

# Stručný obsah

## Část I

---

<b>Seznámení s jazykem Microsoft Visual C# a aplikací Microsoft Visual Studio 2010</b>	<b>27</b>
1. Vítejte v C#	29
2. Práce s proměnnými, operátory a výrazy	51
3. Psaní metod a aplikace oboru platnosti	69
4. Rozhodovací příkazy a jejich použití	91
5. Používáme složená přiřazení a iterační příkazy	107
6. Ošetřování chyb a výjimek	123

## Část II

---

<b>Poznáváme jazyk C#</b>	<b>141</b>
7. Vytváření a správa tříd a objektů	143
8. Poznáváme hodnotové a referenční typy	161
9. Vytváření hodnotových typů pomocí výčtů a struktur	181
10. Používáme pole a kolekce	197
11. Seznámení s poli parametrů	223
12. Pracujeme s dědičností	233
13. Tvorba rozhraní a abstraktních tříd	253
14. Úklid paměti a správa prostředků	275

## Část III

---

<b>Vytváření komponent</b>	<b>289</b>
15. Implementace vlastností pro přístup k datovým složkám	291
16. Indexery a jejich používání	309
17. Přerušování toku programu a zpracování událostí	323
18. Úvod do generických typů	345
19. Enumerace kolekcí	371
20. Dotazování na paměťová data pomocí dotazových výrazů	383
21. Přetěžování operátorů	405

## Část IV

---

### **Tvorba grafických aplikací v subsystému WPF** **425**

- |     |   |     |
|-----|---|-----|
| 22. | Seznámení s grafickým subsystémem Windows Presentation Foundation | 427 |
| 23. | Získávání vstupu od uživatele                                     | 459 |
| 24. | Ověřování zadaných hodnot   | 487 |

## Část V

---

### **Správa dat** **509**

- |     |  |     |
|-----|--|-----|
| 25. | Dotazování na informace v databázi                                     | 511 |
| 26. | Zobrazení a úprava dat pomocí rozhraní Entity Framework a datové vazby | 539 |

## Část VI

---

### **Tvorba profesionálních řešení ve Visual Studiu 2010** **569**

- |     |   |     |
|-----|---|-----|
| 27. | Seznámení s knihovnou TPL (Task Parallel Library) | 571 |
| 28. | Paralelní přístup k datům                         | 617 |
| 29. | Tvorba a použití webových služeb                  | 649 |
| P.  | Spolupráce s dynamickými jazyky                   | 679 |

# Obsah

<b>Poděkování</b>	<b>15</b>
<b>Úvod</b>	<b>17</b>
<b>Komu je kniha určena</b>	<b>17</b>
<b>Najděte si optimální výchozí bod</b>	<b>17</b>
<b>Konvence používané v této knize</b>	<b>18</b>
Konvence	18
Další prvky	19
<b>Předběžné vydání softwaru</b>	<b>19</b>
<b>Systémové požadavky</b>	<b>19</b>
<b>Cvičební soubory</b>	<b>20</b>
Instalace cvičebních souborů	20
Používání cvičebních souborů	20
<b>Zpětná vazba od čtenářů</b>	<b>24</b>
<b>Dotazy</b>	<b>25</b>
<b>Errata</b>	<b>25</b>

## Část I

### **Seznámení s jazykem Microsoft Visual C# a aplikací Microsoft Visual Studio 2010**

#### Kapitola 1

---

<b>Vítejte v C#</b>	<b>29</b>
<b>Začínáme programovat v prostředí Visual Studia 2010</b>	<b>29</b>
<b>Píšeme první program</b>	<b>34</b>
<b>Používání oborů názvů</b>	<b>39</b>
<b>Vytvoření grafické aplikace</b>	<b>41</b>
<b>Stručné shrnutí kapitoly 1</b>	<b>50</b>

#### Kapitola 2

---

<b>Práce s proměnnými, operátory a výrazy</b>	<b>51</b>
<b>Příkazy</b>	<b>51</b>
<b>Používáme identifikátory</b>	<b>52</b>
Klíčová slova	52
<b>Proměnné</b>	<b>53</b>
Pojmenování proměnných	53
Deklarování proměnných	54
<b>Práce s primitivními datovými typy</b>	<b>54</b>
Nepřiřazené lokální proměnné	55
Zobrazení hodnot primitivních datových typů	55
<b>Aritmetické operátory</b>	<b>59</b>
Operátory a typy	59
Seznámení s aritmetickými operátory	60

Jak zacházet s prioritou operátorů	63
Použití asociativity pro vyhodnocování výrazů	63
Asociativita a operátor přiřazení	64
<b>Zvyšování a snižování hodnoty proměnných</b>	<b>64</b>
Prefixový a postfixový operátor	65
<b>Deklarace implicitně typovaných lokálních proměnných</b>	<b>65</b>
<b>Stručné shrnutí kapitoly 2</b>	<b>67</b>

## Kapitola 3

<b>Psaní metod a aplikace oboru platnosti</b>	<b>69</b>
<b>Vytváříme metody</b>	<b>69</b>
Deklarace metody	69
Vracení dat z metod	70
Volání metod	72
Syntaxe volání metody	72
<b>Obor platnosti</b>	<b>74</b>
Definice lokálního oboru platnosti	74
Definice třídního oboru platnosti	75
Přetěžování metod	76
<b>Psaní metod</b>	<b>76</b>
<b>Volitelné parametry a pojmenované argumenty</b>	<b>82</b>
Definování volitelných parametrů	84
Předávání pojmenovaných argumentů	84
Řešení nejednoznačností souvisejících s volitelnými parametry a pojmenovanými argumenty	85
<b>Stručné shrnutí kapitoly 3</b>	<b>89</b>

## Kapitola 4

<b>Rozhodovací příkazy a jejich použití</b>	<b>91</b>
<b>Deklarace logických proměnných</b>	<b>91</b>
<b>Používáme logické operátory</b>	<b>92</b>
Operátor rovnosti a relační operátory	92
Podmínkové logické operátory	93
Zkrácené vyhodnocování	93
Shrnutí priority a asociativity operátorů	94
<b>Rozhodování pomocí příkazů if</b>	<b>95</b>
Syntaxe příkazu if	95
Seskupování příkazů do bloků	96
Kaskádové příkazy if	96
<b>Jak používat příkazy switch</b>	<b>101</b>
Syntaxe příkazu switch	101
Pravidla pro používání příkazu switch	102
<b>Stručné shrnutí kapitoly 4</b>	<b>105</b>

## Kapitola 5

<b>Používáme složená přiřazení a iterační příkazy</b>	<b>107</b>
<b>Operátory složeného přiřazení</b>	<b>107</b>
<b>Psaní příkazů while</b>	<b>108</b>
<b>Psaní příkazů for</b>	<b>112</b>
Obor platnosti příkazu for	113

