

Poglavlja

	Uvod	xxv
1	Dobro došli u projektne obrasce: <i>uvod u projektne obrasce</i>	1
2	Neka objekti budu u toku: <i>obrazac Observer</i>	37
3	Dekorisanje objekata: <i>obrazac Decorator</i>	79
4	Dobrobiti OO dizajna: <i>obrazac Factory</i>	109
5	Jedinstveni objekti: <i>obrazac Singleton</i>	169
6	Kapsuliranje poziva: <i>obrazac Command</i>	191
7	Biti adaptivan: <i>obraci Adapter i Facade</i>	237
8	Kapsuliranje algoritama: <i>obrazac Template Method</i>	277
9	Kolekcije kojima se dobro upravlja: <i>obraci Iterator i Composite</i>	317
10	Stanje stvari: <i>obrazac State</i>	381
11	Upravljanje pristupom objektima: <i>obrazac Proxy</i>	425
12	Obrasci obrazaca: <i>kombinovani obrasci</i>	493
13	Obrasci u stvarnosti: <i>bolji život sa obrascima</i>	563
14	Dodatak: <i>preostali obrasci</i>	597

Sadržaj

Uvod

Vaš mozak o projektnim obrascima. I evo vas, pokušavate da naučite nešto, a vaš mozak vam čini uslugu starajući se da vam ništa od toga ne ostane u glavi. Vaš mozak misli, „Bolje da ostavim mesta za bitnije stvari, kao što su divlje životinje koje treba izbegavati i procenu da li je snoubording bez odeće loša zamisao.“ Pa, kako ćete prevatiti svoj mozak da misli da vam život zavisi od poznavanja projektnih obrazaca?

Kome je knjiga namenjena?	xxvi
Znamo šta mislite.	xxvii
I znamo šta misli vaš mozak.	xxvii
Čitaoca knjige „Um caruje” smatramo učenikom.	xxviii
Metakognicija: razmišljanje o razmišljanju	xxix
Evo šta smo MI uradili:	xxx
Evo šta VI možete da uradite da biste pokorili svoj mozak	xxxi
Pročitaj me	xxxii
Recenzenti	xxxiv
Zahvalnice	xxxvi

Dobro došli u projektne obrasce

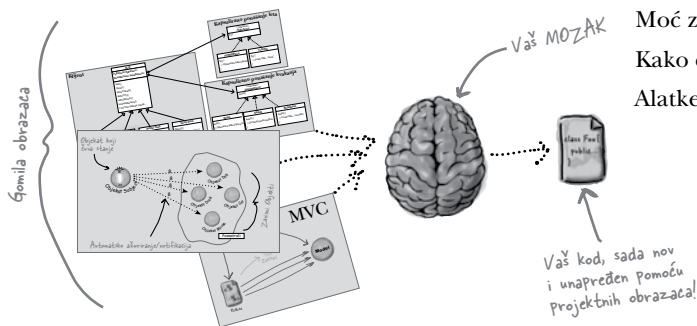
1

Neko je već rešio vaše probleme. U ovom poglavlju ćete naučiti zašto (i kako) da koristite mudrosti i lekcije do kojih su došli drugi programeri koji su išli istim putem rešavanja problema pri projektovanju i preživeli. Pre nego što završimo, razmotrićemo upotrebu i koristi od projektnih obrazaca, pogledati neke principe objektno orijentisanog (OO) projektovanja i proći kroz primer rada jednog obrasca. Najbolji način da koristite obrasce jeste da njima *ispunite svoju glavu* i da potom *prepoznate mesta* u svojim projektima i postojećim aplikacijama gde biste mogli da ih *primenite*. Umesto ponovnog korišćenja koda, obrasci vam pružaju ponovno korišćenje *iskustva*.

Upamti, poznavanje koncepta kao što su apstrakcija, nasleđivanje i polimorfizam ne čine te dobrim objektno orijentisanim projektantom. Projektni guru razmišlja o tome kako da napravi prilagodljivije projekte koji su održivi i mogu da se izbere sa promenom.



