

365 Bit 对接文档 1.0

产品说明

365Bit 是一家交易易平台，为商户提供了简单快速的一键买卖服务。主要提供以下 API:

- 请求支付/充值接口
- 请求提现/代付接口
- 商户异步回调接口

接入流程（重要）

商户调用充值接口下单，下单成功得到支付码链接，商户将得到的支付链接生成二维码，展示给会员，会员截图识别二维码，使用微信或者支付宝识别打开二维码，打开的页面中快速支付或者识别二维码进行支付。（请商户按照此方式接入，防止其他异常情况）

所有接口使用 HTTP 方式请求接口，所有响应数据以 JSON 格式返回：

status	状态(Integer)	接口处理的状态值
message	消息(String)	接口处理的消息
Data	数据体	接口返回的数据体

其中 status 为 0 时 表示接口处理成功，data 有数据内容，当 status 不为 0 时 表示接口处理失败，具体错误信息看 message 返回

当订单金额与交易员收到的实际金额不一致时不会回调该笔订单，也不会补发。此外使用订单查询接口查询该笔订单时除了判断订单状态为订单完成(complete)以外，还要判断实际收到的金额（payActualAmount）与订单是否一致，不一致时判定为异常订单，应不给会员充值。

签名规则

- 1、将待签名字符串要求按照参数进行排序（首先比较所有参数名的第一个字母，按 abcd 顺序排列，若遇到相同首字母，则看第二个字母，以此类推）

例如：对于如下的参数进行签名

```
{
  thirdMerchantId:"1",
  thirdTradeNo:"abcdef1234567890",
  thirdUserId:"abcdef1234567890",
  payChannel:"alipay",
  payAccount:"123@qq.com",
  payAmount:100,
  paySymbol:"CNY",
  buySymbol:"USD",
  requestTime:1552911381611,
  requestSign:"98fe30babe3209f0fe5f5f828de941c8",
}
```

2、生成【待签名字符串】

```
{"buySymbol":"USD", "payAccount":"123@qq.com", "payAmount":100, "payChannel":"alipay", "paySymbol":"CNY", "requestTime":1552911381611, "thirdMerchantId":"1", "thirdTradeNo":"abcdef1234567890", "thirdUserId":"abcdef1234567890"}
```

3、将【待签名字符串】&【apiKey】生成【最终待签名字符串】

```
{"buySymbol":"USD", "payAccount":"123@qq.com", "payAmount":100, "payChannel":"alipay", "paySymbol":"CNY", "requestTime":1552911381611, "thirdMerchantId":"1", "thirdTradeNo":"abcdef1234567890", "thirdUserId":"abcdef1234567890"} + apiKey
```

4、最后是利用 32 位 md5 算法，对【最终待签名字符串】进行签名运算，从而得到签名结果字符串（该字符串赋值于参数 sign），最后值全都转成小写字母进行使用。

API 列表

1、USDT 兑换 CNY 比例

1.1 请求地址

<http://47.56.177.238/trade/recharge/rateUSDTTCNY>

1.2 请求方式

采用 POST 请求方式

1.3 请求头部

Content-type	application/JSON
--------------	------------------

1.4 响应示例（JSON）

```
{
  "status": 0,
  "message": "success",
  "data": 6.86
}
```

2、充值接口

2.1 请求地址

```
http://47.56.177.238/trade/recharge/prepareRechargeTrade
```

2.2 请求头部

Content-type	application/JSON
--------------	------------------

2.3 Body 参数

参数名	类型	Reuired	Desc
buySymbol	String	True	买入的法币币种(USDT)
payAccount	String	True	支付账号
payAmount	Number	True	支付金额
payChannel	String	True	支付渠道(alipay/wechat/union/gather)
paySymbol	String	True	支付法币币种(CNY)
requestSign	String	True	请求签名
requestTime	String	True	时间戳毫秒数
thidMerchantId	String	True	商户 ID
thirdTradeNo	String	True	第三方订单号
thirdUserId	String	True	第三方用户 ID

