

# OBSAH

<b>Předmluva</b>	<b>xiv</b>
<b>O autorech</b>	<b>xvi</b>
Přispěli do publikace	xv

---

## **ČÁST I**

### **Začínáme** **1**

#### **KAPITOLA 1**

### **O jazyce Python** **3**

**Proč používat Python?** **3**

**Stručné hodnocení programovacích jazyků** **4**

**Srovnání Pythonu s jinými jazyky** **6**

Python versus C, Pascal, C++ a Java 6

Python versus Visual Basic 7

Python versus Tcl 9

Python versus Perl 10

**V čem je háček?** **11**

**Soužití Pythonu s jiným programovacím jazykem** **12**

**Python a otevřený software** **13**

**Shrnutí** **14**

#### **KAPITOLA 2**

### **O knize** **15**

**Jak s knihou pracovat** **15**

Členění knihy 15

Poznámky po stranách 16

Upozornění 16

**Výuka Pythonu pomocí příkladů** **16**

**Pravidla používaná při formátování kódu příkladů** **16**

Interaktivní ukázky 17

Neinteraktivní příklady 18

**Co v knize najdete** **19**

**Kde naleznete další informace** **22**

**Zpětná vazba** **23**

<b>KAPITOLA 3</b>	
<b>Začínáme</b>	<b>25</b>
Instalace jazyka Python	25
Prostředí IDLE a jeho základní interaktivní režim	27
Základní interaktivní režim	27
Integrované vývojové prostředí IDLE	28
Ahoj, světe !	29
Práce v okně Python Shell prostředí IDLE	30
<hr/>	
<b>ČÁST II</b>	
<b>ZÁKLADY</b>	<b>33</b>
<b>KAPITOLA 4</b>	
<b>Stručný přehled jazyka Python</b>	<b>35</b>
O této kapitole	35
Přehled jazyka Python	36
Interní datové typy	36
Čísla	36
Seznamy	38
N-tice	39
Řetězce	40
Slovníky	41
Souborové objekty	41
Struktury řízení chodu programu	42
Logické hodnoty a výrazy	42
Příkaz if-elif-else	42
Cyklus while	43
Cyklus for	43
Definice funkcí	44
Výjimky	44
Tvorba modulů	45
Objektově orientované programování	47
Shrnutí	48
<b>KAPITOLA 5</b>	
<b>Základy jazyka</b>	<b>49</b>
Odsazování textu a bloková struktura kódu	49
Komentáře	51
Proměnné a přiřazení	51
Výrazy	52
Řetězce	53
Čísla	54
Vestavěné matematické funkce	55
Další matematické funkce	55
Matematické výpočty	56
Komplexní čísla	56

<b>Hodnota None</b>	<b>57</b>
<b>Vestavěné operátory</b>	<b>57</b>
<b>KAPITOLA 6</b>	
<b>Seznamy a n-tice</b>	<b>59</b>
<b>Seznamy jsou jako pole</b>	<b>60</b>
<b>Indexování seznamu</b>	<b>60</b>
<b>Modifikace seznamů</b>	<b>62</b>
<b>Třídění</b>	<b>64</b>
Uživatelské třídění	65
<b>Další běžné operace se seznamy</b>	<b>67</b>
Zjištění existence prvku v seznamu pomocí operátoru in	67
Zřetězení seznamu operátorem +	67
Inicializace seznamu operátorem *	67
Nalezení nejmenšího a největšího prvku seznamu pomocí funkcí min a max	68
Hledání indexu prvku v seznamu	68
Počet výskytů prvku v seznamu	69
<b>Vnořené seznamy a hluboké kopie</b>	<b>69</b>
<b>N-tice</b>	<b>71</b>
Základy n-tic	72
Jednoprvková n-tice	73
Skládání a rozklad n-tice	73
Převody mezi seznamy a n-ticemi	74
<b>Shrnutí</b>	<b>75</b>
<b>KAPITOLA 7</b>	
<b>Řetězce</b>	<b>77</b>
<b>Řetězce jakožto sekvence znaků</b>	<b>77</b>
<b>Základní řetězcové operace</b>	<b>78</b>
<b>Speciální znaky a escape sekvence</b>	<b>78</b>
Základní escape sekvence	78
Numerické (osmičkové a šestnáctkové) escape sekvence	79
Tisk versus vyhodnocení řetězce se speciálními znaky	80
<b>Modul string</b>	<b>80</b>
Funkce string.split a string.join	81
Převod řetězců na čísla	82
Jak se zbavit přebytečných prázdných míst	83
Hledání řetězců	84
Úpravy řetězců	85
Úpravy řetězců prostřednictvím seznamů	87
Užitečné konstanty	87
<b>Převod objektů na řetězce</b>	<b>88</b>
<b>Formátování řetězců</b>	<b>89</b>
Použití formátovacích sekvencí	90
Pojmenované parametry a formátovací sekvence	91

