C# 11 i .NET 7 dla programistów aplikacji wieloplatformowych : twórz aplikacje, witryny www oraz serwisy sieciowe za pomocą ASP.NET Core 7, Blazor i EF Core 7 / Mark J. Price. – Gliwice, copyright © 2023

Spis treści

O autorze **23**

O korektorze merytorycznym **23**

Wstęp **25**

ROZDZIAŁ 1
Cześć, C#! Witaj, .NET! **31**
Konfigurowanie środowiska programistycznego 33
Wybieranie narzędzia i typu aplikacji właściwych do nauki 34
Instalowanie na wielu platformach 37
Pobieranie i instalowanie Visual Studio 2022 dla Windows 38
Pobieranie i instalowanie Microsoft Visual Studio Code 39
Poznawanie .NET 43
Poznawanie .NET Framework 43
Poznawanie projektów Mono, Xamarin i Unity 43
Poznawanie .NET Core 44
Droga do jednej platformy .NET 44
Plany obsługi platformy .NET 46
Co się zmienia w nowoczesnym .NET? 48
Poznawanie .NET Standard 50
Platformy .NET i narzędzia używane w tym wydaniu 52
Poznawanie języka IL 53
Porównanie technologii .NET 53
Tworzenie aplikacji konsoli za pomocą Visual Studio 2022 54
Zarządzanie wieloma projektami w Visual Studio 2022 54
Pisanie kodu za pomocą Visual Studio 2022 54
Kompilowanie i uruchamianie kodu w Visual Studio 56
Programy najwyższego poziomu 58
Dodawanie drugiego projektu w Visual Studio 2022 60
Tworzenie aplikacji konsoli za pomocą Visual Studio Code 62
Zarządzanie wieloma projektami w Visual Studio Code 62
Pisanie kodu za pomocą Visual Studio Code 62
Kompilowanie i uruchamianie kodu za pomocą polecenia dotnet 66
Dodawanie drugiego projektu w Visual Studio Code 67
Badanie kodu w interaktywnych notatnikach .NET 69
Używanie interaktywnych notatników .NET do pisania kodu z tej książki 69
Przeglądanie folderów i plików projektów 70
Wspólne foldery i pliki 70

Kod w repozytorium GitHuba 71
Wykorzystywanie repozytorium GitHuba w tej książce 72
Pobieranie kodu rozwiązań z repozytorium GitHuba 72
Używanie systemu Git w Visual Studio Code 73
Gdzie znaleźć pomoc? 74
Przeglądanie dokumentacji Microsoftu 74
Uzyskiwanie pomocy dla narzędzia dotnet 74
Przeglądanie definicji typów i ich elementów 75
Poszukiwanie odpowiedzi na Stack Overflow 78
Poszukiwanie odpowiedzi za pomocą Google 78
Subskrybowanie blogów 79
Filmy Scotta Hanselmana 79
Praktyka i ćwiczenia 79
Ćwiczenie 1.1 — sprawdź swoją wiedzę 79
Ćwiczenie 1.2 — ćwicz C# gdzie się da 80
Ćwiczenie 1.3 — dalsza lektura 80
Ćwiczenie 1.4 — motywy kolorystyczne w nowoczesnym .NET 80
Podsumowanie 81

ROZDZIAŁ 2
Mówimy w C# 82
Wprowadzenie do języka C# 82
Rozpoznawanie wersji oraz funkcji języka 83
Standardy języka C# 87
Odczytywanie wersji używanego kompilatora C# 88
Poznawanie gramatyki i słownictwa języka C# 91
Wyświetlanie numeru wersji kompilatora 91
Gramatyka języka C# 93
Instrukcje 93
Komentarze 93
Bloki 95
Przykłady instrukcji i bloków 95
Słownictwo języka C# 95
Porównanie języków programowania do języków ludzkich 96
Zmiana schematu kolorów składni języka C# 96
Pomoc przy pisaniu kodu 97
Importowanie przestrzeni nazw 98
Niejawne i globalne importowanie przestrzeni nazw 99
Czasowniki jako metody 103
Rzeczowniki to typy, pola i zmienne 103
Ujawnienie wielkości słownika języka C# 104
Praca ze zmiennymi 107
Nazywanie zmiennych i przypisanie wartości 107
Literały 108
Przechowywanie tekstu 108
Przechowywanie liczb 111
Przechowywanie liczb rzeczywistych 114