Počelo je jednostavnom aplikacijom SimUDuck	2
Ali patke sada treba da LETE	3
Ali, nešto je pošlo naopako...	4
Džo razmišlja o nasleđivanju...	5
A šta je sa interfejsom?	6
Šta biste vi uradili da ste Džo?	7
Jedina konstanta u razvoju softvera	8
Fokusiranje na problem...	9
Odvajanje onoga što se menja od onoga što ostaje isto	10
Projektovanje ponašanja pataka	11
Implementiranje pačjih ponašanja	13
Integrisanje pačjih ponašanja	15
Testiranje koda Duck	18
Dinamičko zadavanje ponašanja	20
Šira slika kapsuliranih ponašanja	22
IMA može biti bolje nego JESTE	23
Kad smo kod projektnih obrazaca...	24
Načuli smo u lokalnom restoranu...	26
Načuli smo za susednim stolom...	27
Moć zajedničkog rečnika obrazaca	28
Kako da koristim projektne obrasce?	29
Alatke za projektovanje	32

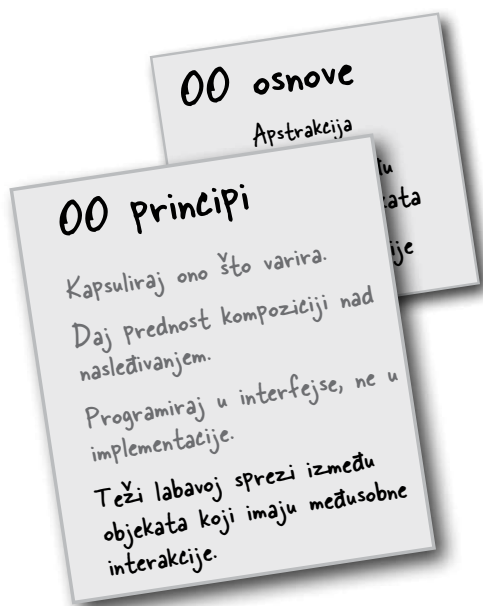


obrazac Observer

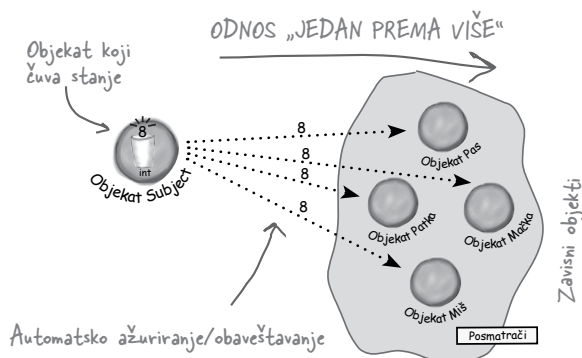
2

Neka objekti budu u toku

Ne želite da propustite neko zanimljivo dešavanje, zar ne? Imamo obrazac koji *upućuje* objekte kada se dešava nešto *do čega im je stalo*. To je obrazac Observer. To je jedan od najčešće korišćenih projektnih obrazaca, i neverovatno je koristan. Razmotrićemo sve zanimljive aspekte Posmatrača, kao što su *veza jedan-prema-više* i *labava sprega*. Imajući te koncepte na umu, kako možete odoleti da ne budete najzanimljiviji na Zabavi obrazaca?



Pregled aplikacije za praćenje vremena	39
Upoznajte obrazac Observer	44
Izdavači + pretplatnici = obrazac Observer	45
Definicija obrasca Observer	51
Moć labavog povezivanja	54
Projektovanje meteorološke stanice	57
Implementiranje meteorološke stanice	58
Pokretanje meteorološke stanice	61
Kodiranje aplikacije koja menja život	66
U međuvremenu, u Weather-O-Rami	69
Testiranje novog koda	71
Alatke za projektovanje	72
Izazov principa projektovanja	73

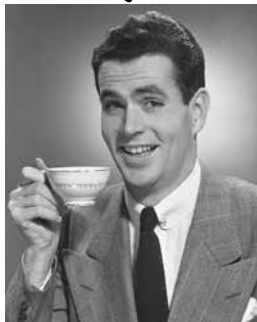


Dekorisanje objekata

3

Samo nazovite ovo poglavlje „Dizajnersko oko za naslednog tipa“. Preispitaćemo tipično preterano korišćenje nasleđivanja i saznaćete kako da dekorišete klase za vreme izvršavanja, koristeći jedan vid kompozicije objekata. Zašto? Kada upoznate tehnike dekorisanja, moći ćete da date svojim (ili tuđim) objektima nova zaduženja, *nepraveći nikakve izmene koda osnovnih klasa*.

