

# 支付宝容器jsAPI文档手册PDF版本

本版本为实验版本，为线下独立查看使用，受制于生成PDF程序的限制，样式问题还没有很好的解决，例如分页切图、代码高亮。  
推荐使用在线版本，方便实时查看jsAPI运行效果。  
如需搜索，使用阅读PDF软件自带功能即可。  
更多细节样式调整和目录索引探索建设中。  
生成时间: 2017-06-08 05:40:30

Since 8.0

开发者在h5页面唤起支付请使用开放平台 alipay.trade.create (<https://doc.open.alipay.com/docs/api.htm?spm=a219a.7629065.0.0.yjpXrR&apild=1046&docType=4>)  
接口生成交易号

## 1. 使用方法

```
// 调用此jsapi的时候，不要在当前页面有监听resume事件并在resume事件中强制刷新页面。  
// 调用方式1: 通过传入交易号唤起快捷调用方式  
AlipayJSBridge.call("tradePay",{  
  tradeNO: "201209071234123221"  
}, function(result){  
});  
// 调用方式2: 通过传入传入交易的字符串唤起快捷调用方式, 只需要这一个参数  
AlipayJSBridge.call("tradePay",{  
  orderStr: "xxxxxxxxxxxxxxxxxxxxxxx"  
}, function(result){  
});
```

## 2. 代码演示

### 2.1. 交易号唤起收银台

```
<h1>点击以下按钮唤起收银台支付</h1>
<a href="javascript:void(0)" class="tradeno">交易号唤起</a>
<script>
function ready(callback) {
    // 如果jsbridge已经注入则直接调用
    if (window.AlipayJSBridge) {
        callback && callback();
    } else {
        // 如果没有注入则监听注入的事件
        document.addEventListener('AlipayJSBridgeReady', callback, false);
    }
}
ready(function(){
    document.querySelector('.tradeno').addEventListener('click', function(){
        AlipayJSBridge.call("tradePay",{
            tradeNO: "201209071234123221"
        }, function(result){
            alert(JSON.stringify(result));
        });
    });
});
</script>
```

## 2.2. 唤起支付密码校验（不发起支付，只做身份确认）

```
<h1>点击下面按钮唤起密码校验</h1>
<a href="javascript:void(0)" class="tradeStr">唤起密码校验</a>
<script>
function ready(callback) {
    // 如果jsbridge已经注入则直接调用
    if (window.AlipayJSBridge) {
        callback && callback();
    } else {
        // 如果没有注入则监听注入的事件
        document.addEventListener('AlipayJSBridgeReady', callback, false);
    }
}
ready(function(){
    document.querySelector('.tradeStr').addEventListener('click', function(){
        AlipayJSBridge.call("tradePay",{
            orderStr: "uuid=95a76812-0b4c-46a6-b4ae-cc48c97b5fc5"+ // uuid每次唯一
            "&extern_token=GZ00UIvFe6xfGj9asLXhWkUeUJoXyGmobilegwGZ00"+ // extern_token登入sessionId
            "&app_name=alipay"+
            "&biz_type=openservice"+
            "&apiname=com.alipay.paypwd.validate"+
            "&apiservice=com.alipay.loan.agreement.sign"+
            "&tip=需要在密码框上显示的文案"
        }, function(result){
            alert(JSON.stringify(result));
        });
    });
});
</script>
```

