

55. 异常处理

学习要点：

1. 异常处理

本节课我们来学习一下异常处理的方式，以及如何设置错误页面。

一. 异常处理

1. 系统输出的异常信息比 PHP 原生的要人性化的多，但需要开启调试模式；

2. 如果你想更改异常页面的样式、布局之类的，可以修改这个页面：

`thinkphp/tpl/think_exception.tpl`

3. 如果你想要直接替换掉异常页面，换成别的，可以在 `app.php` 中进行设置：

`// 异常页面的模板文件`

```
'exception_tpl' => Env::get('think_path') .  
                        'tpl/think_exception.tpl',
```

4. 默认情况下，对所有 PHP 的错误都会抛出异常信息，可以用错误级别关闭；

`error_reporting(0);`

5. 系统的异常抛出是自动进行的，并不需要你手动抛出，但也支持手动；

`throw new Exception('异常消息', 10086);`

6. 我们可以使用 `try...catch` 对可能发生的异常进行手动捕获或抛出；

```
try {  
    echo 0/0;  
} catch (ErrorException $e)  
{  
    echo '发生错误：' . $e->getMessage();  
}
```

7. 我们可以抛出 HTTP 异常，所谓 HTTP 异常比如 404 错误，500 错误之类；

`throw new HttpException(404, '页面不存在');`

8. 系统提供了一个助手函数 `abort()` 方法简化 HTTP 异常抛出；

`abort(404, '页面不存在');`

9. 如果系统关闭了调试模式，进入部署环境下，可以设置 HTTP 错误页面，比如 404；

```
'http_exception_template' => [  
    // 定义 404 错误的模板文件地址  
    404 => Env::get('app_path') . '404.html',  
]
```