

PHP 的常用工具函数

作者: [barryzpc](#)

原文链接: <https://ld246.com/article/1593743051754>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)



记录平时开发中经常使用的小工具

跳转链接

```
/**  
 * @param string $url 请求地址  
 */  
static public function location($url){  
    echo "<script language='javascript'>window.location.href='{$url}';</script>";  
}
```

请求链接获取数据

```
/**  
 * @author Barry  
 * @date 2017-10-24  
 * @param string $url 请求地址  
 * @param array $param 请求参数  
 * @param string $httpMethod 请求方法GET或者POST  
 * @return boolean  
 */  
static public function makeRequest($url, $param, $httpMethod = 'GET', $header = []) {  
    $oCurl = curl_init();  
    if (stripos($url, "https://") !== FALSE) {  
        curl_setopt($oCurl, CURLOPT_SSL_VERIFYPeer, FALSE);  
        curl_setopt($oCurl, CURLOPT_SSL_VERIFYHOST, FALSE);  
    }  
    if ($httpMethod == 'GET') {  
        curl_setopt($oCurl, CURLOPT_URL, $url . "?" . http_build_query($param));  
        curl_setopt($oCurl, CURLOPT_RETURNTRANSFER, 1);  
    }  
    $content = curl_exec($oCurl);  
    curl_close($oCurl);  
    return $content;  
}
```

```

} else {
    curl_setopt($oCurl, CURLOPT_URL, $url);
    curl_setopt($oCurl, CURLOPT_RETURNTRANSFER, 1);
    curl_setopt($oCurl, CURLOPT_POST, 1);
    curl_setopt($oCurl, CURLOPT_POSTFIELDS, http_build_query($param));
}

if (!empty($header)) {
    $headers = [];
    foreach ($header as $k=>$v){
        $headers[] = $k.":".$v;
    }
    curl_setopt($oCurl, CURLOPT_HTTPHEADER, $headers);
}

$sContent = curl_exec($oCurl);
// print_r($sContent);exit;
// $aStatus = curl_getinfo($oCurl);
// print_r($aStatus);die;
curl_close($oCurl);
return json_decode($sContent, true);
}

```

获取客户端IP地址

```

/**
 * 获取客户端IP地址
 */
static public function getClientIP() {
    static $ip = NULL;
    if ( $ip !== NULL )
        return $ip;
    if ( isset( $_SERVER['HTTP_X_FORWARDED_FOR'] ) ) {
        $arr = explode( ',', $_SERVER['HTTP_X_FORWARDED_FOR'] );
        $pos = array_search( 'unknown', $arr );
        if ( false !== $pos )
            unset( $arr[$pos] );
        $ip = trim( $arr[0] );
    } elseif ( isset( $_SERVER['HTTP_CLIENT_IP'] ) ) {
        $ip = $_SERVER['HTTP_CLIENT_IP'];
    } elseif ( isset( $_SERVER['REMOTE_ADDR'] ) ) {
        $ip = $_SERVER['REMOTE_ADDR'];
    }
    // IP地址合法验证
    $ip = ( false !== ip2long( $ip ) ) ? $ip : '0.0.0.0';
    return $ip;
}

```

循环创建目录

```

/**
 * @instruction 循环创建目录
 * @param $dir 目录

```

```

* @param $mode 权限
* @author Barry
* @date 2017-10-24
* @version v1.0
*/
static public function mkdir( $dir, $mode = 0777 ) {

    $aimUrl = str_replace( ' ', '/', $dir );
    $aimDir = '';
    $arr = explode( '/', $aimUrl );
    $result = true;
    foreach ( $arr as $str ) {
        $aimDir .= $str . '/';
        if ( !file_exists( $aimDir ) ) {
            $result = mkdir( $aimDir, $mode );
        }
    }
    return $result;
}

