

# 安徽快3 Dapp接口说明

假定当前使用的节点信息是<http://192.168.2.58:4096>

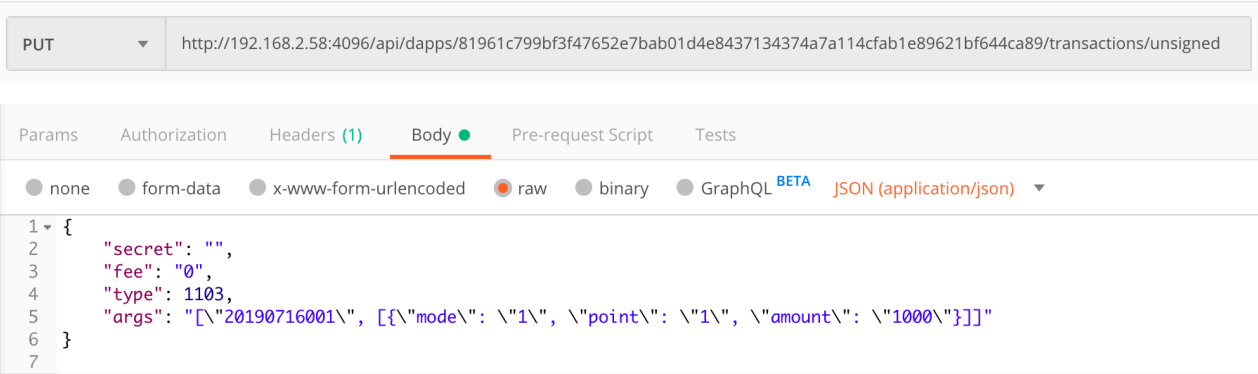
当前使用的安徽快3 Dapp的id  
是“81961c799bf3f47652e7bab01d4e8437134374a7a114cfab1e89621bf644ca89”

则文档说明使用节点接口并以"/"开头的路径时，表示前面加上<http://192.168.2.58:4096>;

文档说明使用Dapp接口并以"/"开头的路径时，表示在前面加上<http://192.168.2.58:4096/api/dapps/81961c799bf3f47652e7bab01d4e8437134374a7a114cfab1e89621bf644ca89>;

## 交易

### (PUT)注(Dapp接口/transactions/unsigned)



如上图所示：

使用Dapp接口/transactions/unsigned可以发出下注交易

字段	必填	类型	说明
secret	是	string	下注账号对应的secret值
fee	是	String	交易所需交易费，测试默认使用"0"
type	是	number	交易类型，固定为1103(下注交易)
args	是	string(使用JSON.stringify进行处理)	下注参数详情，参见"下注参数详情"

### 下注参数详情

下注参数为JSON.stringify处理后的字符串，是一个json数组，数组需要有两个item([期号, 下注定单列表])

期号：游戏当前进行到的期号值，设定了一天40期，格式YYYYMMDDXXX(Y表示年，M表示月，D表示天，X表示当天的累加次数)，如20190716001表示2019年7月16号第1期

