

is_callable() 函数

- 语法：

```
bool is_callable ( callable $name [, bool $syntax_only = false [, string &$amp;$callable_name ]] )
```

```
<?php
if (is_callable(array($object, 'SomeMethod')) {
    $object->SomeMethod($this, TRUE);
}
```

- 功能：函数用于检测函数在**当前环境**中是否可调用。
- 参数：
 - \$name：要检查的回调函数。
 - \$syntax_only：如果设置为 TRUE，这个函数仅仅验证 name 可能是函数或方法。它仅仅拒绝非字符，或者未包含能用于回调函数的有效结构。有效的应该包含两个元素，第一个是一个对象或者字符，第二个元素是个字符。
 - \$callable_name：接受"可调用的名称"。
- 返回值：
 - 如果 name 可调用则返回 TRUE，否则返回 FALSE。
- 例子：

```

<?php
class foo
{
    public function a()
    {    // 公有方法a
        echo 'a';
    }
    protected function b()
    {
        // 受保护的方法b
        echo 'b';
    }
    private function c()
    {
        // 私有方法c
        echo 'c';
    }
    public function d()
    {
        /*
            is_callable() 函数的当前环境是foo类。
        */
        if (is_callable([$this, 'c'])) {
            echo 'true';
        } else {
            echo 'false';
        }
    }
}

class aoo extends foo
{
    public static function test(array $callbacks)
    {
        foreach ($callbacks as $callback) {
            /*
                is_callable() 函数的当前环境是aoo类，aoo类继承自foo类。
            */
            if (is_callable($callback)) {
                echo 'aoo类中可以调用' . $callback[1];
            } else {
                echo 'aoo类中不可以调用' . $callback[1];
            }
            echo '<br/>';
        }
    }
}

class Main
{
    public static function client()
    {
        $a = new foo();
    }
}

```

```

        $callbacks = array(
            [$a, 'a'],
            [$a, 'b'],
            [$a, 'c']
        );
        foreach ($callbacks as $callback) {
            /*
             is_callable() 函数的当前环境是Main类。
            */
            var_dump(is_callable($callback));
            echo '<br/>';
        }
        aoo::test($callbacks);
        $a->d();
    }
}
Main::client();
/*
输出:
bool(true)           // 在Main类中, foo的实例可以调用公有方法a。
bool(false)          // 在Main类中, foo的实例不能调用受保护方法b。
bool(false)          // 在Main类中, foo的实例不能调用公私有方法c。
string(4) "true"      // 在aoo类中, foo的实例可以调用公有方法a。
string(4) "true"      // 在aoo类中, foo的实例可以调用受保护方法b。
string(5) "false"     // 在aoo类中, foo的实例不能调用公私有方法c。
true                 // 在foo类中, foo自身可以调用公私有方法c。
*/

```

note路径

- D:\phpStudy\PHPTutorial\WWW\index\myframe2\note\isCallable

详情:

https://blog.csdn.net/qq_28602957/article/details/65444356