

81. 服务系统

学习要点：

1. 服务系统

本节课我们来学习一下要使用一下系统提供的服务系统。

一. 服务系统

1. 服务系统，可以将一个类的对象注册到容器中去，方便调用执行；
2. 服务的执行优先级较高，在执行主体程序前就已经完成依赖注入；
3. 它的作用可以做一些初始化，配置一些参数，扩展插件等等均可；
4. 验证码扩展类就使用了服务系统，我们自己创建一个简单的服务；
5. 在 common 目录下创建一个 Shut.php 类，这个类是被服务的类；

//定义一个属性字段

```
protected static $name = 'Mr.Lee';
```

//设置

```
public static function setName($name)
```

```
{
```

```
    self::$name = $name;
```

```
}
```

//获取

```
public function run()
```

```
{
```

```
    halt(self::$name.'提醒您，系统已关闭...');
```

```
}
```

6. 使用命令行，生成一个对 Shut.php 服务的服务类 ShutService.php;

```
php think make:service ShutService
```

7. 服务类有两个方法，一个是服务注册 register()，一个服务启动 boot()；

```
public function register()
```

```
{
```

```
    //绑定到容器，将被服务的类注册到容器中去
```

```
    $this->app->bind('shut', Shut::class);
```

```
}
```

```
public function boot()
```

```
{
```

```
    //执行
```

```
    Shut::setName('Mr.Wang');
```

```
}
```

8. 最后一部，将系统服务配置到全局定义文件里，`service.php`;

```
return [  
    \app\service\ShutService::class,  
];
```

9. 最后在任意控制器测试即可，可以容易依赖注入或容器标识执行系统服务;

```
public function index(Shut $shut)  
{  
    //依赖注入调用  
    $shut->run();  
    //容器标识调用  
    $this->app->shut->run();  
    return 'index';  
}
```