```

字符截取，多出的字符串显示省略号

```

/**
 * 字符截取
 *
 * @param $string
 * @param $length
 * @param $dot
 */
public static function cutstr( $string, $length, $dot = '...', $charset = 'utf-8' ) {
    if ( strlen( $string ) <= $length )
        return $string;

    $pre = chr( 1 );
    $end = chr( 1 );
    $string = str_replace( array( '&' , '"' , '<' , '>' ), array( $pre . '&' . $end , $re . '"' . $end , $pre . '<' . $end , $pre . '>' . $end ), $string );

    $strcut = '';
    if ( strtolower( $charset ) == 'utf-8' ) {

        $n = $tn = $noc = 0;
        while ( $n < strlen( $string ) ) {

            $t = ord( $string[$n] );
            if ( $t == 9 || $t == 10 || ( 32 <= $t && $t <= 126 ) ) {
                $tn = 1;
                $n++;
                $noc++;
            } elseif ( 194 <= $t && $t <= 223 ) {
                $tn = 2;
                $n += 2;
                $noc += 2;
            } elseif ( 224 <= $t && $t <= 239 ) {
                $tn = 3;
                $n += 3;
                $noc += 3;
            } else {
                $tn = 1;
                $n++;
                $noc++;
            }
            $strcut .= substr( $string, $n - $tn, 1 );
        }
    }
    return $strcut . $dot . substr( $string, -$length - 3 );
}

```

```

$tn = 3;
$n += 3;
$noc += 2;
} elseif ( 240 <= $t && $t <= 247 ) {
    $tn = 4;
    $n += 4;
    $noc += 2;
} elseif ( 248 <= $t && $t <= 251 ) {
    $tn = 5;
    $n += 5;
    $noc += 2;
} elseif ( $t == 252 || $t == 253 ) {
    $tn = 6;
    $n += 6;
    $noc += 2;
} else {
    $n++;
}

if ( $noc >= $length ) {
    break;
}

}

if ( $noc > $length ) {
    $n -= $tn;
}

$strcut = substr( $string, 0, $n );

} else {
    for ( $i = 0; $i < $length; $i ++ ) {
        $strcut .= ord( $string[$i] ) > 127 ? $string[$i] . $string[++ $i] : $string[$i];
    }
}

$strcut = str_replace( array ( $pre . '&' . $end , $pre . '"' . $end , $pre . '<' . $end , $pre . '>' . $end ), array ( '&' , '"' , '<' , '>' ), $strcut );

$pos = strrpos( $strcut, chr( 1 ) );
if ( $pos !== false ) {
    $strcut = substr( $strcut, 0, $pos );
}
return $strcut . $dot;
}

```

获得来源类型 post get

```

static public function method() {
    return strtoupper( isset( $_SERVER['REQUEST_METHOD'] ) ? $_SERVER['REQUEST_METHOD'] : 'GET' );
}

```

转换XML文档为数组

```
/**  
 * 转换XML文档为数组  
 *  
 * @author Barry  
 * @date 2017-10-24  
 * @param string xml内容  
 * @return mixed 返回的数组, 如果失败, 返回false  
 */  
static public function xmlWx2array($xml) {  
    $xml = (Array)simplexml_load_string($xml, "SimpleXMLElement", LIBXML_NOCDATA);  
    return $xml;  
}
```

正则验证url是否合法

```
static public function checkUrl($url){  
    if(!preg_match('/^(http|https|ftp):\/\/[\w.]+[\w\.\-]*[\w.]*\?[\w=&\+%\%]*$/is',$url)){  
        return false;  
    }  
    return true;  
}
```

数组转xml

```
public function arrayToXml($arr) {
    $xml = "<xml>";
    foreach ($arr as $key=>$val) {
        if (is_numeric($val)) {
            $xml.= "<".$key.">".$val."</".$key.">";
        }
        else {
            $xml.= "<".$key."><![CDATA[\".$val.\"]]></".$key.">";
        }
    }
    $xml.= "</xml>";
    //echo $xml;
    return $xml;
}
```