**KAPITOLA 8**

<b>Slovníky</b>	<b>93</b>
Co je slovník?	93
Proč se slovníky nazývají slovníky?	95
Další operace se slovníky	95
Počítání slov	97
Co všechno může být klíčem?	98
Řídké matice	99
Slovník jako vyrovnávací paměť	100
Efektivita slovníků	101

**KAPITOLA 9**

<b>Řízení běhu programu</b>	<b>103</b>
Cyklus while	103
Konstrukce if-elif-else	104
Cyklus for	105
Funkce range	106
Cyklus for a rozbalení prvků n-tic	107
Příkazy, bloky a odsazování	107
Logické hodnoty a výrazy	110
Většina objektů Pythonu může být pravdivostní	111
Porovnávání a logické operátory	111

**KAPITOLA 10**

<b>Funkce a procedury</b>	<b>113</b>
Základní definice funkcí a procedur	113
Přiřazení funkcí proměnným	114
Lambda funkce	114
Možnosti funkčních parametrů	115
Implicitní hodnoty	115
Předávání argumentů jménem parametru	116
Proměnlivý počet argumentů	117
Kombinace obou metod předávání argumentů	118
Měnitelné objekty jako argumenty	118
Lokální a globální proměnné	119
Shrnutí	120

**KAPITOLA 11**

<b>Moduly a pravidla rozsahu</b>	<b>121</b>
Co je modul?	121
První modul	122
Příkaz import	125
Hledání cesty k modulu	126
Kam umístit vaše vlastní moduly	126

Chráněná jména v modulech	128
Knihovna a moduly třetích výrobců	129
Pythonová pravidla rozsahu a prostor jmen	130
<b>KAPITOLA 12</b>	
<b>Práce se souborovým systémem</b>	<b>137</b>
<b>Cesty a jejich popis</b>	<b>138</b>
Absolutní a relativní cesty	138
Aktuální pracovní adresář	139
Manipulace s popisy cest	141
Užitečné konstanty a funkce	143
<b>Jak získat informace o souborech</b>	<b>145</b>
<b>Další operace se souborovým systémem</b>	<b>146</b>
<b>Zpracování všech souborů v adresářovém podstromu</b>	<b>148</b>
<b>Shrnutí</b>	<b>149</b>
<b>KAPITOLA 13</b>	
<b>Čtení a zápis do souborů</b>	<b>151</b>
<b>Otevření souborů a souborových objektů</b>	<b>151</b>
<b>Uzavření souborů</b>	<b>152</b>
<b>Otevření souborů v režimu zápisu nebo jiném</b>	<b>152</b>
<b>Funkce pro čtení a zápis textu nebo binárních dat</b>	<b>153</b>
<b>Funkce vstupu a výstupu na obrazovku a přesměrování</b>	<b>155</b>
<b>Modul struct</b>	<b>158</b>
<b>Nakládání objektů do souboru</b>	<b>160</b>
<b>Umístění objektů v souboru</b>	<b>163</b>
<b>Shrnutí</b>	<b>165</b>
<b>KAPITOLA 14</b>	
<b>Výjimky</b>	<b>167</b>
<b>Úvod do výjimek</b>	<b>167</b>
Obecná filozofie ošetření chyb a výjimek	168
Formálnější definice výjimek	171
Uživatелеm definované výjimky	171
<b>Výjimky v jazyce Python</b>	<b>172</b>
Typy výjimek v jazyce Python	172
Vyvolávání výjimek	173
Zachycení a obsluha výjimek	174
Definování nových výjimek	175
Ladění programů pomocí příkazu assert	176
Hierarchie při dědění výjimek	177
Příklad: náš program pro zápis na disk v jazyce Python	177
Příklad: použití výjimek při běžných výpočtech	178
Kde používat výjimky	180

