**OSS PHP SDK使用教程 )­T5h\ZO`;**

最近看论坛里好多朋友在问OSS PHP SDK的使用问题。今天有空，把之前使用OSS PHP SDK的经验和大家分享下。由于小弟水平有限，难免会有不足的地方，不足的地方请大家不吝赐教。那我们就开始漫长的SDK使用之旅吧。 '/]fZ|   
XO8 H]   
第一步 ： 去OSS官方注册账户，并且开通OSS服务，当OSS服务开通后，进入管理中心获取相应的API密钥。如下截图： U:T5o]P<   
图片:GET\_API\_KEY.jpg

b5iJ  
m-]#WX|0''^   
第二步： 在获得API密钥后，获取OSS PHP SDK，从官方下载压缩包，解压后，我们会看到一个conf.inc.php的文件，把我们第一步获得的API密钥填入相应的位置，代码如下： 5  
p(t")   
//个人测试 AyI}LQm]u   
//ACCESS\_ID gE]a\*TOZk   
define('OSS\_ACCESS\_ID', ‘这里填入ACCESS ID’); ‑joSr,'x   
^LQ lfd   
//ACCESS\_KEY 1v8:,!C   
define('OSS\_ACCESS\_KEY', '这里填入ACCESS KEY'); R}K5'`[%ZY   
\_dRn0<#1(k   
第三步：当完成以上两步以后，我们就可以开始我们神奇的OSS PHP SDK使用之旅了，接下来我们会详细介绍每个API的使用方法，并且提供详细的测试代码以及文字说明 dwsy(g7   
Q& un­A3   
1. 加载SDK 并且初始化相关参数 3YZ3fhpw   
/\*\* |M{,}.\*CU   
\* 加载sdk包以及错误代码包 tMs|UC [.yx2@W

|  |
| --- |
| \*/ <:\_]Yl  require\_once '../sdk.class.php'; |

nms<6kfzL   
//实例化OSS Service +~v3D^L1­5 3=eGS

|  |
| --- |
| $oss\_sdk\_service = new ALIOSS(); |

KE|u}M@v6   
//设置是否打开curl调试模式,该模式主要是为了方便调试使用，可以设置TRUE|FALSE Ee?K|\_\${ -uY:2

|  |
| --- |
| $oss\_sdk\_service->set\_debug\_mode(FALSE); |

r‑^  
Rcjyc1   
2. list\_bucket 方法使用 T\_|fb)G+{   
/\*\* `^4>^   
\* 获取bucket列表,该方法主要获取该账户下所有的bucket列表 G^eFS;   
\* 使用方法如下： 8  
~L.6c5U ‑l`w|o

|  |
| --- |
| \* $bucket\_list = $oss\_sdk\_service->list\_bucket(); (;S]{z%  \*/ pU)3\*9?cIl  $bucket\_list = $oss\_sdk\_service->list\_bucket(); uu46'aT  print\_r($bucket\_list);die(); |

%:8q7P­N|   
3. get\_bucket\_acl方法使用 /3:­IE%o   
/\*\* >C d&K9H   
\* 获得指定bucket的ACL )-Sl‑/G   
\* 调用方法如下： HFCFEamBMP

|  |
| --- |
| \* $get\_bucket\_acl = $oss\_sdk\_service->get\_bucket\_acl($bucket,$options); |

>j~70 ?   
\* 其中需要传入的参数为 某一$bucket的名称,$options为一个关联数组，该方法可以传入Content-Type,使用系统内置的定义方式 4O,a`:d1$6 aNf3 R;\*

|  |
| --- |
| \*/ RgzSaP;;  $bucket = 'php-sdk-1330835674'; U JRT4>G  $options = array( ^xu)~:} i  ALIOSS::OSS\_CONTENT\_TYPE => 'text/xml', \(7#N<-  ); #i#.tc  $get\_bucket\_acl = $oss\_sdk\_service->get\_bucket\_acl($bucket,$options); LCm}v&~%A  print\_r($get\_bucket\_acl);die(); |

YyBq+6nq5   
4. set\_bucket\_acl方法使用 MM)/B>cQt   
/\*\* N#-\­JlJ)   
\* 设置指定bucket的ACL,目前只有三种acl private,public-read,public-read-write ‑DB=  
cc   
\* 调用方法如下： &=t(NI$

|  |
| --- |
| \* $set\_bucket\_acl = $oss\_sdk\_service->set\_bucket\_acl($bucket, $acl); |

~w}[ .\_'#M   
\* 其中需要指定$bucket,以及$acl的值,$acl的取值只能是private,public-read,public-read-write其中之一,$acl使用系统内置的定义方式 xM"XNT6b   
\* 对应的值为${:$jX[

|  |
| --- |
| private => ALIOSS::OSS\_ACL\_TYPE\_PRIVATE ,public-read=>  ALIOSS::OSS\_ACL\_TYPE\_PUBLIC\_READ ,public-read-write=> ALIOSS::OSS\_ACL\_TYPE\_PUBLIC\_READ\_WRITE K O­HH74}\_  \*/ ^R­DXX+  $bucket = 'php-sdk-1329818657'; z\_0lMX`  $acl = ALIOSS::OSS\_ACL\_TYPE\_PUBLIC\_READ; +P;D}1B#I?  $set\_bucket\_acl = $oss\_sdk\_service->set\_bucket\_acl($bucket, $acl); &7nfTc  print\_r($set\_bucket\_acl);die(); |

