

Spis treści

O autorze	23
O recenzencie	24
Część I Wprowadzenie do ASP.NET Core	25
Rozdział 1. ASP.NET Core w szerszym kontekście	27
Poznajemy ASP.NET Core	27
Poznajemy frameworki aplikacji	27
Poznajemy frameworki narzędziowe	29
Poznajemy platformę ASP.NET Core	30
Co powinienem wiedzieć?	31
Co jest potrzebne do wykonania przykładów zamieszczonych w książce?	31
Jaka platforma jest odpowiednia do wykonania przykładów zamieszczonych w książce?	31
Co zrobić w przypadku problemów podczas wykonywania przykładów?	31
Co zrobić w sytuacji, gdy znajdę błąd w książce?	32
Jaka jest struktura książki?	32
Część I. Wprowadzenie do ASP.NET Core	32
Część II. Platforma ASP.NET Core	32
Część III. Aplikacje ASP.NET Core	33
Część IV. Funkcje zaawansowane ASP.NET Core	33
Czego nie znajdę w książce?	33
Jak mogę skontaktować się z autorem?	33
Co zrobić, jeśli lektura książki sprawiła mi przyjemność?	34
Co zrobić, jeśli ta książka mnie zdenerwowała i chciałbym się poskarżyć?	34
Podsumowanie	34

Rozdział 2. Rozpoczęcie pracy z ASP.NET Core	35
Wybór edytora kodu źródłowego	35
Instalacja Visual Studio	36
Instalacja Visual Studio Code	39
Tworzenie projektu ASP.NET Core	43
Otworzenie projektu w Visual Studio	45
Otworzenie projektu w Visual Studio Code	46
Uruchomienie aplikacji ASP.NET Core	46
Poznajemy punkty końcowe	49
Poznajemy trasy	51
Generowanie stron WWW	51
Połączenie wszystkiego w całość	56
Podsumowanie	56
Rozdział 3. Pierwsza aplikacja ASP.NET Core	57
Przygotowanie sceny	57
Utworzenie projektu	57
Projektowanie modelu danych	59
Utworzenie drugiej metody akcji i widoku	60
Łączenie metod akcji	61
Budowanie formularza	63
Obsługa formularzy	64
Dodanie widoku Thanks	67
Wyświetlenie odpowiedzi	68
Dodanie kontroli poprawności danych	71
Nadanie stylu zawartości	76
Podsumowanie	82
Rozdział 4. Używanie narzędzi programistycznych	83
Utworzenie projektu ASP.NET Core	83
Utworzenie projektu za pomocą narzędzi działających w wierszu polecenia	84
Utworzenie projektu za pomocą Visual Studio	88
Dodanie kodu i treści do projektu	90
Poznajemy proces tworzenia elementów szkieletowych	91
Budowanie i uruchamianie projektu	93
Budowanie i uruchamianie projektu z poziomu wiersza poleceń	94
Budowanie i uruchamianie projektu za pomocą Visual Studio Code	94
Budowanie i uruchamianie projektu za pomocą Visual Studio	95
Zarządzanie pakietami	95
Zarządzanie pakietami NuGet	96
Zarządzanie pakietami narzędziowymi	97
Zarządzanie pakietami działającymi po stronie klienta	97
Zarządzanie pakietami za pomocą Visual Studio	99
Debugowanie projektu	99
Podsumowanie	102

Rozdział 5. Najważniejsze cechy języka C#	103
Utworzenie przykładowego projektu	104
Otworzenie projektu	104
Włączenie frameworka MVC	104
Utworzenie komponentów aplikacji ASP.NET Core	105
Użycie operatora warunkowego null	108
Łączenie operatorów warunkowych null	109
Łączenie operatorów: warunkowego i koalescencji	111
Użycie automatycznie implementowanych właściwości	113
Użycie automatycznie implementowanych metod inicjalizacyjnych właściwości	114
Utworzenie automatycznie implementowanych właściwości tylko do odczytu	115
Interpolacja ciągu tekstowego	116
Użycie inicjalizatorów obiektów i kolekcji	118
Użycie inicjalizatora indeksu	119
Dopasowanie wzorca	120
Dopasowanie wzorca w konstrukcji switch	121
Użycie metod rozszerzających	123
Stosowanie metod rozszerzających do interfejsów	124
Tworzenie filtrujących metod rozszerzających	126
Użycie wyrażeń lambda	128
Definiowanie funkcji	129
Użycie wyrażeń lambda w postaci metod i właściwości	132
Użycie inferencji typów i typów anonimowych	134
Użycie typów anonimowych	135
Użycie implementacji domyślnych w interfejsach	137
Użycie metod asynchronicznych	140
Bezpośrednia praca z zadaniami	140
Użycie słów kluczowych async i await	141
Używanie wyliczenia asynchronicznego	143
Pobieranie nazw	146
Podsumowanie	148
Rozdział 6. Testy jednostkowe w aplikacji ASP.NET Core	149
Utworzenie przykładowego projektu	150
Otworzenie projektu	150
Wybór numeru portu HTTP	151
Włączenie frameworka MVC	151
Utworzenie komponentów aplikacji ASP.NET Core	152
Uruchomienie przykładowej aplikacji	154
Utworzenie projektu testów jednostkowych	154
Usunięcie domyślnej klasy testu	155
Tworzenie i wykonywanie testów jednostkowych	156
Wykonywanie testów w oknie Eksplorator testów w Visual Studio	158
Wykonywanie testów w Visual Studio Code	158
Wykonywanie testów w wierszu poleceń	159
Poprawienie testu jednostkowego	160