二维数组排序

```
/**\n * 二维数组排序\n * <B>说明: </B>\n * <pre>\n * 略\n *</pre>\n * <B>示例: </B>\n * <pre>\n * $array = array(\n
```

```

*      array('userId'=>'3','name'=>'admin3'),
*      array('userId'=>'1','name'=>'admin1'),
*      array('userId'=>'2','name'=>'admin2')
*)
* ArrayList::arraySort($array,'userId','desc');
* 结果
*      * $array = array(
*          array('userId'=>'3','name'=>'admin3'),
*          array('userId'=>'2','name'=>'admin2')
*          array('userId'=>'1','name'=>'admin1'),
*)
* </pre>
* @param array $arr 源数组
* @param string $key 数据项key
* @param string $order 排序规则 desc 降序 asc 升序
* @return int
*/
static public function arraySort($arr, $key, $order = 'desc') {
    $keysvalue = $new_array = array();
    foreach ($arr as $k => $v) {
        $keysvalue[$k] = $v[$key];
    }

    if ($order == 'asc') {
        asort($keysvalue);
    } else {
        arsort($keysvalue);
    }

    reset($keysvalue);
    foreach ($keysvalue as $k => $v) {
        $new_array[$k] = $arr[$k];
    }
}

return $new_array;
}

```

判断是否是微信客户端请求

```

static public function isWeixin() {
    if ( strpos($_SERVER['HTTP_USER_AGENT'], 'MicroMessenger') !== false ) {
        return true;
    }
    return false;
}

```

计算两组经纬度坐标之间的距离

```

/**
 * 计算两组经纬度坐标 之间的距离
 * @params $lat1 纬度1;
 * @params $lng1 经度1;
 * @params $lat2 纬度2;

```

```

* @params $lng2 经度2;
* @params $len_type (1:m or 2:km);
* @return m or km
* @version 1.0
*/
public static function getDistance($lat1, $lng1, $lat2, $lng2, $len_type = 1, $decimal = 2) {
    $EARTH_RADIUS=6378.137;
    $PI=3.1415926;
    $radLat1 = $lat1 * $PI / 180.0;
    $radLat2 = $lat2 * $PI / 180.0;
    $a = $radLat1 - $radLat2;
    $b = ($lng1 * $PI / 180.0) - ($lng2 * $PI / 180.0);
    $s = 2 * asin(sqrt(pow(sin($a/2),2) + cos($radLat1) * cos($radLat2) * pow(sin($b/2),2)));
    $s = $s * $EARTH_RADIUS; $s = round($s * 1000);
    if ($len_type > 1) {
        $s /= 1000;
    }
    return round($s,$decimal);
}

```

将时间戳显示为刚刚、几分钟前、几小时前、昨天、前天等

```

/**
 * 时间格式化 传入时间戳~~~~
 */
static public function formatDate($time){
    $rtime = date ( "m-d H:i", $time );
    $htime = date ( "H:i", $time );

    $time = time () - $time;

    if ($time < 60) {
        $str = '刚刚';
    } elseif ($time < 60 * 60) {
        $min = floor ( $time / 60 );
        $str = $min . '分钟前';
    } elseif ($time < 60 * 60 * 24) {
        $h = floor ( $time / (60 * 60) );
        $str = $h . '小时前' . $htime;
    } elseif ($time < 60 * 60 * 24 * 3) {
        $d = floor ( $time / (60 * 60 * 24) );
        if ($d == 1)
            $str = '昨天' . $rtime;
        else
            $str = '前天' . $rtime;
    } else {
        $str = $rtime;
    }
    return $str;
}

```

求两个日期之间相差的天数

```

/***
 * 求两个日期之间相差的天数
 * @param string $datetime1
 * @param string $datetime2
 * @return number
 */
static public function diffBetweenTwoDays($datetime1, $datetime2) {
    $second1 = strtotime($datetime1);
    $second2 = strtotime($datetime2);

    if ($second1 < $second2) {
        $tmp = $second2;
        $second2 = $second1;
        $second1 = $tmp;
    }
    return bcdiv(bcsub($second1, $second2, 3), 86400, 3);
}

