configs详解——之requests

requests表示当前正在爬取的网站的对象，下面介绍了可以调用的函数

**requests成员**

**input\_encoding**

输入编码  
明确指定输入的页面编码格式(UTF-8,GB2312,…..)，防止出现乱码,如果设置null则自动识别

**String类型 可选设置**

**input\_encoding默认值为null，即程序自动识别页面编码**

举个栗子:

requests::$input\_encoding = 'GB2312';

**output\_encoding**

输出编码  
明确指定输出的编码格式(UTF-8,GB2312,…..)，防止出现乱码,如果设置null则为utf-8

**String类型 可选设置**

**output\_encoding默认值为utf-8, 如果数据库为gbk编码，请修改为gb2312**

举个栗子:

requests::$output\_encoding = 'GB2312';

**requests方法**

**set\_timeout($timeout)**

一般在on\_start回调函数（在[爬虫进阶开发——之回调函数](https://doc.phpspider.org/callback.html)中会详细描述）中调用, 设置请求超时时间

@param $timeout 需添加的timeout

**默认值为5，即5秒超时**

栗子1:  
传入单一的值作为timeout，会同时设置connect和read

$spider->on\_start = function($phpspider)

{

requests::set\_timeout(10);

};

栗子2：  
传入数组，会分别设置connect和read二者的timeout

requests::set\_timeout( array(3, 27) );

栗子3：  
如果远端服务器很慢，你可以让requests永远等待，传入一个 0 作为 timeout 值，然后就冲咖啡去吧。

requests::set\_timeout(0);

**set\_proxy($proxy)**

一般在on\_start回调函数（在[爬虫进阶开发——之回调函数](https://doc.phpspider.org/callback.html)中会详细描述）中调用, 设置请求代理

@param $proxy 需添加的代理，用于破解防采集，支持字符串和数组类型传入

栗子1：

字符串类型

$spider->on\_start = function($phpspider)

{

requests::set\_proxy('http://user:pass@host:port);

};

栗子2:

数组类型，代理有多个

$spider->on\_start = function($phpspider)

{

// 代理如果有多个，请求时会随机采用

$proxy = array(

'http://user1:pass1@host:port',

'http://user2:pass2@host:port',

);

requests::set\_proxy($proxy);

};

**set\_useragent($useragent)**

一般在on\_start回调函数（在[爬虫进阶开发——之回调函数](https://doc.phpspider.org/callback.html)中会详细描述）中调用, 设置浏览器useragent

@param $useragent 需添加的useragent

**默认使用useragent: requests/2.0.0**

[点击查看“常见浏览器useragent大全”](http://taro.iteye.com/blog/1634253)

栗子1:

$spider->on\_start = function($phpspider)

{

requests::set\_useragent("Mozilla/4.0 (compatible; MSIE 6.0; ) Opera/UCWEB7.0.2.37/28/");

};

栗子2:

随机伪造useragent，传递数组即可，爬虫在请求的时候就会随机取一个useragent访问对方网站，让对方网站对useragent的反爬虫限制失

$spider->on\_start = function($phpspider)

{

requests::set\_useragent(array(

"Mozilla/4.0 (compatible; MSIE 6.0; ) Opera/UCWEB7.0.2.37/28/",

"Opera/9.80 (Android 3.2.1; Linux; Opera Tablet/ADR-1109081720; U; ja) Presto/2.8.149 Version/11.10",

"Mozilla/5.0 (Android; Linux armv7l; rv:9.0) Gecko/20111216 Firefox/9.0 Fennec/9.0"

));

};

**set\_referer($referer)**

一般在on\_start回调函数（在[爬虫进阶开发——之回调函数](https://doc.phpspider.org/callback.html)中会详细描述）中调用, 设置请求来路URL

@param $referer 需添加的来路URL，用于破解防采集

举个栗子:

$spider->on\_start = function($phpspider)

{

requests::set\_referer("http://www.qiushibaike.com");

};

**set\_header($key, $value)**

一般在on\_start回调函数（在[爬虫进阶开发——之回调函数](https://doc.phpspider.org/callback.html)中会详细描述）中调用, 用来添加一些HTTP请求的Header

@param $key Header的key, 如User-Agent,Referer等  
@param $value Header的值

举个栗子:  
Referer是HTTP请求Header的一个属性，http://www.9game.cn/kc/是Referer的值

$spider->on\_start = function($phpspider)

{

requests::set\_header("Referer", "http://www.9game.cn/kc/");

};

**set\_cookie($key, $value, $domain='')**

一般在on\_start回调函数（在[爬虫进阶开发——之回调函数](https://doc.phpspider.org/callback.html)中会详细描述）中调用, 用来添加一些HTTP请求的Cookie

