

Spis treści

O autorze	15
O recenzencie technicznym	16
Podziękowania	17
Wprowadzenie	18

Część I. Obiekty 19

Rozdział 1. PHP — projektowanie	21
Problem	21
PHP a inne języki programowania	23
O książce	25
Obiekty	25
Wzorce	26
Narzędzia	26
Nowości w szóstym wydaniu	27
Podsumowanie	28

Rozdział 2. PHP i obiekty	29
Nieoczekiwany sukces obiektów w PHP	29
PHP/FI — u zarania języka	29
PHP3 — składniowy lukier	30
Cicha rewolucja — PHP4	30
PHP5 — nieuchronne zmiany	32
PHP7 — doganianie reszty świata	33
PHP8 — dalsza konsolidacja	33
Debata obiektowa — za czy przeciw?	33
Podsumowanie	34

Rozdział 3.	Obiektowy elementarz	35
	Klasy i obiekty	35
	Pierwsza klasa	35
	Pierwszy obiekt (lub dwa)	36
	Definiowanie składowych klasy	37
	Metody	39
	Metoda konstrukcji obiektu	41
	Promocja składowych konstruktora	43
	Argumenty domyślne i nazwane	44
	Typy argumentów metod	45
	Typy elementarne	45
	Niektóre inne funkcje kontroli typów	49
	Deklaracje typów — typy obiektowe	49
	Deklaracje typów — typy elementarne	51
	Typ mixed	53
	Unia	55
	Typy przyjmujące wartość null	56
	Deklaracje typów zwracanych	57
	Dziedziczenie	57
	Problemy, które można rozwiązać poprzez dziedziczenie	58
	Stosowanie dziedziczenia	62
	Zarządzanie dostępem do klasy — słowa public, private i protected	68
	Podsumowanie	74
Rozdział 4.	Zaawansowana obsługa obiektów	75
	Metody i składowe statyczne	75
	Składowe stałe	79
	Klasy abstrakcyjne	80
	Interfejsy	82
	Cechy typowe	84
	Zadanie dla cech typowych	85
	Definiowanie i stosowanie cechy typowej	86
	Stosowanie wielu cech typowych	87
	Łączenie cech z interfejsami	88
	Unikanie kolizji nazw metod za pomocą słowa insteadof	88
	Aliasy przesłoniętych metod cech typowych	89
	Cechy typowe z metodami statycznymi	90
	Dostęp do składowych klasy włączającej	91
	Definiowanie metody abstrakcyjnej cechy typowej	92
	Zmiana dostępności metod cech typowych	93
	Późne wiązanie statyczne: słowo static	93
	Obsługa błędów	97
	Wyjątki	99
	Klasy i metody finalne	106
	Klasa do obsługi błędów wewnętrznych	107
	Przechwytywanie chybionych wywołań	108

Definiowanie destruktorów	115
Wykonywanie kopii obiektów	116
Reprezentacja obiektu w ciągach znaków	119
Wywołania zwrotne, funkcje anonimowe i domknięcia	120
Klasy anonimowe	126
Podsumowanie	127

Rozdział 5. Narzędzia obiektowe128

PHP a pakiety	128
Pakiety i przestrzenie nazw w PHP	129
Automatyczne wczytywanie kodu	138
Klasy i funkcje pomocnicze	142
Szukanie klasy	143
Badanie obiektów i klas	144
Pozyskiwanie ciągu pełnej nazwy klasy	145
Badanie metod	146
Badanie składowych	148
Badanie relacji dziedziczenia	149
Badanie wywołań metod	149
Reflection API	151
Zaczynamy	151
Pora zakasać rękawy	152
Badanie klasy	154
Badanie metod	156
Badanie argumentów metod	157
Zastosowanie Reflection API	159
Atrybuty	163
Podsumowanie	167

Rozdział 6. Obiekty168

Czym jest projektowanie?	168
Programowanie obiektowe i proceduralne	169
Odpowiedzialność	173
Spójność	173
Sprzęganie	174
Ortogonalność	174
Zasięg klas	175
Polimorfizm	176
Hermetyzacja	177
Nieważne jak	178
Cztery drogowskazy	179
Zwielokrotnianie kodu	180
Przemądrzałe klasy	180
Złota rączka	180
Za dużo warunków	180