z:,!yU c e<wj5:M|‑   
5. create\_bucket方法使用 \_BA\_lkN+D   
/\*\* "TNUw&ih   
\* 创建bucket w)>z3Lm   
\* 调用方法如下： L|'ME| '   
\* $create\_bucket = $oss\_sdk\_service->create\_bucket($bucket, $acl); vF\zZ<R/   
\* 其中需要传入$bucket,$acl是可选的，如果不指定ACL，则默认会是private, xE[tD? M{   
\*$acl的取值只能是private,public-read,public-read-write其中之一,$acl使用系统内置的定义方式 <"P '"SC   
\* 对应的值为:QSW^x Q]:%Jj2

|  |
| --- |
| private => ALIOSS::OSS\_ACL\_TYPE\_PRIVATE ,public-read=>  ALIOSS::OSS\_ACL\_TYPE\_PUBLIC\_READ ,public-read-write=> ALIOSS::OSS\_ACL\_TYPE\_PUBLIC\_READ\_WRITE +>%+r  \*/ SJr­:  $bucket = 'oss-php-sdk-'.time(); &H>dE]Hq,  $acl = ALIOSS::OSS\_ACL\_TYPE\_PUBLIC\_READ; 8"rX;5 vP  $create\_bucket = $oss\_sdk\_service->create\_bucket($bucket, $acl);  eBKIdR%k  print\_r($create\_bucket);die(); |

QQ8W;x   
6. delete\_bucket方法使用 }iloX#   
/\*\* 6b)UoJxj   
\* 删除bucket 6-YR'ikU   
\* 调用方法如下： LX&P]{q  
KS   
\* $delete\_bucket = $oss\_sdk\_service->delete\_bucket($bucket); a‑[rUU'8   
\* 需要传入$bucket参数,其中$bucket内不能有内容，如果有内容则本次删除请求不会成功 v%Q7\X(   
\*/ n?tAa|\_ Jjq%cA

|  |
| --- |
| $bucket = 'oss-php-sdk-1330928803'; ;a­)\5Uy  $delete\_bucket = $oss\_sdk\_service->delete\_bucket($bucket); ,99G2E v4c  print\_r($delete\_bucket);die(); |

4!qDG+m   
7. list\_object方法使用 |O2PcYNu   
/\*\* .?j8{>‑   
\* 获得bucket下的object列表 Zi)8KO[/0   
\* 调用方法如下： P9^h>sV   
\* $list\_object = $oss\_sdk\_service->list\_object($bucket,$options); Ee&$9 )t   
\* 其中需要传入$bucket参数,$options为可选参数，如果需要传入$options,必须为数组，其中数据的key必须为指定的，否则参数无效 Q7­%#3ML   
\* $options = array( l d9#4D[#   
\* 'max-keys' => max-keys用于限定此次返回object的最大数，如果不设定，默认为100，max-keys取值不能大于100。 :W.(,65c   
\* 'prefix' => 限定返回的object key必须以prefix作为前缀。注意使用prefix查询时，返回的key中仍会包含prefix。 va;wQ~&   
\* 'delimiter' => 是一个用于对Object名字进行分组的字符。所有名字包含指定的前缀且第一次出现delimiter字符之间的object作为一组元素 Nf'9]I   
\* 'marker' => 用户设定结果从marker之后按字母排序的第一个开始返回。 YR.'JF`C   
\* ) Q(3x"+   
\* 其中 prefix，marker用来实现分页显示效果，参数的长度必须小于256字节。 5al{[mi Vba}RF[b

|  |
| --- |
| \*/ N8vl< Mq  $bucket = 'php-sdk-1330835674'; ;N|>pSzmL  $options = array( W'$~mK\  'delimiter' => '/', 6u}NI!he  'prefix' => '', -9"Ls?Cu  'max-keys' => 10, (?[cDw/{J:  'marker' => 'myobject-1330850469.pdf', D ]eF3a.G  ); 1gJ!!SHPo  $list\_object = $oss\_sdk\_service->list\_object($bucket,$options); -y( V-  print\_r($list\_object);die(); |

^S|^1   
8. get\_object方法使用 jw[`\h}8   
/\*\* ]FBfh.#X@   
\* 获得$bucket下的某个object,$object为文件，不能为目录 A9gl|II   
\* 调用方法如下： 8<xy\*=%

|  |
| --- |
| \* $get\_object = $oss\_sdk\_service->get\_object($bucket,$object,$options); |

V‑p$ckr   
\* 其中需要传入$bucket,$object参数，参数为必须,$options为可选参数，是一个key=>value格式的关联数组，其中的key定义如下 VJaL$Wv)H   
\* $options = array( L)5nb-qp   
\* 'lastmodified' => 设置请求header 'If-Modified-Since' urB.K<5ZA   
\* 'etag' => 设置请求header Etag < lrw7T  
   