Nekada sam mislio da pravi muškarcu prave potklase svega. Bilo je to dok nisam upoznao moć proširivanja za vreme izvršavanja, umesto za vreme kompajliranja. I vidite me sad!



Dobro došli na Starbuzz kafu	80
Princip otvorenosti i zatvorenosti	86
Upoznajte obrazac Decorator	88
Konstruisanje porudžbine pomoću dekoratora	89
Definicija obrasca Decorator	91
Dekorisanje napitaka	92
Pisanje koda za Starbuzz	95
Kodiranje napitaka	96
Kodiranje dodataka	97
Posluživanje kafe	98
Stvarni dekoratori: Java I/O	100
Dekorisanje java.io klasa	101
Pisanje Java I/O dekoratora	102
Testirajte svoj prvi Java I/O dekorator	103
Alatke za projektovanje	105

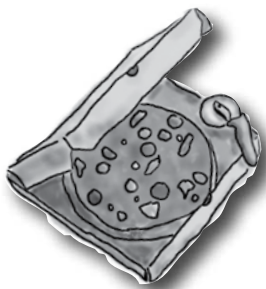
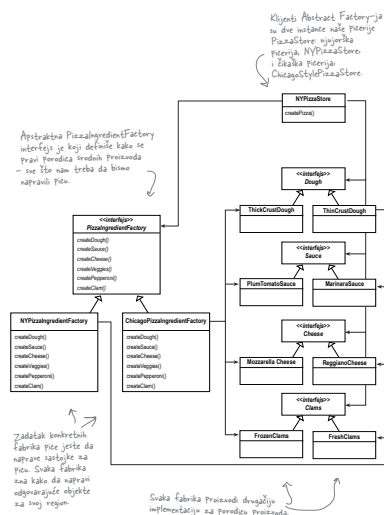
obrazac Factory

Dobrobiti OO dizajna

4

Spremite za pravljenje slabo povezanih OO projekata.

Pravljenje objekata ne svodi se samo na korišćenje operatora **new**. Naučićete da je instanciranje aktivnost koju ne bi uvek trebalo da radite u javnosti i da ono često može voditi do *problema sa povezivanjem*. A to ne želimo, zar ne? Saznajte kako obrasci Factory mogu da vas poštede sramotnih zavisnosti.

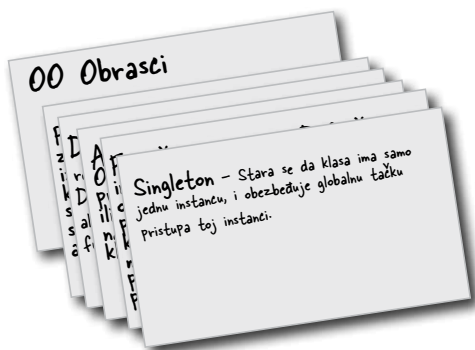


Prepoznavanje aspekata koji variraju	112
Kapsuliranje pravljenja objekata	114
Pravljenje proste fabrike pica	115
Definicija Simple Factory-ja	117
Radni okvir za piceriju	120
Dopuštanje potklasa da odlučuju	121
Deklarisanje fabričke metode	125
Došlo je vreme da upoznate obrazac Factory Method	131
Paralelan pregled kreatora i proizvoda	132
Definicija obrasca Factory Method	134
Razmatranje zavisnosti objekata	138
Princip inverzije zavisnosti	139
Primenjivanje principa	140
Porodice sastojaka...	145
Izgradnja fabrika sastojaka	146
Prepravljanje pica...	149
Ponovna poseta picerijama	152
Šta smo uradili?	153
Definicija obrasca Abstract Factory	156
Poređenje Factory Method i Abstract Factory	160
Alatke za projektovanje	162

5 Jedinstveni objekti

Naša sledeća stanica je obrazac Singleton, naša karta za pravljenje jedinstvenih objekata za koje će uvek postojati samo jedna instanca. Možda će vas obradovati da znate da je od svih obrazaca, obrazac Singleton (eng. *Singleton Pattern*) najjednostavniji po pitanju dijagrama klasa; štaviše, dijagram ima samo jednu klasu! Ali nemojte se previše opuštati; uprkos jednostavnosti iz perspektive dizajna klasa, on zahteva ozbiljno objektno orijentisano razmišljanje za svoju implementaciju. Zato, udahnite duboko, pa da krenemo.

