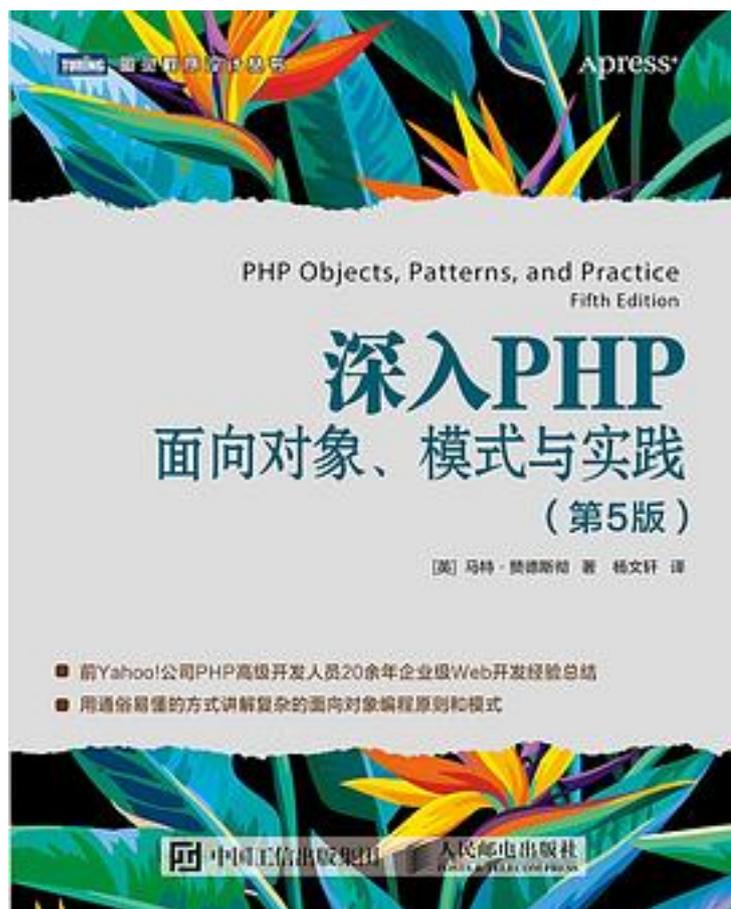


深入PHP：面向对象、模式与实践（第5版）



[深入PHP：面向对象、模式与实践（第5版）_下载链接1](#)

著者:[英] 马特·赞德斯彻

出版者:人民邮电出版社

出版时间:

装帧:平装

isbn:9787115512338

本书是PHP经典图书升级版，它既是一本关于面向对象设计与编程的书，也是一本关于如何使用工具管理PHP代码（从协作到部署）的书。书中讲解了PHP的新特性，例如匿名类以及标量参数提示和返回值类型。第5版重写了Composer和Packagist库的相关内容，并增加了关于Git版本控制的篇幅。示例代码全面更新，符合PSR-1和PSR-2标准。

阅读本书能够帮你构建实现既定目标且易于协同开发的系统，并让你的代码优雅、简洁且易于理解。

作者介绍:

作者简介:

马特·赞德斯彻 (Matt Zandstra)

20余年Web开发经验，曾担任Yahoo!公司高级开发人员，现在是自由顾问和作家，为Linux Magazine、Zend.com、IBM DeveloperWorks和php[architect] Magazine等撰写过文章。

译者简介:

杨文轩

华中科技大学硕士，擅长Web开发，有丰富的对日开发经验。现就职于日本方正股份有限公司。译作有《图解基础设施设计模式》《C++性能优化指南》《图解设计模式》等。

目录: 第一部分 对象

第1章 PHP：设计和管理 2

1.1 问题 2

1.2 PHP与其他语言 3

1.3 关于本书 5

1.3.1 对象 5

1.3.2 模式 6

1.3.3 实践 6

1.3.4 第5版新增内容 7

1.4 小结 8

第2章 PHP和对象 9

2.1 PHP对象的偶然成功 9

2.1.1 起源：PHP/FI 9

2.1.2 语法糖：PHP 3 9

2.1.3 一场静悄悄的革命：PHP 4 10

2.1.4 拥抱变化：PHP 5 11

2.1.5 迎头追赶：PHP 7 12

2.2 拥护和顾虑：关于对象的争辩 12

2.3 小结 13

第3章 对象基础 14

3.1 类和对象 14

3.1.1 第一个类 14

3.1.2 一个（或两个）对象 15

3.2 设置类中的属性 16

3.3 使用方法 18

3.4 参数和类型 21

3.4.1 基本类型 21

3.4.2 接受提示：对象类型 24

3.5 继承 28

3.5.1 继承问题 28

3.5.2 使用继承 33

3.5.3 public、private和protected：管理类的访问 38

3.6 小结 43

第4章 高级特性 44

4.1 静态方法和属性 44

4.2 常量属性 47

4.3 抽象类 48

4.4 接口 50

4.5 trait 52

4.5.1 trait可以解决的问题 52

4.5.2 定义和使用trait 53

4.5.3 使用多个trait 54

4.5.4 组合使用trait与接口 55

4.5.5 通过insteadof管理方法名冲突 56

4.5.6 使用别名重写trait的方法 57

4.5.7 在trait中使用静态方法 58

4.5.8 访问宿主类的属性 59

4.5.9 在trait中定义抽象方法 59

4.5.10 改变trait中方法的访问权限 60

4.6 延迟静态绑定：static关键字 61

4.7 错误处理 65

4.8 final类和方法 73

4.9 内部错误类 74

4.10 使用拦截器 75

4.11 定义析构方法 81

4.12 使用__clone()复制对象 82

4.13 定义对象的字符串值 85

4.14 回调、匿名函数和闭包 86

4.15 匿名类 90

4.16 小结 92

第5章 对象工具 93

5.1 PHP和包 93

5.1.1 PHP包和命名空间 93

5.1.2 自动加载 101

5.2 类函数和对象函数 105

5.2.1 查找类 106

5.2.2 检查对象或类 106

5.2.3 得到指向类的完全限定的字符串引用 107

5.2.4 检查方法 108

5.2.5 检查类属性 110

5.2.6 检查继承 110

5.2.7 方法调用 111

5.3 反射API 112

5.3.1 入门 112

5.3.2 是时候大干一场了 113

5.3.3 检查类 115

5.3.4 检查方法 117

5.3.5 检查方法参数 118

5.3.6 使用反射API 120

5.4 小结 123

第6章 对象与设计 124

6.1 定义代码设计 124

6.2 面向对象编程与面向过程编程 125

6.2.1 职责 129

6.2.2 内聚 129

6.2.3 耦合	129
6.2.4 正交	129
6.3 选择类	130
6.4 多态	131
6.5 封装	132
6.6 忘记细节	133
6.7 四个方向标	133
6.7.1 代码重复	134
6.7.2 类知道太多	134
6.7.3 万能的类	134
6.7.4 条件语句	134
6.8 UML	134
6.8.1 类图	135
6.8.2 序列图	140
6.9 小结	142
第二部分 模式	
第7章 什么是设计模式，为什么要使用设计模式	144
7.1 什么是设计模式	144
7.2 设计模式概要	146
7.2.1 名称	146
7.2.2 问题	146
7.2.3 解决方案	147
7.2.4 效果	147
7.3 《设计模式》的格式	147
7.4 为什么使用设计模式	148
7.4.1 设计模式定义了问题	148
7.4.2 设计模式定义了解决方案	148
7.4.3 设计模式与编程语言无关	148
7.4.4 模式定义了一组词汇	148
7.4.5 模式是经过测试的	149
7.4.6 模式为协作而设计	149
7.4.7 设计模式促进优秀设计	149
7.4.8 流行的框架都使用了设计模式	150
7.5 PHP与设计模式	150
7.6 小结	150
第8章 一些模式原则	151
8.1 模式的启示	151
8.2 组合与继承	152
8.2.1 问题	152
8.2.2 使用组合	155
8.3 解耦	157
8.3.1 问题	157
8.3.2 解耦	159
8.4 针对接口编程，而不是针对实现编程	161
8.5 概念在变化	162
8.6 不要盲从模式	162
8.7 模式	163
8.7.1 用于生成对象的模式	163
8.7.2 用于组织对象和类的模式	163
8.7.3 面向任务的模式	163
8.7.4 企业设计模式	163
8.7.5 数据库模式	163
8.8 小结	163
第9章 生成对象	164

9.1 生成对象的问题和解决方案	164
9.2 单例模式	168
9.2.1 问题	169
9.2.2 实现	169
9.2.3 效果	171
9.3 工厂方法模式	172
9.3.1 问题	172
9.3.2 实现	175
9.3.3 效果	177
9.4 抽象工厂模式	177
9.4.1 问题	177
9.4.2 实现	178
9.4.3 效果	180
9.5 原型模式	181
9.5.1 问题	182
9.5.2 实现	183
9.6 推向边缘：服务定位器	186
9.7 完全隔离：依赖注入	187
9.7.1 问题	187
9.7.2 实现	188
9.7.3 效果	191
9.8 小结	191
第10章 使面向对象编程更加灵活的模式	192
10.1 构造可灵活创建对象的类	192
10.2 组合模式	192
10.2.1 问题	193
10.2.2 实现	195
10.2.3 效果	199
10.2.4 组合模式小结	202
10.3 装饰器模式	202
10.3.1 问题	202
10.3.2 实现	205
10.3.3 效果	209
10.4 外观模式	209
10.4.1 问题	209
10.4.2 实现	211
10.4.3 效果	211
10.5 小结	212
第11章 执行及描述任务	213
11.1 解释器模式	213
11.1.1 问题	213
11.1.2 实现	214
11.1.3 解释器模式的问题	222
11.2 策略模式	222
11.2.1 问题	222
11.2.2 实现	223
11.3 观察者模式	227
11.4 访问者模式	235
11.4.1 问题	235
11.4.2 实现	236
11.4.3 访问者模式的问题	241
11.5 命令模式	242
11.5.1 问题	242
11.5.2 实现	242