Rozdział 2.	Izolowanie komponentów poddawanych testom jednostkowym	161
	Użycie pakietu imitacji	166
	Utworzenie obiektu imitacji	166
	Podsumowanie	168
Rozdział 7.	SportsStore — kompletna aplikacja	169
	Utworzenie projektów	170
	Utworzenie projektu testów jednostkowych	170
	Utworzenie katalogów projektu aplikacji	171
	Otworzenie projektów	171
	Przygotowanie usług aplikacji i potoku żądania	172
	Konfiguracja silnika widoku Razor	173
	Utworzenie kontrolera i widoku	175
	Tworzenie modelu danych	176
	Sprawdzenie i uruchomienie aplikacji	176
	Dodanie danych do aplikacji	177
	Instalowanie pakietów narzędzi Entity Framework Core	177
	Definiowanie ciągu tekstowego połączenia	177
	Utworzenie klasy kontekstu bazy danych	178
	Konfigurowanie Entity Framework Core	179
	Tworzenie repozytorium	180
	Utworzenie i zastosowanie migracji bazy danych	183
	Tworzenie danych początkowych	183
	Wyświetlanie listy produktów	186
	Przygotowanie kontrolera	187
	Uaktualnienie widoku	189
	Uruchamianie aplikacji	189
	Dodanie stronicowania	190
	Wyświetlanie łączy stron	192
	Ulepszanie adresów URL	200
	Dodawanie stylu	202
	Instalacja pakietu Bootstrap	203
	Zastosowanie w aplikacji stylów Bootstrap	203
	Tworzenie widoku częściowego	205
	Podsumowanie	207
Rozdział 8.	SportsStore — nawigacja i koszyk na zakupy	208
	Dodawanie kontrolerek nawigacji	208
	Filtrowanie listy produktów	208
	Ulepszanie schematu URL	213
	Budowanie menu nawigacji po kategoriach	217
	Poprawianie licznika stron	225
	Budowanie koszyka na zakupy	228
	Konfigurowanie Razor Pages	228
	Utworzenie Razor Pages	230
	Tworzenie przycisków koszyka	231
	Włączenie obsługi sesji	233

Implementowanie funkcjonalności koszyka	234
Dokończenie pracy z Razor Page	238
Podsumowanie	244
Rozdział 9. SportsStore — ukończenie koszyka na zakupy	245
Dopracowanie modelu koszyka za pomocą usługi	245
Tworzenie klasy koszyka obsługującej magazyn danych	246
Rejestrowanie usługi	248
Uproszczenie klasy Cart frameworka Razor Pages	249
Kończenie budowania koszyka	251
Usuwanie produktów z koszyka	251
Dodawanie podsumowania koszyka	254
Składanie zamówień	256
Utworzenie klasy modelu	256
Dodawanie procesu składania zamówienia	258
Utworzenie kontrolera i widoku	258
Implementowanie mechanizmu przetwarzania zamówień	260
Zakończenie pracy nad kontrolerem koszyka	265
Wyświetlanie informacji o błędach systemu kontroli poprawności	268
Wyświetlanie strony podsumowania	270
Podsumowanie	271
Rozdział 10. SportsStore — administracja	272
Przygotowanie serwera Blazor	272
Utworzenie pliku importów	274
Utworzenie strony początkowej Razor Page	274
Utworzenie komponentów routingu i układu	275
Utworzenie komponentów Razor	276
Sprawdzenie konfiguracji Blazor	276
Zarządzanie zamówieniami	277
Usprawnienie modelu	278
Wyświetlanie zamówień administratorowi	279
Dodajemy zarządzanie katalogiem	282
Rozszerzenie repozytorium	283
Dodanie kontroli poprawności modelu	284
Tworzenie komponentu listy	285
Tworzenie komponentu widoku szczegółowego	286
Tworzenie komponentu edytora	288
Usuwanie produktu	290
Podsumowanie	293
Rozdział 11. SportsStore — bezpieczeństwo i wdrożenie aplikacji	294
Zabezpieczanie funkcji administracyjnych	294
Utworzenie bazy danych dla systemu Identity	294
Dodanie konwencjonalnej funkcjonalności administracyjnej	300
Zdefiniowanie prostej polityki autoryzacji	302
Utworzenie kontrolera AccountController i widoków	303
Przetestowanie polityki bezpieczeństwa	307