\* 'range' => 设置请求header Range,主要用来实现断点下载 +h/$\_5   
\* 'fileDownload' => 用户写入本地文件，需要有写权限 L8  
J/GVmj T9^i#8-^‑

|  |
| --- |
| \* ) 4`F\*] Ft  \*/ 5;l\_-0=  $bucket = 'php-sdk-1329818657'; Zn//u<D  $object = 'myobject-pdf-1331429369.pdf'; .y!Hw{cq  $filepath = "D:\\download\\myobject-".time().".pdf"; 5 \J;EWTU  $options = array( 2H#vA  'fileDownload' => $filepath, .8m)^ET  ); 7d\*<'k]{,  $get\_object\_response = $oss\_sdk\_service->get\_object($bucket,$object,$options); AVjtK  print\_r($get\_object\_response);die(); |

,3j\*D+   
9. create\_object\_dir方法使用 U#}.r<   
/\*\* s"/8h#!zv   
\* 创建文件夹(是虚拟文件夹) :FSkXe2yy0   
\* 调用方法如下： $4"OD"Z Cq   
\* $create\_object\_dir = $oss\_sdk\_service->create\_object\_dir($bucket,$object); \Mi< ROp5   
\* 其中需要传入$bucket,$object参数，参数为必须 O\w%E@9Fh   
\*/ a\_yV\*N`D UnPSJ]VW

|  |
| --- |
| $bucket = 'efrwerwertyrty'; LHz{\*`22q  $object = '我的OSSbucket-'.time(); ~"N]%Cu  $create\_object\_dir = $oss\_sdk\_service->create\_object\_dir($bucket,$object); R&QT 'i  print\_r($create\_object\_dir);die(); |

]~P?   
10. upload\_file\_by\_content方法使用 ­,Y!)V   
/\*\* q)[gVL   
\* 通过http body上传文件,适用于直接写入内容的上传，比较小的文件 2I7P}=   
\* 调用方法如下： #/:[ho{JQ

|  |
| --- |
| \* $upload\_file\_by\_content = $oss\_sdk\_service->upload\_file\_by\_content($bucket,$object,$upload\_file\_options); |

K6X1a7   
\* 其中的$bucket,$object以及$upload\_file\_options为必选参数，$upload\_file\_options必须为数组，且key必须为规定的值，否则会上传失败, H{BjxZ~)   
\* $object是文件名称，如果上传的文件不是直接位于bucket下，而是位于某一子目录下，则$object = 'dir\_name/dir\_name/file\_name' :\[F=   
\* 其中的content 为文件的内容，$length为文件的大小 w-2]69‑$k jsR1jou6

|  |
| --- |
| \* $upload\_file\_options = array( ^:],JN k  \* 'content' => $content, Rg6>6.fk\*  \* 'length' => $length, c(jA"K[|b  \* ); SnqLF /d  \*/ 4c% :?H@2  $bucket = 'php-sdk-1328520184'; jmIP c3O0  $object = 'oss-file-name-'.time().'.txt'; Eq?o/'e  $content = ''; h^s}8y  for($i = 0;$i<= 10000;$i ){ 'Zzm'pC  $content .= $i."\n"; R Q 8okA  } mF1oY[xa\_  $upload\_file\_options = array( \*8N~Zmz  'content' => $content, x .q‑­%O1  'length' => strlen($content), 2{&" 3dq  ); Gc'H ‑F"w  $upload\_file\_by\_content = $oss\_sdk\_service->upload\_file\_by\_content($bucket,$object,$upload\_file\_options); VltWY'\Wu;  print\_r($upload\_file\_by\_content);die(); )Q8Q#S |

CtfSfSAUuu   
11. upload\_file\_by\_file 方法使用 Qhr:d`@^]   
/\*\* iqU}t2vFrj   
\* 通过文件方式上传,适合小文件上传，大文件上传请使用multipart u/zC$L3B( \DHCf  
4,

|  |
| --- |
| \*/ +)|2$$m  $bucket = 'php-sdk-1328520898'; ,9rT|:N  $file = "d:\\ccc.pdf"; 文件路径 YmwVa s  $object = 'cpp.how.to.program-'.time().'.pdf'; //object名称 ?YS`?Rr  $response\_upload\_file\_by\_file = $oss\_sdk\_service->upload\_file\_by\_file($bucket,$object,$file); {T IGPK  print\_r($response\_upload\_file\_by\_file);die(); |

iTX.?‑\*   
12. copy\_object 方法使用 UJZa1p@L   
/\*\* pM=vW{"I/   
\* Copy Object HDaec`j   
\* 调用方法如下： l]ZUKy   
\* $copy\_object\_result = $oss\_sdk\_service->copy\_object($from\_bucket, $from\_object,$to\_bucket, $to\_object)   
\)v.dQ!   
\* 其中需要传入$from\_bucket,$from\_object,$to\_bucket,$to\_object等参数，均不能为空 =19]a wO"Q{oi+ xs:{%ki

|  |
| --- |
| \*/ B|Omz:c  $from\_bucket = 'aaaaaaaaaaa'; 9\xw}ph  $to\_bucket = 'efrwerwertyrty'; ^\_oLhNoez2  $from\_object = 'CreateLaunchConfigurationRequest.cs'; A=])pYE1  $to\_object = 'copy-object-'.time().'.cs'; -L[K1;Xv"  $copy\_object\_result = $oss\_sdk\_service->copy\_object($from\_bucket, $from\_object,$to\_bucket, $to\_object); +vuW 9  print\_r($copy\_object\_result);die(); |