- 11.6 空对象模式 247
 - 11.6.1 问题 247
 - 11.6.2 实现 249
- 11.7 小结 251
- 第12章 企业设计模式 252
 - 12.1 架构概述 252
 - 12.1.1 模式 252
 - 12.1.2 应用与分层 253
 - 12.2 企业架构外的基础模式 255
 - 12.2.1 注册表 255
 - 12.2.2 实现 256
 - 12.2.3 效果 260
 - 12.3 表示层 260
 - 12.3.1 前端控制器 261
 - 12.3.2 应用控制器 271
 - 12.3.3 页面控制器 283
 - 12.3.4 模板视图和视图助手 288
 - 12.4 业务逻辑层 291
 - 12.4.1 事务脚本 291
 - 12.4.2 领域模型 295
 - 12.5 小结 298
- 第13章 数据库设计模式 299
 - 13.1 数据层 299
 - 13.2 数据映射器 299
 - 13.2.1 问题 300
 - 13.2.2 实现 300
 - 13.2.3 效果 313
 - 13.3 标识映射 315
 - 13.3.1 问题 315
 - 13.3.2 实现 315
 - 13.3.3 效果 318
 - 13.4 工作单元 319
 - 13.4.1 问题 319
 - 13.4.2 实现 319
 - 13.4.3 效果 323
 - 13.5 延迟加载 323
 - 13.5.1 问题 323
 - 13.5.2 实现 324
 - 13.5.3 效果 326
 - 13.6 领域对象工厂 326
 - 13.6.1 问题 326
 - 13.6.2 实现 326
 - 13.6.3 效果 327
 - 13.7 标识对象 329
 - 13.7.1 问题 329
 - 13.7.2 实现 330
 - 13.7.3 效果 335
 - 13.8 选择工厂与更新工厂模式 335
 - 13.8.1 问题 336
 - 13.8.2 实现 336
 - 13.8.3 效果 340
 - 13.9 现在映射器中还剩下什么 340
 - 13.10 小结 342
- 第三部分 实践