Język UML	181
Diagramy klas	181
Diagramy sekwencji	187
Podsumowanie	189
Część II. Wzorce	191
Rozdział 7. Czym są wzorce projektowe?	193
Czym są wzorce projektowe?	193
Wzorzec projektowy	196
Nazwa	196
Problem	196
Rozwiązanie	196
Konsekwencje	197
Format wzorca według Bandy Czworga	197
Po co nam wzorce projektowe?	198
Wzorzec projektowy definiuje problem	198
Wzorzec projektowy definiuje rozwiązanie	198
Wzorce projektowe są niezależne od języka programowania	198
Wzorce definiują słownictwo	199
Wzorce są wypróbowane	199
Wzorce mają współpracować	200
Wzorce promują zasady projektowe	200
Wzorce są stosowane w popularnych frameworkach	200
Wzorce projektowe a PHP	200
Podsumowanie	201
Rozdział 8. Wybrane zasady wzorców	202
Olsnienie wzorcami	202
Kompozycja i dziedziczenie	203
Problem	203
Zastosowanie kompozycji	206
Rozprzęganie	209
Problem	209
Osłabianie sprzężenia	210
Kod ma używać interfejsów, nie implementacji	212
Zmienne koncepcje	214
Nadmiar wzorców	214
Wzorce	215
Wzorce generowania obiektów	215
Wzorce organizacji obiektów i klas	215
Wzorce zadaniowe	215
Wzorce korporacyjne	216
Wzorce baz danych	216
Podsumowanie	216

Rozdział 9. Generowanie obiektów	217
Generowanie obiektów — problemy i rozwiązania	217
Wzorzec Singleton	221
Problem	222
Implementacja	223
Konsekwencje	224
Wzorzec Factory Method	225
Problem	225
Implementacja	228
Konsekwencje	230
Wzorzec Abstract Factory	231
Problem	231
Implementacja	232
Konsekwencje	233
Prototyp	235
Problem	236
Implementacja	236
Naginanie rzeczywistości — wzorzec Service Locator	240
Doskonała izolacja — wstrzykiwanie zależności	241
Problem	242
Implementacja	242
Konsekwencje	255
Podsumowanie	256
 Rozdział 10. Wzorce elastycznego programowania obiektowego	 257
Strukturalizacja klas pod kątem elastyczności obiektów	257
Wzorzec Composite	258
Problem	258
Implementacja	261
Konsekwencje	264
Composite — podsumowanie	268
Wzorzec Decorator	268
Problem	269
Implementacja	271
Konsekwencje	275
Wzorzec Facade	276
Problem	276
Implementacja	278
Konsekwencje	278
Podsumowanie	279
 Rozdział 11. Reprezentacja i realizacja zadań	 280
Wzorzec Interpreter	280
Problem	280
Implementacja	282
Ciemne strony wzorca Interpreter	289

Wzorzec Strategy	290
Problem	290
Implementacja	291
Wzorzec Observer	295
Implementacja	297
Wzorzec Visitor	303
Problem	303
Implementacja	305
Wady wzorca Visitor	309
Wzorzec Command	310
Problem	310
Implementacja	311
Wzorzec Null Object	315
Problem	316
Implementacja	318
Podsumowanie	319
Rozdział 12. Wzorce korporacyjne	320
Przegląd architektury	320
Wzorce	321
Aplikacje i warstwy	321
Małe oszustwo na samym początku	324
Wzorzec Registry	324
Implementacja	326
Warstwa prezentacji	330
Wzorzec Front Controller	330
Wzorzec Application Controller	341
Wzorzec Page Controller	355
Wzorce Template View i View Helper	360
Warstwa logiki biznesowej	363
Wzorzec Transaction Script	363
Wzorzec Domain Model	368
Podsumowanie	372
Rozdział 13. Wzorce bazodanowe	373
Warstwa danych	373
Wzorzec Data Mapper	374
Problem	374
Implementacja	374
Konsekwencje	390
Wzorzec Identity Map	392
Problem	392
Implementacja	393
Konsekwencje	396

Wzorzec Unit of Work	397
Problem	397
Implementacja	397
Konsekwencje	402
Wzorzec Lazy Load	402
Problem	402
Implementacja	403
Konsekwencje	405
Wzorzec Domain Object Factory	405
Problem	406
Implementacja	406
Konsekwencje	407
Wzorzec Identity Object	408
Problem	409
Implementacja	409
Konsekwencje	416
Wzorce Selection Factory i Update Factory	416
Problem	416
Implementacja	416
Konsekwencje	421
Co zostało z wzorca Data Mapper?	421
Podsumowanie	424

Część III. Narzędzia 425

Rozdział 14. Dobre (i złe) praktyki	427
Nie tylko kod	427
Pukanie do otwartych drzwi	428
Jak to zgrać?	430
Uskrzydlenie kodu	431
Standardy	432
Vagrant	433
Testowanie	433
Ciągła integracja	434
Podsumowanie	435

