

## 51. 请求类型和 HTTP 头信息

学习要点：

1. 请求类型
2. HTTP 头信息

本节课我们来学习一下数据请求类型和 HTTP 头信息的内容。

### 一. 请求类型

1. 有时，我们需要判断 **Request** 的请求类型，比如 **GET**、**POST** 等等；
2. 可以使用 **method()** 方法来判断当前的请求类型，当然，还有很多专用的请求判断；

用途	方法
获取当前请求类型	method
判断是否GET请求	isGet
判断是否POST请求	isPost
判断是否PUT请求	isPut
判断是否DELETE请求	isDelete
判断是否AJAX请求	isAjax
判断是否PJAX请求	isPjax
判断是否JSON请求	isJson
判断是否手机访问	isMobile
判断是否HEAD请求	isHead
判断是否PATCH请求	isPatch
判断是否OPTIONS请求	isOptions
判断是否为CLI执行	isCli
判断是否为CGI模式	isCgi

3. 使用普通表单提交，通过 **method()** 方法获取类型；

```
<form action="http://localhost/tp6/public/rely/get" method="post">
  <input type="text" name="name" value="Lee">
  <input type="submit" value="提交">
</form>

return Request::method();
```

4. 在表单提交时，我们也可以设置请求类型伪装，设置隐藏字段 **\_method**；
5. 而在判断请求，使用 **method(true)** 可以获取原始请求，否则获取伪装请求；

```
<input type="hidden" name="_method" value="PUT">

Request::method(true);
```

6. 如果想更改请求伪装变量类型的名称，可以在 `app/Request.php` 中更改；

```
protected $varMethod = '_m';
```

7. AJAX/PJAX 伪装，使用 `?_ajax=1` 和 `?_pjax=1`，并使用 `isAjax()` 和 `isPjax()`；

```
.../rely?_ajax=1
```

```
Request::isAjax();
```

8. 这里需要用 `isAjax()` 和 `isPjax()` 来判断，用 `method` 无法判断是否为 `a(p)jax`；

9. 在 `app.php` 也可以更改 `ajax` 和 `pjax` 的名称；

```
protected $varAjax = '_a';
```

```
protected $varPjax = '_p';
```

## 二. HTTP 头信息

1. 使用 `header()` 方法可以输出 HTTP 头信息，返回是数组类型，也可单信息获取；

```
Request::header();
```

```
Request::header('host');
```