Seciranje klasične implementacije obrasca Singleton	173
Fabrika čokolade	175
Definicija obrasca Singleton	177
Hershey, PA Hjstone , imamo problem...	178
Višenitni rad	180
Možemo li da popravimo višenitni rad?	181
U međuvremenu, u fabrici čokolade...	183
Alatke za projektovanje	186

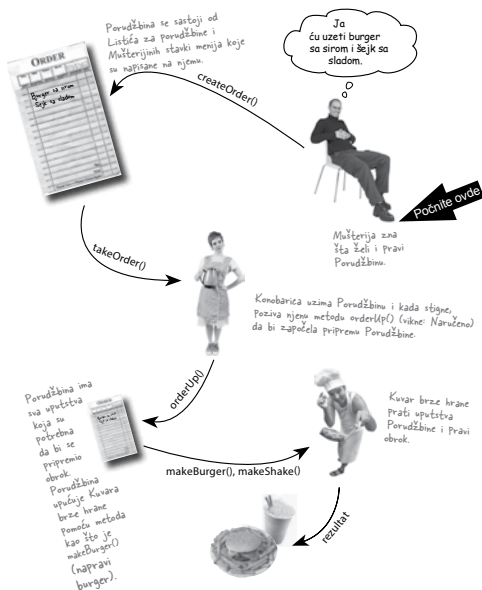


obrazac Command

6

Kapsuliranje poziva

U ovom poglavlju podižemo kapsuliranje na sasvim nov nivo: kapsuliraćemo pozivanje metode. Baš tako – kapsuliranjem pozivanja metode, možemo da iskristališemo delove izračunavanja, pa objekat koji poziva izračunavanje ne mora da brine o tome kako da nešto uradi, već prosto koristi našu iskristalisanu metodu da bi to obavio. Sa kapsuliranim pozivima metoda možemo da uradimo i neke lukave stvari, na primer da ih sačuvamo za evidentiranje ili da ih iznova koristimo da bismo u kodu implementirali funkcionalnost poništavanja akcije.



Automatizacija kuće ili propast	192
Pogled na klase dobavljača	194
Kratak uvod u obrazac Command	197
Od restorana do obrasca Command	201
Naš prvi komandni objekat	203
Korišćenje komandnog objekta	204
Dodeljivanje komandi pozicijama	209
Implementiranje sistema daljinskog upravljanja	210
Implementiranje komandi	211
Provera rada sistema daljinskog upravljanja	212
Vreme je da se napiše dokumentacija...	215
Šta radimo?	217
Vreme je za proveru kvaliteta tastera Undo	220
Korišćenje stanja za implementiranje Undo	221
Dodavanje Undo komandama plafonskog ventilatora	222
Svakom daljinskom treba režim Party!	225
Korišćenje makro komande	226
Još namena obrasca Command: zahtevi na čekanju	229
Još namena obrasca Command: evidencija zahteva	230
Obrazac Command u stvarnosti	231
Alatke za projektovanje	233

7

Biti adaptivan

U ovom poglavlju ćemo pokušati neverovatne poduhvate kao što je umetanje četvrtastog klina u oblu rupu. Zvuči nemoguće? Ne kada imamo projektne obrasce. Sećate se obrasca Decorator? **Omotavali smo objekte** da bismo im davali nova zaduženja. Sada ćemo omotati objekte sa drugačijom namerom: da bi njihovi interfejsi izgledali kao nešto drugo. Zašto bismo to radili? Da bismo dizajn koji očekuje jedan interfejs prilagodili klasi koja implementira drugačiji interfejs. I to nije sve; kad smo već kod toga, pogledaćemo još jedan obrazac koji omotava objekte da bi pojednostavio njihov interfejs.