Przechowywanie wartości logicznych 118
Zapisywanie obiektów dowolnego typu 118
Przechowywanie typów dynamicznych 119
Deklarowanie zmiennych lokalnych 120
Odczytywanie i ustalanie domyślnych wartości typów 124
Dokładniejsze poznawanie aplikacji konsoli 125
Wyświetlanie informacji dla użytkownika 125
Pobieranie danych od użytkownika 129
Uproszczone korzystanie z konsoli 130
Odczytywanie naciśnięć klawiszy 131
Odczytywanie parametrów aplikacji konsoli 132
Ustalanie opcji za pomocą argumentów 135
Obsługiwanie platform nieobsługujących wybranych API 137
Instrukcje async i await 139
Poprawianie reakcji aplikacji konsoli 139
Praktyka i ćwiczenia 141
Ćwiczenie 2.1 — sprawdź swoją wiedzę 141
Ćwiczenie 2.2 — sprawdź swoją wiedzę o typach liczbowych 141
Ćwiczenie 2.3 — poznaj wielkości i zakresy liczb 142
Ćwiczenie 2.4 — dalsza lektura 143
Podsumowanie 143

ROZDZIAŁ 3
Sterowanie przepływem, konwertowanie typów i obsługa wyjątków **144**
Działania na zmiennych 144
Operatory jednoargumentowe 145
Dwuargumentowe operatory arytmetyczne 146
Operatory przypisania 148
Operatory logiczne 148
Warunkowe operatory logiczne 149
Operatory bitowe i operatory przesunięć 151
Operatory różne 153
Instrukcje wyboru 154
Instrukcja if 154
Dopasowywanie wzorców z instrukcją if 155
Instrukcja switch 156
Dopasowywanie wzorców z instrukcją switch 158
Upraszczanie instrukcji switch za pomocą wyrażeń switch 161
Instrukcje iteracji 162
Instrukcja while 162
Instrukcja do 163
Instrukcja for 163
Instrukcja foreach 164
Zapisywanie wielu wartości w tablicy 165
Praca z tablicami jednowymiarowymi 165
Praca z tablicami wielowymiarowymi 167

Praca z tablicami poszarpanymi 169
Dopasowywanie wzorców list w tablicach 171
Podsumowanie tablic 173
Rzutowanie i konwertowanie między typami 174
Jawne i niejawne rzutowanie liczb 174
Używanie typu System.Convert 176
Zaokrąglanie liczb 177
Kontrolowanie sposobu zaokrąglania 178
Konwersja z dowolnego typu na ciąg znaków 179
Konwertowanie obiektu binarnego na ciąg znaków 179
Parsowanie ciągów znaków z liczbami, datami i czasem 180
Obsługa wyjątków 182
Instrukcja try 183
Wykrywanie przepełnień 188
Instrukcja checked 188
Instrukcja unchecked 189
Praktyka i ćwiczenia 191
Ćwiczenie 3.1 —sprawdź swoją wiedzę 191
Ćwiczenie 3.2 — pętle i przepełnienia 191
Ćwiczenie 3.3 — pętle i operatory 192
Ćwiczenie 3.4 — obsługa wyjątków 192
Ćwiczenie 3.5 — sprawdź swoją wiedzę o operatorach 193
Ćwiczenie 3.6 — dalsza lektura 193
Podsumowanie 193

ROZDZIAŁ 4
Pisanie, debugowanie i testowanie funkcji 194
Tworzenie funkcji 194
Programy i funkcje najwyższego poziomu 195
Przykład z tabliczką mnożenia 197
Dygresja na temat argumentów i parametrów 200
Pisanie funkcji zwracającej wartość 201
Rekurencyjne obliczanie silni 205
Dokumentowanie funkcji za pomocą komentarzy XML 209
Używanie wyrażeń lambda w implementacji funkcji 210
Debugowanie tworzonego programu 213
Używanie zintegrowanego terminala Visual Studio Code podczas debugowania 213
Tworzenie aplikacji z celowym błędem 214
Tworzenie punktu przerwania 215
Pasek narzędzi debugowania 218
Okna debugowania 219
Krokowe wykonywanie kodu 219
Dostosowywanie punktów przerwania 221
Przeładowywanie na gorąco w trakcie programowania 223
Przeładowywanie na gorąco w Visual Studio 2022 224
Przeładowywanie na gorąco w Visual Studio Code i w wierszu poleceń 224

