

Содержание

Об авторе	15
О техническом рецензенте	17
Благодарности	17
Введение	19
От издательства	20
Часть I. Введение	21
Глава 1. Проектирование и сопровождение приложений на PHP	23
Проблема	23
PHP и другие языки	25
Об этой книге	28
Объекты	29
Шаблоны	29
Практика	30
Что нового в пятом издании книги	32
Резюме	32
Часть II. Объекты	35
Глава 2. PHP и объекты	37
Неожиданный успех объектов в PHP	37
Вначале был PHP/FI	37
Синтаксические удобства в версии PHP 3	38
Версия PHP 4 и незаметная революция	38
Изменения приняты: PHP 5	41
Восполнение пробела: PHP 7	41
Дебаты сторонников и противников объектов	42
Резюме	43
Глава 3. Основные положения об объектах	45
Классы и объекты	45
Первый класс	45
Несколько первых объектов	46
Установка свойств в классе	47
Работа с методами	51
Создание метода-конструктора	52
Аргументы и типы	54
Простые типы данных	54
Уточнение типов объектов	59
Наследование	64
Проблема наследования	65
Использование наследования	70
Управление доступом к классам: модификаторы <code>public</code> , <code>private</code> и <code>protected</code>	77
Методы доступа	79
Семейство классов <code>ShopProduct</code>	80
Резюме	83

Глава 4. Расширенные средства	85
Статические методы и свойства	86
Постоянные свойства	90
Абстрактные классы	91
Интерфейсы	93
Трейты	96
Проблема, которую позволяют решить трейты	96
Определение и применение трейтов	98
Применение нескольких трейтов	99
Сочетание трейтов с интерфейсами	100
Устранение конфликтов имен методов с помощью ключевого слова <code>insteadof</code>	101
Назначение псевдонимов переопределенным методам трейта	102
Применение статических методов в трейтах	103
Доступ к свойствам базового класса	104
Определение абстрактных методов в трейтах	105
Изменение прав доступа к методам трейта	106
Позднее статическое связывание: ключевое слово <code>static</code>	107
Обработка ошибок	112
Исключения	114
Завершенные классы и методы	123
Внутренний класс <code>Error</code>	125
Работа с методами-перехватчиками	126
Определение методов-деструкторов	134
Копирование объектов с помощью метода <code>__clone()</code>	135
Определение строковых значений для объектов	139
Функции обратного вызова, анонимные функции и механизм замыканий	140
Анонимные классы	147
Резюме	149
Глава 5. Средства для работы с объектами	151
RНР и пакеты	151
Пакеты и пространства имен в RНР	152
Пути включения файлов	160
Автозагрузка	163
Функции для исследования классов и объектов	167
Поиск классов	169
Получение сведений об объекте или классе	170
Получение полностью уточненной строковой ссылки на класс	171
Получение сведений о методах	172
Получение сведений о свойствах	174
Получение сведений о наследовании	174
Вызов методов	175
Интерфейс Reflection API	177
Краткое введение в Reflection API	177
Время засучить рукава	178
Исследование класса	181
Исследование методов	183
Исследование аргументов методов	185
Применение интерфейса Reflection API	187
Резюме	192

8 СОДЕРЖАНИЕ

Глава 6. Объекты и проектирование	193
Определение программного проекта	193
Объектно-ориентированное и процедурное программирование	194
Ответственность	199
Связность	200
Тесная связь	200
Ортогональность	201
Выбор классов	201
Полиморфизм	203
Инкапсуляция	205
Забудьте, как это делается	206
Четыре явных признака недоброкачественного кода	207
Дублирование кода	207
Класс, который слишком много знает	208
На все руки мастер	208
Условные операторы	208
Язык UML	209
Диаграммы классов	209
Диаграмма последовательностей	217
Резюме	219
Часть III. Шаблоны	221
Глава 7. Назначение и применение проектных шаблонов	223
Что такое проектные шаблоны	224
Краткий обзор проектных шаблонов	227
Название	227
Постановка задачи	227
Решение	228
Последствия	228
Формат “Банды четырех”	228
Причины для применения проектных шаблонов	229
Шаблоны определяют задачи	230
Шаблоны определяют решения	230
Шаблоны не зависят от языка программирования	230
Шаблоны определяют словарь	230
Шаблоны проверяются и тестируются	231
Шаблоны предназначены для совместной работы	231
Шаблоны способствуют удачным проектам	232
Шаблоны применяются в распространенных каркасах	232
РНР и проектные шаблоны	232
Резюме	233
Глава 8. Некоторые принципы действия шаблонов	235
Открытие шаблонов	235
Композиция и наследование	236
Проблема	236
Применение композиции	240
Развязка	243
Проблема	243
Ослабление связанности	245
Программируйте на основе интерфейса, а не его реализации	247