@param $key Cookie的key  
@param $value Cookie的值  
@param $domain 默认放到全局Cookie，设置域名后则放到相应域名下

举个栗子:  
cookie是由键-值对组成的，BAIDUID是cookie的key，FEE96299191CB0F11954F3A0060FB470:FG=1则是cookie的值

$spider->on\_start = function($phpspider)

{

requests::set\_cookie("BAIDUID", "FEE96299191CB0F11954F3A0060FB470:FG=1");

// 把Cookie设置到 www.phpspider.org 域名下

requests::set\_cookie("NAME", "phpspider", "www.phpspider.org");

};

**get\_cookie($name, $domain = '')**

一般在on\_start回调函数（在[爬虫进阶开发——之回调函数](https://doc.phpspider.org/callback.html)中会详细描述）中调用, 用来得到某个域名所附带的某个Cookie

@param $name Cookie的名称  
@param $domain configs的domains成员中的元素

举个栗子:  
得到s.weibo.com域名所附带的Cookie, 并将Cookie添加到weibo.com的域名中

$configs = array(

'domains' => array(

's.weibo.com',

'weibo.com'

)

// configs的其他成员

...

);

$spider->on\_start = function($phpspider)

{

$cookie = requests::get\_cookie("SUB", "s.weibo.com");

// 把Cookie设置到 weibo.com 域名下

requests::set\_cookie("SUB", $cookie, "weibo.com");

};

**set\_cookies($cookies, $domain='')**

一般在on\_start回调函数（在[爬虫进阶开发——之回调函数](https://doc.phpspider.org/callback.html)中会详细描述）中调用, 用来添加一些HTTP请求的Cookie

@param $cookies 多个Cookie组成的字符串  
@param $domain 默认放到全局Cookie，设置域名后则放到相应域名下

举个栗子:  
cookies是多个cookie的键-值对组成的字符串，用;分隔。BAIDUID和BIDUPSID是cookie的key，FEE96299191CB0F11954F3A0060FB470:FG=1和FEE96299191CB0F11954F3A0060FB470是cookie的值，键-值对用=相连

$spider->on\_start = function($phpspider)

{

requests::set\_cookies("BAIDUID=FEE96299191CB0F11954F3A0060FB470:FG=1; BIDUPSID=FEE96299191CB0F11954F3A0060FB470;");

// 把Cookie设置到 www.phpspider.org 域名下

requests::set\_cookies("NAME", "www.phpspider.org");

};

**get\_cookies($domain = '')**

一般在on\_start回调函数（在[爬虫进阶开发——之回调函数](https://doc.phpspider.org/callback.html)中会详细描述）中调用, 用来得到某个域名所附带的所有Cookie

@param $domain configs的domains成员中的元素  
@return array 返回的是所有Cookie的数组

举个栗子:  
得到s.weibo.com域名所附带的Cookie, 并将Cookie添加到weibo.com的域名中

$configs = array(

'domains' => array(

's.weibo.com',

'weibo.com'

)

// configs的其他成员

...

);

$spider->on\_start = function($phpspider)

{

$cookies = requests::get\_cookies("s.weibo.com");

// 返回的是数组，可以输出看看所有的Cookie内容

print\_r($cookies);

// 数组转化成String

$cookies = implode(";", $cookies);

// 把Cookie设置到 weibo.com 域名下

requests::set\_cookies($cookies, "weibo.com");

};

**set\_client\_ip($ip)**

一般在on\_start回调函数（在[爬虫进阶开发——之回调函数](https://doc.phpspider.org/callback.html)中会详细描述）中调用, 设置请求伪IP

@param $ip 需添加的伪IP

栗子1:

$spider->on\_start = function($phpspider)

{

requests::set\_client\_ip("192.168.0.2");

};

栗子2:

随机伪造IP，只需要添加数组即可

$spider->on\_start = function($phpspider)

{

$ips = array(

"192.168.0.2",

"192.168.0.3",

"192.168.0.4"

);

requests::set\_client\_ip($ips);

};

**set\_hosts($host, $ips)**

一般在on\_start回调函数（在[爬虫进阶开发——之回调函数](https://doc.phpspider.org/callback.html)中会详细描述）中调用, 设置请求的第三方主机和IP

@param $hosts 需添加的主机和IP，用于采集第三方不同的服务器

举个栗子:

$spider->on\_start = function($phpspider)

{

$host = "www.qiushibaike.com";

$ips = array(

"203.195.143.21",

"203.195.143.22"

);

requests::set\_hosts($host, $ips);

};