Przygotowanie ASP.NET Core do wdrożenia	307
Konfiguracja obsługi błędów	307
Utworzenie produkcyjnych ustawień konfiguracyjnych	309
Utworzenie obrazu Dockera	310
Uruchomienie skonteneryzowanej aplikacji	312
Podsumowanie	313

Część II Platforma ASP.NET Core 315

Rozdział 12. Poznajemy platformę ASP.NET Core 317

Utworzenie przykładowego projektu	318
Uruchomienie przykładowej aplikacji	319
Poznajemy platformę ASP.NET Core	320
Poznajemy oprogramowanie pośredniczące i potok żądania	320
Poznajemy usługi	321
Poznajemy projekt ASP.NET Core	322
Poznajemy punkt wyjścia aplikacji	323
Poznajemy klasę Startup	324
Poznajemy plik projektu	325
Tworzenie własnego oprogramowania pośredniczącego	327
Definiowanie oprogramowania pośredniczącego za pomocą klasy	331
Poznajemy zwrotną ścieżkę dostępu potoku żądania	334
Skrócenie potoku żądania	336
Tworzenie odgałęzienia potoku żądania	338
Utworzenie końcowego oprogramowania pośredniczącego	340
Konfiguracja oprogramowania pośredniczącego	343
Używanie wzorca opcji z bazującym na klasie komponentem oprogramowania pośredniczącego	345
Podsumowanie	347

Rozdział 13. Routing URL 348

Utworzenie przykładowego projektu	349
Poznajemy routing URL	353
Dodanie komponentu oprogramowania pośredniczącego routingu i zdefiniowanie punktu końcowego	353
Poznajemy wzorce adresu URL	357
Używanie zmiennych segmentów we wzorcach URL	358
Generowanie adresów URL na podstawie tras	363
Zarządzanie dopasowaniem adresów URL	367
Dopasowanie wielu wartości z pojedynczego segmentu adresu URL	367
Używanie wartości domyślnych dla zmiennych segmentów	369
Używanie segmentów opcjonalnych we wzorcu adresu URL	369
Używanie zmiennej segmentu catchall	371
Ograniczenia podczas dopasowywania segmentów	372
Definiowanie trasy awaryjnej	376

Zaawansowane funkcje routingu	377
Tworzenie ograniczeń niestandardowych	378
Unikanie wyjątków związanych z niedopasowaniem trasy	379
Uzyskanie dostępu do punktu końcowego z poziomu komponentu oprogramowania pośredniczącego	382
Podsumowanie	384

Rozdział 14. Wstrzykiwanie zależności385

Utworzenie przykładowego projektu	386
Utworzenie komponentu oprogramowania pośredniczącego i punktu końcowego	387
Konfiguracja potoku żądania	388
Poznajemy położenie usługi i ściśle powiązanie komponentów	390
Problem związany z położeniem usługi	391
Problem związany ze ścisłym powiązaniem komponentów	393
Używanie mechanizmu wstrzykiwania zależności	396
Używanie usługi w klasie oprogramowania pośredniczącego	397
Używanie usługi w punkcie końcowym	399
Cykl życiowy usługi	404
Tworzenie usługi tymczasowej	405
Unikanie problemów związanych z ponownym używaniem usługi tymczasowej	406
Używanie usługi zasięgu	410
Inne funkcje mechanizmu wstrzykiwania zależności	416
Utworzenie łańcucha zależności	416
Dostęp do usług w metodzie ConfigureServices()	418
Używanie funkcji fabryki usługi	419
Tworzenie usługi z wieloma implementacjami	421
Używanie usługi pozbawionej typu	424
Podsumowanie	425

