

Stručný obsah

Předmluva	19
1. Úvod do jazyka C#	23
2. C# a CLR	33
3. Přehled syntaxe jazyka C#	41
4. Třídy, struktury a objekty	65
5. Rozhraní a smluvní vztahy	159
6. Přetěžování operátorů	187
7. Zabezpečení zdrojového kódu proti výjimkám a jejich ošetřování	203
8. Práce s řetězci	237
9. Pole, kolekce a iterátory	265
10. Delegáty, anonymní funkce a události	301
11. Genericita	327
12. Práce s podprocesy v jazyku C#	379
13. Hledání kanonických tvarů v jazyce C#	445
14. Rozšiřující metody	503
15. Lambda-výrazy	529
16. LINQ: Dotazy integrované do jazyka	553
17. Dynamické typy	587

Obsah

O autorovi	17
Odborný korektor	17
Poděkování.....	18
Předmluva	19
O této knize.....	20
Kapitola 1	
Úvod do jazyka C#	23
Rozdíly mezi jazyky C# a C++.....	23
C#.....	23
C++	24
Automatická správa paměti (garbage collector) v CLR	25
Příklad programu v jazyce C#.....	26
Přehled nových vlastností C# 2.0	27
Přehled nových vlastností C# 3.0	28
Přehled nových vlastností C# 4.0	30
Shrnutí	30
Kapitola 2	
C# a CLR.....	33
Překladač JIT a CLR.....	34
Sestavení a zavaděč sestavení	35
Minimalizace množiny stránek aplikace	36
Názvy sestavení.....	37
Načítání sestavení.....	37
Metadata.....	38
Kompatibilita napříč jazyky	39
Shrnutí	40

Kapitola 3

Přehled syntaxe jazyka C# 41

C# je silně typový jazyk	41
Výrazy	42
Příkazy a výrazy	44
Typy a proměnné	44
Hodnotové typy	46
Referenční typy	49
Výchozí inicializace proměnných	50
Implicitně typované lokální proměnné	51
Konverze typů	53
Operátory as a is	55
Genericita	57
Jmenné prostory	58
Deklarace jmenných prostorů	59
Používání jmenných prostorů	60
Řízení toku programu	62
if-else, while, do-while a for	62
switch	62
foreach	63
break, continue, goto, return a throw	64
Shrnutí	64

Kapitola 4

Třídy, struktury a objekty 65

Definice tříd	67
Složky	68
Konstruktory	71
Metody	72
Vlastnosti	74
Zapouzdření	78
Přístupnost	82
Rozhraní	84
Dědění	85
Zapečetěné třídy	92
Abstraktní třídy	93
Vnořené třídy	94
Indexery	97
Částečné třídy	99

Částečné metody.....	100
Statické třídy.....	101
Vyhrazené názvy členů.....	103
Definice hodnotových typů.....	104
Konstruktory.....	105
Význam klíčového slova this	107
Finalizéry.....	109
Rozhraní	110
Anonymní typy	110
Inicializátory objektů	113
Balení a vybalování hodnotových typů	116
Kdy dojde k zabalení	120
Efektivita a zmatek	122
System.Object	122
Rovnost a její význam	124
Rozhraní IComparable	124
Vytváření objektů	125
Klíčové slovo new	125
Inicializace složek.....	126
Statické konstruktory (konstruktory tříd)	127
Konstruktor instance a pořadí operací při jejím vytváření.....	130
Destrukce objektů	134
Finalizéry.....	134
Deterministická destrukce	136
Ošetřování výjimek	137
Jednorázové objekty	137
Rozhraní IDisposable.....	137
Klíčové slovo using.....	140
Typy parametrů metod.....	141
Hodnotové parametry	142
Parametry předávané odkazem (ref)	142
Výstupní parametry (out)	144
Proměnný počet parametrů (params)	144
Přetěžování metod	145
Volitelné parametry	145
Pojmenované parametry	146
Dědění a virtuální metody.....	150
Virtuální a abstraktní metody.....	150
Metody s modifikátory override a new	150
Zapečetěné metody (sealed)	152

Několik posledních slov o virtuálních metodách v jazyce C#	153
Dědění, skládání a delegace	153
Volba mezi použitím rozhraní a děděním tříd	153
Delegace a kompozice vs. dědění	155
Shrnutí	157

Kapitola 5

Rozhraní a smluvní vztahy..... 159

Rozhraní definují typy	160
Definice rozhraní	161
Co může v rozhraní být	162
Dědičnost rozhraní a skrývání členů	162
Implementace rozhraní	165
Implicitní implementace rozhraní	165
Explicitní implementace rozhraní	165
Překrývání implementací rozhraní v odvozené třídě	167
Pozor na vedlejší účinky hodnotových typů, které implementují rozhraní	171
Pravidla přiřazování členům rozhraní	172
Explicitní implementace rozhraní a hodnotové typy	176
Práce s verzemi	178
Kontrakty	179
Kontrakty implementované třídami	179
Kontrakt rozhraní	181
Výběr mezi rozhraními a třídami	181
Shrnutí	185