第14章 优秀（以及糟糕）的实践 346

14.1 超越代码 346

14.2 借轮子 347

14.3 合作愉快 348

14.4 为代码插上翅膀 349

14.5 标准 350

14.6 Vagrant 350

14.7 测试 351

14.8 持续集成 351

14.9 小结 352

第15章 PHP标准 353

15.1 为什么需要标准 353

15.2 什么是PSR 354

15.2.1 为什么选择PSR 354

15.2.2 哪些人需要PSR 355

15.3 编码风格 355

15.3.1 PSR-1基础编码规范 356

15.3.2 PSR-2编码风格规范 358

15.3.3 检查和修改代码 360

15.4 PSR-4自动加载规范 362

15.5 小结 365

第16章 通过Composer使用和创建组件 366

16.1 什么是Composer 366

16.2 安装Composer 367

16.3 安装一个（组）包 367

16.3.1 通过命令行安装包 368

16.3.2 版本 368

16.3.3 require-dev元素 369

16.4 Composer与自动加载 370

16.5 创建自己的包 371

16.5.1 添加包信息 371

16.5.2 平台软件包 372

16.6 通过Packagist分发包 373

16.7 私有包 376

16.8 小结 377

第17章 用Git进行版本控制 378

17.1 为什么进行版本控制 378

17.2 安装Git 379

17.3 使用在线Git代码库 380

17.4 配置Git服务器 382

17.5 启动项目 384

17.6 更新与提交 387

17.7 文件和目录的添加与移除 390

17.7.1 添加文件 390

17.7.2 删除文件 390

17.7.3 添加目录 391

17.7.4 删除目录 391

17.8 标记一次发布 392

17.9 创建分支 393

17.10 小结 398

第18章 使用PHPUnit进行测试 399

18.1 功能测试与单元测试 399

18.2 手动测试 400

18.3 引入PHPUnit 402

- 18.3.1 创建测试用例 402
- 18.3.2 断言方法 405
- 18.3.3 测试异常 406
- 18.3.4 运行测试套件 407
- 18.3.5 约束 407
- 18.3.6 mock和stub 409
- 18.3.7 失败是成功之母 412
- 18.4 编写Web测试 415
 - 18.4.1 为测试重构Web应用 415
 - 18.4.2 简单的Web测试 417
 - 18.4.3 引入Selenium 419
- 18.5 警告 424
- 18.6 小结 426
- 第19章 使用Phing进行自动化构建 427
 - 19.1 Phing是什么 427
 - 19.2 获取和安装Phing 428
 - 19.3 编写构建文档 428
 - 19.3.1 目标 430
 - 19.3.2 属性 432
 - 19.3.3 类型 438
 - 19.3.4 任务 443
 - 19.4 小结 446
- 第20章 Vagrant 447
 - 20.1 问题 447
 - 20.2 设置 448
 - 20.3 挂载本地目录到Vagrant镜像 450
 - 20.4 配置 451
 - 20.4.1 设置Web服务器 452
 - 20.4.2 设置MySQL 452
 - 20.4.3 配置主机名 453
 - 20.5 结束语 455
 - 20.6 小结 455
- 第21章 持续集成 456
 - 21.1 什么是持续集成 456
 - 21.1.1 准备一个持续集成项目 458
 - 21.1.2 安装Jenkins插件 467
 - 21.1.3 设置Git公钥 468
 - 21.1.4 创建新项目 469
 - 21.1.5 运行第一次构建 472
 - 21.1.6 配置报告 472
 - 21.1.7 触发构建 474
 - 21.2 小结 476
- 第22章 对象、模式和实践 477
 - 22.1 对象 477
 - 22.1.1 选择 478
 - 22.1.2 封装与委托 478
 - 22.1.3 解耦 478
 - 22.1.4 可复用性 479
 - 22.1.5 美学 479
 - 22.2 模式 479
 - 22.2.1 模式给我们带来了什么 480
 - 22.2.2 模式与设计原则 480
 - 22.3 实践 482
 - 22.3.1 测试 482

- 22.3.2 标准 483
- 22.3.3 版本控制 483
- 22.3.4 自动构建 483
- 22.3.5 持续集成 484
- 22.3.6 我们遗漏了什么 484
- 22.4 小结 485
- 附录A 参考文献 486
- 附录B 一个简单的解析器 488
- • • • • ([收起](#))

[深入PHP：面向对象、模式与实践（第5版）_下载链接1](#)

标签

PHP

面向对象

设计模式

软件开发

计算机

★自学编程

●2020

评论

这本书太酷了，讲得很全面，非常适合有点PHP基础的同学去研究，最后的实践非常棒。

对象的高级特性 中部模式 应用性很强 循序渐进的建立一个框架 后面的实践 DevOps

Phing 自动化构建 PHP的一整套方案 都非常棒

[深入PHP：面向对象、模式与实践（第5版）_下载链接1](#)

书评

每个段落先提出问题, 给出实现, 并讨论成效,
对于OO入门有一定帮助,能够帮助开拓思路,对OO老鸟有参考价值,可以换换空气,让脑子
清空一下,听听别人说什么,对开发新程序有一定作用
内容并不能说新颖,毕竟内容已经是2007的了,不过设计模式并不会随着技术的改进而有
多少变化,毕竟理...

另外一本是PHP in Action。
PHP架构中常用的设计模式不多, 书中基本都谈到了。我觉得学习设计模式最好是和框
架一起进行, 一个是理论, 一个是实践, 而且流行的框架基本代表了设计的最新思想,
设计模式没有好坏之分, 所以有空都应该学学。

看到有人说这本书没有达到书名的目标, 可能“深入”这个词让他产生的误解了吧, 这
本书更像一本实实在在的PHP进阶指南。
本书全文分为三个方面: PHP面向对象思想, PHP设计模式, PHP实践。这三个方面对
于初级PHP工程师进阶来说都是很重要的内容。PHP OOP, 一般非直接通过PHP入门...

读第一遍读到数据库模式, 感觉吃不消了, 所以跳过去直接读后面的实践部分。
目前在读第二遍, 希望这次能吃透作者讲的数据库模式。
这绝对是一本每读一遍都会受益一便的好书,
虽然书中讲的各种模式目前看来没有应用到工作中的机会, 但是通过作者的讲解,
你会看到这些模式一旦应用...

设计模式一直以来很难懂, 之前遇到很大的瓶颈, 买回来这本书, 读起来基本一目十行
, 不是因为内容太简单, 而是该做的我都已经做过了, 只是在模式上认识还不够清晰,

