

PHP 反射在设计模式中的应用分享试题

1、填空题（每空 2 分，共 40 分）

- 1、ReflectionClass 的 getMethod 方法是否可以访问类的一个 private 方法，
如果不可以访问说明原因 ____是____，如果可以访问，是否可以对获取的方法执行
invoke 操作____否____。
- 2、反射是指 PHP 在____运行____状态中，动态获取____类____、____方法____、____属性____、
____参数____、____注释____等信息和动态调用对象的方法的功能。
- 3、依赖注入通过以____构造函数____参数，____方法____或____属性____等方式将具体组件传递给依赖方。
- 4、策略模式中体现了面向对象三大特性的____多态性____。
- 5、注解是允许加在源代码中的一种特殊格式的语法元数据，PHP 并没有专门的语言特性来支持注解，可以通过反射 API 中____getDocComment____方法来实现。
- 6、现有调用第三方发送消息的类，要实现在每次请求之前记录请求参数的日志，在不改变发送消息类方法的情况下，可以采用____代理模式____实现。
- 7、在现实生活中，我们想去某个地方的话。一般会在地图 APP 中输入起点和终点。然后选择出行方式，而每种方式价格可能不一样，可以用设计模式中的____策略模式____来实现。
- 8、PHP 可以用魔术方法____call____方法的方式来实现远程调用，很容易实现动态代理模式。
- 9、在____文档生成器____，依赖注入容器等地方广泛应用反射 API，可以通过反射深入到类的内部，但反射也会破坏对象的____封装性____，使不该暴露的方法被强制暴露了。
- 10、PHP 内置函数 get_class 可以用 ReflectionClassAPI 的 __getName__ 代替，
getMethods____能____用 get_class_vars 代替。

2、简答题（第 1，2，3，4 题每题 10 分，第 4 题 20 分，共 60 分）

1、PHP 如何验证类的属性是私有的，判断 User 类的 \$name 属性（写出一种方式即可）

(1)、\$prop = get_class_vars(User::class);

array_key_exists('name', \$prop);

(2)、\$ref = new ReflectionClass(User::class);

\$prop = \$ref->getProperty('name');

\$prop->isPrivate();

2、简要说明策略模式的使用场景？

(1)、多个类只区别在表现行为不同，在运行时动态选择具体要执行的行为

(2)、需要在不同情况下使用不同的策略

(3)、出现同一抽象类有多个子类，而又需要使用多个条件语句来选择具体子类时

3、简要描述装饰器模式优缺点。

优点:

(1)、和继承的共同特点就是扩展对象的功能，而装饰器模式比继承更加灵活，可以在不改变

原类文件和使用继承的情况下，动态的扩展一个对象的功能

(2)、通过使用不同的具体装饰器类，及其不同的排列组合，可以产生出大量不同的组合

缺点:

(1)、比继承更加复杂.

(2)、会出现一些小类，过度使用会使程序变得复杂

3、装饰器模式和代理模式的区别？

(1)、代理模式，注重对对象某一功能的流程把控和辅助。它可以控制对象做某些事，重心是为了借用对象的功能完成某一流程，而非对象功能如何。

(2)、装饰模式，注重对对象功能的扩展，它不关心外界如何调用，只注重对对象功能的加强，装饰后还是对象本身。

对于代理类，如何调用对象的某一功能是思考重点，而不需要兼顾对象的所有功能；

对于装饰类，如何扩展对象的某一功能是思考重点，同时也需要兼顾对象的其它功能，因为再怎么装饰，本质也是对象本身，要担负起对象应有的职责。

5、请求地址 `/index.php?controller=user&action=list` ,访问到 `UserController` 下到 `list` 方法，用代码简单写出请求实现过程。

(1)、`$class = new ReflectionClass(ucfirst($controller).'Controller');`

`$controller = $class->newInstance();`

`if ($class->hasMethod($method)) {`

`$method = $class->getMethod($method);`

`$method->invoke($controller);`

`}`

(2)、`$controller=ucfirst($controller).'Controller'`

`if (class_exists($controller)) {`

`$controller= new $controller;`

`$controller->$action();`

`}`