu średniozaawansowanym

## Czas trwania:

2 dni

## Wiedza teoretyczna i praktyczna:

- Mechanika nowoczesnych systemów JavaScriptowych
- WebWorkers w szczegółach
- Service Worker
- Progressive enhancement
- Manifest JSON
- Nieblokujące operacje
- Integracja z systemem operacyjnym

## Umiejętności:

undefined

# Node.js: kurs podstawowy (dla programistów front-end)

## Opis szkolenia:

Celem szkolenia jest przekazanie wiedzy osobom, które do tej pory pracowały tylko we frontendzie, na temat tego w jaki sposób język programowania z którym mają do czynienia na co dzień (JavaScript) może zostać użyty na serwerze. Uczestnicy poznają środowisko, rozumieją jego mechanikę oraz nauczą się pisać proste aplikacje po stronie serwera. Dzięki praktycznym ćwiczeniom nauczą się pracy z bazami danych oraz zarządzania kolejkami asynchronicznych funkcji.

## Profil uczestników:

Szkolenie adresowane jest do programistów aplikacji frontendowych nie mających doświadczenia w tworzeniu aplikacji backendowych.

## Czas trwania:

2 dni

## Wiedza teoretyczna i praktyczna:

- JavaScript po stronie serwera
- Eventy i event-driven-development
- Asynchroniczność
- Routing
- Node API
- Security
- Buffers
- Error Handling
- Express.js
- Połączenie z bazami typu NoSQL
- Połączenie z bazami SQL
- Testowanie
- Node + RxJS

### Umiejętności:

Po skończonym szkoleniu uczestnicy będą znali założenia środowiska Node.js, jego architekturę oraz nauczą się tworzenia prostych programów komunikujących się z bazami danych.

# Object-oriented programming w JavaScript

## Opis szkolenia:

Celem szkolenia jest przekazanie informacji na temat mechanizmów języka JavaScript pozwalających na tworzenie skalowalnego i wydajnego oprogramowania stosując techniki OOP.

## Profil uczestników:

Szkolenie adresowane jest do programistów, którzy znają podstawy języka JavaScript - posiadają wiedzę na poziomie ukończonego szkolenia JavaScript kurs podstawowy (dla programistów).

## Czas trwania:

2 dni

## Wiedza teoretyczna i praktyczna:

- Difference between OOP and Functional Programming
- OOP and FP real word examples
- Exploring the OOP concepts
- Types of JavaScript functions
- Object creation - object literal, Object.create(), constructor function, class
- Factory pattern
- typeof vs instanceof, duck typing
- 'This' in JavaScript
- Binding the context
- Prototypal chain
- prototype and Object.setPrototypeOf()
- Function constructors and prototypes
- ES6 classes and prototypes
- Closures and module pattern
- Encapsulation and classes
- Methods, static methods and polymorphism

- Composition over inheritance
- Dependency Injection
- SOILD
- Calling constructor functions (constructor composition pattern)
- The hidden advantage of class methods
- ES6 class mixins (dynamic inheritance)
- new.target and abstract class pattern
- Thinking in an OOP way
- Proxy, Reflect
- property descriptors

### Umiejętności:

Po ukończonym szkoleniu uczestnicy będą znać szczegółowo zasady działania mechanizmów pomagających w tworzeniu programów w JavaScript w podejściu obiekowym

# JavaScript: Programowanie funkcyjne i reaktywne

## Opis szkolenia:

Paradygmaty programowania reaktywnego i funkcyjnego są coraz częściej, używane w językach programowania. Strumienie, czyste" + " funkcje, niemutowalność stanu, składanie funkcji, funkcje wyższego rzędu, monady.... Zrozumienie podejścia" + " funkcyjnego daje programiście dużo większe możliwości, a kod zaimplementowany w ten sposób jest" + " zdecydowanie bardziej ekspresywny, skalowalny i łatwiejszy do testowania."

