

模拟股市管理员端后台接口文档：

一、登录

请求Url: <http://ip:8080/login>

请求方式: POST

请求参数:

参数名字	参数类型	备注
account	String	登录名
password	String	登陆密码

请求示例:

POST http://localhost:8080/login?account=admin&password=123456

Params Authorization Headers (10) Body Pre-request Script Tests

Query Params

KEY	VALUE
<input checked="" type="checkbox"/> account	admin
<input checked="" type="checkbox"/> password	123456
Key	Value

返回示例:

登陆成功:

```
1 {
2   "status": 200,
3   "msg": "登录成功",
4   "data": null
5 }
```

登录失败:

```
{
  "status": 200,
  "msg": "用户名或密码错误",
  "data": null
}
```

二、获取当前用户（登录后保持登录状态，所以下一次访问要知道是谁来访问）

请求Url: <http://ip:8080/getAccount>

请求方式: GET

请求参数: 无

请求示例:

GET

http://localhost:8080/getAccount

Params

Authorization

Headers (10)

Body ●

Pre-request Script

Tests

Query Params

	KEY	VALUE
	Key	Value

请求成功:

```
1 {
2   "status": 200,
3   "msg": "ok",
4   "data": {
5     "account": "admin"
6   }
7 }
```

其他情况:

GET

http://localhost:8080/getAccount

Params

Authorization

Headers (10)

Body ●

Pre-request Script

Query Params

	KEY
	Key

Body

Cookies (1)

Headers (3)

Test Results

Pretty

Raw

Preview

JSON

```
1 {
2   "msg": "用户未登录",
3   "status": "401"
4 }
```

三、退出登录

请求Url: <http://ip:8080/logout>

请求方式: GET

请求参数: 无

请求示例:

GET
▼
http://localhost:8080/logout

Params
Authorization
Headers (10)
Body ●
Pre-req

Query Params

	KEY
	Key

Body
Cookies (1)
Headers (4)
Test Results

Pretty
Raw
Preview
JSON ▼

```

1 {
2   "status": 200,
3   "msg": "注销成功",
4   "data": null
5 }
```

其他情况:

```

1 {
2   "msg": "用户未登录",
3   "status": "401"
4 }
```

四、增加比赛

请求Url: <http://ip:8080/addMatch>

请求方式: POST

请求参数:

▼ Headers (1)		
	KEY	VALUE
<input checked="" type="checkbox"/>	Content-Type	application/json

参数	参数类型	备注
match_name	String	比赛名称
match_detail	String	比赛细节
match_rule	String	比赛规则
start_time	String	开始时间（需要转换成时间戳）
sign_time	String	比赛注册时间（需要转换成时间戳）
end_time	String	结束时间（需要转换成时间戳）
init_money	String	各用户初始金额

请求示例：

POST

http://localhost:8080/addMatch

Params

Authorization

Headers (10)

Body

Pre-request Script

Tests

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL BETA

JSON (application/json)

1

{

2

"match_name": "测试13",

3

"match_detail": "测试细节",

4

"match_rule": "测试规则",

5

"start_time": "1568044800",

6

"sign_time": "1568044800",

7

"end_time": "1568044800",

8

"init_money": "10000"

9

}

返回示例：

```
1  {
2    "status": 200,
3    "msg": "新增比赛成功",
4    "data": null
5  }
```

其他情况：

```
1  {
2      "status": 400,
3      "msg": "该比赛已存在",
4      "data": {
5          "id": 15,
6          "matchName": "测试12",
7          "matchDetail": "测试细节",
8          "matchRule": "测试规则",
9          "startTime": 1568044800,
10         "signTime": 1568044800,
11         "endTime": 1568044800,
12         "initMoney": 10000
13     }
14 }
```

五、获取所有比赛（一页十条）

请求Url: <http://ip:8080/getAllMatch>

请求方式: GET

请求参数:

请求参数	参数类型	备注
page	String	页码

请求示例:

GET

http://localhost:8080/getAllMatch?page=1

Params

Authorization

Headers (10)

Body

Pre-request Script

Query Params

	KEY
<input checked="" type="checkbox"/>	page
	Key

Body

Cookies (1)

Headers (3)

Test Results

Pretty

Raw

Preview

JSON

```

1  {
2      "status": 200,
3      "msg": "ok",
4      "data": [
5          {
6              "id": 1,
7              "matchName": "测试0",
8              "matchDetail": "测试细节0",
9              "matchRule": "测试规则0",
10             "startTime": 1568044800,
11             "signTime": 1568044800,
12             "endTime": 1568044800,
13             "initMoney": 10000
14         },
15         {
16             "id": 2,
17             "matchName": "测试1",
18             "matchDetail": "测试细节1",
19             "matchRule": "测试规则1",
20             "startTime": 1568044800,
21             "signTime": 1568044800,
22             "endTime": 1568044800,
23             "initMoney": 10000
24         },
25         {
26             "id": 3,
27             "matchName": "测试2",
28             "matchDetail": "测试细节",
29             "matchRule": "测试规则"
30         }
31     ]
32 }
```