Ax\Fg 5   
13. get\_sign\_url方法使用 k >MgrtJI   
/\*\* 81g&WQ‑'   
\* 获取带签名的外链URL ZibODs=f;   
\* 调用方法如下： \kRBJ1)|f   
\* $get\_sign\_url = $oss\_sdk\_service->get\_sign\_url($bucket,$object,$timeout) QXrK-&fj­u   
\* 其中需要传入$bucket,$object不能为空，$timeout为过期的秒数，从当前时间往后多少秒过期，若不传，则使用默认值60 =SdWU}xn2 ‑uZ^i8;i

|  |
| --- |
| \*/ P~#jvm!  $bucket = 'php-sdk-1330835674'; c:­iMbJOn#  $object = 'Cpp.How.to.Program.7-1330855122.pdf'; Mqpo S  $timeout = 3600; $ p0s  $get\_sign\_url = $oss\_sdk\_service->get\_sign\_url($bucket,$object,$timeout); `+gF|o9  print\_r($get\_sign\_url);die(); |

lC2xl(#!  
   
14. is\_object\_exist方法使用 AE$)RhY`   
/\*\* zb4g\H 0   
\* 检测Object是否存在 spI{d!c   
\* 调用方法如下： D3Mce|t^   
\* $is\_object\_exist = $oss\_sdk\_service->is\_object\_exist($bucket, $object); ­<Du\*Re6g   
\* 其中需要传入$bucket,$object参数，参数为必须 07#!b~N @Q^;qMy

|  |
| --- |
| \*/ '-x%?Ll  $bucket = 'php-sdk-1330835674'; xq{4i|d)  $object = 'Cpp.How.to.Program.7-1330855122.pdf'; A g­‑Pg0(G  $is\_object\_exist = $oss\_sdk\_service->is\_object\_exist($bucket, $object); ks;%f34  var\_dump($is\_object\_exist?'Exist':'Not Exist');die(); |

Jq!($PdA   
15 . get\_object\_url方法使用 e`oc#Od&x]   
/\*\* `=\*svrmS   
\* 获得某一个object的URL i!iODt3k   
\* 调用方法如下： -K!-a'J   
\* $get\_object\_url = $oss\_sdk\_service->get\_object\_url($bucket, $object); E%e-R6gl   
\* 其中需要传入$bucket,$object参数，参数为必须,$object为文件 !d^`­YEfE

|  |
| --- |
| \*/ M]'AA Uo8  $bucket = 'php-sdk-1330835674'; ­Y­U`{  $object = 'Cpp.How.to.Program.7-1330855122.pdf'; ..t=Y#  $get\_object\_url = $oss\_sdk\_service->get\_object\_url($bucket, $object); RoTT%c P\_  var\_dump($get\_object\_url?$get\_object\_url:'Not Exist');die(); |

J]UH  
q$B ,?qS#B+>   
16. get\_object\_meta方法使用 ^#o.WL%4/B   
/\*\* c>g%oE   
\* 获得object的meta #59zv=   
\* 调用方法如下： nAzr!$qbNv   
\* $get\_object\_meta = $oss\_sdk\_service->get\_object\_meta($bucket, $object); D  
H^^$)   
\* 其中需要传入$bucket,$object参数，参数为必须,如果object为文件夹，则需要添加'/' fCfY.vd5

|  |
| --- |
| \*/ Ie~~L‑U  $bucket = 'php-sdk-1330835674';  0#JBz\  $object = 'Cpp.How.to.Program.7-1330855122.pdf'; - \QtE}|4  $get\_object\_meta = $oss\_sdk\_service->get\_object\_meta($bucket, $object); eBB D9SI  print\_r($get\_object\_meta);die(); |

FuKNH~MevQ eG v"&kr   
17. delete\_object方法使用 NEk[0   
/\*\* 9E^piLA   
\* 删除object {>Px.%[<   
\* 调用方法如下： if'4­MDl   
\* $delete\_object = $oss\_sdk\_service->delete\_object($bucket,$object); =X\*E(.6Ip   
\* 其中需要传入$bucket,$object参数，如果object为文件，则会直接删除，如果为文件夹，若文件夹不为空，则不能删除 7h2bL6Y88

|  |
| --- |
| \*/ =m:W  $bucket = 'php-sdk-1330835674'; 0"`skYJ@  $object = 'Cpp.How.to.Program.7-1330855122.pdf'; 2. \_cEY34‑  $delete\_object = $oss\_sdk\_service->delete\_object($bucket,$object); F?B=:8,}  print\_r($delete\_object);die(); |

St`m52V(5X q4{tH   
18. delete\_objects方法使用 .Dm{mV@\*T   
/\*\* @XmkIm   
\* 批量删除某一bucket下的objects Xhi?b|   
\* 调用方法如下： t3G'x1

|  |
| --- |
| \* $delete\_objects\_response = $oss\_sdk\_service->delete\_objects($bucket,$objects,$options); |