## Profil uczestników:

Szkolenie adresowane jest do programistów, którzy znają przynajmniej podstawy języka Java lub JavaScript

## Czas trwania:

2 dni

## Wiedza teoretyczna i praktyczna:

- Programowanie funkcyjne i obiektowe
- Programowanie imperatywne i deklaratywne
- Czyste funkcje
- Funktory i monady
- Niemutowalne struktury danych
- callback
- memoization
- higher-order functions
- currying
- partially applied functions
- Transductions
- Reactive programming
- Data streams
- Rx operators
- Creating operators



- Composing streams
- Observables and observers
- Subjects
- Modules
- Model-Viewer-Intent architecture
- Testing
- Practical exercises

### Umiejętności:

Po ukończonym szkoleniu uczestnicy będą znać szczegółowo zasady podejścia reaktywnego i funkcyjnego.

# Architektura aplikacji frontendowych

## Opis szkolenia:

Celem szkolenia jest nauczenie uczestników w jaki sposób tworzyć skalowalne architektury aplikacji oraz jak wyglądają nowoczesne narzędzia JS umożliwiające tworzenie i utrzymywanie dużych systemów frontendowych.

## Profil uczestników:

Szkolenie adresowane jest do programistów znających język JavaScript.

## Czas trwania:

3 dni

## Wiedza teoretyczna i praktyczna:

- Podejście obiektowe i funkcyjne
- Paradygmaty programowania: imperatywny, deklaratywny, reaktywny
- Asynchroniczność:
- Events and custom events
- Generatory
- Async/await
- Strumienie
- ES6 Observables
- Communicating sequential processes (CSP)
- Domain Driven Design
- Test-driven development
- Modele architektoniczne:
- Model-view-controller
- Model-view-viewmodel
- Model-view-presenter
- Model-view-intent
- CQRS
- Flux

- Redux
- Static analysis tools
- Architektura informacji
- Podstawy UX
- Accessibility
- Semantic Web
- Debugowanie
- Developer tools
- Optymalizacja
- Critical render path
- Progressive Web App
- Web workers
- Service workers

#### Umiejętności:

Po ukończonym szkoleniu uczestnicy będą znali zagadnienia związane z architekturą aplikacji. Pozną również szereg narzędzi, które pozwalają na dużo łatwiejsze pisanie programów frontendowych

# ES6+

## Opis szkolenia:

Celem szkolenia jest przekazanie wiedzy na temat nowych featurów języka JavaScript, które zostały zaimplementowane w standardach EcmaScript nowszych niż EcmaScript 5. Uczestnicy nauczą się podstawowych mechanizmów takich jak zmienne blokowe jak również tych bardziej skomplikowanych takich jak iteratory, generatory czy metaprogramowanie przy użyciu proxy

## Profil uczestników:

Szkolenie adresowane jest do programistów znających język JavaScript w wersji EcmaScript 5 chcących poznać możliwości języka, które zostały dodane w standardach EcmaScript 2015 i kolejnych (czyli ES6+).

## Czas trwania:

8 dni

## Wiedza teoretyczna i praktyczna:

- Zmienne z zasięgiem blokowym
- Arrow functions
- Nowe funkcje Array i Array.prototype
- Destructuring i deep destructuring
- enhanced object literal
- Object.assign()
- Rest/spread operator
- default parameters
- Map
- Set
- WeakMap
- WeakSet
- Symbols
- Generators
- Iterables

- String templates
- String templates tags
- Classes
- Polymorphism
- EcmaScript Modules
- Proxy
- Reflect
- Asynchroniczne generatory
- Async for of loop

### Umiejętności:

Po zakończonym szkoleniu uczestnicy będą sprawnie posługiwać się nowymi featurami języka JavaScript.