Rozdział 15. Używanie funkcjonalności platformy — część I426

Utworzenie przykładowego projektu	427
Używanie usługi konfiguracji	429
Poznajemy plik konfiguracyjny dla danego środowiska	430
Uzyskiwanie dostępu do ustawień konfiguracyjnych	431
Używanie danych konfiguracyjnych w usługach	433
Poznajemy plik ustawień początkowych	435
Ustalanie środowiska za pomocą kodu w klasie Startup	441
Przechowywanie kluczy tajnych użytkownika	442
Używanie usługi rejestrowania danych	445
Generowanie komunikatów	446
Określenie minimalnego poziomu rejestrowanych komunikatów	450
Używanie treści statycznej i pakietów działających po stronie klienta	451
Dodanie komponentu oprogramowania pośredniczącego odpowiedzialnego za obsługę treści statycznej	452
Używanie pakietów po stronie klienta	457
Podsumowanie	460

Rozdział 16. Używanie funkcjonalności platformy — część II	461
Utworzenie przykładowego projektu	462
Używanie mechanizmu ciasteczek	463
Obsługa zgody na użycie ciasteczek	466
Zarządzanie zgodą na używanie ciasteczek	468
Mechanizm sesji	471
Konfiguracja usługi sesji i komponentu oprogramowania pośredniczącego	471
Używanie danych sesji	474
Praca z połączeniami HTTPS	476
Włączanie obsługi połączeń HTTPS	476
Wykrywanie żądań HTTPS	478
Wymuszanie obsługi żądań HTTPS	480
Włączenie protokołu HTTP Strict Transport Security	481
Obsługa wyjątków i błędów	484
Zwrot odpowiedzi HTML zawierającej informacje o błędzie	486
Usprawnianie odpowiedzi zawierających kody stanu	489
Filtrowanie żądań za pomocą nagłówka Host	491
Podsumowanie	494
Rozdział 17. Praca z danymi	495
Utworzenie przykładowego projektu	497
Buforowanie danych	499
Buforowanie wartości danych	501
Używanie współdzielonego i trwałego bufora danych	505
Buforowanie odpowiedzi	509
Entity Framework Core	512
Instalacja Entity Framework Core	513
Tworzenie modelu danych	514
Konfiguracja usługi bazy danych	515
Tworzenie i stosowanie migracji bazy danych	517
Przygotowanie bazy danych	517
Używanie danych w punkcie końcowym	520
Podsumowanie	523
Część III Aplikacje ASP.NET Core	525
Rozdział 18. Utworzenie przykładowego projektu	527
Utworzenie projektu	527
Dodawanie modelu danych	528
Dodawanie pakietów NuGet do projektu	528
Tworzenie modelu danych	529
Przygotowanie bazy danych	531
Konfiguracja usług Entity Framework Core i komponentu oprogramowania pośredniczącego	533
Tworzenie i stosowanie migracji	534

Dodawanie frameworka CSS	535
Konfigurowanie potoku żądania	535
Uruchomienie przykładowej aplikacji	537
Podsumowanie	537

Rozdział 19. Tworzenie usługi sieciowej RESTful

Utworzenie przykładowego projektu	539
Poznajemy usługi sieciowe RESTful	540
Poznajemy metody i adresy URL żądania	540
Poznajemy format JSON	541
Tworzenie usługi sieciowej za pomocą niestandardowego punktu końcowego	542
Utworzenie usługi sieciowej za pomocą kontrolera	545
Włączenie obsługi frameworka MVC	545
Tworzenie kontrolera	547
Usprawnienie usługi sieciowej	557
Używanie akcji asynchronicznych	558
Uniknięcie zbędnego dołączania modelu	560
Używanie wyniku akcji	561
Sprawdzanie poprawności danych	568
Stosowanie atrybutu kontrolera API	570
Pomijanie właściwości null	571
Podsumowanie	574

Rozdział 20. Funkcje zaawansowane usługi sieciowej

Utworzenie przykładowego projektu	576
Usunięcie bazy danych	577
Uruchomienie przykładowej aplikacji	577
Praca z powiązаныmi ze sobą danymi	578
Usunięcie odwołania cyklicznego w danych	580
Obsługa metody HTTP PATCH	581
Poznajemy standard JSON Patch	581
Instalacja i konfiguracja pakietu JSON Patch	582
Definiowanie metody akcji	583
Formatowanie treści	584
Poznajemy domyślną politykę treści	584
Poznajemy negocjację treści	586
Określanie formatu danych akcji	590
Pobranie formatu danych z adresu URL	592
Ograniczanie formatów otrzymywanych przez metodę akcji	593
Dokumentowanie i analizowanie usług sieciowych	595
Rozwiązywanie konfliktów metod akcji	595
Instalacja i konfiguracja pakietu Swashbuckle	596
Dopracowanie opisu API	599
Podsumowanie	603

