

Obsah

Předmluva k aktuálnímu vydání	15
1 Úvod k prvnímu vydání z roku 2000	16
Typografické a syntaktické konvence	20
2 Základní pojmy	21
2.1 Trocha historie nikoho nezabije	21
2.2 Způsob zpracování programu v Javě	21
2.3 Různé typy programů	22
2.4 Vývojové nástroje a jejich verze	22
2.4.1 Vývoj a verze JDK	23
2.5 Používané pojmy	25
2.5.1 Java Core API	25
2.5.2 Java Platforma	26
2.5.3 Technologie hot-spot a JIT kompilátor	26
2.6 Co byste měli vědět, než skutečně začnete programovat	28
2.6.1 Jak přeložit a spustit program	28
2.6.2 Běžné problémy	29
3 Základní dovednosti v Javě	33
3.1 Komentáře	33
3.2 Způsob zápisu identifikátorů	35
3.3 Anotace	36
3.4 Hlavní program	40
3.5 Základní datové typy	41
3.5.1 Celočíselné typy a jejich konstanty	42

3.5.2	Znakový typ a jeho konstanty	43
3.5.3	Řetězcové konstanty (literály)	45
3.5.4	Logický typ a jeho konstanty	46
3.5.5	Reálné datové typy a jejich konstanty	46
3.6	Deklarace proměnných	48
3.6.1	Deklarace proměnných s konstantní hodnotou	49
3.6.2	Deklarace výčtového typu	51
3.7	Operátor přiřazení	54
3.8	Operátor přetypování	55
3.8.1	Rozšiřující konverze	56
3.8.2	Zužující konverze	58
3.8.3	Rozšiřující konverze se ztrátou přesnosti	58
3.9	Aritmetické výrazy	59
3.9.1	Unární operátory	60
3.9.2	Binární operátory	61
3.9.3	Přiřazovací operátory	62
3.10	Relační operátory	64
3.10.1	Zkrácené vyhodnocování logických součtů a součinů	64
3.10.2	Úplné vyhodnocování logických součtů a součinů . .	66
3.11	Bitové operace	67
3.11.1	Bitový součin	68
3.11.2	Bitový součet	69
3.11.3	Negace bit po bitu	70
3.11.4	Bitový exkluzivní součet	71
3.11.5	Operace bitového posunu doleva	71
3.11.6	Operace bitového posunu doprava znaménkově . . .	71
3.11.7	Operace bitového posunu doprava neznaménkově .	72
3.12	Priority vyhodnocování operátorů	73
4	Terminálový vstup a výstup	76
4.1	Formátovaný výstup pomocí <code>System.out.print()</code>	76
4.2	Komplexní řešení výstupu	79
4.2.1	Výpis celého čísla v desítkové soustavě	81

4.2.2	Výpis celého čísla v jiných soustavách	83
4.2.3	Výpis znaku	83
4.2.4	Výpis reálného čísla	83
4.2.5	Výpis řetězce	84
4.3	Formátovaný vstup	85
4.3.1	Základní způsoby čtení	85
4.4	Problém vyprázdnění vstupu	87
5	Řídící struktury	92
5.1	Příkaz if a příkaz if-else	92
5.1.1	Neúplná podmínka	92
5.1.2	Úplná podmínka	93
5.2	Podmíněný výraz – ternární operátor	94
5.3	Návěští	96
5.4	Příkazy break a continue	97
5.5	Iterační příkazy – cykly	97
5.5.1	Příkaz while	97
5.5.2	Příkaz do-while	99
5.5.3	Příkaz for	100
5.6	Příkaz switch	106
5.7	Příkaz return	111
6	Metody	114
6.1	Deklarace metody	115
6.2	Metoda bez parametrů	116
6.3	Metoda bez návratového typu – procedura	117
6.3.1	Procedura bez parametrů	117
6.4	Metoda s více parametry různých typů	118
6.5	Rekurzivní metody	119
6.6	Konverze skutečných parametrů a návratové hodnoty metody	120
6.7	Způsoby předávání skutečných parametrů metod	121
6.8	Přetížené metody	121
6.9	Místa deklarací metod	123

6.10	Proměnné z pohledu místa deklarace	124
6.10.1	Nelokální proměnné – “globální” proměnné	124
6.10.2	Proměnné metod – lokální proměnné	126
6.10.3	Zastínění nelokálních proměnných lokálními	127
7	Pole	129
7.1	Pojem referenční datový typ	129
7.2	Deklarace pole	130
7.3	Délka pole	131
7.4	Inicializované pole	132
7.5	Speciality při práci s poli	133
7.5.1	Průchod všemi prvky pole	133
7.5.2	Výpis celého pole najednou	134
7.5.3	Seřazení pole	134
7.6	Dvourozměrná pole	135
7.6.1	Inicializace dvourozměrného pole	136
7.7	Trojrozměrná pole	136
7.8	Více rozměrů v jednorozměrném poli	137
8	Třídy a objekty – základní dovednosti	139
8.1	Deklarace třídy	140
8.2	Vytvoření objektu	141
8.3	Přímý přístup k datům objektu	142
8.4	Práce s metodami	142
8.5	Konstruktory	144
8.5.1	Implicitní konstruktor	147
8.6	Využití <code>this</code> pro přístup k proměnným	147
8.7	Přetížení metod a konstruktorů	148
8.8	Využití <code>this</code> pro přístup ke konstruktoru	150
8.9	Volání metod jinými metodami téže třídy nebo konstruktorem	151
8.10	Použití proměnné třídy v objektech	151
8.11	Použití statických metod v objektech	154
8.11.1	Použití statické metody ze třídy z Java Core API	154