Меняющаяся концепция	249
Проблемы применения шаблонов	250
Шаблоны	250
Шаблоны для формирования объектов	251
Шаблоны для организации объектов и классов	251
Шаблоны, ориентированные на задачи	251
Промышленные шаблоны	251
Шаблоны баз данных	251
Резюме	251
Глава 9. Формирование объектов	253
Формирование объектов: задачи и решения	253
Шаблон Singleton	259
Проблема	260
Реализация	260
Выводы	263
Шаблон Factory Method	263
Проблема	263
Реализация	267
Выводы	269
Шаблон Abstract Factory	270
Проблема	270
Реализация	271
Выводы	274
Шаблон Prototype	276
Проблема	276
Реализация	277
Доведение до крайности: шаблон Service Locator	281
Блестящее одиночество: шаблон Dependency Injection	283
Проблема	283
Реализация	284
Выводы	288
Резюме	289
Глава 10. Шаблоны для программирования гибких объектов	291
Структурирование классов для повышения гибкости объектов	291
Шаблон Composite	292
Проблема	292
Реализация	295
Выводы	300
Краткие выводы по шаблону Composite	304
Шаблон Decorator	305
Проблема	305
Реализация	308
Выводы	313
Шаблон Facade	313
Проблема	314
Реализация	315
Выводы	316
Резюме	317
Глава 11. Выполнение задач и представление результатов	319
Шаблон Interpreter	319
Проблема	320

10 СОДЕРЖАНИЕ

Реализация	321
Трудности реализации шаблона Interpreter	331
Шаблон Strategy	332
Проблема	332
Реализация	333
Шаблон Observer	337
Реализация	340
Шаблон Visitor	347
Проблема	347
Реализация	349
Трудности реализации шаблона Visitor	355
Шаблон Command	355
Проблема	355
Реализация	356
Шаблон Null Object	361
Проблема	362
Реализация	365
Резюме	366
Глава 12. Шаблоны корпоративных приложений	369
Краткий обзор архитектуры	370
Шаблоны	370
Приложения и уровни	371
Нарушение правил с самого начала	374
Шаблон Registry	374
Уровень представления данных	381
Шаблон Front Controller	382
Шаблон Application Controller	395
Шаблон Page Controller	410
Шаблоны Template View и View Helper	417
Уровень логики приложения	421
Шаблон Transaction Script	421
Шаблон Domain Model	427
Резюме	431
Глава 13. Шаблоны баз данных	433
Уровень хранения данных	433
Шаблон Data Mapper	434
Проблема	434
Реализация	435
Результаты	452
Шаблон Identity Map	453
Проблема	453
Реализация	454
Результаты	458
Шаблон Unit of Work	458
Проблема	458
Реализация	459
Результаты	463
Шаблон Lazy Load	464
Проблема	464
Реализация	465
Результаты	467
Шаблон Domain Object Factory	467
Проблема	468