<b>Psaní příkazů do</b>	<b>114</b>
<b>Stručné shrnutí kapitoly 5</b>	<b>121</b>

## Kapitola 6

<b>Ošetřování chyb a výjimek</b>	<b>123</b>
<b>Jak se vypořádat s chybami</b>	<b>123</b>
<b>Příkazy try a catch</b>	<b>124</b>
Neošetřené výjimky	125
Používání několika obsluh catch	126
Zachytávání výjimek několika typů	127
<b>Příkazy pro kontrolu přetečení celočíselných aritmetických operací</b>	<b>131</b>
Bloky s klíčovými slovy checked a unchecked	132
Výrazy s klíčovými slovy checked a unchecked	132
<b>Vyvolávání výjimek</b>	<b>134</b>
<b>Blok finally</b>	<b>137</b>
<b>Stručné shrnutí kapitoly 6</b>	<b>139</b>

## Část II

## Poznáváme jazyk C#

## Kapitola 7

<b>Vytváření a správa tříd a objektů</b>	<b>143</b>
<b>Klasifikace</b>	<b>143</b>
<b>Účel zapouzdření</b>	<b>144</b>
<b>Definování a použití třídy</b>	<b>144</b>
<b>Řízení přístupnosti</b>	<b>145</b>
Práce s konstruktory	147
Přetěžování konstruktorů	147
<b>Statické metody a data</b>	<b>154</b>
Vytvoření sdílené datové složky	155
Vytvoření statické datové složky pomocí klíčového slova const	155
Statické třídy	156
Anonymní třídy	158
<b>Stručné shrnutí kapitoly 7</b>	<b>159</b>

## Kapitola 8

<b>Poznáváme hodnotové a referenční typy</b>	<b>161</b>
<b>Kopírování proměnných hodnotových typů a tříd</b>	<b>161</b>
<b>Hodnoty null a nulovatelné typy</b>	<b>165</b>
Nulovatelné typy	166
Vlastnosti nulovatelných typů	167
<b>Předávání parametrů odkazem a výstupní parametry</b>	<b>168</b>
Parametry předávané odkazem (ref)	168
Výstupní parametry (out)	169
<b>Jak je uspořádána paměť v počítači</b>	<b>171</b>
Jak používat zásobník a haldu	172
<b>Třída System.Object</b>	<b>172</b>

<b>Zabalování</b>	<b>173</b>
<b>Rozbalování</b>	<b>174</b>
<b>Bezpečné přetypování dat</b>	<b>175</b>
Operátor is	176
Operátor as	176
<b>Stručné shrnutí kapitoly 8</b>	<b>178</b>

## Kapitola 9

<b>Vytváření hodnotových typů pomocí výčtů a struktur</b>	<b>181</b>
<b>Pracujeme s výčtem</b>	<b>181</b>
Deklarace výčtu	181
Praktické použití výčtu	182
Volba hodnot literálů výčtů	183
Volba interního typu prvků výčtu	183
<b>Práce se strukturami</b>	<b>185</b>
Deklarování struktury	187
Rozdíly mezi strukturami a třídami	187
Deklarace proměnných struktury	189
Jak probíhá inicializace struktury	189
Kopírování proměnných struktury	193
<b>Stručné shrnutí kapitoly 9</b>	<b>196</b>

## Kapitola 10

<b>Používáme pole a kolekce</b>	<b>197</b>
<b>Co je pole</b>	<b>197</b>
Deklarace proměnné typu pole	197
Vytvoření instance pole	198
Inicializace proměnných typu pole	199
Tvorba implicitně typovaných polí	199
Přístup k jednotlivým prvkům pole	200
Procházení pole	201
Kopírování polí	202
Vícerozměrná pole	203
Použití polí pro hraní karet	203
<b>Co jsou kolekce</b>	<b>210</b>
Třída ArrayList (pole)	211
Třída Queue (fronta)	213
Třída Stack (zásobník)	214
Třída Hashtable (hashovací tabulka)	215
Třída SortedList (seřazený seznam)	216
Inicializační prvky kolekcí	217
Srovnání polí a kolekcí	217
Implementace karetní hry pomocí kolekcí	218
<b>Stručné shrnutí kapitoly 10</b>	<b>221</b>

## Kapitola 11

<b>Seznámení s poli parametrů</b>	<b>223</b>
<b>Argumenty ve formě polí</b>	<b>224</b>
Deklarace polí s modifikátorem params	224
Pole objektů s modifikátorem params	226

Použití polí s modifikátorem params	227
<b>Porovnání polí parametrů a volitelných parametrů</b>	<b>230</b>
<b>Stručné shrnutí kapitoly 11</b>	<b>232</b>

## Kapitola 12

<b>Pracujeme s dědičností</b>	<b>233</b>
<b>Co je dědičnost?</b>	<b>233</b>
<b>Používáme dědičnost</b>	<b>234</b>
Volání konstruktoru báze třídy	235
Přiřazování tříd	236
Deklarování metod s klíčovým slovem new	238
Deklarování virtuálních metod	239
Deklarování metod s klíčovým slovem override	240
Chráněný přístup	242
<b>Rozšiřující metody</b>	<b>247</b>
<b>Stručné shrnutí kapitoly 12</b>	<b>250</b>

## Kapitola 13

<b>Tvorba rozhraní a abstraktních tříd</b>	<b>253</b>
<b>Rozhraní</b>	<b>253</b>
Definování rozhraní	254
Implementování rozhraní	254
Odkazování na třídu prostřednictvím rozhraní	255
Práce s více rozhraními	256
Explicitně implementovaná rozhraní	256
Omezení platná pro rozhraní	258
Definujeme a používáme rozhraní	258
<b>Abstraktní třídy</b>	<b>266</b>
Abstraktní metody	268
<b>Zapečetěné třídy</b>	<b>268</b>
Zapečetěné metody	268
Implementujeme a používáme abstraktní třídy	269
<b>Stručné shrnutí kapitoly 13</b>	<b>273</b>

## Kapitola 14

<b>Úklid paměti a správa prostředků</b>	<b>275</b>
<b>Doba existence objektu</b>	<b>275</b>
Tvorba destruktoru	276
Význam čističe paměti	278
Fungování čističe paměti	279
Doporučení	279
<b>Správa zdrojů</b>	<b>279</b>
Vyřazovací metody	280
Vyřazení bezpečné vzhledem k výjimkám	280
Příkaz using	281
Volání vyřazovací metody z destrukturu	282
<b>Implementace vyřazení bezpečného kódu vzhledem k výjimkám</b>	<b>284</b>
<b>Stručné shrnutí kapitoly 14</b>	<b>287</b>