8.11.2	Použití statické metody z téže třídy	156
8.12	Inicializace proměnných tříd	158
8.13	Rušení objektů	160
8.14	Ukončení práce s objekty	161
9	Řetězce a znaky	165
9.1	Vytvoření řetězce	166
9.1.1	Inicializované pole řetězců	167
9.2	Práce s celými řetězci	167
9.2.1	Porovnávání	167
9.2.2	Převody na malá či velká písmena	168
9.2.3	Spojení řetězců	169
9.2.4	Náhrada všech znaků v řetězci	169
9.3	Práce s částí řetězce	169
9.3.1	Získání části řetězce	169
9.3.2	Práce se začátkem a koncem řetězce	170
9.3.3	Oříznutí bílých znaků na okrajích	170
9.4	Práce s jednotlivými znaky řetězce	171
9.4.1	Získání znaku	171
9.4.2	Hledání znaku	171
9.5	Konverze základních datových typů na řetězec	172
9.5.1	Tisk do řetězce	173
9.6	Konverze řetězce na základní datové typy	174
9.7	Vyvolání více metod jedním příkazem	175
9.8	Dělení řetězce na části pomocí <code>split()</code>	175
9.8.1	Základní použití	176
9.8.2	Použití regulárních výrazů	178
9.9	Metoda <code>toString()</code>	180
9.10	Třída <code>StringBuffer</code>	182
9.10.1	Vytvoření instance	182
9.10.2	Délka řetězce	182
9.10.3	Změny celého řetězce	183
9.10.4	Změny části řetězce	183

9.10.5 Konverze na <code>String</code>	185
9.11 Třída <code>Character</code> – práce s jednotlivými znaky	185
9.11.1 Rozpoznávání druhu znaků	185
9.11.2 Změna velikosti písmene	186
9.11.3 Převod jednotlivých znaků na čísla	187
10 Třídy a objekty – pokračování	188
10.1 Modifikátory deklarace třídy	188
10.1.1 Třídy s modifikátorem <code>public</code>	189
10.2 Kompozice objektů	190
10.3 Autorizovaný přístup k datům	192
10.4 Pole objektů	195
10.4.1 Pole objektů obalovacích tříd a automatické převody	197
10.5 Předávání skutečných parametrů metodám	198
10.5.1 Předávání primitivních datových typů	198
10.5.2 Předávání objektů	200
10.5.3 Předávání polí jako skutečných parametrů metodám	202
10.5.4 Předávání vícerozměrných polí metodám	204
11 Dědičnost	206
11.1 Úvodní poznámky	206
11.2 Realizace dědičnosti	207
11.3 Problémy s neimplicitními konstruktory rodičovské třídy .	210
11.4 Nechceme, aby bylo možné metodu překrýt – finální metody	211
11.5 Chceme, aby bylo nutné metodu překrýt – abstraktní metody a třídy	212
11.6 Nechceme, aby bylo možné třídu zdědit – finální třídy . .	214
11.7 Překrytí proměnné	215
11.8 Základem je <code>Object</code>	216
11.8.1 Metoda <code>clone()</code>	217
11.8.2 Metoda <code>equals()</code>	223
11.8.3 Metoda <code>hashCode()</code>	224
11.8.4 Metoda <code>getClass()</code>	225

12 Balíky	228
12.1 Import balíků	229
12.1.1 Statický import balíků	230
12.2 Vytváření balíků	232
12.2.1 Celosvětově platná konvence pro pojmenování . . .	234
12.3 Přístupová práva	234
12.3.1 Specifikátor <code>private</code>	235
12.3.2 Specifikátor <code>protected</code>	237
12.3.3 Specifikátor <code>public</code>	239
12.3.4 Specifikátor neuveden (je “přátelský”)	239
12.4 Při dědění nelze zeslabit přístupová práva	240
13 Rozhraní (interface)	241
13.1 Konstrukce rozhraní	242
13.2 Použití jednoho rozhraní	243
13.3 Použití rozhraní jako typu referenční proměnné	244
13.4 Implementace více rozhraní jednou třídou	245
13.5 Instance rozhraní může využívat jen metody rozhraní . . .	246
13.6 Implementované rozhraní se dědí beze změny	247
13.7 Dědění třídy a současná implementace rozhraní	248
13.8 Dědění rozhraní a konstanty rozhraní	250
13.9 Využití operátoru <code>instanceof</code>	251
14 Polymorfismus	253
14.1 Využití abstraktní třídy	254
14.2 Použití neabstraktních tříd	257
14.3 Použití rozhraní	259
15 Vnořené třídy	261
15.1 Vnitřní třídy	262
15.1.1 Implementace rozhraní pomocí metody využívající vnitřní třídu	263
15.1.2 Implementace rozhraní pomocí metody využívající anonymní vnitřní třídu	264

