

# Sadržaj

<b>Predgovor</b>	<b>xiii</b>
<b>Zahvalnice</b>	<b>xvii</b>
<b>O autoru</b>	<b>xix</b>
<b>Poglavlje 1 Pythonsko razmišljanje</b>	<b>1</b>
Tema 1: Saznajte koju verziju Pythona koristite	1
Tema 2: Pridržavajte se smernica za stil PEP 8	2
Tema 3: Naučite razliku između bytes i str	4
Tema 4: Dajte prednost interpoliranim f-šablonima u odnosu na šablone formata u stilu C-a i str.format metode	10
Tema 5: Pišite pomoćne funkcije umesto složenih izraza	21
Tema 6: Dajte prednost raspakivanju sa višestrukom dodelom u odnosu na indeksiranje	23
Tema 7: Dajte prednost enumerate u odnosu na range	27
Tema 8: Koristite zip da paralelno obrađujete iteratore	29
Tema 9: Izbegavajte blokove else posle petlji for i while	32
Tema 10: Sprečite ponavljanje pomoću izraza dodele	34
<b>Poglavlje 2 Liste i rečnici</b>	<b>41</b>
Tema 11: Naučite da izvadite deo niza	41
Tema 12: Izbegavajte koraćanje i isecanje u jednom izrazu	44
Tema 13: Dajte prednost raspakivanju sa hvatanjem u odnosu na isecanje	46
Tema 14: Sortirajte po složenim kriterijumima korišćenjem parametra key	50

## viii Sadržaj

Tema 15: Budite oprezni kada se oslanjate na redosled umetanja u dict	56
Tema 16: Za nedostajuće ključeve rečnika dajte prednost metodi get u odnosu na in i KeyError	64
Tema 17: Za nedostajuće stavke u unutrašnjem stanju klase dajte prednost klasi defaultdict u odnosu na setdefault	68
Tema 18: Naučite da koristeći __missing__ konstruišete početne vrednosti koje zavise od ključa	71
<b>Poglavlje 3 Funkcije</b>	<b>75</b>
Tema 19: Nikada ne raspakujte više od tri promenljive kada funkcije vraćaju više vrednosti	75
Tema 20: Dajte prednost generisanju izuzetaka u odnosu na vraćanje None	78
Tema 21: Naučite kako međusobno deluju zatvaranja i opseg promenljive	81
Tema 22: Smanjite vizuelni šum pomoću promenljivog broja pozicionih argumenata	85
Tema 23: Obezbedite opciono ponašanje pomoću argumenata po imenu	88
Tema 24: Koristite None i Docstring da zadate početne dinamičke argumente	92
Tema 25: Budite jasniji korišćenjem argumenata po imenu i po poziciji	95
Tema 26: Definišite dekoratore funkcija pomoću functools.wraps	100
<b>Poglavlje 4 Sagledavanje i generatori</b>	<b>105</b>
Tema 27: Koristite sagledavanje umesto funkcija map i filter	105
Tema 28: Izbegavajte više od dva kontrolna podizraza u sagledavanju	107
Tema 29: Izbegavajte ponavljanje posla pri sagledavanju pomoću izraza dodele	109
Tema 30: Razmotrite korišćenje generatora umesto vraćanja liste	112
Tema 31: Budite uzdržani kada prolazite kroz argumente iteracijom	115
Tema 32: Razmotrite korišćenje generatorskih izraza za sagledavanja velikih lista	120
Tema 33: Spajajte više generatora koristeći izraz yield from	122
Tema 34: Izbegavajte ubrizgavanje podataka u generatore metodom send	125

Tema 35: Izbegavajte promene stanja u generatorima nastale sa throw	131
Tema 36: Koristite itertools za rad sa iteratorima i generatorima	135
<b>Poglavlje 5 Klase i interfejsi</b>	<b>141</b>
Tema 37: Spajajte klase umesto da ugnežđujete više nivoa ugrađenih tipova	141
Tema 38: Prihvatajte funkcije umesto klasa za jednostavne interfejse	148
Tema 39: Koristite polimorfizam @classmethod da biste generički konstruisali objekte	151
Tema 40: Inicijalizujte roditeljsku klasu koristeći super	156
Tema 41: Razmislite o spajanju funkcionalnosti pomoću mix-in klasa	161
Tema 42: Dajte prednost javnim atributima u odnosu na privatne	166
Tema 43: Za namenske tipove kontejnera nasleđujte od collections.abc	171
<b>Poglavlje 6 Metaklase i atributi</b>	<b>177</b>
Tema 44: Koristite čiste attribute umesto metoda za zadavanje i očitavanje vrednosti	177
Tema 45: Razmotrite korišćenje dekoratora @property umesto prerađivanja atributa	182
Tema 46: Koristite deskriptore za višekratne metode @property	186
Tema 47: Koristite __getattr__, __getattribute__ i __setattr__ za spore attribute	191
Tema 48: Proverite potklasu sa __init_subclass__	197
Tema 49: Registrujte postojanje klase sa __init_subclass__	205
Tema 50: Obeležite attribute klase sa __set_name__	210
Tema 51: Dajte prednost dekoratorima klase u odnosu na metaklase za sastavljiva proširenja klase	215
<b>Poglavlje 7 Konkurentnost i paralelnost</b>	<b>221</b>
Tema 52: Koristite subprocess da biste upravljali procesima potomcima	222
Tema 53: Koristite niti za blokiranje U/I, a izbegavajte za paralelnost	226
Tema 54: Koristite Lock da biste sprečili utrkivanje podataka u nitima	231