<b>KAPITOLA 15</b>	
<b>Skripty</b>	<b>181</b>
<b>Vytvoření nejjednoduššího skriptu</b>	<b>182</b>
Spuštění skriptu z příkazového řádku	182
Argumenty předávané z příkazového řádku	182
Přesměrování vstupu a výstupu skriptu	183
Modul getopt	184
Použití modulu fileinput	185
<b>Co udělat, aby byl skript v systému UNIX přímo spustitelný</b>	<b>187</b>
<b>Možnosti spuštění skriptu v systému Windows</b>	<b>187</b>
Spouštění skriptu jako dokumentu nebo zástupce	187
Spouštění skriptu z dialogového okna Spustit	189
Spouštění skriptu z příkazového okna MS-DOS	189
Další možnosti ve Windows	190
<b>Skripty v systému Windows versus skripty v systému UNIX</b>	<b>191</b>
<b>Skripty a moduly</b>	<b>193</b>
<b>Vytvoření spustitelného programu pomocí nástroje freeze</b>	<b>197</b>
<b>Shrnutí</b>	<b>198</b>
<b>KAPITOLA 16</b>	
<b>Třídy a objektově orientované programování</b>	<b>199</b>
<b>Definování tříd</b>	<b>199</b>
Použití instance třídy jako struktury nebo záznamu	200
Objekty, instance tříd a dalších typů jazyka Python	201
<b>Proměnné instance</b>	<b>201</b>
<b>Metody</b>	<b>202</b>
<b>Proměnné třídy</b>	<b>203</b>
Úskalí při použití proměnných tříd	204
<b>Metody třídy</b>	<b>206</b>
<b>Dědění</b>	<b>207</b>
<b>Dědění s proměnnými tříd a instancí</b>	<b>209</b>
<b>Soukromé proměnné a metody</b>	<b>211</b>
<b>Pravidla rozsahu platnosti a jmenné prostory pro instance tříd</b>	<b>212</b>
<b>Destruktory a správa paměti</b>	<b>216</b>
<b>Vícenásobná dědičnost</b>	<b>219</b>
<b>Shrnutí</b>	<b>221</b>
<b>KAPITOLA 17</b>	
<b>Grafická uživatelská rozhraní a knihovna Tk</b>	<b>223</b>
<b>Instalace knihovny Tk</b>	<b>224</b>
<b>Spuštění knihovny Tk a použití modulu Tkinter</b>	<b>225</b>
<b>Principy knihovny Tk</b>	<b>225</b>
Grafické prvky	225
Pojmenované atributy	226
Správa geometrie a rozmisťování prvků	227

<b>Jednoduchá aplikace s modulem Tkinter</b>	<b>228</b>
<b>Tvorba grafických prvků</b>	<b>229</b>
<b>Rozmístování grafických prvků</b>	<b>231</b>
<b>Co dalšího se dá s modulem Tkinter dělat?</b>	<b>233</b>
Obsluha událostí	234
Plátno a text	235
Doplňky knihovny Tk a modulu Tkinter	236
<b>Alternativy ke knihovně Tk a modulu Tkinter</b>	<b>237</b>
<b>Shrnutí</b>	<b>238</b>
Poznámka	238

## ČÁST III

# **Pokročilé vlastnosti jazyka** **239**

### KAPITOLA 18

## **Balíčky** **241**

<b>Co je to balíček</b>	<b>241</b>
<b>První ukázka</b>	<b>241</b>
<b>Skutečný příklad</b>	<b>243</b>
Základní použití balíčku matproj	244
Zavedení vnořených balíčků a modulů	245
Příkazy import uvnitř balíčků	246
Soubory __init__.py	246
<b>Atribut __all__</b>	<b>247</b>
<b>Správné použití balíčků</b>	<b>248</b>

### KAPITOLA 19

## **Datové typy jako objekty** **249**

<b>Typy jsou také objekty</b>	<b>249</b>
<b>Použití typů</b>	<b>250</b>
<b>Modul types</b>	<b>250</b>
<b>Typy a uživatelské třídy</b>	<b>251</b>

### KAPITOLA 20

## **Speciální metody** **255**

<b>Co je to speciální metoda?</b>	<b>255</b>
<b>Jak vytvořit objekt tak, aby se choval jako seznam</b>	<b>257</b>
Speciální metoda __getitem__	258
Jak to pracuje	259
Implementace všech vlastností seznamu	259
<b>Vzorový problém 2</b>	<b>260</b>
<b>Kdy speciální metody použít</b>	<b>262</b>

**KAPITOLA 21**

<b>Regulární výrazy</b>	<b>265</b>
Co je regulární výraz?	265
Regulární výrazy s metaznaký	266
Regulární výrazy a neupravené řetězce	267
Neupravené řetězce – naše záchrana	268
Získání odpovídajícího textu z řetězce	269
Nahrazování textu pomocí regulárních výrazů	272
Co dalšího lze s regulárními výrazy dělat?	274