Protokołowanie błędów 225
Dostępne opcje protokołowania 226
Wykorzystywanie typów Debug i Trace 226
Konfigurowanie obiektów nasłuchujących 228
Przełączanie poziomów śledzenia 230
Protokołowanie informacji o kodzie źródłowym 235
Testy jednostkowe 237
Różne rodzaje testów 238
Tworzenie biblioteki klas wymagającej testowania 238
Tworzenie testów jednostkowych 241
Rzucanie i wychwytywanie wyjątków w funkcjach 243
Rozróżnienie błędów użycia i błędów wykonania 244
Wyjątki często rzucane w funkcjach 244
Czym jest stos wywołań? 245
Gdzie należy wychwytywać wyjątki? 249
Ponowne rzucanie wyjątku 249
Implementowanie wzorca tester-wykonawca 251
Praktyka i ćwiczenia 252
Ćwiczenie 4.1 — sprawdź swoją wiedzę 252
Ćwiczenie 4,2 — tworzenie funkcji z wykorzystaniem debugowania i testów jednostkowych 253
Ćwiczenie 4.3 — dalsza lektura 253
Podsumowanie 254

ROZDZIAŁ 5
Tworzenie własnych typów w programowaniu obiektowym 255
Programowanie obiektowe 256
Tworzenie bibliotek klas 257
Tworzenie biblioteki klas 257
Definiowanie klasy w przestrzeni nazw 258
Elementy klasy 260
Tworzenie obiektów 260
Importowanie przestrzeni nazw 262
Poznawanie obiektów 264
Przechowywanie danych w polach 265
Definiowanie pól 265
Modyfikatory dostępu 266
Ustalanie i wypisywanie wartości pól 267
Zapisywanie wartości za pomocą słowa kluczowego enum 268
Przechowywanie wielu wartości w typie enum 269
Zapisywanie wielu wartości za pomocą kolekcji 270
Kolekcje generyczne 271
Tworzenie pól statycznych 272
Tworzenie stałych pól 273
Tworzenie pól tylko do odczytu 275
Inicjowanie pól w konstruktorach 275
Tworzenie i wywoływanie metod 277

Zwracanie wartości z metody 277
Łączenie wielu wartości za pomocą krotki 278
Sterowanie przekazywaniem parametrów 282
Przeciążanie metod 282
Parametry opcjonalne i nazywane 283
Sposoby przekazywania parametrów 285
Zwracanie wartości ze słowem kluczowym ref 287
Dzieleni.e klas na części 287
Kontrola dostępu za pomocą właściwości i indekserów 288
Definiowanie właściwości tylko do odczytu 288
Definiowanie właściwości z możliwością przypisania 289
Wymaganie podania wartości właściwości przy tworzeniu obiektu 291
Definiowanie indekserów 294
Więcej informacji o metodach 296
Implementowanie działań w metodzie 296
Implementowanie działań za pomocą operatora 300
Definiowanie funkcji lokalnych 302
Dopasowywanie wzorców z obiektami 303
Definiowanie listy pasażerów 303
Rozszerzenia dopasowywania wzorców w C# 9 i nowszych 305
Praca z rekordami 306
Właściwości wyłącznie inicjalizowane 306
Rekordy 307
Pozycyjne elementy danych w rekordach 308
Praktyka i ćwiczenia 309
Ćwiczenie 5.1 — sprawdź swoją wiedzę 309
Ćwiczenie 5.2 — dalsza lektura 310
Podsumowanie 310

ROZDZIAŁ 6
Implementowanie interfejsów i dziedziczenie klas 311
Konfigurowanie biblioteki klas i aplikacji konsoli 312
Wykorzystywanie typów generycznych 313
Praca z typami niegenerycznymi 313
Praca z typami generycznymi 315
Wywoływanie i obsługa zdarzeń 316
Wywoływanie metod za pomocą delegatów 316
Definiowanie i obsługa delegatów 318
Definiowanie i obsługiwanie zdarzeń 320
Implementowanie interfejsów 322
Typowe interfejsy 322
Porównywanie obiektów podczas sortowania 323
Porównywanie obiektów za pomocą osobnej klasy 327
Jawne i niejawne implementowanie interfejsów 329
Definiowanie interfejsów z domyślnymi implementacjami 330
Zarządzanie pamięcią za pomocą typów referencyjnych i typów wartości 331
Definiowanie typów referencyjnych i typów wartości 332