Kapitola 6

Přetěžování operátorů 187

To, že můžete, ještě neznamená, že byste měli	187
Typy a formáty přetížených operátorů	188
Operátory by neměly měnit své operandy	189
Záleží na pořadí parametrů?	189
Přetížení operátoru pro sčítání	190
Operátory, které lze přetěžovat	191
Porovnávací operátory	192
Konverzní operátory	195
Logické operátory	198
Shrnutí	201

Kapitola 7

Zabezpečení zdrojového kódu proti výskytu výjimek a jejich ošetřování 203

Jak CLR nakládá s výjimkami	204
Mechanismus ošetřování výjimek v CLR	204
Vývolávání výjimek	204
Změny ve způsobu zpracování neošetřených výjimek ve verzi .NET 2.0	205
Přehled syntaxe příkazů try, catch a finally	206
Opětovné vyvolávání výjimek a překlad výjimek	208
Výjimky vyvolané v blocích finally	210
Výjimky vyvolané ve finalizérech	211
Výjimky ve statických konstruktorech	212
Kdo by měl výjimky ošetřovat?	213
Vyhnete se využívání výjimek k řízení běhu programu	214
Jak docílit neutrálního chování při výskytu výjimek	214
Základní struktura zdrojového kódu, který se chová při výskytu výjimek neutrálně	215
Omezené prováděcí oblasti	221
Kritické finalizéry a třída SafeHandle	223
Vytváření vlastních tříd výjimek	227
Práce s alokovanými prostředky a výjimkami	229
Implementace zrušení změn („rollback“)	233
Shrnutí	236

Kapitola 8

Práce s řetězci 237

Základní údaje o typu String	237
Řetězcové literály	238
Specifikátory formátu a globalizace	239
Object.ToString, IFormattable a CultureInfo	240
Vytváření a registrace uživatelských typů CultureInfo	241
Formátovací řetězce	243
Metody Console.WriteLine a String.Format	244
Příklady formátování řetězců v uživatelských typech	245
Rozhraní ICustomFormatter	247
Porovnávání řetězců	249
Práce s řetězci z vnějších zdrojů	250
StringBuilder	253
Prohledávání řetězců pomocí regulárních výrazů	254
Vyhledávání regulárních výrazů	255

Vyhledávání a seskupování.....	256
Nahrazení textu regulárním výrazem	260
Možnosti vytváření regulárních výrazů	262
Shrnutí	264

Kapitola 9

Pole, kolekce a iterátory 265

Úvod do problematiky polí	265
Implicitně typovaná pole.....	267
Přetypování a kovariance	269
Třídění a prohledávání.....	270
Synchronizace.....	271
Vektory vs. pole	271
Pravouhlá multidimenzionální pole	273
Nepravidelná multidimenzionální pole.....	275
Typy kolekcí	276
Srovnání rozhraní ICollection<T> a ICollection.....	277
Synchronizace kolekcí.....	278
Seznamy.....	279
Slovníky.....	280
Množiny.....	281
Jmenný prostor System.Collections.ObjectModel	281
Efektivita	284
Rozhraní IEnumerable<T>, IEnumerator<T>, IEnumerable a IEnumerator	285
Typy, které produkují kolekce	289
Iterátory	289
Dopředné, zpětné a obousměrné iterátory.....	294
Inicializátory kolekcí	298
Shrnutí	299

Kapitola 10

Delegáty, anonymní funkce a události 301

Přehled problematiky delegátů.....	301
Vytváření a používání delegátů.....	303
Samostatný delegát.....	303
Řetězení delegátů.....	304
Iterování řetězcem delegátů	306
Otevřené delegáty (unbound, open instance)	308
Události	310

Anonymní metody	314
Zachycené proměnné a uzávěry	317
Pozor na překvapení, které vám může přichystat zachycená proměnná	319
Anonymní metody jako nástroj pro vazbu parametrů delegátů	322
Návrhový vzor strategie	325
Shrnutí	326

Kapitola 11

Genericita	327
Rozdíl mezi generickými typy a šablonami v C++	328
Efektivita a typová bezpečnost generických konstrukcí	329
Definice generických typů a konstruované typy	331
Generické třídy a struktury	332
Generická rozhraní	335
Generické metody	335
Generické delegáty	337
Konverze generických typů	341
Výrazy výchozích hodnot	342
Nulovatelné typy	343
Přístupnost konstruovaných typů	345
Genericita a dědičnost	345
Omezení	347
Omezení netřídových typů	351
Kovariance a kontravariance	352
Kovariance	354
Kontravariance	357
Invariance	359
Variance a delegáty	359
Generické systémové kolekce	363
Generická systémová rozhraní	365
Vybrané problémy a jejich řešení	366
Konverze a operátory v generických typech	367
Dynamické vytváření konstruovaných typů	376
Shrnutí	378

Kapitola 12

Práce s podprocesy v jazyku C#	379
Práce s podprocesy v jazyce C# a v prostředí .NET	380
Spouštění podprocesů	380