**ČÁST IV**

<b>Pokročilá témata a aplikace</b>	<b>275</b>
------------------------------------	------------

**KAPITOLA 22**

<b>Python, Windows a objekty COM</b>	<b>277</b>
Úvod	277
Jak používat Python s objekty COM	278
Instalace a nastavení	280
Anatomie serveru COM v Pythonu	281
Import modulů pro práci s objekty COM	281
Nastavení třídy serveru COM	282
Psaní metod třídy serveru	283
Registrace třídy serveru Pythonu pro použití s ostatními objekty COM	283
Testovací kód	284
Vytvoření těla modulu	284
Šablona pro server COM Pythonu vcelku	285
Vytvoření a použití serveru z jazyka Visual Basic	286
Řešení problémů se serverem	287
Použití serveru	288
Zastavení serveru	288
Výměna dat se serverem	289
Čísla	289
Řetězce	289
Získání seznamu	290
Předání seznamu	290
Získání tabulkových dat	290
Více o polích typu Variant	291
Nepravidelné seznamy	292
Předání objektů	292
Zpětná volání	293
Program Excel jako klient	294
Nastavení sešitu programu Excel pro práci se serverem	295
Vývolání dat	296
Co dál?	297
Distribuované objekty COM	297
Klientská strana objektů COM – automatizace programů Office	298



Vyvolání dat z databáze Access	298
Aktualizace záznamu v databázi Access	299
Spuštění a vložení dat do programu Excel	299
Spuštění a vložení dat do programu Word	300
<b>Typové knihovny</b>	<b>300</b>
<b>Pokyny</b>	<b>301</b>
<b>Další specialitky</b>	<b>302</b>
Hostitelské skriptování ve Windows	302
ODBC	303
Použití s knihovnou MFC	304
Další rozhraní API Windows	304
Volání libovolné knihovny DLL	304
<b>Zdroje informací</b>	<b>305</b>
<b>KAPITOLA 23</b>	
<b>Rozšíření Pythonu pomocí jazyků C a C++</b>	<b>307</b>
Účel této kapitoly	307
Podrobnosti o kompilování	308
Jaký kompilátor?	308
Statické a dynamické rozšiřující moduly	308
Podrobnosti o kompilování v systému	309
První příklad	309
Vytvoření šablony	313
Správa paměti	314
Potřeba správy paměti	314
Jednoduchý příklad	315
Kdy je čítač odkazů snížen?	316
Počítání odkazů je věda	317
<b>Jak psát rozšiřující moduly v jazyce C++</b>	<b>317</b>
<b>Co dál</b>	<b>317</b>
<b>KAPITOLA 24</b>	
<b>Integrace s javovským virtuálním strojem: JPython</b>	<b>319</b>
<b>Co je to JPython</b>	<b>319</b>
<b>Prolog: falešná iluze o jediném jazyce</b>	<b>320</b>
A co Java?	321
<b>JPython, bomba mezi skriptovacími jazyky</b>	<b>322</b>
<b>Stažení a instalace překladače JPython</b>	<b>324</b>
Test funkčnosti JPythonu	325
<b>Použití Javy z jazyka JPython</b>	<b>326</b>
Rozšiřování javových tříd	329
<b>Použití jazyka JPython z Javy</b>	<b>330</b>
<b>Kompilování tříd jazyka JPython</b>	<b>334</b>
Poznámky k používání skriptu jpythone	337
<b>Společné použití Javy a jazyka JPython</b>	<b>337</b>
<b>Závěr</b>	<b>338</b>

**KAPITOLA 25**

<b>HTML a Python – balíček HTMLgen</b>	<b>339</b>
<b>K čemu můžeme knihovnu HTMLgen použít</b>	<b>340</b>
Příklad použití knihovny HTMLgen	340
<b>Jak knihovna HTMLgen zapisuje objekty</b>	<b>346</b>
<b>Dokumentový model</b>	<b>346</b>
Hierarchie tříd dokumentu	346
<b>Tabulky</b>	<b>349</b>
Tabulka rychle, ale nepěkně	349
Tabulka se vším všudy	350
<b>Seznamy</b>	<b>351</b>
<b>Rámce</b>	<b>352</b>
<b>Obrázky</b>	<b>354</b>
<b>Obrazové mapy</b>	<b>354</b>
<b>Značka &lt;A&gt;</b>	<b>355</b>
<b>Volání třídy jako funkce</b>	<b>356</b>
<b>Podpora kaskádových stylů CSS1</b>	<b>357</b>
Pokročilé vlastnosti stylů	359
<b>Skripty CGI a formuláře</b>	<b>359</b>
Skripty CGI a třída StickyForm	363
Skripty CGI a aplikační server Zope	363
<b>Doplňkové moduly</b>	<b>363</b>
<b>Budoucnost</b>	<b>364</b>