Britanska utičenica



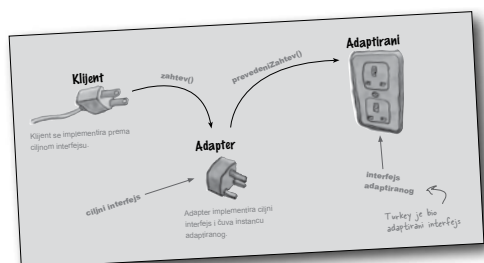
Adapter za naizmeničnu struju



Standardni američki utikač



Adapteri su svuda oko nas	238
Objektno orijentisani adapteri	239
Ako hoda kao patka i kvače kao patka, onda to mora može biti patka čurka umotana u pačji adapter...	240
Proba adaptera	242
Objašnjenje obrasca Adapter	243
Definicija obrasca Adapter	245
Adapteri objekta i klase	246
Adapteri u stvarnom svetu	250
Adaptiranje enumeracije na iterator	251
„Dome, slatki dome“ bioskop	257
Gledanje filma (na teži način)	258
Svetla, kamera, Facade!	260
Konstruisanje fasade kućnog bioskopa	263
Implementiranje pojednostavljenog interfejsa	264
Vreme je za gledanje filma (na lakši način)	265
Definicija obrasca Facade	266
Princip najmanjeg znanja	267
Kako NE steći prijatelje i uticati na objekte	268
Obrazac Facade i princip najmanjeg znanja	271
Alatke za projektovanje	272



obrazac Template Method

8

Kapsuliranje algoritama

Zahuktali smo se sa kapsuliranjem; kapsulirali smo pravljenje objekata, pozivanje metoda, složene interfejsse, patke, pice... šta bi moglo da bude sledeće?

Bacićemo se na kapsuliranje delova algoritama, tako da klase mogu da se prikače na izračunavanje kad god poželeva. Učičemo i o principu projektovanja koji je inspirisan Holivudom. Da počnemo...



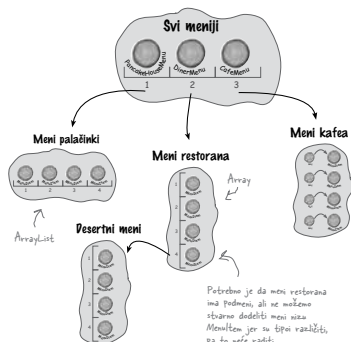
Vreme je za još kofeina	278
Mučkanje kafe i čaja (u Javi)	279
Apstrahujmo Coffee i Tea	282
Dalje doterivanje projekta...	283
Apstrahovanje prepareRecipe()	284
Šta smo uradili?	287
Upoznajte Template Method	288
Šta nam je doneo Template Method?	290
Definicija obrasca Template Method	291
Priključeni na šablonsku metodu...	294
Korišćenje priključaka	295
Holivudski princip i Template Method	299
Šablonske metode u divljini	301
Sortiranje pomoću Template Method	302
Treba da sortiramo neke patke...	303
Šta je compareTo()?	303
Poređenje pataka i pataka	304
Hajde da sortiramo patke	305
Izrada mašine za sortiranje pataka	306
Razmahivanje sa okvirima	308
Namenske liste pomoću AbstractList	309
Alatke za projektovanje	313

Kolekcije kojima se dobro upravlja

9

Ima mnogo načina da se objekti naguraju u kolekciju.

Smestite ih u bilo koji niz, stek, listu, heš mapu – izaberite sami. Svaki način ima svoje prednosti i nedostatke. Ali u nekom trenutku, vaši klijenti će želeti da prave iteracije nad tim objektima, a kada se to desi, da li ćete im otkriti svoju implementaciju? Iskreno se nadamo da nećete! To prosto ne bi bilo profesionalno. Pa, ne morate da rizikujete svoju karijeru; u ovom poglavlju ćete videti kako možete klijentima dozvoliti da prolaze kroz objekte, a da nikada ne vide na koji način skladištite objekte. Naučićete i kako da napravite vrhunske kolekcije objekata koje u jednom skoku mogu da preskoče impresivne strukture podataka. A ako to nije dovoljno, naučićete i ponešto o odgovornostima objekata.