## x Sadržaj

Tema 55: Koristite Queue da biste koordinisali rad niti	234
Tema 56: Naučite da prepoznate kada je potrebna konkurentnost	243
Tema 57: Izbegavajte pravljenje novih instanci Thread za raspodelu po zahtevu	248
Tema 58: Razumite zašto upotreba klase Queue za konkurentnost zahteva refaktorisanje	252
Tema 59: Uzmite u obzir klasu ThreadPoolExecutor kada su niti neophodne za konkurentnost	259
Tema 60: Postignite visoko konkurentan U/I pomoću korutina	262
Tema 61: Naučite kako da prebacite U/I u nitima na asyncio	266
Tema 62: Kombinujte niti i korutine da biste olakšali prelaz na asyncio	277
Tema 63: Za što veću odzivnost izbegavajte blokiranje petlje događaja asyncio	283
Tema 64: Razmislite o korišćenju modula concurrent.futures za pravu paralelnost	286

## **Poglavlje 8 Robusnost i performanse** **293**

Tema 65: Iskoristite prednosti svakog try/except/else/finally bloka	293
Tema 66: Razmotrite naredbe contextlib i with za ponovljivo ponašanje try/finally	298
Tema 67: Koristite datetime umesto time za lokalne časovnike	302
Tema 68: Učinite pickle pouzdanim pomoću copyreg	307
Tema 69: Koristite modul decimal kada je neophodna preciznost	313
Tema 70: Profilišite pre optimizovanja	316
Tema 71: Dajte prednost klasi deque za redove čekanja proizvođač-korisnik	321
Tema 72: Razmotrite pretraživanje sortiranih nizova pomoću bisect	328
Tema 73: Naučite kako da koristite heapq za prioritetne redove čekanja	331
Tema 74: Razmotrite memoryview i bytearray za interakcije bez kopija sa instancama bytes	341

## **Poglavlje 9 Testiranje programa i otklanjanje grešaka** **347**

Tema 75: Za otklanjanje grešaka u rezultatu, koristite znakovne nizove repr	348
Tema 76: Proverite povezana ponašanja u potklasama TestCase	351

Tema 77: Izolujte testove jedne od drugih pomoću <code>setUp</code> , <code>tearDown</code> , <code>setUpModule</code> i <code>tearDownModule</code>	358
Tema 78: Koristite simulatore za testiranje koda sa složenim zavisnostima	361
Tema 79: Kapsulirajte zavisnosti da biste olakšali simuliranje i testiranje	369
Tema 80: Razmotrite interaktivno otklanjanje grešaka pomoću <code>pdb</code>	373
Tema 81: Upotrebite <code>tracemalloc</code> da biste razumeli za šta se koristi i gde curi memorija	377
<b>Poglavlje 10 Saradnja</b>	<b>381</b>
Tema 82: Saznajte kako da pronađete module koje pravi zajednica korisnika	381
Tema 83: Koristite virtuelna okruženja za izolovane i reprodukovane zavisnosti	382
Tema 84: Pišite Docstring dokumentaciju za svaku funkciju, klasu i modul	388
Tema 85: Koristite pakete da biste organizovali module i obezbedili stabilne API-je	393
Tema 86: Razmotrite kod u opsegu modula za konfigurisanje upotrebnih okruženja	397
Tema 87: Definišite korenski <code>Exception</code> da biste izolovali pozivaoce od API-ja	400
Tema 88: Naučite kako da prekinete kružne zavisnosti	404
Tema 89: Razmotrite <code>warnings</code> u refaktorisanju i migraciji	409
Tema 90: Razmotrite statičku analizu putem modula <code>typing</code> da biste preduhitрили greške	416
<b>Indeks</b>	<b>425</b>
<b>Spisak termina korišćenih u knjizi</b>	<b>435</b>