

Obsah

Předmluva	19
Úvod	24
<i>Java 2</i>	22
Předpoklady	24
Studium <i>Javy</i>	25
Záměry	25
Dokumentace online	26
Kapitoly	27
Praktická cvičení	32
Multimediální CD ROM	32
Zdrojový kód	32
Standardy, dodržované při vytváření zdrojového kódu	34
Verze jazyka <i>Java</i>	34
Semináře a poradenství	35
Chyby	35
Poznámky k návrhu obálky	35
Poděkování	36
Spolupracovníci v síti Internet	38
1. Úvod do objektů	39
1.1. Postup při zobecňování	39
1.2. Objekt má svoje rozhraní	41
1.3. Ukrytá implementace	43
1.4. Opětne používání implementace	44
1.5. Dědičnost – opětne použití rozhraní	45
Vztahy typu je... versus podobá se	49
Zaměnitelné objekty a polymorfismus	50
1.6. Abstraktní bázevé třídy a rozhraní	54
1.7. Obor platnosti a životnost objektu	54
Kolekce a jejich procházení	55
Hierarchie s jedním kořenovým objektem	57
Knihovny kolekci a podpora snadného používání kolekci	58
Přetypování směrem dolů versus obecné šablony	58
Dilema hospodaření: kdo by měl uklidit?	59
Sběrače neplatných objektů versus efektivita a pružnost programového kódu	59
1.8. Ošetření výjimek: zacházení s chybami	60
1.9. Multithreading	61
1.10. Trvalé objekty (persistence)	62
1.11. <i>Java</i> a Internet	62
Co je to síť WWW?	62
Architektura klient/server	63
Síť WWW jako obří server	63
Programování na straně klienta	64
Doplňky	65
Skriptovací jazyky	66
<i>Java</i>	66
ActiveX	67

Zabezpečení	68
Internet versus intranet	69
Programování na straně serveru	70
Samostatná aréna: aplikace	70
1.12. Analýza a návrh	71
Fáze 0: Vytvořte si plán	73
Fáze 1: Co vytváříte?	73
Fáze 2: Jak budeme své řešení tvořit?	76
Pět vývojových stupňů návrhu objektů	78
Metodické pokyny pro projektování objektů	79
Fáze 3: Sestavení jádra	80
Fáze 4: Procházení případů užití	80
Fáze 5: Evoluce	81
Plánovat se vyplatí	82
1.13. Extrémní programování	83
Nejdříve napiš test	83
Programování ve dvojicích	84
1.14. V čem vězí úspěch Javy	85
Snadnější vyjadřování a porozumění systémům	85
Maximální využití knihoven	85
Ošetřování chyb	86
Programování ve velkém	86
1.15. Strategie přechodu	86
Metodické pokyny	87
Překážky ze strany vedení	88
Počáteční náklady	88
Otázky výkonu	89
Běžné chyby doby návrhu	89
1.16. Java versus C++?	90
1.17. Shrnutí	91
2. Všechno je objekt	93
2.1. S objekty se manipuluje pomocí odkazů	93
2.2. Všechny objekty musíte vytvořit	94
Kde je úložiště	95
Zvláštní případ: primitivní typy	96
Pole v Javě	97
2.3. Už nikdy nemusíte mazat objekty	98
Obor platnosti	98
Obor platnosti objektů	99
2.4. Vytváření nových datových typů: třída	99
Datové položky (fields) a metody	100
Implicitní hodnoty primitivních datových členů	101
2.5. Metody, argumenty a návratové hodnoty	102
Seznam argumentů	102
2.6. Vytvoření programu v jazyce Java	104
Viditelnost názvu	104
Používání ostatních komponent	104
Klíčové slovo static	105
2.7. Váš první program v jazyce Java	107
Překlad a spouštění	108
2.8. Komentáře a vložená dokumentace	109
Poznámková dokumentace	110