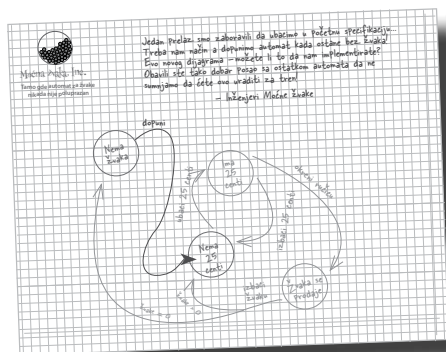


Najnovija vest: spajanje restorana Objektgrad i palačinkarnice Objektgrad	318
Pogledajte stavke menija	319
Implementiranje specifikacija: prvi pokušaj	323
Možemo li da kapsuliramo iteraciju?	325
Upoznajte obrazac Iterator	327
Dodavanje iteratora u DinerMenu	328
Prepravljanje menija restorana pomoću Iterator-a	329
Popravljanje koda Waitress	330
Testiranje koda	331
Analiza trenutnog dizajna...	333
Raščišćavanje pomoću java.util.Iterator	335
Definicija obrasca Iterator	338
Struktura obrasca Iterator	339
Princip jedne odgovornosti	340
Upoznajte Javin interfejs Iterable	343
Javina poboljšana petlja for	344
Pogled na meni kafea	347
Iteratori i kolekcije	353
Da li je konobarica spremna za udarni termin?	355
Definicija obrasca Composite	360
Dizajniranje menija sa Composite-om	363
Implementacija MenuComponent	364
Implementacija MenuItem	365
Implementiranje kompozitne klase Menu	366
A sad testiranje...	369
Alatke za projektovanje	376

Stanje stvari

Manje poznata činjenica: obrasci Strategy i State su blizanci razdvojeni na rođenju.

Pomislili biste da će oni živeti sličnim životima, ali obrazac Strategy je nakon toga stvorio veoma uspešan biznis širom međusobno zamenjivih algoritama, dok je obrazac State pošao, možda, plemenitijim putem pomaganja objektima da upravljaju svojim ponašanjem putem menjanja svog unutrašnjeg stanja. Ipak, koliko god da su njihovi putevi postali različiti, u pozadini ćete pronaći gotovo potpuno isti dizajn. Kako je to moguće? Kao što ćete videti, strategija i stanje imaju veoma različite svrhe. Pozabavimo se prvo onim o čemu se radi u obrascu State, a njihov odnos ćemo istražiti na kraju poglavlja.



Java žvaka	382
Osnove mašina stanja	384
Pisanje koda	386
Provera pre isporuke	388
Znali ste da će se desiti... promena zahteva!	390
Zbrkano STANJE stvari...	392
Novi dizajn	394
Definisanje interfejsa i klasa State	395
Prepravljanje automata za žvake	398
Pogledajmo sada celu klasu GumballMachine...	399
Implementiranje još stanja	400
Definicija obrasca State	406
Treba da završimo igru 1 od 10	409
Dovršavanje igre	410
Demo za direktora Moćne žvake, Inc.	411
Zdravorazumska provera...	413
Zamalo da zaboravimo!	416
Alatke za projektovanje	419