Rozdział 21. Używanie kontrolerów z widokami — część I	604
Utworzenie przykładowego projektu	605
Usunięcie bazy danych	606
Uruchomienie przykładowej aplikacji	607
Rozpoczęcie pracy z widokiem	607
Konfiguracja aplikacji	607
Utworzenie kontrolera HTML	609
Tworzenie widoku Razor	612
Wybór widoku na podstawie nazwy	614
Praca z widokami Razor	618
Wybór typu modelu widoku	621
Używanie pliku poleceń importujących widoki	624
Poznajemy składnię Razor	626
Poznajemy dyrektywy	627
Poznajemy wyrażenia treści	627
Zdefiniowanie elementu treści	628
Zdefiniowanie wartości atrybutu	629
Użycie konstrukcji warunkowych	630
Wyświetlanie sekwencji	634
Używanie bloków kodu Razor	636
Podsumowanie	637
Rozdział 22. Używanie kontrolerów z widokami — część II	638
Utworzenie przykładowego projektu	639
Usunięcie bazy danych	640
Uruchomienie przykładowej aplikacji	641
Używanie obiektu ViewBag	641
Używanie danych tymczasowych	643
Praca z układami	646
Konfiguracja układu za pomocą obiektu ViewBag	648
Używanie pliku ViewStart	650
Nadpisanie układu domyślnego	651
Zastosowanie sekcji układu	655
Użycie widoków częściowych	661
Włączanie widoków częściowych	661
Tworzenie widoku częściowego	662
Zastosowanie widoku częściowego	662
Poznajemy kodowanie treści	665
Kodowanie treści HTML	666
Kodowanie JSON	668
Podsumowanie	669
Rozdział 23. Strony Razor	670
Utworzenie przykładowego projektu	671
Uruchomienie przykładowej aplikacji	672

Poznajemy strony Razor	672
Konfiguracja stron Razor	673
Tworzenie strony Razor	674
Poznajemy routing stron Razor	678
Określanie wzorca routingu na stronie Razor	680
Dodawanie tras stron Razor	682
Poznajemy klasę modelu strony	684
Używanie pliku klasy ukrytej	684
Poznajemy wynik akcji na stronie Razor	688
Obsługa wielu metod HTTP	692
Wybór metody procedury obsługi	694
Poznajemy widok strony Razor	696
Utworzenie układu dla strony Razor	696
Używanie widoków częściowych na stronach Razor	698
Tworzenie strony Razor bez modelu strony	699
Podsumowanie	701

Rozdział 24. Komponenty widoku 702

Utworzenie przykładowego projektu	703
Usunięcie bazy danych	705
Uruchomienie przykładowej aplikacji	706
Poznajemy komponent widoku	706
Utworzenie komponentu widoku	707
Zastosowanie komponentu widoku	708
Poznajemy wynik działania komponentu widoku	711
Zwrot widoku częściowego	712
Zwrot fragmentów kodu HTML	715
Pobieranie danych kontekstu	718
Użycie argumentów do przekazania kontekstu z widoku nadrzędnego	719
Tworzenie asynchronicznego komponentu widoku	722
Utworzenie klasy komponentu widoku	725
Utworzenie hybrydowej klasy kontrolera	727
Podsumowanie	730

Rozdział 25. Poznajemy atrybuty pomocnicze znaczników 731

Utworzenie przykładowego projektu	732
Usunięcie bazy danych	734
Uruchomienie przykładowej aplikacji	734
Utworzenie atrybutu pomocniczego znacznika	735
Zdefiniowanie klasy atrybutu pomocniczego znacznika	736
Rejestrowanie atrybutu pomocniczego znacznika	739
Użycie atrybutu pomocniczego znacznika	739
Zawężanie zasięgu atrybutu pomocniczego znacznika	741
Rozszerzenie zasięgu atrybutu pomocniczego znacznika	742
Zaawansowane funkcje atrybutu pomocniczego znacznika	744
Tworzenie elementów skrótu	745
Programowe tworzenie elementów	747