## Část III

**Vytváření komponent**

## Kapitola 15

<b>Implementace vlastností pro přístup k datovým složkám</b>	<b>291</b>
<b>Implementace zapouzdření pomocí metod</b>	<b>291</b>
<b>Co jsou vlastnosti?</b>	<b>293</b>
Jak používat vlastnosti	295
Vlastnosti jen ke čtení	295
Vlastnosti jen pro zápis	296
Přístupnost vlastností	296
<b>Omezení vlastností</b>	<b>297</b>
<b>Deklarace vlastností v rozhraní</b>	<b>299</b>
Jak používat vlastnosti v aplikaci Windows	300
<b>Generování automatických vlastností</b>	<b>301</b>
<b>Inicializace objektů pomocí vlastností</b>	<b>303</b>
<b>Stručné shrnutí kapitoly 15</b>	<b>307</b>

## Kapitola 16

<b>Indexery a jejich používání</b>	<b>309</b>
<b>Co je indexer</b>	<b>309</b>
Příklad bez indexerů	309
Stejný příklad přepsaný pomocí indexerů	311
Přístupové metody indexerů	312
Srovnání indexerů a polí	313
<b>Indexery v rozhraní</b>	<b>315</b>
<b>Používání indexerů v aplikaci Windows</b>	<b>316</b>
<b>Stručné shrnutí kapitoly 16</b>	<b>321</b>

## Kapitola 17

<b>Přerušování toku programu a zpracování událostí</b>	<b>323</b>
<b>Deklarace a používání delegátů</b>	<b>323</b>
Příklad automatizované továrny	324
Implementace továrny bez delegátů	324
Implementace továrny pomocí delegátů	325
Používáme delegáty	327
<b>Lambda výrazy a delegáty</b>	<b>332</b>
Vytvoření adaptéru metody	332
Lambda výrazy jako adaptéry	332
Formy lambda výrazů	333
<b>Oznamování pomocí událostí</b>	<b>335</b>
Deklarování události	335
Přihlášení k odběru události	336
Odhlášení odběru události	336
Vyvolání události	336
<b>Události uživatelského rozhraní WPF</b>	<b>337</b>
Používáme události	338
<b>Stručné shrnutí kapitoly 17</b>	<b>342</b>



## Kapitola 18

<b>Úvod do generických typů</b>	<b>345</b>
<b>Problémy s typem object</b>	<b>345</b>
<b>Řešení na bázi generických typů</b>	<b>346</b>
Generické třídy vs zobecněné třídy	348
Generické typy a omezení	349
<b>Vytvoření generické třídy</b>	<b>349</b>
Teorie binárních stromů	349
Vytvoření generické třídy binárního stromu	352
<b>Vytvoření generické metody</b>	<b>360</b>
Definování generické metody pro vytvoření binárního stromu	361
<b>Variance a generická rozhraní</b>	<b>363</b>
Kovariantní rozhraní	364
Kontravariantní rozhraní	366
<b>Stručné shrnutí kapitoly 18</b>	<b>368</b>

## Kapitola 19

<b>Enumerace kolekcí</b>	<b>371</b>
<b>Procházení prvků v kolekci</b>	<b>371</b>
Ruční implementace enumerátoru	372
Implementace rozhraní IEnumerable	376
<b>Implementace enumerátoru pomocí iterátoru</b>	<b>378</b>
Jednoduchý iterátor	378
Definujeme enumerátor třídy Strom<TPrvěk> pomocí iterátoru	379
<b>Stručné shrnutí kapitoly 19</b>	<b>382</b>

## Kapitola 20

<b>Dotazování na paměťová data pomocí dotazových výrazů</b>	<b>383</b>
<b>Jazyk LINQ</b>	<b>383</b>
<b>Používáme LINQ v aplikacích jazyka C#</b>	<b>384</b>
Vybrání dat	386
Filtrování dat	388
Řazení, seskupování a agregování dat	388
Spojování dat	390
Používáme dotazové operátory	392
Dotazování dat v objektech typu Strom<TPrvěk>	393
LINQ a odložené vyhodnocení	398
<b>Stručné shrnutí kapitoly 20</b>	<b>402</b>

## Kapitola 21

<b>Přetěžování operátorů</b>	<b>405</b>
<b>Operátory</b>	<b>405</b>
Omezení operátorů	405
Přetížené operátory	406
Vytváření symetrických operátorů	407
<b>Vyhodnocování složených přiřazení</b>	<b>409</b>
<b>Deklarace operátorů inkrementace a dekrementace</b>	<b>410</b>
<b>Srovnání operátorů ve strukturách a třídách</b>	<b>411</b>
<b>Jak definovat operátory v párech</b>	<b>411</b>

<b>Implementujeme operátory</b>	<b>412</b>
<b>Seznámení s převodními operátory</b>	<b>418</b>
Vestavěné převody typů	419
Implementace vlastních převodních operátorů	419
Vytváření symetrických operátorů v revidované podobě	420
Píšeme převodní operátory	421
<b>Stručné shrnutí kapitoly 21</b>	<b>424</b>

## Část IV

### Tvorba grafických aplikací v subsystému WPF

#### Kapitola 22

#### Seznámení s grafickým subsystémem

<b>Windows Presentation Foundation</b>	<b>427</b>
<b>Aplikace WPF</b>	<b>427</b>
Vytváříme aplikaci WPF	428
<b>Přidávání ovládacích prvků na formulář</b>	<b>440</b>
Používáme ovládací prvky knihovny WPF	440
Dynamická změna vlastností	448
<b>Obsluha událostí v rámci formuláře WPF</b>	<b>451</b>
Zpracování událostí ve formuláři	452
<b>Stručné shrnutí kapitoly 22</b>	<b>456</b>

#### Kapitola 23

#### Získávání vstupu od uživatele

<b>Doporučení pro návrh nabídek a výběr jejich stylu</b>	<b>459</b>
<b>Nabídky a jejich události</b>	<b>460</b>
Vytvoření nabídky	460
Obsluha událostí nabídky	466
<b>Místní nabídky</b>	<b>471</b>
Vytváříme místní nabídky	471
<b>Standardní dialogová okna systému Windows</b>	<b>474</b>
Třída SaveFileDialog	475
<b>Zlepšení responzivnosti v aplikaci WPF</b>	<b>477</b>
<b>Stručné shrnutí kapitoly 23</b>	<b>486</b>