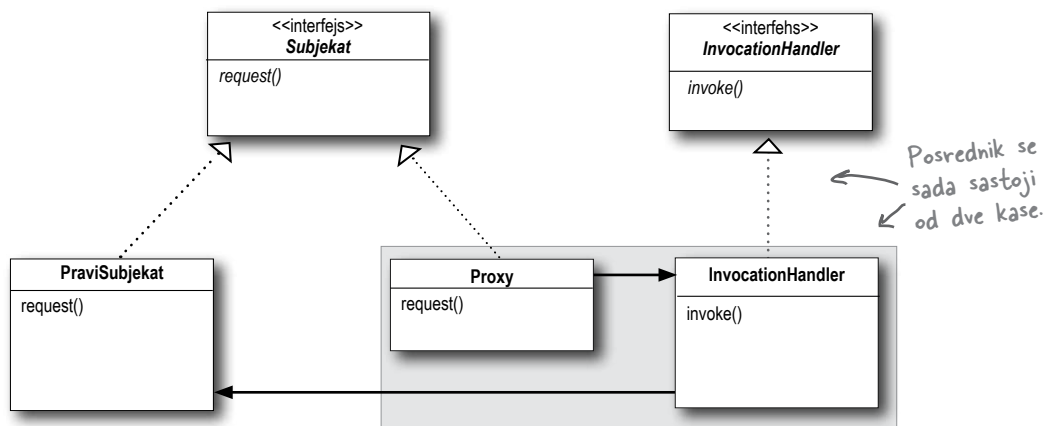
11

Upravljanje pristupom objektima

Da li ste se ikada igrali dobrog i lošeg policajca? Vi ste dobar policajac i pružate sve svoje usluge na fin i prijateljski način, ali ne želite da vam svako traži usluge, pa imate lošeg policajca koji kontroliše koliko ćete biti dostupni. To je ono što rade posrednici: kontrolišu i upravljaju pristupom. Videćete da, postoji mnogo načina na koje posrednici zamenjuju objekte kojima posreduju. Posrednici su poznati po tome da za objekte kojima posreduju mogu da preuzmu čitave pozive za metode preko interneta; i da strpljivo odmenjuju neke prilično lenje objekte.



Kodiranje monitora	427
Testiranje monitora	428
Osnove udaljenih metoda	433
Pripremanje GumballMachine da bude udaljeni servis	446
Registrowanje u registru RMI...	448
Definicija obrasca Proxy	455
Spremite se za virtuelnog posrednika	457
Projektovanje virtuelnog posrednika za korice albuma	459
Pisanje klase ImageProxy	460
Korišćenje Proxyja Java API-ja za pravljenje zaštitnog posrednika	469
Upoznavanje u Objektgradu	470
Implementacija interfejsa Person	471
Petominutna drama: zaštita subjekata	473
Šira slika: pravljenje dinamičkog posrednika za Person	474
Zoološki vrt posrednika	482
Alatke za projektovanje	485
Kod za prikazivač korica albuma	489



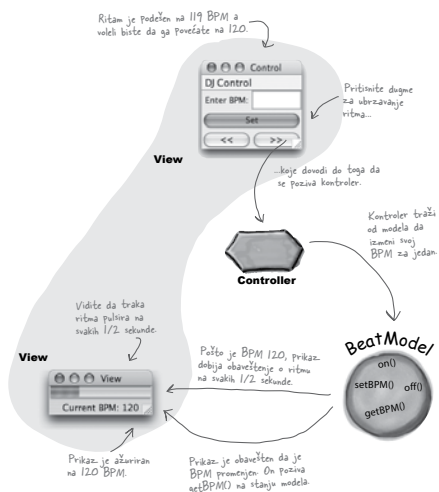
kombinovani obrasci

12

Obrasci obrazaca

Ko bi ikad pomislio da bi projektni obrasci mogli da rade zajedno? Već ste bili svedoci oštrog Časkanja uz vatricu (a čak niste ni videli

rubrike Borbe obrazaca na život i smrt, jer nas je urednik primorao da ih uklonimo iz knjige), pa ko bi mogao pomisliti da obrasci zapravo mogu dobro da rade jedan s drugim? Verovali ili ne, neki od najmoćnijih OO projekata koriste istovremeno više obrazaca. Spremite se da svoje veštine korišćenja obrazaca podignete na naredni nivo; vreme je za kombinovane obrasce.



Udruženi rad	494
Ponovno okupljanje pataka	495
Šta smo uradili?	517
Pogled iz pačje perspektive: dijagram klasa	518
Kralj kombinovanih obrazaca	520
Upoznajte Model-View-Controller	523
Detaljniji pogled...	524
Razumevanje MVC-a kao skupa obrazaca	526
Upotreba MVC-a za upravljanje ritmom...	528
Izgradnja delova	531
Pogledajmo sada konkretnu klasu BeatModel	532
Prikaz	533
Implementiranje prikaza	534
A sada kontroler	536
Sklapanje svega...	538
Istraživanje strategije	539
Prilagođavanje modela	540
A sada proba...	542
Alatke za projektovanje	545

