

## 51. Facade

学习要点：

1. 创建静态调用
2. 核心类库

本节课我们来学习一下 Facade 的使用方式，以及系统内置的 Facade 核心类库。

### 一. 创建静态调用

1. Facade，即门面设计模式，为容器的类提供了一种静态的调用方式；
2. 在之前的很多课程中，我们大量的引入 Facade 类库，并且通过静态调用；
3. 比如请求 Request::?，路由 Route::?，数据库 Db::?等等，均来自 Facade；
4. 下面我们手工来创建一个自己的静态调用类库，来了解一下流程；
5. 首先，在应用目录下创建 common 公共类库文件夹，并创建 Test.php；

```
namespace app\common;
class Test
{
    public function hello($name)
    {
        return 'Hello, '.$name;
    }
}
```

6. 再在同一个目录下创建 facade 文件夹，并创建 Test.php，用于生成静态调用；

```
namespace app\Facade;
use think\Facade;
class Test extends Facade
{
    protected static function getFacadeClass()
    {
        return 'app\common\Test';
    }
}
```

7. 然后在控制器端，就可以和之前系统提供的静态调用一样调用了；

```
return Test::hello('Mr.Lee!');
```

8. 除了在 getFacadeClass() 方法显示绑定，也可以在应用公共函数文件进行绑定；
9. 这里的绑定后，就不需要 getFacadeClass() 方法了，还可以进行批量统一绑定；

```
// 应用公共文件
use think\Facade;
Facade::bind('app\facade\Test', 'app\common\Test');
```

```
Facade::bind([
    'app\\facade\\Test' => 'app\\common\\Test',
]);
```

## 二. 核心类库

### 1. 以下是系统提供的常用 Facade 核心类库表：

(动态)类库	Facade类
think\\App	think\\facade\\App
think\\Build	think\\facade\\Build
think\\Cache	think\\facade\\Cache
think\\Config	think\\facade\\Config
think\\Cookie	think\\facade\\Cookie
think\\Debug	think\\facade\\Debug
think\\Env	think\\facade\\Env
think\\Hook	think\\facade\\Hook
think\\Lang	think\\facade\\Lang
think\\Log	think\\facade\\Log
think\\Middleware	think\\facade\\Middleware
think\\Request	think\\facade\\Request
think\\Response	think\\facade\\Response
think\\Route	think\\facade\\Route
think\\Session	think\\facade\\Session
think\\Url	think\\facade\\Url
think\\Validate	think\\facade\\Validate
think\\View	think\\facade\\View

### 2. 在真正使用 Facade 核心类库时，直接使用提供的别名即可，具体别名如下：

别名类	对应Facade类
App	think\\facade\\App
Build	think\\facade\\Build
Cache	think\\facade\\Cache
Config	think\\facade\\Config
Cookie	think\\facade\\Cookie
Db	think\\Db
Debug	think\\facade\\Debug
Env	think\\facade\\Env
Hook	think\\facade\\Hook
Lang	think\\facade\\Lang
Log	think\\facade\\Log
Middleware	think\\facade\\Middlew
Request	think\\facade\\Request
Response	think\\facade\\Respons
Route	think\\facade\\Route
Session	think\\facade\\Session
Url	think\\facade\\Url
Validate	think\\facade\\Validate
View	think\\facade\\View