**KAPITOLA 26**

<b>Použití systému Zope</b>	<b>365</b>
<b>Úvod</b>	<b>365</b>
<b>Publikování objektů</b>	<b>366</b>
Co je to publikování objektu?	367
Průchod k objektu: od adresy URL k volání objektu	367
Publikování objektu v detailech	369
Jednoduchý příklad	370
Návštěvní kniha a generátor reklamy	372
<b>Pokročilé vlastnosti systému Zope</b>	<b>377</b>
Generování kódu HTML pomocí balíčku DocumentTemplate	378
Podmínky, posloupnosti a výrazy	380
Tvorba trvalých objektů pomocí databáze BoboPOS	382
Další vlastnosti	384
Nabídka zaměstnání na Internetu	385

**KAPITOLA 27**

<b>Co ještě Python umí?</b>	<b>393</b>
-----------------------------	------------

---

**PŘÍLOHA**

<b>Stručná referenční příručka jazyka Python</b>	<b>395</b>
Konvence použité v příručce	396
Speciální identifikátory	397
Operátory	397
Řazení porovnávání	399
Čísla	400
Řetězce	404
Seznamy	410
N-tice (datový typ tuple)	413
Slovníky	415
Příkazy, řízení chodu a definice funkce	416
Moduly, skripty a balíčky	419
Soubory a adresáře	423
Třídy	432
Speciální metody	432
Výjimky	436
Regulární výrazy	439
Další užitečné příkazy, funkce a moduly	443
Stručný přehled interaktivního režimu překladače Pythonu	447
 <b>Rejstřík</b>	 <b>451</b>

# Předmluva

Před sebou máte jednoduchou, avšak velmi obsažnou knihu o jednoduchém, přitom však rozhodně velmi schopném jazyce – Pythonu. Napsali jsme ji, protože jsme chtěli, aby jeho naučení bylo stejně snadné jako je vlastní práce s ním. Naším cílem je naučit vás základy programování v Pythonu a seznámit vás s některými jeho skutečně užitečnými knihovny.

Význačnými charakteristickými rysy Pythonu jsou snadnost používání a obdivuhodné možnosti, jaké nenabízí žádný z dnes rozšířených jazyků. Je vhodný jak pro krátké skripty, tak i pro rozsáhlé programy, a pokud jde o produktivitu programátora, snadno předčí Perl, C, C++, Javu a všechny ostatní jazyky, které jsme dosud měli možnost poznat. A učit se Python je proto také opravdu příjemně strávený čas. Tento jazyk je k dispozici již ve zralé a stabilní podobě pro všechny významnější platformy a je vybaven rozhraním umožňujícím použití toolkitu s multiplatformním grafickým uživatelským rozhraním (GUI). Dodává se s vlastním integrovaným vývojovým prostředím a k dispozici začínají být mnohé další nové knihovny. Má za sebou již i úspěšný debut v podobě integrace do jazyka Java.

Abyste se Python z této knihy naučili, měli byste už znát docela zevrubně některý z počítačových jazyků. Jazyky C, C++, Pascal, Perl, Visual Basic, Java či mnohé jiné vám k tomu dávají dostatečnou praxi. Nemusíte v nich být expertem; bude vám stačit, chápete-li jejich základy jako jsou řetězce, pole, cykly, podmíněné příkazy (typu „if-then“) apod.

Výhodou přitom pro vás bude, ovládáte-li objektové programování. I když však nebudete o OOP nic znát (a třeba ani nebudete ani chtít), neměli byste mít s pochopením Pythonu problémy. Python je totiž také vynikajícím procedurálním jazykem.

Python je zřejmě také jedním z nejsnadněji pochopitelných programovacích jazyků. Tuto knihu píšeme tak, abyste byli schopni Python co nejrychleji zvládnout. Naším záměrem je poskytnout vám více než pouhého začátečnického průvodce. Snažili jsme se o to, aby vám kniha i poté, co se stanete zkušenějšími pythonisty, byla velmi užitečným pomocníkem jakožto indexovaná vyrovnávací paměť obsahující informace, které si při návrhu a konstrukci programů budete potřebovat nejčastěji připomenout.