

Predgovor

Programski jezik Python ima jedinstvene snage i šarm koje je ponekad teško sagledati. Mnogi programeri kojima su poznati drugi jezici često pristupaju Pythonu sa ograničenim stavovima umesto da prihvate njegovu punu izražajnost. Neki programeri idu predaleko u drugom smeru, preterano koristeći Pythonove mogućnosti, što kasnije može izazvati velike probleme.

Ova knjiga pruža uvid u *pythonski* način pisanja programa: najbolji način da se koristi Python. Ona nadograđuje osnovno razumevanje jezika za koje pretpostavljam da već imate. Programeri početnici će naučiti najbolje strategije za Pythonove mogućnosti. Iskusni programeri će naučiti kako da samouvereno prihvate snage novog alata.

Moj cilj je da vas pripremim za postizanje velikog uticaja sa Pythonom.

Koje oblasti knjiga pokriva

Svako poglavlje knjige sadrži širok, ali povezan skup tema. Slobodno skačite sa teme na temu i pratite sopstvena interesovanja. Svaka tema sadrži koncizne i specifične smernice koje objašnjavaju kako da efikasnije pišete programe na Pythonu. Teme sadrže savete o tome šta da radite, šta da izbegavate, kako da postignete ravnotežu i zbog čega je to najbolji izbor. Teme referenciraju jedna drugu da biste lakše popunili praznine dok čitate.

Ovo drugo izdanje knjige fokusirano je isključivo na Python 3 (Tema 1: Saznajte koju verziju Pythona koristite), zaključno sa verzijom 3.8. Većina originalnih tema iz prvog izdanja prerađena je i uključena u knjigu, ali mnoge su pretrpele znatno ažuriranje. Za neke teme, moji saveti su se sasvim promenili između dva izdanja knjige, jer su se najbolje strategije razvijale kako je Python sazrevao. Ukoliko i dalje uglavnom koristite Python 2, uprkos njegovom kraju zakazanom za 1. januar 2020, prethodno izdanje knjige možda će vam biti korisnije.

Python za svoju standardnu biblioteku ima pristup „baterije su priložene“, za razliku od mnogih drugih programskih jezika koji se dobijaju sa malim brojem uobičajenih paketa i zahtevaju da bitne funkcionalnosti

tražite na drugim mestima. Mnogi od tih ugrađenih paketa toliko su čvrsto prepleteni sa idiomatskim Pythonom da bi lako mogli biti i deo specifikacije jezika. Pun skup standardnih modula je prevelik da bi se pokrio u ovoj knjizi, ali uključio sam one koje smatram ključnim za poznavanje i korišćenje.

Poglavlje 1: Pythonsko razmišljanje

Zajednica korisnika Pythona počela je da koristi pridev *pythonski* da bi opisala kod koji se pridržava određenog stila. Pythonovi idiomi su vremenom izronili kroz iskustva korišćenja jezika i rada sa drugima. Ovo poglavlje se bavi najboljim načinom da se u Pythonu obave najuobičajenije stvari.

Poglavlje 2: Liste i rečnici

Najčešći način da se u Pythonu organizuju informacije jeste niz vrednosti uskladišten u listi (`list`). Prirodna dopuna liste je rečnik (`dict`) koji skladišti ključeve za pretraživanje uparene sa odgovarajućim vrednostima. Ovo poglavlje objašnjava kako da izgradite programe od tih svestranih gradivnih blokova.

Poglavlje 3: Funkcije

Funkcije u Pythonu imaju razne dodatne mogućnosti koje programerima olakšavaju život. Neke su slične mogućnostima u drugim programskim jezicima, ali mnoge su jedinstvene za Python. Ovo poglavlje objašnjava kako da koristite funkcije da razjasnite nameru, promovirate višekratnost i smanjite programske greške.

Poglavlje 4: Sagledavanje i generatori

Python ima posebnu sintaksu za brze iteracije kroz liste, rečnike i skupove da bi se generisale izvedene strukture podataka. On omogućava i da funkcija postepeno vraća tok vrednosti nad kojima se mogu praviti iteracije. Ovo poglavlje objašnjava kako te mogućnosti mogu obezbediti bolje performanse, smanjiti utrošak memorije i poboljšati preglednost.

Poglavlje 5: Klase i interfejsi

Python je objektno orijentisan jezik. Završavanje stvari u Pythonu često zahteva da pišete nove klase i definišete njihove interakcije kroz njihove interfejsne i hijerarhijske. Ovo poglavlje objašnjava kako se klase koriste za izražavanje namernih ponašanja pomoću objekata.

