这篇文章主要介绍了PHP版微信公众平台微信API类，目前主要实现了微信红包接口，陆续会继续进行更新，非常的实用，这里推荐给小伙伴们，有需要的朋友可以参考下。

重写了一下PHP下面的微信API接口，

微信红包支持，JSAPI的动态参数接口支持

http://git.oschina.net/youkuiyuan/yky\_test/blob/master/class/wxapi.class.php

微信API类 - 增加红包支持

1. <?php
2. /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*
3. \* @author Kyler You <QQ:2444756311>
4. \* @link http://mp.weixin.qq.com/wiki/home/index.html
5. \* @version 2.0.1
6. \* @uses $wxApi = new WxApi();
7. \* @package 微信API接口 陆续会继续进行更新
8. \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/
10. **class** WxApi {
11. **const** appId = "";
12. **const** appSecret = "";
13. **const** mchid = ""; //商户号
14. **const** privatekey = ""; //私钥
15. **public** $parameters = **array**();
17. **public** **function** \_\_construct(){
19. }
21. /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*
22. \* 微信提交API方法，返回微信指定JSON
23. \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/
25. **public** **function** wxHttpsRequest($url,$data = null){
26. $curl = curl\_init();
27. curl\_setopt($curl, CURLOPT\_URL, $url);
28. curl\_setopt($curl, CURLOPT\_SSL\_VERIFYPEER, FALSE);
29. curl\_setopt($curl, CURLOPT\_SSL\_VERIFYHOST, FALSE);
30. **if** (!empty**empty**($data)){
31. curl\_setopt($curl, CURLOPT\_POST, 1);
32. curl\_setopt($curl, CURLOPT\_POSTFIELDS, $data);
33. }
34. curl\_setopt($curl, CURLOPT\_RETURNTRANSFER, 1);
35. $output = curl\_exec($curl);
36. curl\_close($curl);
37. **return** $output;
38. }
40. /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*
41. \* 微信带证书提交数据 - 微信红包使用
42. \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/
44. **public** **function** wxHttpsRequestPem($url, $vars, $second=30,$aHeader=**array**()){
45. $ch = curl\_init();
46. //超时时间
47. curl\_setopt($ch,CURLOPT\_TIMEOUT,$second);
48. curl\_setopt($ch,CURLOPT\_RETURNTRANSFER, 1);
49. //这里设置代理，如果有的话
50. //curl\_setopt($ch,CURLOPT\_PROXY, '10.206.30.98');
51. //curl\_setopt($ch,CURLOPT\_PROXYPORT, 8080);
52. curl\_setopt($ch,CURLOPT\_URL,$url);
53. curl\_setopt($ch,CURLOPT\_SSL\_VERIFYPEER,false);
54. curl\_setopt($ch,CURLOPT\_SSL\_VERIFYHOST,false);
56. //以下两种方式需选择一种
58. //第一种方法，cert 与 key 分别属于两个.pem文件
59. //默认格式为PEM，可以注释
60. curl\_setopt($ch,CURLOPT\_SSLCERTTYPE,'PEM');
61. curl\_setopt($ch,CURLOPT\_SSLCERT,getcwd().'/apiclient\_cert.pem');
62. //默认格式为PEM，可以注释
63. curl\_setopt($ch,CURLOPT\_SSLKEYTYPE,'PEM');
64. curl\_setopt($ch,CURLOPT\_SSLKEY,getcwd().'/apiclient\_key.pem');
66. curl\_setopt($ch,CURLOPT\_CAINFO,'PEM');
67. curl\_setopt($ch,CURLOPT\_CAINFO,getcwd().'/rootca.pem');
69. //第二种方式，两个文件合成一个.pem文件
70. //curl\_setopt($ch,CURLOPT\_SSLCERT,getcwd().'/all.pem');
72. **if**( count($aHeader) >= 1 ){
73. curl\_setopt($ch, CURLOPT\_HTTPHEADER, $aHeader);
74. }
76. curl\_setopt($ch,CURLOPT\_POST, 1);
77. curl\_setopt($ch,CURLOPT\_POSTFIELDS,$vars);
78. $data = curl\_exec($ch);
79. **if**($data){
80. curl\_close($ch);
81. **return** $data;
82. }
83. **else** {
84. $error = curl\_errno($ch);
85. echo "call faild, errorCode:$error\n";
86. curl\_close($ch);
87. **return** false;
88. }
89. }
91. /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*
92. \* 微信获取AccessToken 返回指定微信公众号的at信息
93. \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/
95. **public** **function** wxAccessToken($appId = NULL , $appSecret = NULL){
96. $appId = is\_null($appId) ? self::appId : $appId;
97. $appSecret = is\_null($appSecret) ? self::appSecret : $appSecret;
98. //echo $appId,$appSecret;
99. $url = "https://api.weixin.qq.com/cgi-bin/token?grant\_type=client\_credential&appid=".$appId."&secret=".$appSecret;
100. $result = $this->wxHttpsRequest($url);
101. //print\_r($result);
102. $jsoninfo = json\_decode($result, true);
103. $access\_token = $jsoninfo["access\_token"];
104. **return** $access\_token;
105. }
107. /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*
108. \* 微信通过OPENID获取用户信息，返回数组
109. \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/
111. **public** **function** wxGetUser($openId){
112. $wxAccessToken = $this->wxAccessToken();