%\*OJRL`   
\* 其中需要传入$bucket,$object参数，如果objects为bucket下的object数组 M6iKl   
\* $options 可以设置返回模式，目前提供两种返回模式：详细(verbose)模式和简单(quiet)模式，默认为verbose模式,模式的设置如下 ­$}$@)!-   
\* $options = array('quite'=>true|false),true 为quiet模式，false为verbose模式 Xdt+\}\

|  |
| --- |
| \*/ @ni~ij  $bucket = 'myossbucket002'; V{{x~Q9  $objects = array( 27:x5g?  'AliUpdate.dll','libGLESv2.dll' ;\+0H$  ); ‑G.8b\E~  =\*jcO119L  $options = array( LS'­=>s"  'quiet' => true, U@nwSfp:G  ); aehMLl9cl  $delete\_objects\_response = $oss\_sdk\_service->delete\_objects($bucket, $objects,$options); sA}R!  print\_r($delete\_objects\_response);die(); |

@XC97kGWp n~j[­P­w   
19. create\_object\_group方法使用 ISew]R2   
/\*\* c­ ?‑XUb[   
\* 创建object group Z"X\*FzFo   
\* 调用方法如下： G`"Cqs<

|  |
| --- |
| \* $create\_object\_group = $oss\_sdk\_service->create\_object\_group($bucket,$object\_group,$object\_group\_array); |

"6Uj:9   
\* 其中参数$object\_group,$bucket,$object\_group\_array不为空，且$object\_group\_array内的object必须位于同一bucket下 1o>R\g3

|  |
| --- |
| \*/ oE1]vX  $bucket = 'php-sdk-1330835674'; D|9fHMg%  $object\_group = 'object-group-'.time(); :'~ gLW>j  $object\_group\_array = array( uFZB8+  'Cpp.How.to.Program.7-1330855122.pdf', g/p9"eBpq  'copy-object-1330929729.pdf', !U:‑s.^{  'hh\_axur-setup.pdf', XWpnZFjE  ); p]Zabky   twmJ  $create\_object\_group = $oss\_sdk\_service->create\_object\_group($bucket,$object\_group,$object\_group\_array); K-C-+RB  print\_r($create\_object\_group);die(); |

+  
t4m\/y i0Q \_f!j   
20. get\_object\_group方法使用 8p‑{   
/\* @n\*‑  
D>g   
\* 获取object group B@inH]wq   
\* 调用方法如下： &xroms"S=   
\* $get\_object\_group = $oss\_sdk\_service->get\_object\_group($bucket,$object\_group); iLcadX   
\* 其中参数$object\_group,$bucket为必须参数 oh0|2IrM

|  |
| --- |
| \*/ )+4}‑Ix/q  $bucket = 'php-sdk-1330835674'; ,­9wenr  $object\_group = 'object-group-1330930277'; % 3-\3qx\*  $get\_object\_group = $oss\_sdk\_service->get\_object\_group($bucket,$object\_group); Ej09RO"pB  print\_r($get\_object\_group);die(); |

:;q\_f+U :\*:fun   
21. get\_object\_group\_index方法使用 /3  
Ix,7   
/\*\* 'O2#1SWe   
\* 获取object group index ZQ-`‑l:G   
\* 调用方法如下： ^@lg5d3F

|  |
| --- |
| \* $get\_object\_group\_index = $oss\_sdk\_service->get\_object\_group\_index($bucket,$object\_group); |

?J@qg20z   
\* 其中参数$object\_group,$bucket为必须参数 vq:O‑H H a05:iFoJ

|  |
| --- |
| \*/ G4%M$LJ h  $bucket = 'php-sdk-1330835674'; \*a,.E6C\*  $object\_group = 'object-group-1330930277'; GL =XiBt  $get\_object\_group\_index = $oss\_sdk\_service->get\_object\_group\_index($bucket,$object\_group); qL5#.bR  print\_r($get\_object\_group\_index);die(); |

eO  
PCYyN   
22. get\_object\_group\_meta方法使用 gq\_7\_Y/   
/\*\* >l< ~Z;   
\* 获取object group meta (<R\   
\* 调用方法如下： P,;b'-5C

|  |
| --- |
| \* $get\_object\_group\_meta = $oss\_sdk\_service->get\_object\_group\_meta($bucket,$object\_group); |

NQiecxvt=   
\* 其中参数$object\_group,$bucket为必须参数 T  
~~[a|bLa a}qse5Fr !P60[\*>

|  |
| --- |
| \*/ MRg\FR 2>1  $bucket = 'php-sdk-1330835674'; `Ivw`}L  $object\_group = 'object-group-1330930277'; =#Cf5s6qt  $get\_object\_group\_meta = $oss\_sdk\_service->get\_object\_group\_meta($bucket,$object\_group); 4 ^~zN"6]  print\_r($get\_object\_group\_meta);die(); |

‑ J"Y  
  
   
23. delete\_object\_group方法使用 E7NbPNd   
/\*\* ZCE%‑38E N   
\* 删除object group q"LJwV}W   
\* 调用方法如下： AJ?}Hel[0

|  |
| --- |
| \* $delete\_object\_group = $oss\_sdk\_service->delete\_object\_group($bucket,$object\_group); |