# Testy automatyczne aplikacji webowych dla programistów

## Opis szkolenia:

Celem szkolenia jest nauczenie uczestników czym jest testowalny kod, jak pisać testy jednostkowe oraz testy end-to-end. Uczestnicy dowiedzą się również czym jest podejście typu Test-Driven Development (TDD), Behaviour Driven Development (BDD) oraz jakie są wady i zalety podejść tego typu.

## Profil uczestników:

Podstawowa wiedza z zakresu programowania.

## Czas trwania:

2 dni

## Wiedza teoretyczna i praktyczna:

- Jak pisać testowalny kod
- Różne typy testów - unit tests, integration test, system tests, acceptance tests
- Różnica w podejściach typu TDD oraz BDD
- Zasada Arrange-Act-Assert
- Tworzenie testów w Jasmine
- tworzenie suite's za pomocą describe i it
- używanie wbudowanych matcher'ów i asymetrycznych matcher'ów
- negacje matcher'ów
- Arrange-Act-Assert w Jasmine
- słowo kluczowe this w suite
- ręczne wywoływanie niepowodzeń
- disabled suites
- pending suites
- function spies
- mockowanie funkcji
- mockowanie czasu i daty

- testowanie asynchronicznego kodu za pomocą callbacków, promisów, async/await
- mockowanie wywołań eventów
- jasmine-ajax i mockowanie odpowiedzi z serwera
- custom matchers
- custom equality
- custom asymmetric equality
- inne narzędzia jasmine (reporters, custom boot)
- pisanie czystego kodu w czasie tworzenia testów
- Selenium
- WebDriverJS
- Testy end-to-end z wykorzystaniem frameworka Protractor:
- Setup środowiska
- Locators
- Page Object
- Debugging
- Skąd czerpać dalszą wiedzę

### Umiejętności:

Po ukończonym szkoleniu uczestnicy będą posiadać wiedzę na temat pisania testów jednostkowych oraz testów typu end-to-end. Będą wiedzieć jak pisać testowalny kod co zwiększy jego przejrzystość i jakość.

# Wstęp do programowania z JavaScript

## Opis szkolenia:

Celem szkolenia jest nauczenie uczestników podstaw programowania w języku JavaScript. Jest to kurs dla osób, które dopiero chcą zacząć swoją przygodę z programowaniem. Uczestnicy poznają podstawy programowania takie jak instrukcje warunkowe, zmienne czy pętle, zrozumieją na czym polegają różne podejścia programistyczne i dostaną szereg narzędzi pozwalających na późniejszy samodzielny rozwój.

## Profil uczestników:

Szkolenie adresowane jest dla osób, które nie potrafią jeszcze programować w żadnym języku.

## Czas trwania:

3 dni

## Wiedza teoretyczna i praktyczna:

- Czym jest program komputerowy
- Podział front-end i back-end
- Podstawy HTML
- Podstawy CSS
- Czym są poziomy abstrakcji w programowaniu
- Zmienne i typy zmiennych
- Czym są obiekty i jak ich używać
- Jak używać JavaScript do interakcji ze stroną internetową
- Komunikacja z serwerem
- Skąd czerpać wiedzę i jak się dalej rozwijać

## Umiejętności:

undefined



# Wykorzystywanie kodu Java, Rust, C++ w przeglądarce - wprowadzenie do WebAssembly

## Opis szkolenia:

W czasie szkolenia uczestnicy zdobędą szereg informacji na temat języka webassembly. Webassembly pozwala na użycie kodu źródłowego napisanego w innych językach programowania niż JavaScript, w aplikacjach front-endowych, co daje olbrzymie możliwości

## Profil uczestników:

Szkolenie adresowane jest dla osób, które znają podstawy języka JavaScript oraz dowolnego innego języka programowania

## Czas trwania:

2 dni

## Wiedza teoretyczna i praktyczna:

- Czym jest WebAssembly
- WebAssembly API
- Wydajność
- Wywoływanie funkcji cpp z poziomu JS
- Eksportowanie funkcji cpp
- Wywoływanie funkcji cpp w JS
- Integracja z Rust
- Integracja kodu Java z JWebAssembly
- Emscripten
- Security