#### Kapitola 24

#### Ověřování zadaných hodnot

<b>Ověřování platnosti dat</b>	<b>487</b>
Strategie pro ověřování zadaných údajů	487
<b>Příklad – objednávka vstupenek na události</b>	<b>488</b>
Ověřování dat pomocí datové vazby	489
Změna místa, v němž dochází k ověřování platnosti zadaných dat	503
<b>Stručné shrnutí kapitoly 24</b>	<b>507</b>

## Část V

### Správa dat

#### Kapitola 25

<b>Dotazování na informace v databázi</b>	<b>511</b>
<b>Dotazování nad databází pomocí knihovny ADO.NET</b>	<b>511</b>
Databáze Northwind	512
Tvorba databáze	512
Zjišťování údajů o objednávkách pomocí knihovny ADO.NET	514
<b>Dotazování nad databází pomocí jazyka LINQ pro SQL</b>	<b>523</b>
Definování entitní třídy	523
Tvorba a spouštění dotazů jazyka LINQ pro SQL	525
Odložené a okamžité načítání	526
Spojování tabulek a vytváření vztahů	527
Odložené a okamžité načítání podruhé	531
Definování vlastní třídy DataContext	532
Zjišťování údajů o objednávkách pomocí jazyka LINQ pro SQL	533
<b>Stručné shrnutí kapitoly 25</b>	<b>537</b>

#### Kapitola 26

<b>Zobrazení a úprava dat pomocí rozhraní Entity Framework a datové vazby</b>	<b>539</b>
<b>Používáme datovou vazbu s rozhraním Entity Framework</b>	<b>540</b>
<b>Modifikace dat pomocí datové vazby</b>	<b>555</b>
Aktualizace stávajících dat	555
Ošetření konfliktních aktualizací	556
Přidávání a mazání dat	558
<b>Stručné shrnutí kapitoly 26</b>	<b>567</b>

## Část VI

### Tvorba profesionálních řešení ve Visual Studiu 2010

#### Kapitola 27

<b>Seznámení s knihovnou TPL (Task Parallel Library)</b>	<b>571</b>
<b>Proč provádět multitasking pomocí paralelního zpracování?</b>	<b>572</b>
Vzestup vícejádrových procesorů	573
<b>Implementace multitaskingu v desktopové aplikaci</b>	<b>574</b>
Úlohy, vlákna a fond vláken	574
Vytváření, spouštění a řízení úloh	575
Použití třídy Task pro implementaci paralelismu	579
Abstrahování úloh pomocí třídy Parallel	586
Kdy nepoužívat třídu Parallel	590
Vrácení hodnoty z úlohy	593
<b>Současné použití úloh a vláken uživatelského rozhraní</b>	<b>597</b>
<b>Rušení úloh a obsluha výjimek</b>	<b>600</b>

Mechanismus kooperativního zrušení úlohy	601
Obsluha výjimek úloh pomocí třídy AggregateException	609
Použití kontinuaace se zrušenými a nestandardně ukončenými úlohami	612

**Stručné shrnutí kapitoly 27** **613**

Kapitola 28

**Paralelní přístup k datům** **617**

**Paralelizace deklarativního přístupu k datům pomocí rozšíření PLINQ** **618**

Použití rozšíření PLINQ pro zlepšení výkonu při procházení kolekce	618
Nastavení pro dotazy rozšíření PLINQ	622
Zrušení dotazu rozšíření PLINQ	623

**Synchronizace souběžného imperativního přístupu k datům** **623**

Zamykání dat	626
Synchronizační primitiva knihovny TPL	627
Rušení a synchronizační primitiva	633
Třídy představující kolekce se souběžným přístupem	634
Implementace bezpečného vícevláknového přístupu k datům pomocí kolekce se souběžným přístupem a zámku	636

**Stručné shrnutí kapitoly 28** **645**

Kapitola 29

**Tvorba a použití webových služeb** **649**

**Co je webová služba?** **650**

Role frameworku WCF	650
---------------------	-----

**Architektura webových služeb** **650**

Webové služby modelu SOAP	651
Webové služby modelu REST	653

**Tvorba webových služeb** **654**

Webová služba modelu SOAP s názvem InformaceOProduktu	654
Webové služby modelu SOAP, klienti a zástupné třídy	661
Konzumace webové služby modelu SOAP s názvem InformaceOProduktu	662
Webová služba modelu REST s názvem DetailyProduktu	667
Konzumace webové služby modelu REST s názvem DetailyProduktu	673

**Stručné shrnutí kapitoly 29** **678**

Příloha

**Spolupráce s dynamickými jazyky** **679**

**Prostředí DLR** **680**

**Klíčové slovo dynamic** **681**

**Příklad: IronPython** **681**

**Příklad: IronRuby** **684**

**Shrnutí** **686**

**Rejstřík** **687**

# Poděkování

Možná znáte ten příběh o dělnících, kteří měli zajištěnu práci na celý život natíráním železničního mostu Forth Railway Bridge, což je rozsáhlá samonosná konstrukce z viktoriánského období, která se klene nad zálivem Firth of Forth severně od Edinburghu. Podle tohoto mýtu jim natření mostu z jednoho konce na druhý zabralo několik let, a když skončili, tak začali zase znovu. Nejsem si jist, zda to bylo způsobeno nevlídností skotského počasí, nebo citlivostí používané barvy, i když moje dcera je přesvědčena, že je to jednoduše tím, že radní města Edinburgh se stále nemohli shodnout na barvě, kterou by na mostě rádi viděli. Někdy mám pocit, že tato kniha má podobné rysy. Jakmile totiž dokončím edici a uvidím ji na pultech obchodů, tak společnost Microsoft vydá další skvělou aktualizaci pro Visual Studio a jazyk C# a moji přátelé v nakladatelství Microsoft Press se na mě obrátí s otázkou, jaké jsou mé plány pro další edici. Nicméně na rozdíl od mostu Forth Railway Bridge je práce na nové edici tohoto textu pokaždé zábavnou činností, která ve srovnání s pokusy o nalezení nového způsobu, jak držet natěračský štětec, nabízí mnohem větší prostor pro nové nápady. Vždy je zde něco nového, co je potřeba se naučit, a dále inovativní technologie, s níž je možné si pohrát. V této edici jsem se věnoval novým prvkům jazyka C# 4.0 a rozhraní .NET Framework 4.0, které jsou pro vývojáře neocenitelnou pomůckou při tvorbě aplikací, jež mohou využít vzrůstající sílu stále dostupnějšího výkonného hardwaru. Přestože se tedy tato práce jeví jako nikdy nekončící činnost, je vždy plodná a příjemná.