,;cel^.b   
\* 其中参数$object\_group,$bucket为必须参数 "gb­nLKs

|  |
| --- |
| \*/ ()nKug`.@  $bucket = 'php-sdk-1330835674'; vJj:9KcP>h  $object\_group = 'object-group-1330930277'; ua$k^m7m5  $delete\_object\_group = $oss\_sdk\_service->delete\_object\_group($bucket,$object\_group); k99gjL`  print\_r($delete\_object\_group);die(); 2c"N-c&A  3r-VxP 5n  I`"-$99|t1  /\*%\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*%\*/ |

pqH( Tbjq   
//Mulit Part相关 ，适合大文件上传 ])­QO%   
i{|lsd(+   
/\*\*   
iFnOl\*TC   
\* 使用Multi-Part上传文件，该操作需要经过如下的步骤 {8{t]LK<   
\* 1. 初始化上传，调用initiate\_multipart\_upload，其中$bucket,$object为必选 \*dG}R#9Nv   
\* 2. 上传part,调用upload\_part，$bucket, $object, $uploadId为必选，还需要传入一个关联数组作为参数，该数组如下定义： UVT>7   
\* array( @[5\_C­?2   
\* 'fileUpload' => $filepath, 文件路径 OK M\"A4   
\* 'partNumber' => ($i 1), 文件编号 N /sEec

|  |
| --- |
| \* 'seekTo' => (integer) $part['seekTo'], 读取文件位置 |.Nr.4Yp  \* 'length' => (integer) $part['length'], 该部分part文件长度 @~$F;M=.\*  \* ) Wkg\*J3O  \* |

o  
<D3Y95b   
\* 其中的seekTo,length会有程序自动生成MK-a  
$~<   
\* JNxW6 cK   
\* 3. 完成上传,调用complete\_multipart\_upload，其中$bucket, $object, $uploadId, $upload\_parts为必选 g{A3W) [ b   
\*/ k1f3?l vlU ERfd7­V<c>   
//步骤1 ，初始化multipart MfXt‑+c`r r~Y>+ln. ‑'GV&]

|  |
| --- |
| $bucket = 'php-sdk-1329818657'; w69>tC‑  $object = 'Cpp.How.to.Program.7-'.time().'.pdf'; zdLVxL>87  $filepath = "D:\\ccc.pdf"; G}=`V­YK  $response\_initiate\_multipart\_upload = $oss\_sdk\_service->initiate\_multipart\_upload($bucket,$object); y3={NB+   if(!$response\_initiate\_multipart\_upload->isOK()){ bK$/,,0=X/  die('initial multipart upload failed....');  } |

< 7‑2s7\*Rv   
//解析返回，获取uploadId ,\_.@l+BM. q4sl=`L5Sp

|  |
| --- |
| $xml = new SimpleXmlIterator($response\_initiate\_multipart\_upload->body); %PQldPL8  $uploadId = (string)$xml->UploadId; |

FasI'Ulk   
//步骤2 上传part o5N]((9   
//首先获取文件可以分多少个part tXf}jU}

|  |
| --- |
| $parts = $oss\_sdk\_service->get\_multipart\_counts(filesize($filepath), 5242880); (@%XWg  $response\_upload\_part = array(); 4r\_!>['`"  foreach ($parts as $i => $part){ |

P<>[e­9|   
//开始上传part Rz.i/wg}

|  |
| --- |
| $response\_upload\_part[] = $oss\_sdk\_service->upload\_part($bucket, $object, $uploadId, array( #'J~Xk  'fileUpload' => $filepath, f\*{M3"$E  'partNumber' => ($i 1), %‑$/=4f.j  'seekTo' => (integer) $part['seekTo'], `xISkW4%  'length' => (integer) $part['length'], Rz`@N`U  )); zo\_k\K`{@  } 4|zd84g  \_n< LVd­E  ‑Z -pyF‑K\  $upload\_parts = array(); `[X6#`<  $result = true; |

xh{mca>?G   
AU0$A403   
//获取上传Part返回结果 nURvy}<r J=@hk@Nq#

|  |
| --- |
| foreach ($response\_upload\_part as $i=>$response){ Vn)%C\_-]A  $result = $result && $response->isOk(); uAp-$?  } w>v5oy8s-  NV&;e[z  if(!$result){ H!Wis3S3G  $oss\_sdk\_service->abort\_multipart\_upload($bucket, $object, $uploadId); ` !zQ  die('any part upload failed...'); ,`D/sNP ,q  } |

{VBR/M(q   
//构造upload part b+W)2rFO AeN 3<|RN

|  |
| --- |
| foreach ($response\_upload\_part as $i=>$response){ nARxn#<+  $upload\_parts[] = array( O-i4\_YdVt  'PartNumber' => ($i 1), S2Zx &D/\_  'ETag' => (string) $response->header['etag'] %'$f ?y­  ); DD2adu^  } |

)r=9]0=   
//步骤3 完成multipart upload -y%QRO(

|  |
| --- |
| $complete\_multipart\_upload\_response = $oss\_sdk\_service->complete\_multipart\_upload($bucket, $object, $uploadId, $upload\_parts); 1JIG+ZNmd  print\_r($complete\_multipart\_upload\_response);die(); >+L7k^[,0  JK[T]|G  F@g17aa  P\_ZM'[  /\*\* |