Sposób przechowywania w pamięci typów referencyjnych i typów wartości 333
Równość typów 334
Definiowanie typu kategorii struct 336
Praca z typami record struct 337
Zwalnianie niezarządzanych zasobów 338
Wymuszanie wywołania metody Dispose 340
Praca z wartościami null 341
Przekształcanie typu wartości w typ nullowalny 341
Inicjalizowanie typów nullowalnych 343
Poznawanie nullowalnych typów referencyjnych 344
Sterowanie funkcją ostrzeżeń dla typów nullowalnych 345
Deklarowanie nienullowalnych zmiennych i parametrów 345
Sprawdzanie wartości null 348
Kontrolowanie wartości null w parametrach metod 349
Dziedziczenie klas 350
Rozbudowywanie klasy 351
Ukrywanie elementów 351
Pokrywanie elementów klasy 352
Dziedziczenie po klasach abstrakcyjnych 353
Blokowanie dziedziczenia i pokrywania 355
Polimorfizm 355
Rzutowanie w ramach hierarchii dziedziczenia 357
Rzutowanie niejawne 357
Rzutowanie jawne 358
Obsługa wyjątków rzutowania 358
Dziedziczenie i rozbudowywanie typów .NET 360
Dziedziczenie po wyjątku 360
Rozszerzanie typów, po których nie można dziedziczyć 362
Tworzenie lepszego kodu 364
Traktowanie ostrzeżeń jak błędów 364
Fale ostrzegawcze 367
Stosowanie analizatorów, aby tworzyć lepszy kod 369
Praktyka i ćwiczenia 376
Ćwiczenie 6.1 — sprawdź swoją wiedzę 376
Ćwiczenie 6.2 — tworzenie hierarchii dziedziczenia 377
Ćwiczenie 6.3 — dalsza lektura 377
Podsumowanie 377

ROZDZIAŁ 7
Poznawanie typów .NET **379**
Wprowadzenie do .NET 7 379
NET Core 1.0 381
.NET Core 1.1 381
.NET Core 2.0 381
.NET Core 2.1 381
.NET Core 2.2 382

.NET Core 3.0 382
.NET Core 3.1 382
.NET 5.0 382
.NET 6.0 383
.NET 7.0 384
Poprawki wydajności między .NET 5 a nowszymi wersjami 384
Sprawdzanie dostępności aktualizacji .NET SDK 384
Zestawy i przestrzenie nazw 385
Zestawy, pakiety i przestrzenie nazw 385
Poznawanie pakietów SDK dla projektów .NET 386
Przestrzenie nazw i typy w zestawach 387
Pakiety NuGet 387
Czym są frameworki? 388
Importowanie przestrzeni nazw w celu użycia typu 389
Związki słów kluczowych języka C# z typami .NET 389
Wieloplatformowe współdzielenie kodu z bibliotekami klas .NET Standard 392
Domyślne ustawienia bibliotek klas w różnych wersjach SDK 393
Tworzenie biblioteki klas .NET Standard 2.0 394
Kontrolowanie wersji .NET SDK 395
Publikowanie własnych aplikacji 396
Tworzenie aplikacji konsoli do publikacji 397
Poznawanie polecenia dotnet 399
Pobieranie informacji na temat platformy .NET i jej środowiska 400
Zarządzanie projektami 401
Publikowanie samodzielnej aplikacji 402
Publikowanie aplikacji jednoplikowej 404
Zmniejszanie wielkości aplikacji 405
Dekompilowanie zestawów 406
Dekompilowanie za pomocą rozszerzenia ILSpy w Visual Studio 2022 406
Przeglądanie oryginalnych źródeł w Visual Studio 2022 412
Nie, nie można zablokować możliwości dekompilowania 413
Przygotowywanie własnych pakietów NuGet 414
Dodawanie odwołania do pakietu 414
Tworzenie pakietu dla NuGet 416
Przeszukiwanie pakietów NuGet 420
Testowanie pakietu 421
Przenoszenie kodu z .NET Framework do .NET Core 422
Co można przenieść? 423
Co należy przenieść? 424
Różnice między .NET Framework i nowoczesnym .NET 424
Korzystanie z programu .NET Portability Analyzer 425
Asystent uaktualniania programu .NET 425
Używanie bibliotek spoza .NET 425
Praca z proponowanymi funkcjami 427
Wymaganie proponowanych funkcji 428
Włączanie proponowanych funkcji 428

Praktyka i ćwiczenia 429
Ćwiczenie 7.1 — sprawdź swoją wiedzę 429
Ćwiczenie 7.2 — dalsza lektura 429
Ćwiczenie 7.3 — PowerShell 430
Podsumowanie 430