Umieszczanie treści przed elementem i po nim	748
Pobieranie danych kontekstu widoku	752
Praca z modelem widoku	755
Koordinacja między atrybutami pomocniczymi znaczników	759
Zawieszenie wygenerowania elementu	761
Używanie komponentów atrybutu pomocniczego znacznika	763
Utworzenie komponentu atrybutu pomocniczego znacznika	763
Zwiększenie puli elementów obsługiwanych przez komponent atrybutu pomocniczego znacznika	766
Podsumowanie	768
Rozdział 26. Używanie wbudowanych atrybutów pomocniczych znaczników	769
Przygotowanie przykładowego projektu	770
Dodawanie pliku obrazu	772
Instalowanie pakietu działającego po stronie klienta	772
Usunięcie bazy danych	773
Uruchomienie przykładowej aplikacji	773
Włączanie wbudowanych atrybutów pomocniczych znaczników	774
Przekształcanie znaczników <a>	774
Używanie znaczników <a> na stronach Razor	776
Używanie atrybutów pomocniczych znaczników obsługujących pliki JavaScript i CSS	777
Zarządzanie plikami JavaScript	778
Zarządzanie arkuszami stylów CSS	787
Praca ze znacznikiem <image>	790
Użycie buforowanych danych	792
Określenie czasu wygaśnięcia buforowanej treści	794
Używanie atrybutu pomocniczego znacznika <environment>	797
Podsumowanie	798
Rozdział 27. Użycie atrybutów pomocniczych znaczników formularza	799
Przygotowanie przykładowego projektu	800
Usunięcie bazy danych	801
Uruchomienie przykładowej aplikacji	802
Poznajemy wzorzec obsługi formularza	803
Tworzenie kontrolera do obsługi formularza	803
Tworzenie strony Razor przeznaczonej do obsługi formularza	806
Używanie atrybutów pomocniczych znaczników do usprawnienia formularzy	808
Praca ze znacznikami formularza HTML	808
Transformacja przycisków formularza	810
Praca ze znacznikami <input>	812
Transformacja atrybutu type elementu <input>	813
Formatowanie wartości danych	815
Wyświetlanie w znaczniku <input> wartości z powiązanych ze sobą danych	818
Praca ze znacznikiem <label>	823
Praca ze znacznikami <select> i <option>	824
Dodanie treści do znacznika <select>	826
Praca ze znacznikiem <textarea>	828

Zabezpieczenie przed atakami typu CSRF	830
Włączanie w kontrolerze zabezpieczenia przed atakami typu CSRF	831
Włączanie na stronie Razor zabezpieczenia przed atakami typu CSRF	833
Klienty JavaScript i tokeny zabezpieczające przed atakami typu CSRF	834
Podsumowanie	837

Rozdział 28. Dołączanie modelu838

Utworzenie przykładowego projektu	839
Usunięcie bazy danych	840
Uruchomienie przykładowej aplikacji	840
Poznajemy dołączanie modelu	841
Dołączanie typów prostych	843
Dołączanie typów prostych na stronach Razor	844
Poznajemy dołączanie wartości domyślnej	845
Dołączanie typów złożonych	848
Dołączanie do właściwości	850
Dołączanie zagnieżdżonych typów złożonych	852
Selektywne dołączanie właściwości	856
Selektywne dołączanie właściwości w klasie modelu	857
Dołączanie tablic i kolekcji	859
Dołączanie do tablic	859
Dołączanie do prostej kolekcji	862
Dołączanie do słownika	864
Dołączanie kolekcji typów złożonych	866
Określanie źródła dołączania modelu	869
Określanie źródła danych dla właściwości	871
Użycie nagłówków jako źródła danych dla funkcji dołączania modelu	873
Użycie treści żądania jako źródła danych dla funkcji dołączania modelu	874
Ręczne dołączanie modelu	875
Podsumowanie	877

Rozdział 29. Kontrola poprawności danych modelu878

Utworzenie przykładowego projektu	879
Usunięcie bazy danych	881
Uruchomienie przykładowej aplikacji	881
Potrzeba stosowania kontroli poprawności danych modelu	881
Jawna kontrola poprawności modelu	882
Wyświetlenie użytkownikowi błędów podczas kontroli poprawności	886
Wyświetlanie komunikatów kontroli poprawności	888
Wyświetlanie błędów kontroli poprawności na poziomie właściwości	893
Wyświetlanie błędów kontroli poprawności na poziomie modelu	895
Jawne sprawdzanie danych na stronie Razor	897
Definiowanie reguł poprawności za pomocą metadanych	901
Tworzenie własnego atrybutu kontroli poprawności	905
Użycie kontroli poprawności po stronie klienta	910
Wykonywanie zdalnej kontroli poprawności	913
Wykonywanie zdalnej kontroli poprawności na stronie Razor	916
Podsumowanie	918