Velká část potěšení při práci na projektu, jako je tento, je možnost spolupracovat s vysoce motivovanou skupinou talentovaných lidí v nakladatelství Microsoft Press, s vývojáři společnosti Microsoft pracujícími na Visual Studiu 2010 a s lidmi, kteří revidují každou kapitolu a navrhují nejrůznější vylepšení. Zvláště bych chtěl poděkovat Rosemary Capertonové a Stephenu Sagmanovi, kteří neúnavně pracovali, aby veškerá práce šla podle plánu, dále Peru Blomqvistovi, který kontroloval (a opravoval) každou kapitolu, a Rogeru LeBlancovi, jenž se zhostil nevděčného úkolu přepsat rukopis a převést jej do angličtiny. Musím také připojit zvláštní zmínku o Michaelu Blomeovi, který mi poskytnul raný přístup k softwaru a odpověděl na řadu otázek ohledně knihovny Task Parallel Library. Několik členů společnosti Content Master bylo zaměstnáno kontrolou a testováním kódu pro cvičení – děkuji Miku Sumsionovi, Chrisovi Cullymu, Jamesi Millarovi a Louise Perryové. Musím samozřejmě dodatečně poděkovat Jonu Jaggerovi, který se v roce 2001 autorsky podílel na první edici této knihy.

V neposlední řadě musím poděkovat své rodině. Moje žena Diana je úžasným zdrojem inspirace. Při psaní kapitoly 28 o knihovně Task Parallel Library jsem měl mentální blok a musel jsem se jí zeptat, jak by vysvětlila bariérové metody (měl jsem samozřejmě na mysli metody třídy `Barrier`). Žertovně se na mě podívala a odvětila, že je to sice anatomicky správně, pokud bych ovšem byl v ordinaci doktora, a pak dodala, že buď jsem otázku nevyjádřil dostatečně pečlivě, nebo absolutně neporozuměla, na co jsem se ptal! James je už dospělý a brzy se bude muset naučit, co obnáší skutečná práce, pokud se o mě a Dianu chce starat způsobem, na jaký bychom si se ženou v senilitě rádi zvykli. Francesca také vyrostla a vypadá to, že vypilovala strategii pro dosažení všeho, co chce, aniž by musela udělat cokoliv jiného, než se na mě svýma velkýma, zářícíma očima podívat a usmát se.

A konečně, „až do dna!”

—John Sharp



# Úvod

Microsoft Visual C# je výkonný, ale přitom jednoduchý jazyk zaměřený především na vývojáře aplikací na platformě .NET Framework. Zdědil velké množství toho nejlepšího z jazyků C++ a Microsoft Visual Basic, ale jen málo z jejich nesrovnalostí a anachronismů, takže výsledkem je čistší a logičtější jazyk. Jazyk C# 1.0 měl svoji veřejnou premiéru v roce 2001. Jazyk C# 2.0 s Visual Studiem 2005 přináší několik důležitých nových prvků, včetně generických typů, iterátorů a anonymních metod. Jazyk C# 3.0, který byl vydán s Visual Studiem 2008, přidal rozšiřující metody, lambda výrazy a technologii LINQ (Language Integrated Query), jež je ze všech novinek nejzajímavější. Poslední verze jazyka C# s označením 4.0 nabízí další rozšíření, která zlepšují jeho interoperabilitu s ostatními jazyky a technologiemi. Mezi tyto prvky patří podpora pro pojmenované a volitelné argumenty, typ `dynamic`, který říká, že běhové prostředí jazyka by mělo pro daný objekt implementovat pozdní vazbu, a variance, která řeší některé problémy způsobem, v jehož rámci se definují generická rozhraní. Jazyk C# 4.0 těží z poslední verze rozhraní .NET Framework, které je taktéž označeno číslem 4.0. Toto vydání rozhraní .NET Framework obsahuje také řadu přídatků, z nichž jsou pravděpodobně nejvýznamnější třídy a typy, které tvoří knihovnu TPL (Task Parallel Library). Pomocí knihovny TPL můžete vytvářet vysoce škálovatelné aplikace, které dokážou rychle a snadno využít plný potenciál vícejádrových procesorů. Kromě toho byla rozšířena podpora pro webové služby a knihovnu WCF (Windows Communication Foundation). Nyní můžete vytvářet služby, které fungují na bázi modelu REST i na bázi tradičtějšího modelu SOAP.

Ve vývojovém prostředí poskytovaném Visual Studiem 2010 se tyto výkonné novinky dají velmi snadno používat, produktivitu práce vývojáře pak výrazně zvyšují noví průvodci a různá další vylepšení, která jsou součástí Visual Studia 2010.

## Komu je kniha určena

Tato kniha předpokládá, že jejím čtenářem je vývojář, který se chce naučit základům programování v jazyku C#, a to v prostředí Visual Studia 2010 na platformě .NET Framework ve verzi 4.0. Seznámíte se tu se základními rysy jazyka C# a ty pak budete používat ke tvorbě aplikací běžících pod operačními systémy Microsoft Windows. Po prostudování celé knihy budete jazyk C# již dobře znát a budete mít za sebou několik jednoduchých aplikací typu WPF (Windows Presentation Foundation), budete umět přistupovat k databázím umístěným na Microsoft SQL Serveru pomocí knihovny ADO.NET a jazyka LINQ, sestavovat responzivní a škálovatelné aplikace pomocí knihovny TPL a pomocí knihovny WCF (Windows Communication Foundation) vytvářet webové služby modelu REST a SOAP.

## Najděte si optimální výchozí bod

Tato kniha vám pomůže získat potřebné znalosti v mnoha základních oblastech. Kniha se hodí jak pro úplné programátorské nováčky, tak i pro ty, kdo přecházejí z jiných jazyků (např. C, C++, Java nebo Visual Basic). Pomocí následující tabulky si můžete najít svůj nejvhodnější výchozí bod.