Poglavlje 6: Metaklase i atributi

Metaklase i dinamički atributi su moćne Pythonove opcije. Oni omogućavaju i da implementirate izuzetno bizarna i neočekivana ponašanja. Ovo poglavlje pokriva uobičajene idiome za upotrebu tih mehanizama da biste se sigurno pridržavali *pravila najmanjeg iznenađenja*.

Poglavlje 7: Konkurentnost i paralelnost

Python olakšava pisanje konkurentnih programa koji rade više različitih stvari, naizgled istovremeno. Python se može koristiti i za paralelan rad preko sistemskih poziva, potprocesa i C proširenja. Ovo poglavlje objašnjava kako da na najbolji način koristite Python u tim neznatno različitim situacijama.

Poglavlje 8: Robusnost i performanse

Python ima ugrađene mogućnosti i module koji pomažu u ojačavanju programa tako da budu pouzdani. Python sadrži i alate koji vam pomažu da postignete visoke performanse uz minimalan trud. Ovo poglavlje objašnjava kako da koristite Python da biste optimizovali programe za maksimalnu preglednost i efikasnost u produkciji.

Poglavlje 9: Testiranje i otklanjanje programskih grešaka

Uvek treba da testirate kod, bez obzira na jezik kojim je napisan. Međutim, Pythonove dinamičke osobine na jedinstven način mogu uvećati rizik za greške tokom izvršavanja. Srećom, one istovremeno olakšavaju pisanje testova i dijagnostikovanje programa koji ne rade dobro. Ovo poglavlje pokriva Pythonove ugrađene alate za testiranje i otklanjanje programskih grešaka.

Poglavlje 10: Saradnja

Saradnja na Pythonovim programima zahteva da svoj kod pišete promišljeno. Čak i ako radite sami, želećete da razumete kako se koriste moduli koje pišu drugi. Ovo poglavlje se bavi standardnim alatima i najboljim praksama koji omogućavaju ljudima da rade zajedno na Pythonovim programima.

Konvencije korišćene u knjizi

Delovi Pythonovog koda u ovoj knjizi ispisani su neprooporionalnim fontom i imaju istaknutu sintaksu. Kada su redovi dugački, koristim znak (↪) da pokažem gde se prelamaju. Skratio sam neke delove koda koristeći tri tačke (...) da bih označio da se kod nastavlja, ali nije bitan za prikazivanje poente. Treba da preuzmete ceo kod primera (pogledajte dole gde ga možete naći) da bi ti skraćeni delovi koda ispravno radili na vašem računaru.

Malo sam odstupio od Pythonovih smernica za stil da bi primeri koda bolje odgovarali formatu knjige, ili da bih istakao najbitnije delove. Takođe, izostavio sam ugrađenu dokumentaciju da bih skratio primere koda. Preporučujem da ovo ne imitirate u svojim projektima, već da se pridržavate smernica za stil (Tema 2: Pridržavajte se smernica za stil PEP 8) i da pišete dokumentaciju (Tema 84: Pišite docstring dokumentaciju za svaku funkciju, klasu i modul).

Većinu delova koda iz ove knjige prati odgovarajući rezultat dobijen pokretanjem koda. Kada kažem „rezultat“, mislim na izlaz na

konzoli ili terminalu: ono što vidite kada izvršite Pythonov program u interaktivnom interpreteru. Delovi koda sa rezultatom prikazani su neproporcionalnim fontom i prethodi im red `>>>` (Pythonov interaktivni odzivnik). Zamisao je da možete da upišete delove koda u Pythonovo komandno okruženje i reprodukujete očekivani rezultat.

Najzad, ima nekih delova koda ispisanih neproporcionalnim fontom pre kojih ne stoji red `>>>`. Oni predstavljaju rezultat izvođenja programa izvan običnog Pythonovog interpretera. Ti primeri često počinju znakom `$` koji označava da programe izvršavam iz ljuske sa komandnom linijom kao što je Bash. Ako te komande izvršavate na Windowsu ili nekom drugom tipu sistema, možda ćete morati da podesite imena programa i argumente u skladu s tim.

Gde se nalaze kod i ispravke štamparskih grešaka

Korisno je pogledati neke primere iz ove knjige kao cele programe, neprepletene sa tekстом. Tako ćete imati priliku i da sami petljate po kodu i razumete zašto program radi kako je opisano. Izvorni kod za sve delove koda iz knjige možete pronaći na veb lokaciji knjige, <https://effective-python.com>. Ona sadrži i ispravke za knjigu i objašnjava kako da prijavite greške.