## Umiejętności:

undefined

# JavaScript dla programistów Java

## Opis szkolenia:

Cytując klasyka - 'Java is to JavaScript as Car is to Carpet'. Jest to szkolenie profilowane, dla osób ze świata Javy, które chciałby się dowiedzieć w jaki sposób działa JavaScript i jak w nim wygodnie pisać. W czasie tego szkolenia uczestnicy dostaną solidną dawkę wiedzy na temat mechaniki JavaScriptu, dzięki czemu język ten przestanie być (aż tak) upierdliwy.

## Profil uczestników:

Szkolenie adresowane jest do programistów znających język Java.

## Czas trwania:

3 dni

## Wiedza teoretyczna i praktyczna:

- Porównanie podejścia OOP i podejść wykorzystywanych w JS
- Porównanie narzędzi Javowych i JavaScriptowych
- Jak nie popełniać typowych błędów ludzi ze świata OOP
- W jaki sposób wykorzystywać elastyczność języka JavaScript
- Wstęp do TypeScript
- Mechanika JS w szczegółach
- Kilka słów nt. node
- Kilka słów nt. JSowych bibliotek i frameworków (Angular, Vue, React)

## Umiejętności:

Po ukończonym szkoleniu uczestnicy będą się czuć znacznie pewniej w kodzie JavaScript, będą znali jego mechanikę i będą wiedzieć w jaki sposób unikać typowych błędów

# Progresywne przepisywanie aplikacji, zmniejszanie długu technologicznego i zasady clean code

## Opis szkolenia:

Celem szkolenia jest nauczenie uczestników w jaki sposób zabrać się do migracji legacy aplikacji. Szkolenie dotyczyć będzie również zasady czystego kodu, które pozwolą tworzyć aplikacje o znacznie wyższej jakości. W czasie zajęć poruszane będą również tematy dotyczące długu technologicznego oraz techniki na jego minimalizację.

## Profil uczestników:

Od uczestników szkolenia wymagana jest umiejętność programowania w dowolnym języku.

## Czas trwania:

2 dni

## Wiedza teoretyczna i praktyczna:

- Czym jest legacy code
- W jaki sposób wykstrahować zależności, które są legacy
- Po czym poznać, że aplikacja musi zostać przepisana
- Na czym polega skalowalność aplikacji
- W jaki sposób zabrać się za przepisywanie aplikacji
- Zabezpieczenia przed regresjami
- Zasady clean code
- Język dziedzinowy
- Metaprogramowanie
- Wzorce architektoniczne

## Umiejętności:

Uczestnicy po ukończonym szkoleniu będą znać zasady pomagające w łatwiejszym migrowaniu aplikacji.

# GraphQL

## Opis szkolenia:

Celem szkolenia jest przekazanie wiedzy dotyczącej języka zapytań GraphQL będącego alternatywą dla protokołu REST. Uczestnicy po szkoleniu będą w stanie użyć tej technologii w swoich projektach. Szkolenie będzie dotyczyło implementacji GraphQL w języku JavaScript, niemniej jednak sama technologia może być również użyta z innymi językami takimi jak Java, Go i wieloma innymi.

## Profil uczestników:

Szkolenie adresowane jest do programistów znających dowolny język programowania.

## Czas trwania:

8 dni

## Wiedza teoretyczna i praktyczna:

- GraphQL a REST
- Queries
- Mutacje
- Schemes
- Typy
- Walidacja
- Ekosystem
- Apollo Client i Relay
- Security
- Use cases

## Umiejętności:

Po ukończonym szkoleniu uczestnicy będą posiadać wiedzę na temat tego w jaki sposób wykorzystać język zapytań GraphQL w aplikacjach webowych. Jak tworzyć odpowiednie zapytania, w jaki sposób dbać o bezpieczeństwo i jakich narzędzi używać przy implementowaniu aplikacji opierających się o GraphQL'a.