QQ\*gF‑P.Ao   
\* 列出multipart上传中的part，由于在列出part的时候需要有正在进行的multipart，故需要完成multi-part的步骤1和2,调用步骤如下： S3ErH,XB.   
\* 1. 初始化上传，调用initiate\_multipart\_upload，其中$bucket,$object为必选 %WZ$]M?q   
\* 2. 上传part,调用upload\_part，$bucket, $object, $uploadId为必选，还需要传入一个关联数组作为参数，该数组如下定义： E\ tL   
\* array( WmNA5;<Q   
\* 'fileUpload' => $filepath, 文件路径 {JX‑f\*IJ   
\* 'partNumber' => ($i 1), 文件编号 d60c$?"]a(   
\* 'seekTo' => (integer) $part['seekTo'], 读取文件位置 ebzzzmwo   
\* 'length' => (integer) $part['length'], 该部分part文件长度 l@xWQj9   
\* ) 5Wo5n7o   
\* >#  
  
I­NEO   
\* 其中的seekTo,length会有程序自动生成 &i)helXs]   
\* Ha?G=X   
\* 3. 列出parts,$oss\_sdk\_service->list\_parts($bucket, $object, $uploadId,$options);其中$bucket, $object, $uploadId为必选,$options可选 f#1/}Hq/I   
\*/

ti}f&w ICJ   
//步骤1 初始化 \*u}'}jC1X   
 76\_<xUt{

|  |
| --- |
| $bucket = 'php-sdk-1329818657'; >d#3|;RY  $object = 'Cpp.How.to.Program.7-'.time().'.pdf'; k; ;viT  $filepath = "D:\\ccc.pdf"; -'j7SOGk  $response\_initiate\_multipart\_upload = $oss\_sdk\_service->initiate\_multipart\_upload($bucket,$object); D'<$ g  if(!$response\_initiate\_multipart\_upload->isOK()){ ;e‑d#+$Na  die('initial multipart upload failed....'); ­Z4#v~!  } ­zOL;"/R  wE?CvL  $xml = new SimpleXmlIterator($response\_initiate\_multipart\_upload->body); ;>8kPG  $uploadId = (string)$xml->UploadId; |

qIz}$%!A   
获取分片 5\xr?`VZ   
 oXdel Ju?

|  |
| --- |
| $parts = $oss\_sdk\_service->get\_multipart\_counts(filesize($filepath), 5242880); g>\_OuQ|c  $response\_upload\_part = array(); |

?.Q3 pUT   
//步骤2 上传part \X5 3|Y;= NJm-%K

|  |
| --- |
| foreach ($parts as $i => $part){ h D.)M  $response\_upload\_part[] = $oss\_sdk\_service->upload\_part($bucket, $object, $uploadId, array( WtVf wC\_  'fileUpload' => $filepath, Ch t%uzb,  'partNumber' => ($i 1), @7%.7LK  'seekTo' => (integer) $part['seekTo'], [ZD[a6(94  'length' => (integer) $part['length'], $R?@L  )); 0yaMe@&,  } |

Y.NE^Vn0   
/\*列出parts，调用方法($bucket, $object, $uploadId,$options);,其中$bucket, $object, $uploadId为必选参数，$options可选，$options若传，定义方式如下 t0.;nv‑@A0   
\* e^8BV;+c   
\*$options = array( }vY.EEy!   
\* 'max-parts' => 10 一次最多取多少个part 8ljuc5,J   
\* 'part-number-marker' => 1, 从哪个part-number开始 AuXs B‑ CdL‑.?^

|  |
| --- |
| \*) l sr?b   \*/ Yz\ N&0"  m8+:=0|$  $options = array( L#2ZMy  'max-parts' => 10, {[P!$ /  'part-number-marker' => 1, :BD>yOlG  ); |

nmg{%P   
//步骤3 列出parts ~j"3}wXc5   
mX;H((

|  |
| --- |
| $list\_parts\_response = $oss\_sdk\_service->list\_parts($bucket, $object, $uploadId,$options); ­= ^s$<  print\_r($list\_parts\_response);die(); E30Z`$cz:   |IN[uQ  'yr{­^Pek  /\*\* |

`[Sl1saZ$S   
\* 终止multi-part upload，因为有终止进行中的任务，故需要执行步骤1和2 ­i  
?-Y   
\* 1. 初始化上传，调用initiate\_multipart\_upload，其中$bucket,$object为必选 h/?­6=D{   
\* 2. 上传part,调用upload\_part，$bucket, $object, $uploadId为必选，还需要传入一个关联数组作为参数，该数组如下定义： G "brT5:   
\* array( #c:kCZt#   
\* 'fileUpload' => $filepath, 文件路径 !iJipe5   
\* 'partNumber' => ($i 1), 文件编号 bLlH//ZRH   
\* 'seekTo' => (integer) $part['seekTo'], 读取文件位置 ]\dHU.i   
\* 'length' => (integer) $part['length'], 该部分part文件长度 hZU1O   
\* ) M\*`hDd‑S   
\* FM\[].   
\* 其中的seekTo,length会有程序自动生成 m‑9t$­h   
\* iRt\*A6`m+   
\* 3. 终止multi-part,调用$oss\_sdk\_service->abort\_multipart\_upload($bucket, $object, $uploadId);其中$bucket, $object, $uploadId为必选 zJC  
EA   
\*/

