

Spis treści

O autorze	11
O recenzencie	11
Wstęp	12
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 literałów	41
Nazwy zastępcze typów	42

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
Poznanie rodzajów zmiennych w ES6 oraz zasięgów w języku JavaScript	70
Poznanie funkcji strzałkowych	72
Zmianowanie 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 atrap z wykorzystaniem jest.fn	173
Tworzenie atrap 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

Część III. Tworzenie usług internetowych z użyciem Expressa i GraphQL-a 223

Rozdział 8. Prezentacja tworzenia aplikacji serwerowych z wykorzystaniem Node.js i Expressa 225

Wymagania techniczne	226
Wyjaśnienie sposobu działania środowiska Node	226
Pętla zdarzeń	228
Prezentacja możliwości środowiska Node	229
Instalowanie Node	229
Tworzenie prostego serwera Node	233
Żądania i odpowiedzi	236
Trasowanie	239
Debugowanie	241
Jak Express ułatwia pisanie rozwiązań przeznaczonych dla środowiska Node	248
Przedstawienie możliwości frameworka Express	250
Tworzenie internetowego API przy użyciu Expressa	256
Podsumowanie	259

Rozdział 9. Czym jest GraphQL? 260

Wymagania techniczne	260
Czym jest GraphQL?	261
Schematy GraphQL	263
Definicje typów i resolwery	264
Zapytania, mutacje oraz subskrypcje	270
Podsumowanie	277

Rozdział 10. Konfiguracja projektu Expressa z zależnościami od języków TypeScript i GraphQL 278

Wymagania techniczne	279
Tworzenie projektu Expressa tworzonego w języku TypeScript	279
Dodawanie do projektu GraphQL-a i jego zależności	283
Prezentacja pakietów pomocniczych	290
Podsumowanie	292

Rozdział 11. Czego się nauczysz — aplikacja internetowego forum 293

Analiza aplikacji, którą napiszemy — internetowego forum	294
Analiza uwierzytelniania użytkowników forum	295
Analiza zarządzania wątkami	296
Analiza systemu punktacji wątków	297
Podsumowanie	298

Rozdział 12. Tworzenie klienta Reacta na potrzeby aplikacji internetowego forum	299
Wymagania techniczne	299
Tworzenie wstępnej wersji aplikacji Reacta	300
CSS Grid	302
Granice błędów	309
Warstwa usługi danych	311
Menu nawigacyjne	313
Komponenty związane z uwierzytelnianiem	318
Trasowanie i ekrany aplikacji	328
Ekran główny	329
Ekran wątku i jego wpisów	341
Podsumowanie	360
Rozdział 13. Przygotowywanie stanu sesji przy użyciu Expressa i Redisa	361
Wymagania techniczne	362
Czym jest stan sesji?	362
Przedstawienie magazynu danych Redis	363
Tworzenie stanu sesji z wykorzystaniem Expressa i Redisa	369
Podsumowanie	375
Rozdział 14. Przygotowywanie Postgresa oraz warstwy repozytorium przy wykorzystaniu TypeORM	376
Wymagania techniczne	377
Przygotowanie bazy danych Postgres	377
Przedstawienie mechanizmów odwzorowań obiektowo-relacyjnych na przykładzie TypeORM	382
Tworzenie warstwy repozytorium bazującej na Postgresie i TypeORM	383
Podsumowanie	413
Rozdział 15. Dodawanie schematu GraphQL-a — część 1.	414
Wymagania techniczne	414
Tworzenie definicji typów i resolverów dla serwerowego kodu GraphQL	415
System punktacji wątków	424
Integracja mechanizmu uwierzytelniania z resolverami GraphQL-a	428
Przygotowanie hooków Reacta do korzystania z serwera Apollo GraphQL	432
Ekran główny — komponent Main	435
Możliwości związane z uwierzytelnianiem	447
Ekran profilu użytkownika	454
Podsumowanie	460
Rozdział 16. Dodawanie schematu GraphQL-a — część 2.	461
Komponent Thread i jego trasa	461
System punktów	471
Podsumowanie	511

Rozdział 17. Wdrażanie w chmurze AWS

Wymagania techniczne	513
Konfiguracja Ubuntu w chmurze AWS	513
Instalacja Redisa, Postgresa i Node w systemie Ubuntu	520
Instalacja serwera Redis	521
Instalacja Postgresa	522
Instalacja Node	524
Konfiguracja i wdrażanie aplikacji na serwerze NGINX	525
Konfigurowanie projektu super-forum-client	532
Konfiguracja serwera NGINX	534
Rozwiązywanie problemów	542
Podsumowanie	543