Rozdział 30. Filtry	919
Utworzenie przykładowego projektu	920
Włączenie obsługi połączeń HTTPS	921
Usunięcie bazy danych	922
Uruchomienie przykładowej aplikacji	922
Użycie filtrów	923
Użycie filtrów na stronach Razor	927
Poznajemy filtry	929
Tworzenie własnych filtrów	931
Użycie filtrów autoryzacji	931
Używanie filtrów zasobów	933
Użycie filtrów akcji	938
Używanie filtrów strony	943
Używanie filtra wyniku	947
Użycie filtrów wyjątków	952
Utworzenie filtra wyjątku	953
Zarządzanie cyklem życiowym filtra	955
Utworzenie fabryki filtrów	957
Używanie zasięgu mechanizmu wstrzykiwania zależności do zarządzania cyklem życiowym filtrów	959
Użycie filtrów globalnych	961
Poznajemy i zmieniamy kolejność wykonywania filtrów	963
Zmiana kolejności filtrów	965
Podsumowanie	968
Rozdział 31. Utworzenie aplikacji bazującej na formularzach	969
Utworzenie przykładowego projektu	969
Usunięcie bazy danych	972
Uruchomienie przykładowej aplikacji	972
Utworzenie aplikacji bazującej na formularzach MVC	973
Przygotowanie modelu widoku i widoku	973
Odczyt danych	975
Tworzenie danych	977
Edycja danych	981
Usunięcie danych	984
Utworzenie bazującej na formularzach aplikacji stron Razor	986
Tworzenie wspólnej funkcjonalności	988
Definiowanie stron dla operacji CRUD	991
Tworzenie nowych obiektów powiązanych ze sobą danych	993
Dostarczanie w tym samym żądaniu powiązanych ze sobą danych	993
Utworzenie nowych danych w innym miejscu	997
Podsumowanie	1001

Część IV	Funkcje zaawansowane ASP.NET Core	1003
Rozdział 32.	Utworzenie przykładowego projektu	1005
	Utworzenie projektu	1005
	Dodawanie pakietów NuGet do projektu	1006
	Dodawanie modelu danych	1007
	Przygotowanie bazy danych	1008
	Konfiguracja usług Entity Framework Core i komponentu oprogramowania pośredniczącego	1010
	Tworzenie i stosowanie migracji	1012
	Dodawanie frameworka CSS	1012
	Konfigurowanie usług i komponentu oprogramowania pośredniczącego	1013
	Tworzenie kontrolera i widoku	1014
	Tworzenie strony Razor	1016
	Uruchomienie przykładowej aplikacji	1018
	Podsumowanie	1019
Rozdział 33.	Praca z serwerem Blazor — część I	1020
	Utworzenie przykładowego projektu	1021
	Poznajemy serwer Blazor	1022
	Poznajemy zalety serwera Blazor	1024
	Poznajemy wady serwera Blazor	1024
	Wybór między serwerem Blazor i frameworkiem Angular, React lub Vue.js	1024
	Rozpoczęcie pracy z technologią Blazor	1025
	Konfiguracja ASP.NET Core do pracy z serwerem Blazor	1025
	Tworzenie komponentu Razor	1027
	Poznajemy podstawy funkcjonalności komponentu Razor	1032
	Poznajemy zdarzenia Blazor i dołączanie danych	1033
	Wykorzystanie mechanizmu dołączania danych	1041
	Używanie klasy do definiowania komponentu	1047
	Używanie pliku ukrytego kodu	1047
	Definiowanie klasy komponentu Razor	1048
	Podsumowanie	1050
Rozdział 34.	Praca z serwerem Blazor — część II	1051
	Utworzenie przykładowego projektu	1051
	Łączenie komponentów	1052
	Konfiguracja komponentu za pomocą atrybutu	1054
	Tworzenie niestandardowych zdarzeń i dołączania danych	1060
	Wyświetlanie treści potomnej w komponencie	1066
	Tworzenie komponentu szablonu	1068
	Używanie parametrów typu generycznego w komponencie szablonu	1070
	Parametry kaskadowe	1076
	Obsługa błędów	1079
	Błędy dotyczące połączenia	1080
	Obsługa niewychwyconych błędów aplikacji	1082
	Podsumowanie	1085

