# API 文档

**自定义编码**

* 200: 任务成功
* **201**: 任务进行中 (Task is running. Just wait.)
* **203**: 任务已完成 (Task completed. You can redo it after 5 minutes.)
* **500**: 系统错误 (System error. Contact jack.)
* **501**: 任务错误 (Task error. You can retry it after 5 minutes.)
* **502**: 登录失败 (Unable to login, please check your key)
* **510: 单腿错误，严重异常**
* **511：建仓失败，请检查资金余额问题**

## 1. 资金分布情况

\*\*接口\*\*: GET /distribute

\*\*描述\*\*: 获取不同资产的资金分布情况。

\*\*URL\*\*: http://127.0.0.1:5001/distribute

\*\*查询参数\*\*:

| 参数 | 类型 | 描述 |

|------------|--------|----------------|

| api\_key | string | 用于认证的 API 密钥。 |

| api\_secret | string | 用于认证的 API 密钥。 |

\*\*响应\*\*: 返回一个表示资金分布的对象数组。

\*\*响应示例\*\*:

[

{

"money": 387.7547,

"name": "SUI",

"percent": 25.147240333627423,

"totalMoney": 1541.9373850000002

},

{

"money": 389.499915,

"name": "KEY",

"percent": 25.260423593659738,

"totalMoney": 1541.9373850000002

},

{

"money": 389.19972,

"name": "ETH",

"percent": 25.24095490427453,

"totalMoney": 1541.9373850000002

},

{

"money": 375.48305,

"name": "WIF",

"percent": 24.351381168438298,

"totalMoney": 1541.9373850000002

}

]

\*\*响应字段\*\*:

| 字段 | 类型 | 描述 |

|-------------|-------|------------------|

| money | float | 指定资产的金额。 |

| name | string| 资产名称。 |

| percent | float | 占总资金的百分比。|

| totalMoney | float | 所有资产的总金额。|

## 2. 初始化账户，建仓

\*\*接口\*\*: GET /doInitAccount

\*\*描述\*\*: 初始化账户并开设仓位。

\*\*URL\*\*: https://127.0.0.1:5001/doInitAccount

\*\*查询参数\*\*:

| 参数 | 类型 | 描述 |

|------------|--------|----------------|

| api\_key | string | 用于认证的 API 密钥。 |

| api\_secret | string | 用于认证的 API 密钥。 |

| percent | int | 建仓百分比，最小 20，最大 80。|

\*\*响应\*\*: 返回账户初始化的结果和当前状态。

\*\*响应示例\*\*:

{

"code": 203,

"message": "Task completed. You can redo it after 5 minutes."

}

\*\*响应字段\*\*:

| 字段 | 类型 | 描述 |

|-------------|--------|------------------|

| code | int | 返回状态码。 |

| message | string | 返回消息。 |

## 3. 获取账户信息

\*\*接口\*\*: GET /account

\*\*描述\*\*: 获取账户信息，包括余额和持仓情况。

\*\*URL\*\*: https://127.0.0.1:5001/account

\*\*查询参数\*\*:

| 参数 | 类型 | 描述 |

|------------|--------|----------------|

| api\_key | string | 用于认证的 API 密钥。 |

| api\_secret | string | 用于认证的 API 密钥。 |

\*\*响应\*\*: 返回账户的详细信息。

\*\*响应示例\*\*:

{

"asset USDT Balance": 10561.79591268884,

"asset Total": 3204.0211168384,

"Margin Rate": "231.35703056",

"Crypto Assets": [

{"name": "SUI", "quantity": 1053.6, "current\_price": 0.7524, "money": 792.72864, "uTime": 1720075812436},

{"name": "KEY", "quantity": 148230, "current\_price": 0.00537, "money": 798.51501, "uTime": 1720075812436},

{"name": "CFX", "quantity": 35, "current\_price": 0.14804, "money": 5.1814, "uTime": 1720075812436},

{"name": "ETH", "quantity": 0.247, "current\_price": 3209.53, "money": 792.75391, "uTime": 1720075812436},

{"name": "WIF", "quantity": 438.5, "current\_price": 1.8271, "money": 801.18335, "uTime": 1720075812436},

{"name": "SUIUSDT", "quantity": -1053.6, "current\_price": 0.7524, "money": 792.7591, "uTime": 1720075812436}

],

"Contracts": [

{"name": "KEYUSDT", "quantity": -148230, "current\_price": 0.00537, "uTime": 1720075812436},

{"name": "WIFUSDT", "quantity": -438.5, "current\_price": 1.8271, "uTime": 1720075812436},

{"name": "ETHUSDT", "quantity": -0.247, "current\_price": 3209.53, "uTime": 1720075812436}

]

}

## 4. 获取资金费

\*\*接口\*\*: GET /funding\_fee

\*\*描述\*\*: 获取账户的资金费记录。

\*\*URL\*\*: https://127.0.0.1:5001/funding\_fee

\*\*查询参数\*\*:

| 参数 | 类型 | 描述 |

|------------|--------|----------------|

| api\_key | string | 用于认证的 API 密钥。 |

| api\_secret | string | 用于认证的 API 密钥。 |

| startTime | int | 起始时间，默认值为0 |

| endTime | int | 结束时间，默认值为当前时间戳 |

\*\*响应\*\*: 返回账户的资金费记录。

\*\*响应示例\*\*:

[{

"uid": "573713111",

"ts": 17194475203000,

"symbol": "CFXUSDT",

"income": 0.00054173

}]

## 5. 平仓，平仓所有账户

\*\*接口\*\*: GET /doCloseAccount

\*\*描述\*\*: 初始化账户并开设仓位。

\*\*URL\*\*: https://127.0.0.1:5001/doCloseAccount

\*\*查询参数\*\*:

| 参数 | 类型 | 描述 |

|------------|--------|----------------|

| api\_key | string | 用于认证的 API 密钥。 |

| api\_secret | string | 用于认证的 API 密钥。 |

\*\*响应\*\*: 返回账户初始化的结果和当前状态。

\*\*响应示例\*\*:

{

"code": 203,

"message": "Task completed. You can redo it after 5 minutes."