Pokud...	... postupujte takto:
Jste úplnými nováčky v oblasti objektového programování	<ol style="list-style-type: none"> <li>1. Nainstalujte cvičební soubory podle popisu v následující části „Instalace a používání cvičebních souborů“.</li> <li>2. Procházejte postupně kapitoly v první až třetí části knihy v tom pořadí, v jakém jsou uvedeny.</li> <li>3. Podle úrovně svého zájmu a podle toho, jak se budete cítit, prostudujte postupně také kapitoly ve čtvrté až šesté části.</li> </ol>
Znáte procedurální jazyky (např. C), ale v jazyku C# jste úplnými nováčky	<ol style="list-style-type: none"> <li>1. Nainstalujte cvičební soubory podle popisu v následující části „Instalace a používání cvičebních souborů“. Prvních pět kapitol projděte jen letmo, abyste získali obecný přehled o jazyku C# a Visual Studiu 2010, a poté se soustředte na kapitoly 6 až 21.</li> <li>2. Podle úrovně svého zájmu a podle toho, jak se budete cítit, prostudujte postupně také kapitoly ve čtvrté až šesté části.</li> </ol>
Přecházíte z objektově orientovaných jazyků (např. C++ nebo Java)	<ol style="list-style-type: none"> <li>1. Nainstalujte cvičební soubory podle popisu v následující části „Instalace a používání cvičebních souborů“.</li> <li>2. Prvních sedm kapitol projděte jen letmo, abyste získali obecný přehled o jazyku C# a Visual Studiu 2010. Poté se soustředte na kapitoly 8 až 21.</li> <li>3. Informace o tvorbě aplikací s grafickým uživatelským rozhraním a o práci s databázemi najdete ve čtvrté a páté části knihy.</li> <li>4. Informace o tvorbě škálovatelných aplikací a webových služeb najdete v šesté části knihy.</li> </ol>
Přecházíte z Visual Basicu 6	<ol style="list-style-type: none"> <li>1. Nainstalujte cvičební soubory podle popisu v následující části „Instalace a používání cvičebních souborů“.</li> <li>2. Procházejte postupně kapitoly v první až třetí části knihy v tom pořadí, v jakém jsou uvedeny.</li> <li>3. Informace o tvorbě aplikací s grafickým uživatelským rozhraním najdete ve čtvrté části knihy.</li> <li>4. Informace o práci s databázemi najdete v páté části knihy.</li> <li>5. Informace o tvorbě škálovatelných aplikací a webových služeb najdete v šesté části knihy.</li> <li>6. Pročtěte si části „Stručné shrnutí“ na koncích kapitol, kde najdete krátký přehled syntaxe a technik prezentovaných v dané kapitole o specifických konstrukcích a rysech jazyka C# a Visual Studia 2010.</li> </ol>
Jste prošli všechna cvičení a potřebujete něco najít	<ol style="list-style-type: none"> <li>1. Jednotlivá témata a hesla vyhledávejte podle závěrečného rejstříku nebo obsahu knihy.</li> <li>2. Pročtěte si části „Stručné shrnutí“ na koncích kapitol, kde najdete krátký přehled syntaxe a postupů probíraných v dané kapitole.</li> </ol>

## Konvence používané v této knize

Informace v knize jsou uspořádány tak, aby byly snadno srozumitelné a dobře se četly. Než se pustíte do vlastního textu, přečtěte si následující seznam s vysvětlením všech konvencí, s nimiž se zde setkáte, a se zvýrazněním užitečných prvků, které se vám mohou hodit.

### Konvence

- Každé cvičení je tvořeno sérií úloh. Každá úloha je uvedena v posloupnosti očíslovaných kroků (1, 2 atd.). Odrážka (■) naznačuje, že cvičení má jeden jediný krok.





**Tip:** Poznámky s nadpisem „Tip“ obsahují dodatečné informace nebo alternativní způsoby pro úspěšné dokončení probíraného kroku.



**Důležité:** Poznámky s nadpisem „Důležité“ vás upozorňují na to, co je nutné před dalším pokračováním zkontrolovat.

- Text, který máte někam zapsat, je uveden **tučně**.
- Znak (+) mezi dvěma názvy kláves značí, že tyto klávesy musíte stisknout najednou. Například text „stiskněte Alt+Tab“ znamená, že přidržíte klávesu Alt a stisknete tabulátor.

## Další prvky

- Orámované odbočky v textu knihy obsahují podrobnější informace na určité téma a obvykle se vztahují ke cvičení. Mohou nabízet různé doplňující informace, tipy nebo vlastnosti týkající se probíraných témat.
- Každá kapitola končí částí „Stručné shrnutí“, která obsahuje stručnou připomínku toho, jak se provádějí činnosti, které jste se v dané kapitole naučili.

## Předběžné vydání softwaru

Tato kniha byla napsána a otestována proti Visual Studiu 2010 ve verzi Beta 2. Příklady byly kontrolovány a testovány proti finálnímu vydání softwaru. Mezi ostrou verzí a příklady, textem a obrázky v této knihze však mohou být drobné rozdíly.

## Systémové požadavky

Pro práci s cvičebními úlohami v této knize je nutné splnit následující požadavky na hardware a software:

- Microsoft Windows 7 Home Premium, Windows 7 Professional, Windows 7 Enterprise nebo Windows 7 Ultimate. Cvičení půjdou spustit také na operačním systému Microsoft Windows Vista s aktualizací Service Pack 2 nebo novější;
- Microsoft Visual Studio 2010 Standard, Microsoft Visual Studio 2010 Professional nebo Microsoft Visual C# 2010 Express a Microsoft Visual Web Developer 2010 Express;
- Microsoft SQL Server 2008 Express (je poskytován se všemi edicemi Visual Studia 2010 a s aplikacemi Visual C# 2010 Express a Visual Web Developer 2010 Express);
- procesor na frekvenci 1,6 GHz nebo vyšší, kapitoly 27 a 28 vyžadují alespoň dvoujádrový procesor;
- 1 GB fyzické paměti RAM pro 32bitové procesory nebo 2 GB fyzické paměti RAM pro 64bitové procesory;
- monitor s rozlišením alespoň 1024 × 768 s nejméně 256 barvami;
- mechanika DVD-ROM;
- myš od společnosti Microsoft nebo kompatibilní ukazovací zařízení.

Kromě toho budete také potřebovat přístup ke svému počítači s právy Administrator, který je nutný ke konfiguraci serveru SQL Server 2008 Express.

## Cvičební soubory

Doprovodné DVD, přiložené v obálce na konci knihy, obsahuje cvičební soubory, které budete v jednotlivých kapitolách používat. Cvičební soubory vás zbaví nutnosti ručně vytvářet vše, co není v rámci daného cvičení podstatné. Soubory a návody v jednotlivých kapitolách, jež vás vedou krok za krokem, dodržují princip výuky na konkrétních úkolech, takže teoreticky nabyté znalosti snadno a efektivně ověříte v praktických cvičeních.

## Instalace cvičebních souborů

V následujících bodech je popsán způsob instalace cvičebních souborů tak, abyste je mohli používat ve cvičeních v této knize.

1. Vytáhněte DVD z obálky na zadní straně knihy a vložte jej do mechaniky DVD-ROM.
2. Objeví se nabídka s volbami týkajícími se obsahu knihy a doprovodného softwaru.
3. Klepněte na odkaz Zdrojové kódy.
4. Otevře se okno obsahující odkazy na zdrojové kódy.
5. Klepněte na odkaz Prozkoumat DVD.
6. Otevře se okno Průzkumníka s příslušnou složkou obsahující zdrojové kódy.
7. Zkopírujte všechny složky (složky *Kapitola 1* až *Kapitola 29* a složku *Příloha*) do složky *Dokumenty\Visual CSharp 2010 Krok za krokem*.
8. Další text knihy předpokládá, že máte zdrojové kódy příkladů zkopírovány do své složky *Dokumenty\Visual CSharp 2010 Krok za krokem*.