ROZDZIAŁ 8
Używanie typów .NET 431
Praca z liczbami 431
Praca z wielkimi liczbami całkowitymi 432
Praca z liczbami zespolonymi 433
Kwaterniony 434
Generowanie liczb losowych na potrzeby gier i podobnych aplikacji 434
Praca z tekstem 435
Odczytywanie długości ciągu znaków 435
Odczytywanie znaków z ciągu 436
Dzielenie ciągu znaków 436
Pobieranie części ciągu znaków 437
Poszukiwanie tekstu w ciągu 438
Inne elementy klasy string 439
Wydajne tworzenie ciągów znaków 440
Dopasowywanie wzorców za pomocą wyrażeń regularnych 441
Kontrolowanie cyfr wprowadzonych jako tekst 441
Poprawianie wydajności wyrażeń regularnych 443
Składnia wyrażenia regularnego 443
Przykłady wyrażeń regularnych 444
Dzielenie złożonych ciągów znaków rozdzielanych przecinkami 445
Włączanie kolorowania składni wyrażeń regularnych 446
Poprawianie wydajności wyrażeń regularnych za pomocą generatorów kodu 450
Praca z kolekcjami 452
Wspólne funkcje wszystkich kolekcji 453
Poprawianie wydajności przez zdefiniowanie pojemności kolekcji 454
Poznawanie kolekcji 454
Praca z listami 459
Praca ze słownikami 460
Praca z kolejkami 461
Sortowanie kolekcji 464
Używanie specjalizowanych kolekcji 465
Używanie kolekcji niezmiennych 465
Dobre praktyki w pracy z kolekcjami 466
Praca z typem Span, indeksami i zakresami 467
Wydajne korzystanie z pamięci za pomocą typu Span 467
Określanie pozycji za pomocą typu Index 468
Definiowanie zakresów za pomocą typu Range 468
Używanie indeksów i zakresów 469
Praca z zasobami sieciowymi 470

Praca z adresami URI, serwerami DNS i adresami IP 470
Pingowanie serwera 472
Praktyka i ćwiczenia 473
Ćwiczenie 8.1 —sprawdź swoją wiedzę 473
Ćwiczenie 8.2 — wyrażenia regularne 474
Ćwiczenie 8.3 — metody rozszerzające 474
Ćwiczenie 8.4 — dalsza lektura 474
Podsumowanie 475

ROZDZIAŁ 9
Praca z plikami, strumieniami i serializacją 476
Praca z systemem plików 476
Obsługa środowisk i systemów plików na wielu platformach 476
Obsługa napędów 479
Praca z katalogami 480
Praca z plikami 481
Praca ze ścieżkami 483
Odczytywanie informacji o pliku 484
Zarządzanie plikami 485
Odczytywanie i zapisywanie w strumieniach 485
Strumienie abstrakcyjne i konkretne 486
Praca ze strumieniami tekstowymi 488
Praca ze strumieniami XML 490
Uproszczenie zwalniania zasobów za pomocą instrukcji using 492
Strumienie kompresujące 493
Praca z archiwami tar 496
Kodowanie tekstu 501
Kodowanie ciągu znaków jako tablicy bajtów 502
Kodowanie i dekodowanie tekstu w plikach 505
Odczytywanie i zapisywanie w urządzeniach o dostępie swobodnym 505
Serializacja obiektów 506
Serializacja do formatu XML 507
Generowanie kompaktowej struktury XML 510
Deserializacja danych z formatu XML 510
Serializowanie do formatu JSON 511
Wydajne przetwarzanie danych w formacie JSON 513
Kontrolowane przetwarzanie danych JSON 515
Nowe metody rozszerzające, które ułatwiają pracę z odpowiedziami HTTP 518
Przenoszenie kodu z biblioteki Newtonsoft do nowej biblioteki 518
Praktyka i ćwiczenia 518
Ćwiczenie 9.1 — sprawdź swoją wiedzę 518
Ćwiczenie 9.2 — serializowanie do formatu XML 519
Ćwiczenie 9.3 — dalsza lektura 519
Podsumowanie 520

ROZDZIAŁ 10

Praca z bazami danych przy użyciu Entity Framework Core 521
Nowoczesne bazy danych 521
Czym jest Entity Framework? 522
Entity Framework Core 523
Co znaczy „najpierw baza danych" i „najpierw kod"? 524
Usprawnienia wydajności w EF Core 7 524
Tworzenie aplikacji konsoli do pracy z EF Core 524
Używanie przykładowej relacyjnej bazy danych 525
Używanie SQLite 526
Tworzenie przykładowej bazy danych Northwind na serwerze SQLite 527
Zarządzanie przykładową bazą danych Northwind za pomocą SQLiteStudio 529
Używanie lekkiego dostawcy ADO.NET dla SQLite 531
Używanie Microsoft SQL Server w systemie Windows 531
Konfigurowanie EF Core 531
Wybieranie dostawcy danych EF Core 531
Łączenie z bazą danych 532
Definiowanie klasy kontekstu bazy danych Northwind 532
Definiowanie modeli EF Core 534
Konwencje w EF Core 534
Atrybuty EF Core 535
Płynne API EF Core 536
Tworzenie modelu w EF Core 537
Dodawanie tabel do klasy kontekstu bazy danych Northwind 540
Konfigurowanie narzędzia dotnet-ef 541
Tworzenie modeli na podstawie istniejącej bazy danych 542
Dostosowywanie szablonów wstecznej inżynierii 546
Konfigurowanie konwencji 546
Zapytania do modelu EF Core 547
Filtrowanie dołączanych encji 550
Filtrowanie i sortowanie produktów 552
Pobieranie generowanych instrukcji SQL 554
Protokołowanie w EF Core 555
Dopasowywanie wzorców za pomocą instrukcji Like 557
Generowanie liczb losowych w zapytaniach 559
Definiowanie globalnych filtrów 560
Wzorce ładowania w EF Core 561
Chętne ładowanie encji za pomocą metody rozszerzającej Include 561
Włączenie leniwego ładowania 562
Jawne ładowanie encji za pomocą metody Load 563
Manipulowanie danymi w EF Core 566
Wstawianie encji 566
Aktualizowanie encji 569
Usuwanie encji 571
Wydajniejsze aktualizowanie i usuwanie 572
Grupowanie kontekstów baz danych 576