**get($url, $params, $allow\_redirects, $cert)**

可以在任何地方调用, 用来获取某个网页

@param $url 请求URL  
@param $params 请求参数  
@param $allow\_redirects 是否允许获取跳转后的页面  
@param $cert 证书

举个栗子:  
获取 Github 的公共时间线

$json = requests::get("https://github.com/timeline.json");

$data = json\_decode($json, true);

print\_r($data);

**post($url, $params, $files, $allow\_redirects, $cert)**

可以在任何地方调用, 用来获取某个网页

@param $url 请求URL  
@param $params 请求参数  
@param $files 上传文件  
@param $allow\_redirects 是否允许获取跳转后的页面  
@param $cert 证书

栗子1:  
用户登录

$params = array(

'username' => 'test888',

'password' => '123456',

);

$html = requests::post("http://www.domain.com", $params);

栗子2：  
文件上传

$files = array(

'file1' => 'test.jpg',

'file2' => 'test.png'

);

$html = requests::post("http://www.domain.com", NULL, $files);

服务端代码

if ($\_FILES["file1"]["error"] > 0)

{

echo "错误：" . $\_FILES["file1"]["error"] . "<br\>";

}

else

{

echo "上传文件名: " . $\_FILES["file1"]["name"] . "<br\>";

echo "文件类型: " . $\_FILES["file1"]["type"] . "<br\>";

echo "文件大小: " . ($\_FILES["file1"]["size"] / 1024) . " kB<br\>";

echo "文件临时存储的位置: " . $\_FILES["file1"]["tmp\_name"];

}

**put($url, $params, $allow\_redirects, $cert)**

可以在任何地方调用, 用来获取某个网页

@param $url 请求URL  
@param $params 请求参数  
@param $allow\_redirects 是否允许获取跳转后的页面  
@param $cert 证书

举个栗子:  
添加用户 test888

$params="{username:\"test888\",username:\"123456\"}";

$html = requests::put("http://www.domain.com", $params);

**delete($url, $params, $allow\_redirects, $cert)**

可以在任何地方调用, 用来获取某个网页

@param $url 请求URL  
@param $params 请求参数  
@param $allow\_redirects 是否允许获取跳转后的页面  
@param $cert 证书

举个栗子:  
删除用户 test888

$params="{username:\"test888\"}";

$html = requests::delete("http://www.domain.com", $params);

**获取网页编码**

可以使用 requests::$encoding 来获取网页编码

requests::get("http://www.domain.com");

echo requests::$encoding;

// utf-8

当你发送请求时，requests会根据HTTP头部来猜测网页编码，当HTTP头部无法获取时，会继续用网页内容来继续猜测，当你使用 requests::$text 时，requests就会使用这个编码。当然你还可以修改reuqests的编码形式。

requests::$output\_encoding = 'gbk';

requests::get("http://www.domain.com");

echo requests::$encoding;

// gbk

像上面的例子，请求前先指定要输出的编码为gbk，获取到的网页内容就是gbk编码的内容。

**json**

现在很多PHP环境都默认自带json扩展，并不需要像python、golang那样需要去引入新模块，下面是查询IP的API 例子

$json = requests::get('http://ip.taobao.com/service/getIpInfo.php?ip=122.88.60.28');

$rs = json\_decode($json, true);

echo $rs['data']['country'];

// 中国

**获取响应内容**

可通过 requests::$content、requests::$text 来获取转码前后网页的内容。

requests::get("http://www.domain.com");

// 转码前内容

echo requests::$content;

// 转码后内容

echo requests::$text;

**网页状态码**

可以用 requests::$status\_code 来检查网页的状态码

requests::get("http://www.domain.com");

// 状态码

echo requests::$status\_code;

// 如果是302跳转，获取跳转前状态码

echo requests::$history;

**响应头内容**

可以通过 requests::$headers 来获取响应头内容

requests::get("http://www.domain.com");

print\_r(requests::$headers);

echo requests::$headers['Content-Type'];

**请求头内容**

可以通过 requests::$request['headers'] 来获取请求头内容。

requests::get("http://www.domain.com");

print\_r(requests::$request['headers']);

**自定义请求头部**

伪装请求头部是采集时经常用的，我们可以用这个方法来隐藏：

requests::get("http://www.domain.com")

print\_r(requests::$request['headers']['User-Agent']);

// php-requests/1.2.3 PHP/5.6 Windows/XP

requests::$request['headers']['User-Agent'] = 'phpspider';

requests::get("http://www.domain.com");

print\_r(requests::$request['headers']['User-Agent']);

// phpspider

**curl**

curl -X PUT <http://www.domain.com/demo.php> -d "id=1" -d "title=a"