

76. 图像处理功能

学习要点：

1. 图像处理功能

本节课我们来学习一下图像处理功能，这功能是外置的，并非系统内置。

一. 图像处理功能

1. 图像处理功能不是系统内置的功能了，需要通过 `composer` 引入进来；

```
composer require tophink/think-image
```

2. 引入进来之后，首先创建图像处理对象；

```
$image = Image::open('image.png');
```

3. 获得了图像处理对象后，可以得到这张图片的各种属性；

```
//图片宽度
echo $image->width();
//图片高度
echo $image->height();
//图片类型
echo $image->type();
//图片 mime
echo $image->mime();
//图片大小
dump($image->size());
```

4. 使用 `crop()` 方法可以裁剪图片，并使用 `save()` 方法保存到指定路径；

5. 可以点击追踪方法内部，参看源码参数，了解更多的传值方法；

```
//裁剪图片
$image->crop(550,400)->save('crop1.png');
```

6. 使用 `thumb()` 方法，可以生成缩略图，配合 `save()` 把缩略图保存下来；

```
//生成缩略图
$image->thumb(500,500)->save('thumb1.png');
```

7. 这里要注意一个问题，虽然设置了宽和高，但高度变成了 282，说明是等比例的；

8. 可以点击追踪方法内部，第三个参数默认为：`$type = self::THUMB_SCALING`；

9. 而这个常量设置的定义如下：

```
/* 缩略图相关常量定义 */
const THUMB_SCALING = 1; //常量，标识缩略图等比例缩放类型
const THUMB_FILLED = 2; //常量，标识缩略图缩放后填充类型
const THUMB_CENTER = 3; //常量，标识缩略图居中裁剪类型
...
```

10. 使用 `rotate()` 方法，可以旋转图片，默认是 90 度，参数可以设置；

```
$image->rotate(180)->save('rotate1.png');
```

11. `save()` 方法可以配置的参数除了保存文件名的路径，还有以下几个：

12. `save('路径', ['类型', '质量', '是否隔行扫描'])`，追踪到方法查看；

```
save($pathname, $type = null, $quality = 80, $interlace = true)
```

13. `water()` 方法，可以给图片增加一个图片水印，默认位置为右下角，可看源码常量；

```
$image->water('mr.lee.png')->save('water1.png');
```

14. `text()` 方法，可以给图片增加一个文字水印，具体如下：

```
$image->text('Mr.Lee', getcwd().'/1.ttf', 20, '#ffffff')->save('text1.png');
```