2.3. 唤起支付渠道选择列表

```
<h1>点击下面按钮唤起支付渠道选择</h1>
<a href="javascript:void(0)" class="tradeStr">支付渠道</a>
<script>
function ready(callback) {
    // 如果jsbridge已经注入则直接调用
    if (window.AlipayJSBridge) {
        callback && callback();
    } else {
        // 如果没有注入则监听注入的事件
        document.addEventListener('AlipayJSBridgeReady', callback, false);
    }
}
ready(function(){
    document.querySelector('.tradeStr').addEventListener('click', function(){
        AlipayJSBridge.call('getUserInfo', function(info){
            var userId= info.userId;

            AlipayJSBridge.call("tradePay",{
                orderStr: "uuid=95a76812-0b4c-46a6-b4ae-cc48c97b5fc5"+ // uuid每次唯一
                "&extern_token=GZ00cSiae3xXJ9sXjuV2NhnyfW00Pb98mobilegwGZ00"+// extern_token登入sessionId
                "&app_name=alipay"+
                "&biz_identity=userassets10002"+
                "&trade_from=ALIPAY"+
                "&biz_type=switch_channel"
            }, function(result){
                alert(JSON.stringify(result));
            });
        });
    });
});
</script>
```

## 2.4. 唤起添加银行卡

```
<h1>点击下面按钮唤起添加银行卡</h1>
<a href="javascript:void(0)" class="tradeStr">添加银行卡</a>
<script>
function ready(callback) {
    // 如果jsbridge已经注入则直接调用
    if (window.AlipayJSBridge) {
        callback && callback();
    } else {
        // 如果没有注入则监听注入的事件
        document.addEventListener('AlipayJSBridgeReady', callback, false);
    }
}
ready(function(){
    document.querySelector('.tradeStr').addEventListener('click', function(){
        AlipayJSBridge.call("tradePay",{
            orderStr: "uuid=95a76812-0b4c-46a6-b4ae-cc48c97b5fc5"+ // uuid每次唯一
            "&extern_token=GZ00UIvFe6xfGj9asLXhWkUeUJoXyGmobilegwGZ00"+ // extern_token登入sessionId
            "&app_name=alipay"+
            "&biz_type=new_card"+
            "&trade_from=ALIPAY"+
            "&forbid_channel=credit"
        }, function(result){
            alert(JSON.stringify(result));
        });
    });
});
</script>
```

### 3. API

```
AlipayJSBridge.call('tradePay', {
    tradeNO, partnerID, bizType, bizSubType, displayPayResult, bizContext, orderStr
}, fn);
```

#### 3.1. 入参

名称	类型	描述	必选	默认值	版本
tradeNO	string	交易号。多个用”；”分隔	N		8.0
partnerID	string	商户id	N	“”	8.0
bizType	string	交易类型	N	‘trade’	8.0
bizSubType	string	交易子类型	N	“”	8.0

displayPayResult	bool	是否显示支付结果页	N	true	8.0
bizContext	string	支付额外的参数，格式为JSON字符串	N	“	8.2
orderStr	string	完整的一个支付字符串	N	“	8.2
fn	function	收营台关闭之后的结果回调	N		

### 3.2. 出参

回调函数带入的参数result: {resultCode, callbackUrl, memo, result}
--

名称	类型	描述	版本
resultCode	string	支付结果 ‘9000’：订单支付成功； ‘8000’：正在处理中； ‘4000’：订单支付失败； ‘6001’：用户中途取消； ‘6002’：网络连接出错 ‘99’：用户点击忘记密码快捷界面退出(only iOS since 9.5)	8.0
callbackUrl	bool	交易成功后应跳转到的url；一般为空，除非交易有特殊配置	8.0
memo	bool	收银台服务端返回的memo字符串	iOS 8.0, android 8.4
result	bool	收银台服务端返回的result字符串	iOS 8.0, android 8.4

### 3.3. orderStr各个子参数

名称	类型	描述
uuid	string	随机字符串，可自行生成或服务端接口
extern_token	string	钱包信任登录token，即为sessionId，通过服务端接口获取
app_name	string	调用方app标识，alipay代表支付宝钱包
biz_type	string	调用业务类型，openservice代表开放基础服务
apiname	string	调用开放服务api名称，com.alipay.paypwd.validate代表独立密码校验服务
apiservice	string	调用开放服务api业务方，com.alipay.loan.agreement.sign代表小贷签约
biz_identity	string	某个业务类型对应的具体业务参数

user_id	string	某些业务需要的用户id
tip	string	需要在密码框上展示的提示文案