113. $url = "https://api.weixin.qq.com/cgi-bin/user/info?access\_token=".$wxAccessToken."&openid=".$openId."&lang=zh\_CN";
114. $result = $this->wxHttpsRequest($url);
115. $jsoninfo = json\_decode($result, true);
116. **return** $jsoninfo;
117. }
119. /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*
120. \* 微信通过指定模板信息发送给指定用户，发送完成后返回指定JSON数据
121. \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/
123. **public** **function** wxSendTemplate($jsonData){
124. $wxAccessToken = $this->wxAccessToken();
125. $url = "https://api.weixin.qq.com/cgi-bin/message/template/send?access\_token=".$wxAccessToken;
126. $result = $this->wxHttpsRequest($url,$jsonData);
127. **return** $result;
128. }
130. /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*
131. \* 发送自定义的模板消息
132. \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/
134. **public** **function** wxSetSend($touser, $template\_id, $url, $data, $topcolor = '#7B68EE'){
135. $template = **array**(
136. 'touser' => $touser,
137. 'template\_id' => $template\_id,
138. 'url' => $url,
139. 'topcolor' => $topcolor,
140. 'data' => $data
141. );
142. $jsonData = json\_encode($template);
143. $result = $this->wxSendTemplate($jsonData);
144. **return** $result;
145. }
147. /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*
148. \* 微信设置OAUTH跳转URL，返回字符串信息 - SCOPE = snsapi\_base //验证时不返回确认页面，只能获取OPENID
149. \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/
151. **public** **function** wxOauthBase($redirectUrl,$state = "",$appId = NULL){
152. $appId = is\_null($appId) ? self::appId : $appId;
153. $url = "https://open.weixin.qq.com/connect/oauth2/authorize?appid=".$appId."&redirect\_uri=".$redirectUrl."&response\_type=code&scope=snsapi\_base&state=".$state."#wechat\_redirect";
154. **return** $url;
155. }
157. /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*
158. \* 微信设置OAUTH跳转URL，返回字符串信息 - SCOPE = snsapi\_userinfo //获取用户完整信息
159. \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/
161. **public** **function** wxOauthUserinfo($redirectUrl,$state = "",$appId = NULL){
162. $appId = is\_null($appId) ? self::appId : $appId;
163. $url = "https://open.weixin.qq.com/connect/oauth2/authorize?appid=".$appId."&redirect\_uri=".$redirectUrl."&response\_type=code&scope=snsapi\_userinfo&state=".$state."#wechat\_redirect";
164. **return** $url;
165. }
167. /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*
168. \* 微信OAUTH跳转指定URL
169. \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/
171. **public** **function** wxHeader($url){
172. header("location:".$url);
173. }
175. /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*
176. \* 微信通过OAUTH返回页面中获取AT信息
177. \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/
179. **public** **function** wxOauthAccessToken($code,$appId = NULL , $appSecret = NULL){
180. $appId = is\_null($appId) ? self::appId : $appId;
181. $appSecret = is\_null($appSecret) ? self::appSecret : $appSecret;
182. $url = "https://api.weixin.qq.com/sns/oauth2/access\_token?appid=".$appId."&secret=".$appSecret."&code=".$code."&grant\_type=authorization\_code";
183. $result = $this->wxHttpsRequest($url);
184. //print\_r($result);
185. $jsoninfo = json\_decode($result, true);
186. //$access\_token = $jsoninfo["access\_token"];
187. **return** $jsoninfo;
188. }
190. /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*
191. \* 微信通过OAUTH的Access\_Token的信息获取当前用户信息 // 只执行在snsapi\_userinfo模式运行
192. \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/
194. **public** **function** wxOauthUser($OauthAT,$openId){
195. $url = "https://api.weixin.qq.com/sns/userinfo?access\_token=".$OauthAT."&openid=".$openId."&lang=zh\_CN";
196. $result = $this->wxHttpsRequest($url);
197. $jsoninfo = json\_decode($result, true);
198. **return** $jsoninfo;
199. }
201. /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*
202. \* 生成随机字符串 - 最长为32位字符串
203. \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/
204. **public** **function** wxNonceStr($length = 16, $type = FALSE) {