Transakcje 576
Sterowanie transakcjami za pomocą poziomów izolacji 577
Jawne definiowanie transakcji 577
Modele Code First w EF Core 578
Praktyka i ćwiczenia 579
Ćwiczenie 10.1 — sprawdź swoją wiedzę 579
Ćwiczenie 10.2 — eksportowanie danych z wykorzystaniem różnych formatów serializacji 579
Ćwiczenie 10.3 — dalsza lektura 580
Ćwiczenie 10.4— poznawanie baz danych NoSQL 580
Podsumowanie 580

ROZDZIAŁ 11
Odczytywanie danych i manipulowanie nimi za pomocą LINQ 581
Dlaczego LINQ? 581
Porównanie imperatywnych i deklaratywnych funkcji języka 581
Tworzenie wyrażeń LINQ 582
Z czego składa się LINQ? 583
Rozbudowa sekwencji za pomocą klas wyliczeniowych 583
Czym jest opóźnione wykonanie? 586
Filtrowanie encji za pomocą metody Where 588
Korzystanie z metody nazwanej 590
Upraszczanie kodu przez usunięcie jawnego tworzenia delegata 590
Korzystanie z wyrażenia lambda 591
Sortowanie encji 592
Sortowanie według elementów 593
Deklarowanie zapytania za pomocą słowa kluczowego var lub określonego typu 593
Filtrowanie według typu 594
Praca ze zbiorami 595
Używanie LINQ z EF Core 597
Tworzenie modelu danych EF Core 598
Filtrowanie i sortowanie sekwencji 601
Projekcje sekwencji na nowe typy 603
Łączenie i grupowanie 605
Agregowanie sekwencji 608
Uważaj na właściwość Count! 611
Stronicowanie z LINQ 612
Upiększanie składni 617
Używanie wielu wątków w równoległych zapytaniach LINQ 618
Tworzenie własnych metod rozszerzających dla LINQ 618
Próba użycia nowych metod rozszerzających 621
Praca z LINQ to XML 622
Generowanie danych XML za pomocą LINQ to XML 622
Odczytywanie danych XML za pomocą LINQ to XML 623
Praktyka i ćwiczenia 625
Ćwiczenie 11.1 — sprawdź swoją wiedzę 625

Ćwiczenie 11.2 — zapytania LINQ 625
Ćwiczenie 11.3 — dalsza lektura 626
Podsumowanie 626

ROZDZIAŁ 12
Wprowadzenie do aplikacji sieciowych w ASP.NET Core 627
Czym jest ASP.NET Core? 628
Klasyczna ASP.NET kontra ASP.NET Core 629
Tworzenie stron WWW za pomocą ASP.NET Core 630
Tworzenie serwisów sieciowych 632
Nowe funkcje w ASP.NET Core 633
ASP.NET Core 1.0 633
ASP.NET Core 1.1 633
ASP.NET Core 2.0 633
ASP.NET Core 2.1 633
ASP.NET Core 2.2 634
ASP.NET Core 3.0 634
ASP.NET Core 3.1 635
Blazor WebAssembly 3.2 635
ASP.NET Core 5.0 635
ASP.NET Core 6.0 635
ASP.NET Core 7.0 636
Struktury projektów 637
Struktura projektów w rozwiązaniu lub przestrzeni roboczej 637
Tworzenie modelu danych dla bazy danych Northwind 638
Tworzenie biblioteki klas dla modelu encji Northwind 639
Tworzenie biblioteki klas modelu encji dla SQL Server 649
Testowanie bibliotek klas 652
Tworzenie w sieci WWW 654
Protokół HTTP 654
Używanie Google Chrome do wykonywania żądań HTTP 656
Tworzenie oprogramowania dla sieci WWW po stronie klienta 659
Praktyka i ćwiczenia 660
Ćwiczenie 12.1 — sprawdź swoją wiedzę 660
Ćwiczenie 12.2 — znasz te skrótowce? 660
Ćwiczenie 12.3 — dalsza lektura 661
Podsumowanie 661