下注订单列表(数组)：

mode: 游戏类型

point: 下注点数

amount: 下注金额

其中游戏类型和下注点数如下列图表所示

```
Rule1: {
    /// mode: 1 -- 大小/三军
    PointSmall: "0",          // 小
    PointBig: "7",            // 大
    Point1: "1",              // 1点
    Point2: "2",              // 2点
    Point3: "3",              // 3点
    Point4: "4",              // 4点
    Point5: "5",              // 5点
    Point6: "6",              // 6点
},
Rule2: {
    /// mode: 2 -- 围骰/全骰
    PointAll: "777",          // 全骰
    Point111: "111",          // 3个1点
    Point222: "222",          // 3个2点
    Point333: "333",          // 3个3点
    Point444: "444",          // 3个4点
    Point555: "555",          // 3个5点
    Point666: "666",          // 3个6点
},
Rule3: {
    /// mode: 3 -- 点数
    Point4: "4",              // 点数和为4
    Point5: "5",              // 点数和为5
    Point6: "6",              // 点数和为6
    Point7: "7",              // 点数和为7
    Point8: "8",              // 点数和为8
    Point9: "9",              // 点数和为9
    Point10: "10",            // 点数和为10
    Point11: "11",            // 点数和为11
    Point12: "12",            // 点数和为12
    Point13: "13",            // 点数和为13
    Point14: "14",            // 点数和为14
    Point15: "15",            // 点数和为15
    Point16: "16",            // 点数和为16
    Point17: "17",            // 点数和为17
},
Rule4: {
    /// mode: 4 - 长牌
    Point12: "12",            // 点数1、2
    Point13: "13",            // 点数1、3
```

```

    Point14: "14",          // 点数1、4
    Point15: "15",          // 点数1、5
    Point16: "16",          // 点数1、6
    Point23: "23",          // 点数2、3
    Point24: "24",          // 点数2、4
    Point25: "25",          // 点数2、5
    Point26: "26",          // 点数2、6
    Point34: "34",          // 点数3、4
    Point35: "35",          // 点数3、5
    Point36: "36",          // 点数3、6
    Point45: "45",          // 点数4、5
    Point46: "46",          // 点数4、6
    Point56: "56"           // 点数5、6
},
Rule5: {
    /// mode: 5 - 短牌
    Point11: "11",          // 点数1、1
    Point22: "22",          // 点数2、2
    Point33: "33",          // 点数3、3
    Point44: "44",          // 点数4、4
    Point55: "55",          // 点数5、5
    Point66: "66"           // 点数6、6
}

```

## 查询

### (GET)戏账号余额查询(Dapp接口/account/game\_balance)

字段	必填	类型	说明
address	是	String	查询的账号的地址

### (GET)主链ETM余额查询(Dapp接口/account/chain\_balance)

字段	必填	类型	说明
address	是	String	查询的账号地址

### (GET)取用户下注信息(Dapp接口/account/bettings)

字段	必填	类型	说明
address	是	String	账号地址
cond	否	String	条件，默认：“0”，“0”：所有下注记录，“1”：未开奖的下注记录，“2”：已中奖的下注记录，“3”：未中奖的下注记录
offset	否	number	起始索引，默认为0
limit	否	number	最大获取数量，默认为100

### (GET)用户报表(Dapp接口/account/crystal)

字段	必填	类型	说明
address	是	String	账号地址
cond	否	String	条件，默认：“0”，“0”：最近7天，“1”：本周，“2”：上周

### (GET)当前游戏期号查询(Dapp接口/game/period)

参数无

### (GET)完成游戏期号查询(Dapp接口/game/periods)

字段	必填	类型	说明
datetime	是	String	格式为：年(2019)月(07)日(19)[YYYYMMDD],不填表示获取所有的
offset	否	number	起始索引,默认为0
limit	否	number	最大获取数量，默认为40

### (GET)获取已结束的同期的详情(Dapp接口/game/period\_detail)

字段	必填	类型	说明
periodId	是	String	周期值

### (GET)获取已结束同期的下注信息(Dapp接口/game/period\_bets)

字段	必填	类型	说明
periodId	是	String	周期值
address	否	String	下注账号地址
offset	否	number	起始索引,默认为0
limit	否	number	最大数量，默认为100

## 通知

---

### start-period

data: periodId(期号)

### mothball-period

data: periodId(期号)

### end-period

data: {

periodId(期号),

points(骰子点数 array[3]),

hash(产生骰子点数的随机源hash值)

}

## 随机源信息获取

---

随机源域名：<http://47.111.165.42:4567>

获取产生Hash的数据源

GET	▼	http://47.111.165.42:4567/random/getInfo?hash=12320b5778ae344e6540d6c312f3894ea3096056cc11486801cb97827fc420aea8
-----	---	--

Params ● Authorization Headers (7) Body Pre-request Script Tests

### Query Params

	KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/>	hash	12320b5778ae344e6540d6c312f3894ea3096056cc1148680...	
	Key	Value	Description

Body Cookies Headers (5) Test Results

Status: 200 OK Time: 1058 ms

Pretty

Raw

Preview

JSON ▼



## (GET)/random/getInfo

字段	必填	类型	说明
hash	是	String	需要查询的hash值，即每一个period结束时获取到的hash值

字段	类型	说明
random	String	随机数hex字符串值
index	number	随机数index索引值
hashes	Array<{id: string, height: number}>	计算随机数用到的区块信息id:区块Id, height: 区块高度
iter	number	迭代次数
sha256iter	Number	sha256计算迭代次数