2.4 请求示例(JAVA)

```
String apiKey = "123456";//apiKey 可在商户后台查看
PrepareBuyTradeSignData signData = new PrepareBuyTradeSignData();
PrepareBuyTradeDTO prepareTrade = new PrepareBuyTradeDTO();
signData.setBuySymbol("USD");
signData.setPayAccount("123@qq.com");
signData.setPayAmount(new BigDecimal(100));
signData.setPayChannel("alipay");
signData.setPaySymbol("CNY");
signData.setRequestTime(System.currentTimeMillis());
signData.setThirdMerchantId("1");
signData.setThirdTradeNo("1554111400668w04");
signData.setThirdUserId("1554111400668h67");

// 字段按自然顺序转为 JSON string
String json = JSONObject.toJSONString(signData, SerializerFeature.MapSortField);
String requestSign = Md5Utils.getMD5(json + apiKey);
BeanUtils.copyProperties(signData, prepareTrade);
prepareTrade.setRequestSign(requestSign);
ApiResponse apiResponse =
JSONObject.parseObject(HttpClientUtils.postData("http://47.56.177.238/trade/recharge/prepare
RechargeTrade", JSONObject.toJSONString(prepareTrade)), ApiResponse.class);
if (apiResponse.getStatus() == 0) {
    // 买币成功
    apiResponse.getData(); // 数据字段详见返回说明
} else {
    System.out.println(apiResponse.getMessage());
}
```

2.5 响应示例(JSON)

```
{
  "status": 0,
  "message": "success",
  "data": {
    "thirdMerchantId": "1",
    "thirdTradeNo": "1554102555523pwk",
    "thirdUserId": "155410255552381o",
    "payChannel": "alipay",
    "payAccount": "123@qq.com",
    "payAmount": 100,
  }
}
```

```

        "paySymbol": "CNY",
        "buySymbol": "USDT",
        "requestTime": "2019-04-01 15:09:15",
        "tradeId": 567,
        "tradeNo": "1904010709150957",
        "tradeStatus": "create", //订单状态, create:创建成功, paid:已支付, send:已发币, cancel:
已取消, complete:完成
        "buyerReceiptAmount": 14.233567,
        "receiptAccountName": "张三",
        "receiptAccountNumber": "15050491754",
        "receiptQrCode": null,
        "receiptBank": "",
        "receiptBankBranch": "",
        "receiptPayURL": "收款地址"
    }
}

```

3、提现/下发接口

3.1 请求地址

<http://47.56.177.238/trade/recharge/prepareSaleTrade>

3.2 请求头部

Content-type	application/JSON
--------------	------------------

3.3 Body 请求参数

参数名	类型	Reuired	Desc
receiptAccountName	string	True	收款账户户名
receiptAccountNumber	string	True	收款账户账号
receiptBank	string	True	收款银行
receiptBankBranch	String	True	收款支行
receiptSymbol	string	True	收款法币币种(CNY)
requestSign	string	True	请求签名
requestTime	String	True	时间戳毫秒数

SellAmount	Number	True	提现金额
sellAmountSymbol	String	False	金额币种(CNY(默认)/USDT)
sellSymbol	string	True	出售币种(USDT)
thidMerchantId	String	True	商户 ID
thirdTradeNo	string	True	第三方订单号
ThirdUserId	string	True	第三方用户 ID

3.4 请求示例(JAVA)

```
String apiKey = "123456";//apiKey 可在商户后台查看
PrepareSellTradeSignData signData = new PrepareSellTradeSignData();
PrepareSellTradeDTO prepareTrade = new PrepareSellTradeDTO();
signData.setReceiptAccountName("123@qq.com");
signData.setReceiptAccountNumber("8292887348934239");
signData.setReceiptBank("中国银行");
signData.setReceiptBankBranch("园区支行");
signData.setReceiptSymbol("CNY");
signData.setRequestTime(System.currentTimeMillis());
signData.setSellAmount(new BigDecimal(100));
signData.setSellAmountSymbol("CNY");
signData.setSellSymbol("USDT");
signData.setThirdMerchantId("1");
signData.setThirdTradeNo("15541125986764nf");
signData.setThirdUserId("1554112598677s.jp");

// 字段按自然顺序转为 JSON string
String json = JSONObject.toJSONString(signData, SerializerFeature.MapSortField);
String requestSign = Md5Utils.getMD5(json + apiKey);
BeanUtils.copyProperties(signData, prepareTrade);
prepareTrade.setRequestSign(requestSign);
ApiResponse apiResponse =
JSONObject.parseObject(HttpClientUtils.postData("http://47.56.177.238/trade/recharge/prepare
SaleTrade", JSONObject.toJSONString(prepareTrade)), ApiResponse.class);
if (apiResponse.getStatus() == 0) {
    // 卖币成功
    apiResponse.getData();// 数据字段详见返回说明
} else {
    System.out.println(apiResponse.getMessage());
}
```