ROZDZIAŁ 13
Tworzenie witryn WWW przy użyciu ASP.NET Core Razor Pages 662
ASP.NET Core 662
Tworzenie pustego projektu ASP.NET Core 662
Testowanie i zabezpieczanie witryny 665
Kontrola środowiska hostingowego 670
Włączanie plików statycznych 672
Technologia Razor Pages 674
Włączanie technologii Razor Pages 674

Definiowanie strony Razor 675
Używanie wspólnego układu w wielu stronach Razor 677
Używanie plików code-behind w stronach Razor 680
Używanie Entity Framework Core z ASP.NET Core 682
Konfigurowanie Entity Framework Core jako serwisu 682
Manipulowanie danymi na stronach Razor 685
Wstrzykiwanie zależnego serwisu na stronę Razor 687
Używanie bibliotek klas Razor 688
Wyłączanie kompaktowych folderów w Visual Studio Code 688
Tworzenie biblioteki klas Razor 689
Implementowanie widoku cząstkowego do wyświetlania danych pracownika 692
Używanie i testowanie biblioteki klas Razor 693
Konfigurowanie serwisów i potoku obsługi żądań HTTP 694
Routowanie punktów końcowych 694
Konfigurowanie routowania punktów końcowych 695
Przeglądanie konfiguracji routowania punktów końcowych w naszym projekcie 696
Przygotowywanie potoku obsługi żądań HTTP 698
Podsumowanie najważniejszych metod rozszerzających oprogramowania pośredniego 699
Wizualizacja potoku HTTP 700
Implementowanie oprogramowania pośredniego jako anonimowego delegata 701
Włączanie obsługi dekompresji żądań 703
Włączanie obsługi HTTP/3 704
Praktyka i ćwiczenia 706
Ćwiczenie 13.1 — sprawdź swoją wiedzę 706
Ćwiczenie 13.2 — tworzenie witryny obsługującej dane 707
Ćwiczenie 13.3 — zastępowanie aplikacji konsoli stronami WWW 707
Ćwiczenie 13.4 — dalsza lektura 707
Podsumowanie 707

ROZDZIAŁ 14
Tworzenie aplikacji WWW przy użyciu ASP.NET Core MVC 709
Konfigurowanie witryny ASP.NET Core MVC 709
Tworzenie witryny ASP.NET Core MVC 710
Tworzenie bazy danych uwierzytelniania na serwerze SQL Server LocalDB 712
Przeglądanie domyślnej witryny ASP.NET Core MVC 714
Uruchamianie projektu witryny MVC 715
Rejestrowanie użytkowników 716
Przegląd szablonu projektu ASP.NET Core MVC 717
Sprawdzanie bazy danych ASP.NET Core Identity 719
Poznawanie mechanizmów ASP.NET Core MVC 720
Rozruch ASP.NET Core 720
Czym jest domyślna ścieżka? 723

Kontrolery i akcje 724
Konwencja wyszukiwania ścieżek widoku 728
Protokołowanie z wykorzystaniem serwisu zależnego 729
Modele encji i widoków 730
Widoki 733
Jak działa cache busting włączany znacznikami pomocniczymi? 736
Dostosowywanie witryny ASP.NET Core MVC 737
Definiowanie własnych stylów 737
Konfigurowanie obrazków dla kategorii 737
Składnia i wyrażenia Razor 737
Definiowanie typu dla widoku 738
Przekazywanie parametrów przy użyciu wartości ścieżki 742
Wiązanie modeli 745
Sprawdzanie poprawności modelu 749
Poznawanie pomocniczych metod widoku 752
Definiowanie widoków za pomocą znaczników pomocniczych 753
Czym są filtry? 754
Zapisywanie danych wyjściowych w pamięci podręcznej 761
Odczytywanie danych z bazy i używanie szablonów wyświetlania 767
Poprawianie skalowalności za pomocą asynchronicznych zadań 770
Przygotowanie asynchronicznych metod akcji kontrolera 771
Praktyka i ćwiczenia 772
Ćwiczenie 14.1 —sprawdź swoją wiedzę 772
Ćwiczenie 14.2 — implementowanie wzorca MVC w tworzonej stronie danych kategorii 773
Ćwiczenie 14.3 — poprawianie skalowalności przez poznawanie i implementowanie asynchronicznych metod akcji 773
Ćwiczenie 14.4 — testy jednostkowe dla kontrolerów 774
Ćwiczenie 14.5 — dalsza lektura 774
Podsumowanie 774