Předávání dat novým podprocesům	382
Použití delegátového typu ParameterizedThreadStart	383
Návrhový vzor IOU a asynchronní volání metod	384
Stavy podprocesů	384
Ukončování podprocesů	387
Zastavování podprocesů a probouzení spících podprocesů	389
Čekání na ukončení podprocesu	390
Hlavní a vedlejší podprocesy	390
Lokální úložiště podprocesu	391
Jak k sobě pasují neřízené podprocesy a apartmenty COM	395
Synchronizace činnosti mezi podprocesy	396
Odlehčená synchronizace s třídou Interlocked	397
Třída SpinLock	403
Třída Monitor	405
Dejte si pozor na zabalování	409
Metody Pulse a Wait	410
Zámky	414
Třída ReaderWriterLock	414
Třída ReaderWriterLockSlim	417
Mutex	418
Semafor	419
Události	422
Synchronizační objekty rozhraní Win32 a třída WaitHandle	423
Použití třídy ThreadPool	425
Asynchronní volání metod	426
Časovače	434
Souběžné programování	435
Třída Task	436
Třída Parallel	437
Snadný vstup do fondu podprocesů	441
Kolekce bezpečné vzhledem k vícepodprocesovému zpracování	443
Shrnutí	444

Kapitola 13

Hledání kanonických tvarů v jazyce C# 445

Kanonické formy pro referenční typy	446
Třídy standardně zapečetujte	446
Použijte vzor Nevirtuální rozhraní (NVI)	447
Lze objekt klonovat?	450
Lze objekt odstranit?	456

Potřebuje objekt finalizační metodu?	458
Co pro objekt znamená rovnost?	466
Referenční typy a identita	466
Hodnotová rovnost.....	469
Překrytí metody Object.Equals pro referenční typy	470
S metodou Equals překryjte také metodu GetHashCode	473
Podporuje třída řazení?	476
Lze objekt formátovat?	479
Je objekt převoditelný?	483
Vždy dávejte přednost typové bezpečnosti	485
Použití neměnných referenčních typů	489
Kanonické tvary hodnotových typů	492
Překryjte metodu Equals kvůli lepšímu výkonu.....	492
Podporuje daný typ nějaká rozhraní?	496
Implementujte typové bezpečné formy složek rozhraní a odvozených metod	497
Shrnutí	500
Kontrolní seznam pro referenční typy	500
Kontrolní seznam pro hodnotové typy	502

Kapitola 14

Rozšiřující metody 503

Seznámení s rozšiřujícími metodami.....	503
Jak najde překladač rozšiřující metodu?	504
Pod pokličkou	507
Čitelnost kódu versus jeho srozumitelnost	508
Praktická doporučení	509
Zvažte použití rozšiřujících metod místo dědění	509
Vyčleňte rozšiřující metody do samostatného jmenného prostoru	511
Změna kontraktu typu může rozbít rozšiřující metody	512
Transformace	512
Řetězení operací.....	516
Vlastní iterátory	518
Vypůjčeno z funkcionálního programování	519
Vzor Návštěvník.....	525
Shrnutí	528

Kapitola 15

Lambda-výrazy 529

Seznámení s lambda-výrazy	529
Lambda-výrazy a uzávěry	530

Uzávěry v jazyku C# 1.0	533
Uzávěry v jazyce C# 2.0	535
Lambda-příkazy	535
Výrazové stromy	536
Operování na výrazech	539
Funkce jako data	539
Užitečné aplikace lambda-výrazů	540
Revize iterátorů a generátorů	540
Uzávěry (zachycení proměnných) a tabelace	544
Vazba parametrů (curryfikace)	548
Anonymní rekurze	550
Shrnutí	552

Kapitola 16

LINQ: Dotazy integrované do jazyka553

Most k datům	554
Dotazové výrazy	554
Revize rozšiřujících metod a lambda-výrazů	556
Standardní dotazové operátory	557
Dotazová klíčová slova jazyka C#	559
Klauzule from a rozsahové proměnné	559
Klauzule join	561
Klauzule where a filtry	563
Klauzule orderby	563
Klauzule select a projekce	564
Klauzule let	566
Klauzule group	567
Klauzule into a pokračování	570
Přednosti lenosti	571
Iterátory jazyka C# podporují lenost	572
Nižitelé lenosti	573
Okamžité provádění dotazů	574
Revize výrazových stromů	575
Techniky z funkcionálního programování	575
Vlastní standardní dotazové operátory a líné vyhodnocování	576
Nahrazení příkazu foreach	583
Shrnutí	585

Kapitola 17

Dynamické typy 587

Co se skrývá za klíčovým slovem dynamic	587
Jak dynamické typy fungují	590
Veliké sjednocení	592
Místa volání	592
Objekty s vlastním dynamickým chováním	594
Efektivita	597
Zabalování s typem dynamic	598
Dynamické převody	599
Implicitní převod dynamických výrazů	600
Dynamický výběr přetížené varianty	601
Dynamické dědění	603
Od typu dynamic nelze odvozovat nové třídy	603
Nelze implementovat rozhraní s typem dynamic	604
Nové třídy lze odvozovat od dynamických bázevých typů	606
Kachní typování v jazyku C#	607
Omezení typu dynamic	610
Dynamické vytváření objektů pomocí třídy ExpandableObject	610
Shrnutí	614

Rejstřík 615