bolji život sa obrascima

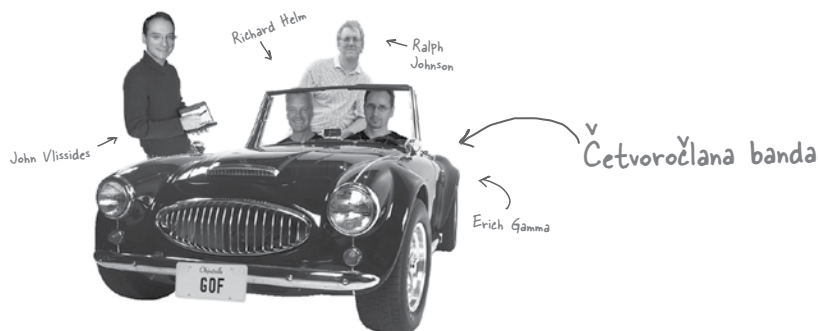
13

Obrasci u stvarnosti

Ahhhhh, sada ste spremni za sjajan novi svet ispunjen projektnim obrascima. Ali, pre nego što počnete da otvarate sva ta vrata mogućnosti, treba da se pozabavimo još nekim pojedinostima na koje ćete nailaziti u stvarnom svetu – tako je, stvari su nešto složenije nego što su ovde, u Objektgradu. Idemo, na narednoj strani imamo fini vodič koji će vam pomoći sa prelaskom...

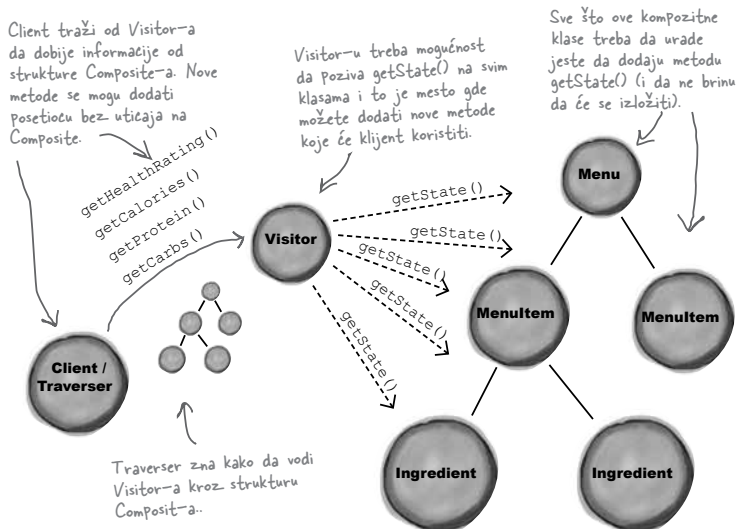


Definicija projektnog obrasca	565
Pažljivije razmatranje definicije projektnog obrasca	567
Neka sila bude sa vama	568
Želiš li biti pisac projektnih obrazaca	573
Organizovanje projektnih obrazaca	575
Razmišljanje u obrascima	580
Vaš um o obrascima	583
Zen um	583
Ne zaboravite moć zajedničkog rečnika	585
Krstarenje Objektgradom sa Četvoročlanom bandom	587
Vaše putovanje je tek počelo...	588
Zoološki vrt obrazaca	590
Uništavanje zla pomoću anti-obrazaca	592
Alatke za projektovanje	594
Napuštamo Objektgrad...	595



14 Dodatak: preostali obrasci

Ne mogu svi biti najpopularniji. Mnogo toga se promenilo u poslednjih 25 i više godina. Otkako je knjiga *Design Patterns: Elements of Reusable Object-Oriented Software* prvi put izdata, programeri su primenili ove obrasce hiljadama puta. Obrasci čiji rezime dajemo u ovom dodatku kompletni su, punopravni i zvanični GoF obrasci, ali se ne koriste toliko često kao obrasci koje smo do sada istraživali. Međutim, ovi obrasci su sami po sebi zadivljujući i ako ih vaše okolnosti zahtevaju, trebalo bi da ih s ponosom primenite. Naš cilj u ovom dodatku je da vam na visokom nivou predstavimo šta su ti obrasci.



Bridge	598
Builder	600
Chain of Responsibility	602
Flyweight	604
Interpreter	606
Mediator	608
Memento	610
Prototype	612
Visitor	614