## Používání cvičebních souborů

V každé kapitole této knihy je vysvětleno, kdy a jak cvičební soubory používat. Ve chvíli, kdy bude nutné použít cvičební soubory, bude uveden také postup pro jejich otevření.

Pro ty, kteří by se rádi dozvěděli podrobnosti již nyní, následuje seznam všech projektů a řešení Visual Studia 2010, seskupených podle složek, v nichž je naleznete. V řadě případů je ke cvičením k dispozici verze s počátečními soubory a dále dokončená verze stejného projektu, kterou můžete použít pro kontrolu. Hotové projekty jsou uloženy ve složce s dovětkem „- Hotové“.

Projekt	Popis
<b>KAPITOLA 1</b>	
Text Ahoj	Úvodní projekt celé knihy, ve kterém projdete jednotlivé kroky tvorby jednoduché aplikace, která vypíše textový pozdrav.
WPF Ahoj	Tento projekt zobrazí pozdrav v okně pomocí knihovny WPF (Windows Presentation Foundation.).
<b>KAPITOLA 2</b>	
PrimitivníDatovéTypy	Projekt předvádí deklaraci proměnných všech primitivních typů a ukazuje, jak jim přiřadit hodnotu a jak tuto hodnotu zobrazit v okně.
MatematickéOperátory	Tento program je úvodem do aritmetických operátorů (+ - * / %).

Projekt	Popis
<b>KAPITOLA 3</b>	
Metody	V tomto projektu přepracujete kód cvičení z minulého projektu a vyzkoušíte si v něm strukturování kódu pomocí metod.
DenniSazba	V tomto projektu si projdete postup vytváření metod, budete metody spouštět a krokovat jejich volání v ladicím nástroji Visual Studio 2010.
DenniSazbaSVolitelnými Parametry	V tomto projektu uvidíte, jak se definují metody, které přijímají volitelné parametry, a jak se volají metody pomocí pojmenovaných argumentů.
<b>KAPITOLA 4</b>	
Výběr	Tento projekt předvádí použití kaskádového příkazu <code>if</code> k implementaci složitější logiky, jako je kupříkladu porovnání dvou kalendářních dat.
PříkazSwitch	Jednoduchý program pro převod znaků do jejich reprezentace v jazyku XML.
<b>KAPITOLA 5</b>	
PříkazWhile	Projekt načítá pomocí příkazu <code>while</code> po řádcích obsah zdrojového souboru a každý řádek zobrazí v textovém poli formuláře.
PříkazDo	Projekt převádí pomocí příkazu <code>do</code> číslo z desítkové do osmičkové soustavy.
<b>KAPITOLA 6</b>	
MatematickéOperátory	Tento projekt dále rozvádí stejnojmenný projekt z druhé kapitoly a demonstrovuje různé výjimky, které dokážou narušit běh programu. Díky klíčovým slovům <code>try</code> a <code>catch</code> je aplikace robustnější, takže její běh už žádná výjimka nenaruší.
<b>KAPITOLA 7</b>	
Třídy	Projekt pokrývá základy definování vlastních tříd, spolu s veřejnými konstruktory, metodami a soukromými datovými složkami. Probíráno je také vytváření instancí tříd pomocí klíčového slova <code>new</code> a způsob definice statických metod a datových složek.
<b>KAPITOLA 8</b>	
Parametry	Program zkoumá rozdíl mezi parametry hodnotového a referenčního typu. Předvádí také použití klíčových slov <code>ref</code> a <code>out</code> .
<b>KAPITOLA 9</b>	
StrukturyAVýčty	Tento projekt definuje strukturu ( <code>struct</code> ) pro reprezentaci kalendářního data.
<b>KAPITOLA 10</b>	
Karty s použitím polí	Tento projekt ukazuje, jak pomocí polí modelovat ruce s kartami v karetní hře.
Karty s použitím kolekcí	Tento projekt ukazuje, jak předělat program karetní hry tak, aby místo polí používal kolekce.

Projekt	Popis
<b>KAPITOLA 11</b>	
PoleParams	Projekt předvádí použití klíčového slova <code>params</code> pro vytvoření jediné metody, která může přijímat libovolný počet argumentů typu <code>int</code> .
<b>KAPITOLA 12</b>	
DopravníProstředky	Tento projekt vytvoří pomocí dědičnosti jednoduchou hierarchii tříd dopravních prostředků. Kromě toho ukazuje, jak definovat virtuální metody.
RozšiřujícíMetoda	Tento projekt ukazuje, jak vytvořit rozšiřující metodu pro typ <code>int</code> , která převádí číselnou hodnotu z desítkové soustavy do soustavy o jiném základu.
<b>KAPITOLA 13</b>	
Kreslení s použitím rozhraní	Tento projekt implementuje část balíčku pro kreslení grafiky. Pomocí rozhraní definuje metody, které kreslicí tvary exponují a implementují.
Kreslení	Tento projekt rozšiřuje projekt Kreslení s použitím rozhraní tak, že společnou funkčnost pro objekty tvarů vyčlení do abstraktních tříd.
<b>KAPITOLA 14</b>	
PříkazUsing	Tento projekt znovu zkoumá malý úryvek kódu z kapitoly 5 a předvádí, že není bezpečný vzhledem k výjimkám. Uvidíte, jak to lze změnit pomocí příkazu <code>using</code> .
<b>KAPITOLA 15</b>	
VlastnostiOkna	Projekt předvádí jednoduchou aplikaci s grafickým uživatelským rozhraním, jež používá několik vlastností pro automatické zobrazování velikosti jejího hlavního okna. Při změně velikosti okna uživatelem se toto zobrazení automaticky aktualizuje.
AutomatickéVlastnosti	Tento projekt ukazuje, jak pro danou třídu vytvořit automatické vlastnosti a jak je použít pro inicializaci instancí této třídy.
<b>KAPITOLA 16</b>	
Indexery	Projekt používá dva indexery: první k vyhledání telefonního čísla osoby daného jména, druhý pro vyhledání jména osoby k zadanému telefonnímu číslu.
<b>KAPITOLA 17</b>	
Hodiny s použitím delegátů	Tento projekt zobrazuje hodiny se světovým časem ukazující místní čas společně s časem v Londýně, New Yorku a Tokiu. Pro spuštění a zastavení displejů hodin používá aplikace delegáty.
Hodiny s použitím událostí	Tato verze hodin se světovým časem používá pro spuštění a zastavení displejů hodin událostí.
<b>KAPITOLA 18</b>	
BinárníStrom	Toto řešení předvádí, jak pomocí generických typů vytvořit typově bezpečnou strukturu, která může obsahovat prvky libovolného typu.
TvorbaStromu	Projekt předvádí, jak pomocí generických typů vytvořit typově bezpečnou metodu, která může přijímat parametry libovolného typu.