# REST - od nowicjusza do geniusza

## Opis szkolenia:

W czasie szkolenia uczestnicy dowiedzą się na jakich zasadach działa architektura REST, jakie są jej założenia i w jaki sposób tworzyć poprawne API oparte o tę architekturę.

## Profil uczestników:

Od uczestników szkolenia wymagana jest umiejętność programowania w dowolnym języku.

## Czas trwania:

1 dzień

## Wiedza teoretyczna i praktyczna:

- Przegląd wzorców architektonicznych
- Protokół HTTP
- Stanowość i bezstanowość
- Komunikacja klient-serwer
- Autoryzacja
- Implementacje REST w różnych językach programowania

## Umiejętności:

Uczestnicy po ukończonym szkoleniu będą znać zasady architektury REST.

# regExp - wyrażenia regularne, zostań mistrzem!

## Opis szkolenia:

Celem szkolenia jest nauczenie uczestników w jaki sposób osiągnąć mistrzostwo w tworzeniu wyrażeń regularnych. Szkolenie jest pełne ćwiczeń dzięki czemu każdy uczestnik będzie w stanie przeciwiczyć nabytą wiedzę

## Profil uczestników:

Szkolenie adresowane jest do programistów w dowolnych językach programowania.

## Czas trwania:

1 dzień

## Wiedza teoretyczna i praktyczna:

- Składnia wyrażeń regularnych
- Kotwice
- metaznaki
- klasy znaków
- kwantyfikatory

## Umiejętności:

Po ukończonym szkoleniu uczestnicy będą potrafili tworzyć zaawansowane wyrażenia regularne

# Business Intelligence, wizualizacja big data w aplikacjach internetowych

## Opis szkolenia:

Tworzenie wizualizacji danych jest skomplikowanym zadaniem, które wymaga nie tylko umiejętności przetworzenia początkowych zbiorów danych, do kształtu, który pragniemy zaprezentować, ale również wykorzystanie odpowiednich narzędzi do wizualizacji danych. Strony WWW oraz aplikacje internetowe, są to doskonałe miejsca prezentacji danych a język do tworzenia aplikacji - JavaScript jest bogaty w narzędzia nie tylko do Business Intelligence ale i do Machine Learning i innych dziedzin data science. Celem szkolenia jest przekazanie wiedzy na temat wizualizacji danych za pomocą takich bibliotek jak d3.js, która daje największe możliwości jeśli chodzi o prezentowanie zależności między danymi. W czasie szkolenia poruszone zostaną również tematy związane z Machine Learning w ciągle niedocenianym języku jakim jest JavaScript.

## Profil uczestników:

Od uczestników szkolenia wymagana jest umiejętność programowania w dowolnym języku.

## Czas trwania:

2 dni

## Wiedza teoretyczna i praktyczna:

- Czym jest Business Intelligence
- Kryteria oceny wizualizacji
- Rodzaje wizualizacji danych
- BI a client-side
- d3.js w szczegółach
- Operacje na danych
- Interaktywne notatniki

### Umiejętności:

Uczestnicy po ukończonym szkoleniu będą znać techniki wizualizacji danych w przeglądarce i będą wiedzieć jak się dalej rozwijać w tym kierunku.



# Sieci neuronowe i uczenie maszynowe z tensorflow.js i brain.js

## Opis szkolenia:

Sieci neuronowe i uczenie maszynowe są to tematy, które w ostatnich latach zdobyły olbrzymią popularność. W czasie szkolenia uczestnicy dowiedzą się w jaki sposób używać dostępnych bibliotek napisanych w JavaScript, aby w prosty sposób przeprowadzać skomplikowane operacje.

## Profil uczestników:

Od uczestników szkolenia wymagana jest umiejętność programowania w dowolnym języku.