Реализация	468
Результаты	469
Шаблон Identity Object	471
Проблема	472
Реализация	472
Результаты	479
Шаблоны Selection Factory и Update Factory	479
Проблема	480
Реализация	480
Результаты	484
Что теперь осталось от шаблона Data Mapper?	485
Резюме	488
Часть IV. Практика	489
Глава 14. Нормы надлежащей и порочной практики	491
Не кодом единым	492
Снова изобретаем колесо	492
Ведите себя хорошо	495
Дайте коду крылья	496
Стандарты	497
Технология Vagrant	498
Тестирование	499
Непрерывная интеграция	500
Резюме	501
Глава 15. Стандарты PHP	503
Зачем нужны стандарты	503
Рекомендованные стандарты PHP	504
Особенности рекомендованных стандартов PSR	505
На кого рассчитаны рекомендации стандартов PSR	506
Программирование в избранном стиле	506
Основные рекомендации стандарта PSR-1 по стилю программирования	507
Рекомендации стандарта PSR-2 по стилю программирования	510
Проверка и исправление исходного кода	514
Рекомендации стандарта PSR-4 по автозагрузке	517
Самые важные правила	518
Резюме	521
Глава 16. Создание и использование компонентов PHP средствами Composer	523
Назначение Composer	524
Установка Composer	524
Установка пакетов	525
Установка пакетов из командной строки	526
Версии пакетов	527
Поле require-dev	528
Composer и автозагрузка	529
Создание собственного пакета	530
Ввод сведений о пакете	530
Пакеты для конкретной платформы	531
Распространение пакетов через сайт Packagist	532
Работа с закрытыми пакетами	535
Резюме	536

12 СОДЕРЖАНИЕ

Глава 17. Контроль версий средствами Git	537
Зачем нужен контроль версий	538
Установка Git	539
Использование онлайн-хранилища Git	540
Конфигурирование сервера Git	542
Создание удаленного хранилища	542
Подготовка хранилища для локальных пользователей	543
Предоставление доступа пользователям	543
Закрытие доступа к системной оболочке для пользователя git	544
Начало проекта	545
Клонирование хранилища	550
Обновление и фиксация изменений	550
Добавление и удаление файлов и каталогов	555
Добавление файла	555
Удаление файла	556
Добавление каталога	557
Удаление каталогов	558
Отметка выпуска	558
Разветвление проекта	559
Резюме	567
Глава 18. Тестирование средствами PHPUnit	569
Функциональные и модульные тесты	570
Тестирование вручную	571
Общее представление о PHPUnit	573
Создание контрольного примера	574
Методы с утверждениями	577
Тестирование исключений	578
Выполнение наборов тестов	579
Ограничения	580
Имитации и заглушки	583
Тесты достигают своей цели, когда завершаются неудачно	587
Написание веб-тестов	590
Реорганизация кода веб-приложения для тестирования	591
Простые веб-тесты	594
Общее представление о Selenium	596
Предупреждения относительно тестирования	603
Резюме	606
Глава 19. Автоматическое построение средствами Phing	607
Назначение Phing	608
Получение и установка Phing	609
Создание документа построения	610
Целевые задания	612
Свойства	615
Типы	623
Задачи	629
Резюме	634
Глава 20. Виртуальная машина Vagrant	635
Задача	635
Простейшая установка	637
Выбор и установка образа операционной системы в Vagrant	637

Монтирование локальных каталогов на виртуальной машине Vagrant	640
Подготовка	641
Настройка веб-сервера	643
Настройка сервера баз данных MySQL	644
Определение имени хоста	645
Краткие итоги	647
Резюме	648
Глава 21. Непрерывная интеграция	649
Что такое непрерывная интеграция	649
Подготовка проекта к непрерывной интеграции	652
Установка сервера Jenkins	664
Установка модулей, подключаемых к серверу Jenkins	666
Установка открытого ключа доступа к системе Git	666
Установка проекта	668
Первое построение проекта	672
Настройка отчетов	672
Автоматический запуск процессов построения проектов	674
Резюме	678
Часть V. Заключение	679
Глава 22. Объекты, шаблоны и практика	681
Объекты	681
Выбор	682
Инкапсуляция и делегирование	682
Развязка	683
Повторное использование кода	684
Эстетика	684
Проектные шаблоны	685
Преимущества шаблонов	686
Шаблоны и принципы проектирования	687
Практика	689
Тестирование	689
Стандарты	690
Контроль версий	690
Автоматическое построение	691
Непрерывная интеграция	691
Что упущено из виду	692
Резюме	694
Часть VI. Приложения	695
Приложение А. Дополнительные источники информации	697
Литература	697
Статьи в Интернете	699
Сайты	699
Приложение Б. Простой синтаксический анализатор	703
Сканер	703
Синтаксический анализатор	712
Предметный указатель	729