

Spis treści

O autorze	11
O recenzencie	11
Wstęp	12
<hr/>	
Część I. Jak zrozumieć TypeScript i poprawić swoją znajomość języka JavaScript	17
Rozdział 1. Zrozumieć TypeScript	19
Wymagania techniczne	20
Czym jest TypeScript?	20
Dlaczego TypeScript jest niezbędny?	21
Typowanie dynamiczne a statyczne	23
Programowanie obiektowe	28
Podsumowanie	31
Rozdział 2. Prezentacja języka TypeScript	33
Wymagania techniczne	34
Czym są typy?	34
Jak działają typy?	35
Wprowadzenie do typów języka TypeScript	36
Typ any	36
Typ unknown	37
Typy przecięć i unii	40
Typy literałowe	41
Nazwy zastępcze typów	42

Spis treści

Typy wyników funkcji.....	42
Funkcje jako typy	44
Typ never	45
Klasy i interfejsy	45
Klasy	46
Interfejsy	52
Dziedziczenie	54
Klasy abstrakcyjne	57
Interfejsy	59
Typy generyczne	61
Prezentacja najnowszych możliwości języka i konfigurowania kompilatora	64
Łączenie opcjonalne	64
Scalanie wartości pustych	65
Konfigurowanie TypeScriptu	66
Podsumowanie	67
Rozdział 3. Tworzenie lepszych aplikacji	
dzięki użyciu możliwości wersji ES6+ języka JavaScript	68
Wymagania techniczne	69
Poznawanie rodzajów zmiennych w ES6 oraz zasięgów w języku JavaScript	70
Poznawanie funkcji strzałkowych	72
Zmienianie kontekstu this	74
Rozproszenie, destrukuryzacja i reszta	76
Rozproszenie, Object.assign oraz Array.concat	77
Destrukuryzacja	79
Reszta	80
Prezentacja wybranych funkcji tablicowych	81
find	81
filter	82
map	83
reduce	84
some oraz every	85
Przedstawienie nowych typów kolekcji	86
Set	86
Map	87
Przedstawienie słów kluczowych async i await	88
Podsumowanie	93
Część II. Nauka tworzenia aplikacji jednostronicowych	
z użyciem frameworka React	95
Rozdział 4. Przedstawienie koncepcji aplikacji jednostronicowych	
oraz ich realizacja z użyciem frameworka React	97
Wymagania techniczne	98
Przedstawienie wcześniejszych sposobów tworzenia witryny WWW	98
Cechy i zalety aplikacji jednostronicowych	100

Jak React pomaga w tworzeniu aplikacji jednostronicowych	101
Atrybuty aplikacji Reacta	102
Podsumowanie	113
Rozdział 5. Tworzenie aplikacji Reacta z wykorzystaniem hooków	114
Wymagania techniczne	115
Wyjaśnienie ograniczeń i problemów	
związanych ze stosowaniem starych komponentów klasowych	115
Stan	116
Metody cyklu życia	117
Prezentacja hooków Reacta i wyjaśnienie,	
dlaczego w stosunku do komponentów klasowych są one usprawnieniem	132
Porównanie stosowania komponentów klasowych i hooków	144
Wielokrotne stosowanie kodu	145
Prostota	145
Podsumowanie	146
Rozdział 6. Przygotowywanie projektu za pomocą create-react-app	
i testowanie go przy użyciu Jest	147
Wymagania techniczne	148
Przedstawienie metod programowania aplikacji Reacta	
i systemu używanego do ich budowania	148
Narzędzia do zarządzania projektami	149
Transpilacja	156
Repozytoria kodu	158
Testowanie aplikacji Reacta po stronie klienta	160
Atrapy	172
Tworzenie atrapy z wykorzystaniem jest.fn	173
Tworzenie atrapy komponentów	178
Prezentacja najpopularniejszych narzędzi oraz praktyk tworzenia aplikacji Reacta	185
Visual Studio Code	185
Prettier	186
Debugger Chrome	187
Alternatywne zintegrowane środowiska programistyczne	190
Podsumowanie	191
Rozdział 7. Redux i React Router	192
Wymagania techniczne	192
Zarządzanie stanem przy użyciu Reduxa	193
Reduktory i akcje	195
React Context	205
Prezentacja frameworka React Router	212
Podsumowanie	221