Syntaxe.....	110
Vložený HTML.....	111
@see: odkazy na ostatní třídy.....	112
Značky v dokumentaci třídy.....	112
Značky dokumentace proměnných.....	113
Značky dokumentace metod.....	113
Příklad dokumentace.....	114
2.9. Styl zápisu kódu.....	115
2.10. Shrnutí.....	115
2.11. Cvičení.....	115
3. Řízení toku programu.....	117
3.1. Používání operátorů jazyka Java.....	117
Priorita.....	117
Přiřazení.....	118
Přezdívaní při volání metod.....	119
Matematické operátory.....	120
Zkrácené vyhodnocování.....	126
Časté chyby při používání operátorů.....	133
Přehled operátorů.....	137
3.2. Řízení běhu programu.....	147
true a false.....	147
Cykly.....	149
3.3. Shrnutí.....	160
3.4. Cvičení.....	161
4. Inicializace a úklid.....	162
4.1. Zaručená inicializace pomocí konstruktoru.....	162
4.2. Přetěžování metod.....	164
Rozlišování přetížených metod.....	166
Přetěžování s primitivními datovými typy.....	166
Přetěžování a návratové hodnoty.....	170
Implicitní konstruktory.....	170
Klíčové slovo this.....	171
Volání konstruktů z konstruktů.....	173
Význam klíčového slova static.....	174
4.3. Úklid: finalizace a shromažďování neplatných objektů.....	174
K čemu je vlastně metoda finalize() dobrá?.....	176
Musíte si po sobě uklidit.....	176
Pozůstalostní podmínky.....	180
Jak funguje sběrač neplatných objektů.....	181
4.4. Inicializace datových složek.....	184
Stanovení inicializace.....	185
Inicializace pomocí konstruktů.....	187
Pořadí inicializace.....	187
Inicializace statických datových členů.....	188
Explicitní inicializace statického členu.....	191
Inicializace nestatické instance.....	192
4.5. Inicializace pole.....	193
Vícerozměrná pole.....	197
4.6. Shrnutí.....	200
4.7. Cvičení.....	200

5. Ukrývání implementace	203
5.1. package: knihovná jednotka	203
Vytváření balíčků s jedinečnými názvy	206
Konflikty	208
Knihovna vlastních nástrojů	209
Používání importu ke změně chování	210
Námítky proti balíčkům	212
5.2. Specifikátory přístupu v jazyce Java	212
Přátelský	212
public: přístup pro rozhraní	213
Implicitní balíček	214
private: ruce pryč!	215
protected: určitý druh přátelského členu	216
5.3. Rozhraní a implementace	217
5.4. Přístup ke třídám	218
5.5. Shrnutí	221
5.6. Cvičení	222
6. Opětné použití tříd	225
6.1. Syntaxe skládání	225
6.2. Syntaxe dědičnosti	228
Inicializace bazové třídy	230
Konstruktory s argumenty	231
Zachycení výjimek implicitního konstruktora	232
6.3. Kombinování mechanismů skládání a dědičnosti	232
Záruky náležitého úklidu	234
Pořadí při shromažďování neplatných objektů	237
Ukrývání názvů	237
6.4. Volba mezi skládáním a dědičností	238
6.5. Klíčové slovo protected	239
6.6. Vývoj po částech	240
6.7. Přetypování na předka	241
Proč se říká „přetypování směrem nahoru“?	242
Ještě jednou o skládání ve vztahu k dědičnosti	242
6.8. Klíčové slovo final	243
Konstantní data	243
Prázdné konstanty	245
Konstantní argumenty	246
Konečné metody	247
Konečné třídy	248
Opatrnost při používání klíčového slova final	249
6.9. Inicializace a zavádění třídy	250
Inicializace a dědičnost	251
6.10. Shrnutí	252
6.11. Cvičení	253
7. Polymorfismus čili mnohotvárnost	255
7.1. Ještě jednou o konverzi na obecnější datový typ	255
Jak zapomenout na typ objektu	256
7.2. Malý úskok	258
Vazby volání metod	258

9.2. Úvod do kontejnerů.....	356
Tisk obsahu kontejneru	357
Zaplnění kontejneru	358
9.3. Nevýhoda kontejneru: neznámý typ	364
Někdy to však stejně funguje.....	366
Vytvoření typově zaměřeného kontejneru ArrayList.....	368
Parametrisované typy	369
9.4. Iterátory	369
Nezamýšlená rekurze.....	371
9.5. Taxonomie neboli systematika kontejnerů.....	372
9.6. Co kolekce umějí	375
9.7. Co umí rozhraní List.....	378
Zřetězený seznam jako zásobník.....	381
Zřetězený seznam jako fronta.....	382
9.8. Co umí rozhraní Set	383
Rozhraní SortedSet	386
9.9. Co umí rozhraní Map.....	386
Rozhraní SortedMap	390
Hešování a hešový kód.....	391
Význam metody hashCode()	393
Různé faktory ovlivňující výkon struktury HashMap.....	398
Překrytí metody hashCode()	398
9.10. Ukládání odkazů	401
Struktura WeakHashMap.....	403
9.11. Ještě jednou iterátory	404
9.12. Volba implementace.....	405
Volba druhu seznamu.....	406
Volba mezi kontejnery typu Set.....	409
Volba mezi typy map.....	411
9.13. Řazení a prohledávání seznamů	413
9.14. Nástroje	414
Vytváření neměnných kolekcí a map	415
Synchronizace kolekcí a map	416
9.15. Rychlé selhání	417
9.16. Nepodporované operace	418
9.17. Kontejnery jazyka Java verze 1.1 a 1.0.....	420
Třídy Vector a Enumeration	420
Třída Hashtable	421
Třída Stack	421
Třída BitSet	422
9.18. Shrnutí.....	424
9.19. Cvičení.....	424