3.5 响应示例

```
{
  "status": 0,
  "message": "success",
  "data": {
    "thirdMerchantId": "1",
    "thirdTradeNo": "1554102398928feu",
    "thirdUserId": "1554102398928pci",
    "payChannel": "union",
    "payAccount": null,
    "payAmount": 685.000000,
    "paySymbol": "CNY",
    "buySymbol": "USDT",
    "requestTime": "2019-04-01 15:06:38",
    "tradeId": 565,
    "tradeNo": "1904010706380955",
    "tradeStatus": "create", //订单状态,create:创建成功,paid:已支付,send:已发币,cancel:
已取消,complete:完成
    "buyerReceiptAmount": 100,
    "receiptAccountName": "123@qq.com",
    "receiptAccountNumber": "1324568",
    "receiptQrCode": null,
    "receiptBank": "中信银行",
    "receiptBankBranch": "北京支行"
  }
}
```

4、查询订单

4.1 请求地址(GET)

```
http://47.56.177.238/trade/recharge/queryRechargeTrade
```

4.2 请求头部

Content-type	application/x-www-form-urlencoded
--------------	-----------------------------------

4.3 请求参数(query)

参数名	类型	Reuired	Desc
thidMerchantId	String	True	商户 ID
thirdTradeNo	string	True	第三方订单号

4.4 请求示例（JAVA）

```
HttpClientUtils.getData("http://47.56.177.238/trade/recharge/queryRechargeTrade?thirdMerchantId=1&thirdTradeNo=1554113082261q94");
```

4.5 响应示例(JSON)

```
{
  "status": 0,
  "message": "success",
  "data": {
    "thirdMerchantId": "1",
    "thirdTradeNo": "1554102555523pwk",
    "thirdUserId": "15541025555238lo",
    "payChannel": "alipay",
    "payAccount": "123@qq.com",
    "payAmount": 100,
    "paySymbol": "CNY",
    "payActualAmount": '100', //实际收到的金额(CNY) 订单金额因此为准
    "buySymbol": "USDT",
    "requestTime": "2019-04-01 15:09:16",
    "tradeId": 567,
    "tradeNo": "1904010709150957",
    "tradeStatus": "create", //订单状态,create:创建成功,paid:已支付,send:已发币,cancel:已取消,complete:完成
    "buyerReceiptAmount": 14.233567,
    "receiptAccountName": "张三",
    "receiptAccountNumber": "15050491754",
    "receiptQrCode": null,
    "receiptBank": "",
    "receiptBankBranch": "",
    "receiptPayURL": "收款地址"
  }
}
```

5、回调通知（POST）

5.1 请求地址

请求地址由商户提供

5.2 请求头部

Content-type	application/JSON
--------------	------------------

5.3 请求参数(query)

参数名	类型	Reuired	Desc
Sign	String	true	请求签名

5.4 Body 响应示例(JSON)

```
{
  "thirdMerchantId": "1",
  "thirdTradeNo": "155410255523pwk",
  "thirdUserId": "15541025552381o",
  "payChannel": "alipay",
  "payAccount": "123@qq.com",
  "payAmount": 100,
  "paySymbol": "CNY",
  "payActualAmount": '100', //实际收到的金额(CNY)
  "buySymbol": "USDT",
  "requestTime": "2019-04-01 15:09:16",
  "tradeId": 567,
  "tradeNo": "1904010709150957",
  "tradeStatus": "create", //订单状态, create:创建成功, paid:已支付, send:已发币, cancel:已取消, complete:完成
  "buyerReceiptAmount": 14.233567,
  "receiptAccountName": "张三",
  "receiptAccountNumber": "15050491754",
  "receiptQrCode": null,
}
```

```
"receiptBank": "",
"receiptBankBranch": "",
"receiptPayURL": "收款地址"
}
```

5.5 返回结果

365 Bit 要求商户在通知处理成功之后给我放服务器返回成功的标识 JSON 字符串:

```
{"status":0,"message":"success"}
```

6、流程图

6.1 充值流程图