Projekt	Popis
TestBinárníhoStromu	Testovací kód, který vytváří instance typu <code>Strom</code> definovaného v projektu <code>BinárníStrom</code> .
<b>KAPITOLA 19</b>	
BinárníStrom	Projekt předvádí implementaci generického rozhraní <code>IEnumerator&lt;T&gt;</code> při tvorbě enumerátoru pro generickou třídu <code>Strom</code> .
IterátorBinárníhoStromu	Pomocí iterátoru vytvoří tento projekt enumerátor pro generickou třídu <code>Strom</code> .
TestEnumerátoru	Testovací kód, který otestuje enumerátor a iterátor pro třídu <code>Strom</code> .
<b>KAPITOLA 20</b>	
DotazováníBinárníhoStromu	Projekt ukazuje, jak pomocí dotazů jazyka LINQ načítat data z objektu binárního stromu.
<b>KAPITOLA 21</b>	
KomplexníČísla	Projekt definuje nový typ, který modeluje komplexní čísla a implementuje operátory, jež jsou pro komplexní čísla běžné.
<b>KAPITOLA 22</b>	
Zvoníci	Jedná se o aplikaci typu WPF (Windows Presentation Foundation) demonstrující způsob definice stylů a použití základních ovládacích prvků WPF.
<b>KAPITOLA 23</b>	
Zvoníci	Rozšíření projektu z předchozí kapitoly, k němuž jsou přidány rozevírací a místní nabídky.
<b>KAPITOLA 24</b>	
ObjednávkaVstupenek	Projekt ukazuje, jak v aplikaci WPF implementovat podniková pravidla pro ověřování vstupu zadaného uživatelem, přičemž jako příklad používá informace o objednávce zákazníka.
<b>KAPITOLA 25</b>	
Objednávky	Projekt předvádí, jak pomocí knihovny ADO.NET přistupovat k databázi. Aplikace načítá údaje z tabulky <code>Orders</code> v databázi <code>Northwind</code> .
ObjednávkyLINQ	Tento projekt ukazuje, jak přistupovat k databázi pomocí jazyka LINQ pro SQL a načítat údaje z tabulky <code>Orders</code> v databázi <code>Northwind</code> .
<b>KAPITOLA 26</b>	
Dodavatelé	Tento projekt demonstruje, jak používat datovou vazbu v rámci aplikace WPF pro zobrazení a naformátování dat načtených z databáze v ovládacích prvcích na formuláři WPF. Aplikace také uživateli umožňuje upravovat údaje tabulky <code>Products</code> v databázi <code>Northwind</code> .
<b>KAPITOLA 27</b>	
Ukázka grafu	Projekt generuje a na formuláři WPF zobrazuje komplexní graf. K provádění výpočtů používá jediné vlákno.
Ukázka grafu s použitím třídy <code>Task</code>	Tato verze projektu <code>Ukázka grafu</code> vytváří více úloh k provedení výpočtů pro graf paralelním způsobem.

Projekt	Popis
Ukázka grafu s použitím úloh vracejících výsledky	Jedná se o rozšířenou verzi projektu Ukázka grafu s použitím třídy <code>Task</code> , která předvádí, jak se z úlohy vracejí data.
Ukázka grafu s použitím třídy <code>Parallel</code>	Tato verze projektu Ukázka grafu používá třídu <code>Parallel</code> pro abstrahování procesu tvorby a správy úloh.
Ukázka grafu s rušením úloh	Projekt ukazuje, jak implementovat rušení pro řízené zastavení úloh před jejich dokončením.
Paralelní cyklus	Tato aplikace poskytuje příklad demonstrující situaci, kdy se pro vytváření a spouštění úloh třída <code>Parallel</code> nehodí.
<b>KAPITOLA 28</b>	
Výpočet PI	Projekt používá algoritmus statistického vzorkování pro výpočet přibližné hodnoty čísla PI. Používá paralelní úlohy.
PLINQ	Projekt ukazuje několik příkladů použití rozšíření PLINQ pro dotazování dat pomocí paralelních úloh.
<b>KAPITOLA 29</b>	
SlužbaInformaceOProduktu	Projekt implementuje webovou službu modelu SOAP sestavenou pomocí knihovny WCF. Webová služba exponuje metodu, která vrací informace o cenách produktů v databázi Northwind.
SlužbaDetailyProduktu	Projekt implementuje webovou službu modelu REST sestavenou pomocí knihovny WCF. Webová služba nabízí metodu, která vrací údaje o zadaném produktu z databáze Northwind.
KontraktyProDetailyProduktu	Projekt obsahuje kontrakty pro služby a data implementovaná webovou službou <code>SlužbaDetailyProduktu</code> .
KlientSlužbyOProduktech	Projekt předvádí, jak vytvářet aplikaci WPF, jež konzumuje webovou službu. Ukazuje, jak vyvolávat webové metody z webových služeb <code>SlužbaInformaceOProduktu</code> a <code>SlužbaDetailyProduktu</code> .

## Zpětná vazba od čtenářů

Nakladatelství a vydavatelství Computer Press, které pro vás tuto knihu přeložilo, stojí o zpětnou vazbu a bude na vaše podněty a dotazy reagovat. Můžete se obrátit na následující adresy:

redakce PC literatury  
 Computer Press  
 Spielberk Office Centre  
 Holandská 3  
 639 00 Brno  
 nebo  
 sefredaktor.pc@cpress.cz

## Dotazy

Máte-li s knihou jakýkoli problém, kontaktujte nás pomocí formuláře na adrese <http://knihy.cpress.cz/K1836>, kde klepněte na odkaz Poslat komentář. Pokusíme se udělat vše, abychom vám ho pomohli vyřešit.

## Errata

Přestože jsme udělali maximum pro to, abychom zajistili přesnost a správnost obsahu, chybám se úplně vyhnout nedá. Pokud v některé z našich knih najdete chybu, ať už chybu v textu nebo v kódu, budeme rádi, pokud nám ji nahlásíte. Ostatní uživatelé tak můžete ušetřit frustrace a pomoci nám zlepšit následující vydání této knihy. Pokud si přejete zadat errata, učiňte tak na adrese <http://knihy.cpress.cz/K1836>, kde klepněte na odkaz Poslat komentář.

Veškerá existující errata zobrazíte na adrese <http://knihy.cpress.cz/K1836> po klepnutí na odkaz Errata.