



<https://darkghosthunter.medium.com/php-10-tips-to-use-for-mockery-33673ba01321>

- 1. 如何测试链式方法
- 2. 如何测试静态方法
- 3. 如何测试 Mockery 的 expects() 和 allows() 方法

在本文中，我们将讨论如何使用 Mockery 来测试链式方法、静态方法和 expects() 和 allows() 方法。

```
$chain = Mockery::mock(Chainable::class);

$chain->expects('foo')->andReturnSelf();
$chain->expects('bar')->with(true)->andReturnSelf();
$chain->expects('baz')->with(10)->andReturnSelf();
```

在上面的代码中，我们使用 Mockery 来模拟 Chainable 类。我们使用 expects() 方法来指定我们期望调用的方法，并使用 with() 方法来指定方法的参数。最后，我们使用 andReturnSelf() 方法来指定方法的返回值。

```
$mock = Mockery::mock(Chainable::class);

$mock->expects('foo->bar->baz')->andReturn('foo');
```

在上面的代码中，我们使用 Mockery 来模拟 Chainable 类。我们使用 expects() 方法来指定我们期望调用的方法，并使用 andReturn() 方法来指定方法的返回值。

```
$mock = Mockery::mock(MyObject::class);

$mock->expects()->first()->ordered();
$mock->expects()->second()->ordered();
```

. 順序は、期待された順序で呼び出されるべきであることを保証します。つまり、期待された順序で呼び出されない場合は、テストは失敗します。

期待された順序で呼び出されない場合は、テストは失敗します。

9. 期待された順序で呼び出されない場合は、テストは失敗します。

期待された順序で呼び出されない場合は、テストは失敗します。

```
$mock = Mockery::spy(Useful::class);

// An object uses the mock.

$mock->shouldHaveReceived('getBody')->times(3);
```

期待された回数で呼び出されない場合は、テストは失敗します。MockeryのandReturnArg()は、期待された回数で呼び出されない場合は、テストは失敗します。

期待された回数で呼び出されない場合は、テストは失敗します。

```
$mock = Mockery::mock(Useful::class);

$mock->expects('foo')
    ->withAny()
    ->andReturnUsing(fn ($string) => 'You wrote: ' . $string)
```

Revision #1

Created 4 August 2024 00:43:31 by 匿名

Updated 4 August 2024 01:05:43 by 匿名