Rozdział 35. Funkcje zaawansowane Blazor	1086
Utworzenie przykładowego projektu	1087
Używanie routingu komponentu	1088
Przygotowania dla stron Razor	1089
Dodawanie tras do komponentów	1090
Poruszanie się między komponentami stosującymi routing	1094
Otrzymywanie danych routingu	1097
Używanie układu do zdefiniowania wspólnej treści	1099
Metody cyklu życiowego komponentu	1101
Używanie metod cyklu życiowego w zadaniach asynchronicznych	1103
Zarządzanie interakcjami komponentu	1106
Używanie odniesień do komponentów potomnych	1106
Praca z komponentami pochodzącymi z innego kodu	1109
Współpraca z komponentami za pomocą JavaScriptu	1114
Podsumowanie	1122
Rozdział 36. Dane i formularze Blazor	1123
Utworzenie przykładowego projektu	1124
Usunięcie bazy danych i uruchomienie aplikacji	1127
Używanie komponentów formularza Blazor	1127
Tworzenie niestandardowych komponentów formularza	1129
Weryfikacja danych formularza	1133
Obsługa zdarzeń formularza	1137
Używanie Entity Framework Core z technologią Blazor	1139
Kwestia zasięgu kontekstu w Entity Framework Core	1140
Problem związany z powtórzonym zapytaniem	1144
Przeprowadzanie operacji CRUD	1150
Utworzenie komponentu List	1150
Utworzenie komponentu Details	1152
Tworzenie komponentu Editor	1153
Rozbudowa funkcjonalności formularza Blazor	1156
Tworzenie niestandardowego ograniczenia weryfikacji	1157
Tworzenie wysyłającego formularz przycisku, który będzie aktywny, jedynie gdy dane są poprawne	1160
Podsumowanie	1162
Rozdział 37. Używanie Blazor Web Assembly	1163
Utworzenie przykładowego projektu	1165
Usunięcie bazy danych i uruchomienie aplikacji	1166
Konfiguracja Blazor WebAssembly	1167
Tworzenie projektu współdzielonego	1167
Tworzenie projektu Blazor WebAssembly	1168
Przygotowanie projektu ASP.NET Core	1169
Dodawanie odwołania do pliku rozwiązania	1169
Otworzenie projektów	1169
Dokończenie konfiguracji Blazor WebAssembly	1169
Testowanie miejsc zarezerwowanych dla komponentów	1173

Tworzenie komponentu Blazor WebAssembly	1173
Importowanie przestrzeni nazw modelu danych	1173
Tworzenie komponentu	1174
Tworzenie układu	1178
Definiowanie stylów CSS	1179
Ukończenie aplikacji Blazor WebAssembly	1179
Tworzenie komponentu Details	1179
Tworzenie komponentu Editor	1181
Podsumowanie	1184

Rozdział 38. Użycie ASP.NET Core Identity 1185

Utworzenie przykładowego projektu	1186
Przygotowanie projektu do użycia ASP.NET Core Identity	1188
Przygotowanie bazy danych ASP.NET Core Identity	1188
Konfigurowanie ciągu tekstowego połączenia z bazą danych	1189
Konfigurowanie aplikacji	1189
Tworzenie migracji bazy danych Identity i jej przeprowadzenie	1191
Tworzenie narzędzi przeznaczonych do zarządzania użytkownikami	1191
Przygotowanie narzędzi przeznaczonych do zarządzania użytkownikami	1192
Wyświetlanie listy kont użytkowników	1193
Tworzenie użytkowników	1195
Edytowanie użytkowników	1204
Usunięcie użytkownika	1206
Tworzenie narzędzi przeznaczonych do zarządzania rolami	1207
Przygotowanie do utworzenia narzędzi przeznaczonych do zarządzania rolami	1209
Wyświetlanie i usuwanie ról	1209
Tworzenie roli	1211
Przypisywanie użytkownika do roli	1212
Podsumowanie	1215

Rozdział 39. Stosowanie ASP.NET Core Identity 1216

Utworzenie przykładowego projektu	1216
Uwierzytelnianie użytkowników	1217
Tworzenie funkcjonalności logowania	1217
Analiza ciasteczka ASP.NET Core Identity	1221
Tworzenie strony wylogowania	1222
Testowanie funkcjonalności uwierzytelniania	1223
Włączanie oprogramowania pośredniczącego odpowiedzialnego za uwierzytelnianie w ASP.NET Core Identity	1223
Autoryzacja dostępu do punktu końcowego	1227
Zastosowanie atrybutu Authorize	1227
Włączenie oprogramowania pośredniczącego odpowiedzialnego za obsługę uwierzytelniania	1228
Utworzenie punktu końcowego informującego o braku dostępu	1228
Przygotowanie danych	1229
Testowanie sekwencji uwierzytelniania	1231