```

根据生日计算年龄

```

static public function Birthday($birthday) {
    $age = $birthday;
    if($age === false) {
        return false;
    }
    list($y1,$m1,$d1) = explode("-",date("Y-m-d",$age));
    $now = strtotime("now");
    list($y2,$m2,$d2) = explode("-",date("Y-m-d",$now));
    $age = $y2 - $y1;
    if((int)($m2.$d2) < (int)($m1.$d1))
        $age -= 1;
    return $age;
}

```

秒转化成时分秒显示

```

/**秒转化成时分秒
 * @author Barry
 * @date 2017-10-24
 * @param $seconds
 * @return string
 */
static public function changeTimeType($seconds){
    if ($seconds > 3600){
        $hours = intval($seconds/3600);
        $minutes = $seconds % 3600;
        $time = $hours.":".gmstrftime('%M:%S', $minutes);
    }else{
        $time = gmstrftime('%H:%M:%S', $seconds);
    }
    return $time;
}

```

删除一个文件夹下面的所有文件以及文件夹

```
//删除一个文件夹下面的所有文件以及文件夹
static public function delDir($path){
    error_reporting(0);
    //如果是目录则继续
    if(is_dir($path)){
        //扫描一个文件夹内的所有文件夹和文件并返回数组
        $p = scandir($path);
        foreach($p as $val){
            //排除目录中的.和..
            if($val != "." && $val != ".."){
                //如果是目录则递归子目录，继续操作
                if(is_dir($path.$val)){
                    //子目录中操作删除文件夹和文件
                    self::deldir($path.$val '/');
                    //目录清空后删除空文件夹
                    @rmdir($path.$val '/');
                }else{
                    //如果是文件直接删除
                    @unlink($path.$val);
                }
            }
        }
        @rmdir($path);
    }
}
```

PHP模拟发送异步请求

```
/**
 * PHP发送异步请求
 * @author Barry
 * @date 2017-10-24
 * @param string $url 请求地址
 * @param array $param 请求参数
 * @param string $httpMethod 请求方法GET或者POST
 * @param array $header
 * @return boolean
 */
static public function makeAsyncRequest($url, $param, $httpMethod = 'GET', $header = []) {
    $oCurl = curl_init();
    if (stripos($url, "https://") !== FALSE) {
        curl_setopt($oCurl, CURLOPT_SSL_VERIFYPeer, FALSE);
        curl_setopt($oCurl, CURLOPT_SSL_VERIFYHOST, FALSE);
    }
    if ($httpMethod == 'GET') {
        curl_setopt($oCurl, CURLOPT_URL, $url . "?" . http_build_query($param));
        curl_setopt($oCurl, CURLOPT_RETURNTRANSFER, 1);
    } else {
        curl_setopt($oCurl, CURLOPT_URL, $url);
        curl_setopt($oCurl, CURLOPT_RETURNTRANSFER, 1);
        curl_setopt($oCurl, CURLOPT_POST, 1);
    }
}
```

```

        curl_setopt($oCurl, CURLOPT_POSTFIELDS, http_build_query($param));
    }

    if (!empty($header)) {
        $headers = [];
        foreach ($header as $k=>$v){
            $headers[] = $k.":".$v;
        }
        curl_setopt($oCurl, CURLOPT_HTTPHEADER, $headers);
    }
    curl_setopt($oCurl, CURLOPT_NOSIGNAL, 1);//必须加，否则下面设置毫秒级会报错
    curl_setopt($oCurl, CURLOPT_TIMEOUT_MS, 18);//设置cURL允许执行的最长毫秒数

    $sContent = curl_exec($oCurl);
//    print_r($sContent);exit;
//    $aStatus = curl_getinfo($oCurl);
    curl_close($oCurl);
    return json_decode($sContent, true);
}

```

隐藏手机号中间4位

```

function blur_phone_number($phoneNum)
{
    $head = substr($phoneNum, 0, 3);
    $tail = substr($phoneNum, -4, 4);
    return ($head . '****' . $tail);
}

```

取毫秒级时间戳

```

//取毫秒级时间戳
function getMillisecond() {
    list($t1, $t2) = explode(' ', microtime());
    return (float)sprintf('%.0f',(floatval($t1)+floatval($t2))*1000);
}

```