‑%\*K;np-q{   
//步骤1 初始化 ="(>>C1- |';7v)CIG

|  |
| --- |
| $bucket = 'php-sdk-1330835674'; B;W(iI  $object = 'Cpp.How.to.Program.7-'.time().'.pdf'; [.&[<!,.­  $filepath = "D:\\ccc.pdf"; vyU!+mlc  $response\_initiate\_multipart\_upload = $oss\_sdk\_service->initiate\_multipart\_upload($bucket,$object); |}Ph"g2D,  if(!$response\_initiate\_multipart\_upload->isOK()){ ^l(^z fsZ  die('initial multipart upload failed....'); V3uXan\_  } }-nU3{1  $xml = new SimpleXmlIterator($response\_initiate\_multipart\_upload->body); 9/3;{`+[a  $uploadId = (string)$xml->UploadId;  p6[ (81  S>t>6&A  $parts = $oss\_sdk\_service->get\_multipart\_counts(filesize($filepath), 5242880); (p,}'I#i\*  $response\_upload\_part = array(); |

D^?\_"wjW   
//步骤2 上传part 9g]M4\*?C9P   
[:B\*6FXMN~

|  |
| --- |
| foreach ($parts as $i => $part){ ]w[ThHRJ  $response\_upload\_part[] = $oss\_sdk\_service->upload\_part($bucket, $object, $uploadId, array( 8zm‑v 5trt  'fileUpload' => $filepath, uJ8{HB  'partNumber' => ($i 1), |Gu‑EGmR  'seekTo' => (integer) $part['seekTo'], w}`3 d@  'length' => (integer) $part['length'], ;\_tO+xL&  )); 7'eh)[T  } |

LNiS`o\   
//步骤3 终止 ?\Q0kr.T%

|  |
| --- |
| $abort\_multipart\_upload\_response = $oss\_sdk\_service->abort\_multipart\_upload($bucket, $object, $uploadId); KlO(o#&N  print\_r($abort\_multipart\_upload\_response);die(); ( 3;`bvYH"  I%xn,u  C\*rd;+1A  /\*\* |

Pfan7fq+   
\* 列出所有的multipart upload Z %EQt   
\* 调用方法 $oss\_sdk\_service->list\_multipart\_uploads($bucket);其中$bucket是必选参数 P4 6,o

|  |
| --- |
| \*/ <'2u ‑a  $bucket = 'php-sdk-1330835674'; GAp!nix6h  $list\_multipart\_uploads\_response = $oss\_sdk\_service->list\_multipart\_uploads($bucket); hk;‑bk?:m  print\_r($list\_multipart\_uploads\_response);die(); 6K 4+0xXv  p;`N\.ld  :hP58 ‑}Q$  /\*\* |

=l,#iYJP8   
\* 一次性完成multi-part upload，由于mulit-part upload涉及三个步骤，所以封装了该方法使得使用multi-part更加方便 ‑oEIqA   
\* 调用方法如下： $?f]ZyZr.

|  |
| --- |
| \* $oss\_sdk\_service->create\_mpu\_object($bucket, $object,$options); |

`~nCbUUee   
\* 其中的$bucket,$object,$options为必须,$options的定义如下 7\%$>< K   
 f{\_)rsqf

|  |
| --- |
| \* $options = array( v{koKQ'Y()  \* 'fileUpload' => $filepath, 文件路径 a))\*F!}c  \* 'partSize' => 5242880, 分片大小为了保证上传的效率，建议该值不要太大，建议使用5M = 5242880 VDiOO  \* ) / O­)6iJ  \*/ kzi|$Gs<  $bucket = 'php-sdk-1329818657'; S@A<6   $object = 'myobject-pdf-'.time().'.rar'; O`(U/?  $filepath = "d:\\hh\_AxureRP-Pro-Setup.rar"; -tyaE  $options = array( xJc.pvV‑Pw  'fileUpload' => $filepath, Q|//Z  'partSize' => 5242880, u+z~  ); 3EOyq^I%  +WLD  $create\_mpu\_object\_response = $oss\_sdk\_service->create\_mpu\_object($bucket, $object,$options); 4Py3I9  print\_r($create\_mpu\_object\_response);die();  |j#^@R  /\*\* |

%zflx~ ]\_#SAhOR)   
\* 上传整个目录，通过multi-part,通过该方式创建的object默认为文件名 6t$N7­8U

|  |
| --- |
| \* $oss\_sdk\_service->create\_mtu\_object\_by\_dir($bucket,$dir,$recursive = false,$exclude = ".|..|.svn",$options = null) |

8&?p   
\* 其中$bucket,$dir为必选参数，$recursive,$exclude,$options可选,其中o Wr#~GFg   
\* $bucket 为bucket名称 yHOqzq56   
\* $dir 某一目录 8­Fx]koP.   
\* $recursive 该参数设置是否递归读取目录。 vX.]hp5~   
\* $exclude 要过滤掉的文件，默认为系统默认生成的.,..,和svn文件.svn DaDUK? b‑"3uD`

|  |
| --- |
| \*/ yxY h?ka  $bucket = 'php-sdk-windows'; vv\*|F  $dir = "C:\\Windows"; !ew6 n‑ I  $recursive = true; hXTYTbTX  $create\_mtu\_object\_by\_dir\_response = $oss\_sdk\_service->create\_mtu\_object\_by\_dir($bucket,$dir,$recursive); D2$^"  print\_r($create\_mtu\_object\_by\_dir\_response);die(); |

|  |
| --- |
|  |