}

\*\*响应字段\*\*:

| 字段 | 类型 | 描述 |

|-------------|--------|------------------|

| code | int | 返回状态码。 |

| message | string | 返回消息。 |

## 6. 用户账户是否是虚拟子账户

\*\*接口\*\*: GET /account\_type

\*\*描述\*\*: 用户账户是否是虚拟子账户

\*\*URL\*\*: https://127.0.0.1:5001/account\_type

\*\*查询参数\*\*:

| 参数 | 类型 | 描述 |

|------------|--------|----------------|

| api\_key | string | 用于认证的 API 密钥。 |

| api\_secret | string | 用于认证的 API 密钥。 |

\*\*响应\*\*:

\*\*响应示例\*\*:

{

"account\_type": "virtual" // virtual 代表是虚拟子账户， main 代表是主账户， 其他类型，待定

}

AIAM

<http://h5.aiam.me>

需要保证，合约和杠杆账户，都有BNB，才能用来充分抵扣手续费

更重要的，需要在杠杆账户留存，因为这样下单的数量才不会衰减

真神奇，重新启动后，就不报 {"code":-2010,"msg":"Account has insufficient balance for requested action."}' 这个错误了， 莫非是那个地方调用了资金归集？

reduceOnly 只减仓

平仓合约之后，合约的盈利，就会转移到合约的钱包，盈利的 USDT

2024-07-12 13:15:37,295 WARNING 400

2024-07-12 13:15:37,295 WARNING b'{"code":-2010,"msg":"Account has insufficient balance for requested action."}

下单还需要考虑 MIN\_NOTIONAL

所以，

平仓，对于合约需要只减仓的参数为 true， 现货不需要。

TODO 现货可能会存在很小的量，这个可能需要调用接口，来实现全部转换为 USDT，或者 BNB

2024-07-12 16:46:54,585 WARNING {'code': -1013, 'msg': 'Filter failure: NOTIONAL'}

TODO 目前还有一个点需要研究下，合约订单的浮亏，会多大程度影响调整后权益，也就是影响联合保证金率

联合保证金率 = 调整后权益 ÷ 维持保证金

调整后权益 = 钱包余额 + 初始保证金

钱包余额 =账户剩余USDT+∑各个资产估值\*资产折扣率）

调整后权益 =账户剩余USDT+初始保证金+∑各个资产估值\*资产折扣率）

联合保证金率 = 账户剩余USDT+初始保证金率+∑各个资产估值\*资产折扣率）÷ 维持保证金

0.0047422 \* 484483 ÷ 20 = 114.62

标记价格 \* 数量 ÷ 20倍 = 保证金（初始保证金）

虽然币安号称是统一账户，但是内部还是分成了3个账户1、全仓杠杆账户2、U本位合约账户3、币本位合约账户里面有些细节我研究透了，也跟大家同步下1、购买现货资产，需要全仓杠杆账户有剩余的USDT（否则下单会失败）2、U本位合约账户获取（失去）的USDT资金费，都会在U本位合约账户中，需要定期转入到全仓杠杆账户，才能用；3、U本位合约账户合约如果未平仓，合约的正向收益是不能用的，平仓后，USDT利润会在U本位合约账户中结算，结算后，转入全仓杠杆账户，再次开新的订单，才能实现我们的终极策略。4、U本位合约账户合约如果未平仓，如果收益是负数，目前还没有看见收取利息

合约账户的 负数 USDT 竟然不会资金划转。 检查下唐森的 账户 是否会收取利息

已经平仓了所有的合约，很神奇

{

"asset": "USDT",

"crossWalletBalance": "-168.90422428",

"crossUnPnl": "0.00000000",

"maintMargin": "0.00000000",

"initialMargin": "0.00000000",

"positionInitialMargin": "0.00000000",

"openOrderInitialMargin": "0.00000000",

"updateTime": 1720839129754

},

所以，平仓正收益的合约，资金转移后，去购买更多的仓位。\*\*\*

选币就 btc（30%）+eth（20%），剩下份额，就用你的算法，排除市值10亿美金以上的收益最高的前10个币。一共12个币。那10个币，我觉得可以平均分配，每人5%

选择10个币， 过滤器选项，市值大于10亿， 收益率最高

每个币种5%， 一万刀就是 500

建仓策略：1、比特币30%，ETH20%2、剩余9个币种，各 4.6%， 建仓达到50%+9\*4.6% = 91.4%3、查询剩余的USDT，全部建仓最后一个币种（因为剩余USDT是动态的，和建仓的币种有关，最终建仓率预估在95%左右）

对，就是定期轮训所有的用户账户， 看下是否有合约盈利超过20%的， 如果有，就平掉这个合约，然后重新建仓这个合约。 当然这个20%可以我们来后期调整