Rozdział 15. Standardy PHP	436
Po co te standardy?	436
Jakie są standardowe rekomendacje PHP?	437
Dlaczego akurat PSR?	438
Rekomendacje PSR — dla kogo?	439
Kodowanie z klasą	439
Podstawowy standard kodowania PSR-1	440
Rozszerzona rekomendacja stylu kodowania PSR-12	442
Sprawdzanie i poprawianie kodu	448

PSR-4 — automatyczne ładowanie	450
Zasady, które są dla nas ważne	450
Podsumowanie	453
Rozdział 16. Używanie i tworzenie	454
Czym jest Composer?	454
Instalowanie Composera	455
Instalowanie (zbioru) pakietów	455
Instalowanie pakietu z poziomu wiersza poleceń	456
Wersje	457
Element require-dev	458
Composer i automatyczne ładowanie kodu	459
Tworzenie własnego pakietu	460
Dodawanie informacji o pakiecie	460
Pakiety systemowe	461
Dystrybucja za pośrednictwem repozytorium Packagist	462
Odrobina prywatności	465
Podsumowanie	466
Rozdział 17. Zarządzanie wersjami projektu	467
Po co mi kontrola wersji?	467
Skąd wziąć klienta Git?	469
Obsługa repozytorium Git online	469
Konfigurowanie serwera Git	471
Tworzenie repozytorium zdalnego	472
Rozpoczynamy projekt	474
Klonowanie repozytorium	477
Wprowadzanie i zatwierdzanie zmian	477
Dodawanie i usuwanie plików i katalogów	481
Dodawanie pliku	481
Usuwanie pliku	481
Dodawanie katalogu	482
Usuwanie katalogów	482
Etykietowanie wersji	482
Rozgałęzianie projektu	484
Podsumowanie	490
Rozdział 18. Testy jednostkowe z PHPUnit	491
Testy funkcjonalne i testy jednostkowe	491
Testowanie ręczne	492
PHPUnit	495
Tworzenie przypadku testowego	495
Metody asercji	497
Testowanie wyjątków	498
Uruchamianie zestawów testów	499
Ograniczenia	500

Atrapy i imitacje	502
Dobry test to obłany test	505
Testy dla aplikacji WWW	509
Refaktoryzacja aplikacji WWW pod kątem testów	509
Proste testy aplikacji WWW	511
Selenium	513
Słowo ostrzeżenia	519
Podsumowanie	521

Rozdział 19. Automatyzacja instalacji z Phingiem522

Czym jest Phing?	523
Pobieranie i instalacja pakietu Phing	523
Montowanie dokumentu kompilacji	524
Różnicowanie zadań kompilacji	526
Właściwości	528
Typy	535
Operacje	540
Podsumowanie	544

Rozdział 20. Vagrant546

Problem	546
Odrobina przygotowań	547
Wybór i instalacja środowiska Vagrant	547
Montowanie lokalnych katalogów w maszynie wirtualnej Vagranta	550
Zaopatrywanie maszyny wirtualnej	551
Konfigurowanie serwera WWW	552
Konfigurowanie MariaDB	553
Konfigurowanie nazwy hosta	554
Kilka słów na koniec	555
Podsumowanie	556

Rozdział 21. Ciągła integracja kodu557

Czym jest ciągła integracja?	557
Przygotowanie projektu do ciągłej integracji	559
Instalowanie rozszerzeń Jenkinsa	569
Konfigurowanie klucza publicznego serwera Git	570
Instalowanie projektu	571
Pierwsza kompilacja	574
Konfigurowanie raportów	574
Automatyzacja kompilacji	577
Podsumowanie	579

Rozdział 22. Obiekty, wzorce, narzędzia580

Obiekty	580
Wybór	581
Hermetyzacja i delegowanie	581
Osłabianie sprzężenia	582

Zdatność do wielokrotnego stosowania kodu	582
Estetyka	583
Wzorce	583
Co dają nam wzorce?	584
Wzorce a zasady projektowe	585
Narzędzia	587
Testowanie	587
Standardy	588
Zarządzanie wersjami	588
Automatyczna kompilacja (instalacja)	588
System integracji ciąglej	589
Co pominęliśmy?	589
Podsumowanie	591

Dodatki 593

Dodatek A Bibliografia	595
Książki	595
Publikacje	596
Witryny WWW	596
Dodatek B Prosty analizator leksykalny	599
Skaner	599
Analizator leksykalny	607