## Czas trwania:

2 dni

## Wiedza teoretyczna i praktyczna:

- Na czym polegają algorytmy uczenia maszynowego
- Filozofia sieci neuronowych
- Tensorflow
- Trenowanie sieci - dobór odpowiednich danych
- Modele danych
- Trenowanie asynchroniczne
- Strumienie
- Transfer learning
- Rozpoznawanie tekstu

## Umiejętności:

Uczestnicy po ukończonym szkoleniu będą znać techniki wizualizacji danych w przeglądarce i będą wiedzieć jak się dalej rozwijać w tym kierunku.

# Prezentowanie treści technicznych

## Opis szkolenia:

Celem szkolenia jest przekazanie praktycznej wiedzy dotyczącej wystąpień publicznych oraz sposobów przekazywania treści technicznych w zrozumiały i ciekawy sposób. Szkolenie skupia się na poprawieniu umiejętności: autoprezentacji, przekazywania wiedzy i podniesienia kompetencji oratorskich.

## Profil uczestników:

Szkolenie adresowane jest do osób, które pragną podnieść jakość swoich wystąpień publicznych, z naciskiem na wystąpienia publiczne związane z tematami technicznymi

## Czas trwania:

1 dzień

## Wiedza teoretyczna i praktyczna:

- Metody walki ze stresem
- Mowa ciała
- Budowanie wizerunku
- Prowadzenie dialogu z uczestnikami
- Praca z głosem - dykcja, intonacja, barwa głosu
- Metody eliminacji parajęzyka (yyy)
- Praca z flipchartem/whiteboardem/tablicą
- Dobieranie treści pod uczestników
- Storytelling
- Przygotowanie materiałów dydaktycznych

## Umiejętności:

Ukończenie szkolenia pomoże uczestnikom nabrać pewności siebie w czasie wystąpień publicznych oraz dostarczy im szereg technik podnoszących jakość wystąpienia.

# Komunikacja z developmentem (dla biznesu)

## Opis szkolenia:

Największym problemem w czasie wytwarzania oprogramowania jest komunikacja. Przez nieodpowiedni sposób porozumiewania się, proces twórczy wydłuża się w nieskończoność. I zarówno część biznesowa jak i deweloperska projektu jest bardzo niezadowolona z warunków pracy. W czasie szkolenia uczestnicy dowiedzą się w jaki sposób komunikować się z deweloperami, jak uzyskać od nich informacje, które są istotne w kontekście biznesu. Uczestnicy będą mieli szansę zrozumieć czemu tak często potrzeby programistów są zupełnie inne niż założone przez management, co wpływa na to, że zaczynają mniej wydajnie pracować czy nawet odchodzą z firmy. W czasie szkolenia przekazana zostanie również wiedza na temat programowania jako konceptu dzięki czemu komunikacja z programistami będzie dużo łatwiejsza.

## Profil uczestników:

Szkolenie będzie prowadzone od absolutnych podstaw. Języki programowania będą tłumaczone w analogii do języków naturalnych. Umiejętność programowania nie jest wymagana.

## Czas trwania:

2 dni

## Wiedza teoretyczna i praktyczna:

- Jak poznać potrzeby programisty
- Skąd się biorą problemy przy wyborze technologii do realizacji projektów
- Skąd się biorą problemy komunikacyjne z deweloperami
- W jaki sposób formułować treści aby dobrze komunikować zespół
- Czemu programista czuje wypalenie zawodowe
- Czym jest programowanie
- Czym jest program
- Czym jest język programowania

- Czym są biblioteki i frameworki
- Na czym polegają różnice między językami programowania
- Podział na frontend i backend
- Czym jest kaczuszka programisty
- Po czym poznać dobrego programistę

#### Umiejętności:

Uczestnicy po ukończonym szkoleniu będą lepiej rozumieć potrzeby programistów, będą wiedzieli w jaki sposób lepiej się z nimi komunikować a także poznają slang, którym się posługują.