

04. 控制器定义

学习要点：

1. 控制器定义
2. 渲染输出

本节课我们来了解一下控制器定义，包括风格、是否需要继承以及各式各样的输出方式。

一. 控制器定义

1. 控制器，即 `controller`，控制器文件存放在 `controller` 目录下；
2. 类名和文件名大小写保持一致，并采用驼峰式（首字母大写）；

```
use think\Controller;
class Index extends Controller
```
3. 继承控制器基类，可以更方便使用功能，但不是必须的；
4. 系统也提供了其它方式，在不继承的情况下完成相同功能；
5. 前面我们知道如果是一个单词，首字母大写，比如 `class Index`；
6. URL 访问时直接 `public/index` 即可；
7. 那么如果创建的是双字母组合，比如 `class HelloWorld`；
8. URL 访问时必须为： `public/hello_world`；
9. 如果你想原样的方式访问 URL，则需要关闭配置文件中自动转换；

```
'url_convert' => false,
```
10. 此时，URL 访问可以为： `public/HelloWorld`；
11. 如果你想改变根命名空间 `app` 为其它，可以在根目录下创建 `.env` 文件；
12. 然后写上配对的键值对即可， `app_namespace=application`；

二. 渲染输出

1. ThinkPHP 直接采用方法内 `return` 返回的方式直接就输出了；
2. 使用 `json` 输出，直接采用 `json` 函数；

```
$data = array('a'=>1, 'b'=>2, 'c'=>3);
return json($data);
```
3. 使用 `view` 输出模版，开启错误提示，可知道如何创建模版；

```
return view();
```
4. 默认输出方式为 `html` 格式输出，如果返回的是数组，则会报错；
5. 可以更改配置文件里的默认输出类型，更改为 `json`；

```
return ['user'=>'Lee', 'age'=>100];
'default_return_type' => 'json',
```

6. 一般来说，正常页面都是 **html** 输出，用于模版，**AJAX** 默认为 **json**；
7. 如果继承了基类控制器，那么可以定义控制器初始化方法：**initialize()**；
8. **initialize()**方法会在调用控制器方法之前执行；

```
protected function initialize()
{
    //parent::initialize();
    echo 'init';
}
```

9. **initialize()**方法不需要任何返回值，输出用 **PHP** 方式，**return** 无效；