205. $chars = "abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789";
206. $str = "";
207. **for** ($i = 0; $i < $length; $i++) {
208. $str .= substr($chars, mt\_rand(0, strlen($chars) - 1), 1);
209. }
210. **if**($type == TRUE){
211. **return** strtoupper(md5(time() . $str));
212. }
213. **else** {
214. **return** $str;
215. }
216. }
218. /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*
219. \* 微信商户订单号 - 最长28位字符串
220. \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/
222. **public** **function** wxMchBillno($mchid = NULL) {
223. **if**(is\_null($mchid)){
224. **if**(self::mchid == "" || is\_null(self::mchid)){
225. $mchid = time();
226. }
227. **else**{
228. $mchid = self::mchid;
229. }
230. }
231. **else**{
232. $mchid = substr(addslashes($mchid),0,10);
233. }
234. **return** date("Ymd",time()).time().$mchid;
235. }
237. /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*
238. \* 微信格式化数组变成参数格式 - 支持url加密
239. \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/
241. **public** **function** wxSetParam($parameters){
242. **if**(is\_array($parameters) && !empty**empty**($parameters)){
243. $this->parameters = $parameters;
244. **return** $this->parameters;
245. }
246. **else**{
247. **return** **array**();
248. }
249. }
251. /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*
252. \* 微信格式化数组变成参数格式 - 支持url加密
253. \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/
255. **public** **function** wxFormatArray($parameters = NULL, $urlencode = FALSE){
256. **if**(is\_null($parameters)){
257. $parameters = $this->parameters;
258. }
259. $restr = "";//初始化空
260. ksort($parameters);//排序参数
261. **foreach** ($parameters **as** $k => $v){//循环定制参数
262. **if** (null != $v && "null" != $v && "sign" != $k) {
263. **if**($urlencode){//如果参数需要增加URL加密就增加，不需要则不需要
264. $v = urlencode($v);
265. }
266. $restr .= $k . "=" . $v . "&";//返回完整字符串
267. }
268. }
269. **if** (strlen($restr) > 0) {//如果存在数据则将最后“&”删除
270. $restr = substr($restr, 0, strlen($restr)-1);
271. }
272. **return** $restr;//返回字符串
273. }
275. /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*
276. \* 微信MD5签名生成器 - 需要将参数数组转化成为字符串[wxFormatArray方法]
277. \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/
278. **public** **function** wxMd5Sign($content, $privatekey){
279. try {
280. **if** (is\_null($key)) {
281. **throw** **new** Exception("财付通签名key不能为空！");
282. }
283. **if** (is\_null($content)) {
284. **throw** **new** Exception("财付通签名内容不能为空");
285. }
286. $signStr = $content . "&key=" . $key;
287. **return** strtoupper(md5($signStr));
288. }
289. catch (Exception $e)
290. {
291. **die**($e->getMessage());
292. }
293. }
295. /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*
296. \* 微信Sha1签名生成器 - 需要将参数数组转化成为字符串[wxFormatArray方法]
297. \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/
298. **public** **function** wxSha1Sign($content, $privatekey){
299. try {
300. **if** (is\_null($key)) {
301. **throw** **new** Exception("财付通签名key不能为空！");
302. }
303. **if** (is\_null($content)) {
304. **throw** **new** Exception("财付通签名内容不能为空");
305. }
306. $signStr = $content . "&key=" . $key;
307. **return** strtoupper(sha1($signStr));
308. }
309. catch (Exception $e)
310. {
311. **die**($e->getMessage());
312. }
313. }
315. /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*
316. \* 将数组解析XML - 微信红包接口
317. \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/
319. **public** **function** wxArrayToXml($parameters = NULL){
320. **if**(is\_null($parameters)){
321. $parameters = $this->parameters;
322. }
324. **if**(!is\_array($parameters) || empty**empty**($parameters)){
325. **die**("参数不为数组无法解析");
326. }
328. $xml = "<xml>";
329. **foreach** ($arr **as** $key=>$val)
330. {
331. **if** (is\_numeric($val))
332. {
333. $xml.="<".$key.">".$val."</".$key.">";
334. }
335. **else**
336. $xml.="<".$key."><![CDATA[".$val."]]></".$key.">";
337. }
338. $xml.="</xml>";
339. **return** $xml;
340. }
342. }

后期还是会增加在一起的把这个CLASS做起来，网上资源很多，但是都是有一定基础的人去看看改改可以，对于没有接触刚刚接触的新手还是需要给予支持的。帮助用户屡屡思路。