ROZDZIAŁ 15
Tworzenie i używanie serwisów sieciowych 775
Tworzenie serwisów w technologii ASP.NET Core Web API 775
Skróty stosowane w serwisach sieciowych 776
Żądania i odpowiedzi HTTP w Web API 776
Tworzenie projektu ASP.NET Core Web API 779
Sprawdzanie funkcji serwisu sieciowego 783
Tworzenie serwisu internetowego dla bazy danych Northwind 785
Tworzenie repozytorium danych dla encji 787
Implementowanie kontrolera Web API 790
Konfigurowanie repozytorium klientów i kontrolera Web API 793
Podawanie szczegółów problemu 797
Kontrola nad serializacją XML 798
Dokumentowanie i testowanie serwisów 799
Testowanie żądań GET za pomocą przeglądarki 799
Testowanie żądań HTTP za pomocą rozszerzenia REST Client 800

Włączanie narzędzia Swagger 804
Testowanie żądań w narzędziu SwaggerUI 805
Włączanie protokołowania HTTP 809
Obsługa protokołowania dodatkowych nagłówków żądań w systemie W3CLogger 812
Używanie serwisu za pomocą klientów HTTP 812
Klasa HttpClient 813
Konfigurowanie klientów HTTP za pomocą klasy HttpClientFactory 813
Pobieranie w kontrolerze listy klientów w formacie JSON 814
Uruchamianie wielu projektów 817
Uruchamianie projektów serwisu sieciowego i klienta MVC 819
Implementowanie zaawansowanych funkcji 820
Tworzenie serwisów sieciowych używających minimalnego API 820
Testowanie minimalnego serwisu pogodowego 823
Dodawanie prognozy pogody do głównej strony witryny Northwind 823
Praktyka i ćwiczenia 826
Ćwiczenie 15.1 — sprawdź swoją wiedzę 826
Ćwiczenie 15.2 — ćwiczenia w tworzeniu i usuwaniu klientów za pomocą HttpClient 827
Ćwiczenie 15.3 — dalsza lektura 827
Podsumowanie 827

ROZDZIAŁ 16
Tworzenie interfejsów użytkownika w technologii Blazor 828
Technologia Blazor 828
JavaScript i podobne 829
Silverlight — C# i .NET w formie wtyczki 829
WebAssembly — podstawa technologii Blazor 830
Różne modele hostowania komponentów Blazora 830
Omówienie komponentów tworzonych za pomocą Blazora 831
Czym różnią się Blazor i Razor? 832
Porównanie szablonów projektów Blazor 833
Przeglądanie szablonu projektu Blazor Server 833
Routowanie do komponentów stronicowych Blazora 840
Uruchamianie szablonu projektu Blazor Server 843
Przeglądanie szablonu projektu Blazor WebAssembly 845
Tworzenie komponentów za pomocą Blazor Server 850
Definiowanie i testowanie prostego komponentu 850
Przekształcanie komponentu w routowalny komponent stronicowy 851
Dodawanie encji do komponentu 852
Tworzenie abstrakcji serwisu dla komponentu Blazora 856
Definiowanie formularzy za pomocą komponentu EditForm 858
Tworzenie i używanie komponentu formularza danych klienta 859
Tworzenie komponentów do tworzenia, edytowania i usuwania klientów 861
Testowanie komponentu formularza danych klienta 863
Tworzenie komponentów za pomocą Blazor WebAssembly 864
Konfigurowanie serwera dla projektu Blazor WebAssembly 865

Konfigurowanie klienta aplikacji Blazor WebAssembly 868
Testowanie komponentów i serwisu Blazor WebAssembly 871
Usprawnianie aplikacji tworzonych za pomocą Blazor WebAssembly 873
Praktyka i ćwiczenia 873
Ćwiczenie 16.1 —sprawdź swoją wiedzę 873
Ćwiczenie 16.2 — przygotowanie komponentu tabliczki mnożenia 874
Ćwiczenie 16.3 — przygotowanie elementu nawigowania według krajów 874
Ćwiczenie 16.4 — dalsza lektura 875
Podsumowanie 875

Epilog **876**
Następne kroki Twojej podróży w świecie C# i .NET 876
Poprawianie swoich umiejętności za pomocą wskazówek projektowych 876
Książka uzupełniająca 877
Z których książek uczyć się dalej? 878
Następne wydanie tej książki 878
Powodzenia! 878

DODATEK A
Odpowiedzi na pytania z testów **879**

oprac. BPK