15.1.3	Proměnná typu rozhraní využívající anonymní vnitřní třídu	266
15.1.4	Vnitřní třída je vytvořená děděním	267
16	Výjimky	269
16.1	Možné druhy výjimek	270
16.1.1	Třída <code>Error</code>	271
16.1.2	Třída <code>RuntimeException</code>	272
16.1.3	Třída <code>Exception</code>	273
16.2	Způsoby ošetření výjimky	274
16.2.1	Předání výjimky výše – deklarace výjimky	274
16.2.2	Kompletní ošetření výjimky	276
16.2.3	Ošetření výjimky a následné předání výše	278
16.2.4	Naprosto nejhorší reakce na výjimku	279
16.2.5	Rozumná reakce na výjimku	280
16.2.6	Elegantní reakce na výjimku	281
16.2.7	Výpis výjimky do okénka	282
16.3	Seskupování výjimek	283
16.3.1	Postupná selekce výjimek	285
16.4	Vyvolání výjimky	286
16.5	Vytvoření a použití vlastní výjimky	287
16.6	Konstrukce <code>try-catch-finally</code>	289
16.6.1	Konstrukce <code>try-finally</code>	291
17	Adresáře a soubory	293
17.1	Nezávislost na operačním systému	293
17.2	Vytvoření instance třídy <code>File</code>	294
17.3	Vytvoření souboru nebo adresáře	295
17.4	Práce se souborem nebo adresářem	296
17.5	Výpis adresáře	297
17.5.1	Selektivní výpis adresáře	298

18 Čtení ze vstupů a zápis na výstupy	301
18.1 Proudly znaků a proudly bajtů	302
18.2 Dva různé typy tříd zděděných od základních tříd	303
18.2.1 Třídy pro práci se zařízeními	304
18.2.2 Třídy vlastností (filtrů)	304
18.3 Čtení ze souboru a zápis do souboru	305
18.3.1 Vstupy a výstupy znaků	305
18.3.2 Vstupy a výstupy bajtů	307
18.3.3 Další dovednosti se soubory	308
18.3.4 Bezpečné zavírání souborů a řešení výjimek	309
18.4 Třídy vlastností	312
18.4.1 Vlastnost: bufferování	313
18.4.2 Vlastnost: čtení po řádcích	313
18.4.3 Vlastnost: výběrové čtení po řádcích	314
18.4.4 Vlastnost: vrácení přečteného znaku	314
18.4.5 Vlastnost: formátování výstupu	315
18.4.6 Vlastnost: formátování výstupu s řádkovým bufferováním	316
18.4.7 Formátovaný vstup pomocí Scanner	316
18.4.8 Formátovaný vstup pomocí FileReader	318
18.4.9 Vlastnost: neformátovaný vstup a výstup základních datových typů	319
18.4.10 Vlastnost: serializace objektů	321
18.4.11 Seskupování vlastností	323
18.5 Vstup a výstup do paměti	324
18.6 Vstup a výstup do roury	325
18.7 Soubory s přímým přístupem	327
19 Systémové akce	333
19.1 Parametry příkazové řádky	333
19.2 Systémové atributy a zdroje	335
19.2.1 Standardní vstupní a výstupní proudy	336
19.2.2 Systémové vlastnosti	338

19.3 Užitečné metody ze třídy <code>System</code>	341
19.3.1 Informace o čase	341
19.3.2 Spuštění garbage collectoru	342
19.3.3 Zjištění velikosti dostupné paměti	343
19.3.4 Spuštění finalizeru	344
19.3.5 Násilné ukončení programu	345
20 Vlákna	347
20.1 Třída <code>Thread</code>	348
20.2 Spolupráce dvou vláken	350
20.2.1 Vlákna se pravidelně střídají	353
20.2.2 Vlákna se střídají nepravidelně	354
20.3 Stavy vlákna a plánovací algoritmus	355
20.3.1 Stavy vlákna	355
20.3.2 Priorita vlákna	356
20.3.3 Sdílení času	357
20.3.4 Praktické ověření plánování a priorit	357
20.4 Rozhraní <code>Runnable</code>	358
20.5 Čekání na vstup či výstup	360
20.6 Synchronizace vláken	362
20.6.1 Vlákno čeká trpělivě na konec jiného vlákna	362
20.6.2 Vlákno čeká netrpělivě na konec jiného vlákna	363
20.6.3 Vlákno ukončí předčasně jiné vlákno	364
20.6.4 Vlákno je násilně probuzeno	366
20.6.5 Kritické sekce – synchronizované metody	367
20.6.6 Kritické sekce – synchronizované bloky	369
20.6.7 Synchronizace časové posloupnosti vláken	370
20.7 Další informace o vláknech	376
20.7.1 Problematika hladovění a uváznutí	376
20.7.2 Skupiny vláken	377
20.7.3 Vlákna typu démon	377
Literatura	380
Rejstřík	381