六、修改比赛

请求Url: <http://ip:8080/updateMatch>

请求方式: POST

请求参数:

▼ Headers (1)		
	KEY	VALUE
<input checked="" type="checkbox"/>	Content-Type	application/json

参数	参数类型	备注
id	String	比赛的id
match_name	String	比赛名称
match_detail	String	比赛细节
match_rule	String	比赛规则
start_time	String	开始时间（需要转换成时间戳）
sign_time	String	比赛注册时间（需要转换成时间戳）
end_time	String	结束时间（需要转换成时间戳）
init_money	String	各用户初始金额

请求示例

```

1 {
2   "id": "1",
3   "match_name": "测试0",
4   "match_detail": "测试细节0",
5   "match_rule": "测试规则0",
6   "start_time": "1568044800",
7   "sign_time": "1568044800",
8   "end_time": "1568044800",
9   "init_money": "10000"
10 }

```

返回示例

```

1 {
2   "status": 200,
3   "msg": "ok",
4   "data": null
5 }

```

其他情况:

```

1 {
2   "status": 400,
3   "msg": "该比赛不存在",
4   "data": null
5 }

```

七、删除比赛

请求Url: <http://ip:8080/deleteMatch>

请求方式: POST

请求参数:

参数	类型	备注
id	String	比赛Id

请求示例

POST

http://localhost:8080/deleteMatch?id=13

Params

Authorization

Headers (1)

Body

Pre-request Script

Tests

Query Params

	KEY	VALUE
<input checked="" type="checkbox"/>	id	13

返回示例:

```
1 {
2   "status": 200,
3   "msg": "ok",
4   "data": null
5 }
```

其他示例:

```
1 {
2   "status": 400,
3   "msg": "该比赛不存在",
4   "data": null
5 }
```

八、查询某个比赛

请求Url: <http://ip:8080/getMatch>

请求方式: GET

请求参数:

参数	类型	备注
match_name	String	比赛名字

请求示例:

GET

http://localhost:8080/getMatch?match_name=测试1

Params

Authorization

Headers (8)

Body

Pre-request Script

Tests

Query Params

	KEY	VALUE
<input checked="" type="checkbox"/>	match_name	测试1
	Key	Value


```
{
  "status": 200,
  "msg": "ok",
  "data": {
    "id": 2,
    "matchName": "测试1",
    "matchDetail": "测试细节1",
    "matchRule": "测试规则1",
    "startTime": 1568044800,
    "signTime": 1568044800,
    "endTime": 1568044800,
    "initMoney": 10000
  }
}
```

其他示例：

```
1  {
2      "status": 200,
3      "msg": "无该比赛记录",
4      "data": null
5  }
```

九、拉黑用户

请求Url: <http://ip:8080/addBlackList>

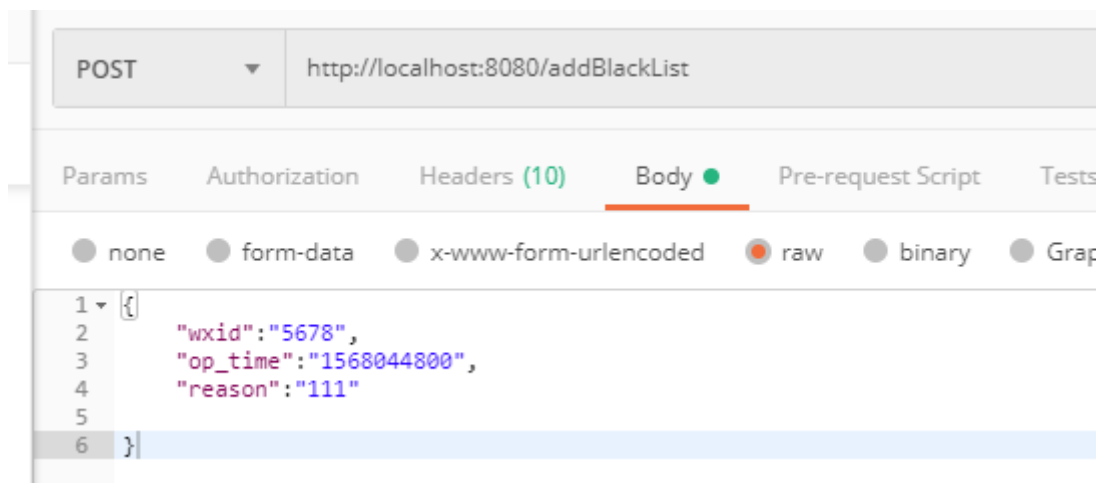
请求方式: POST

请求参数:

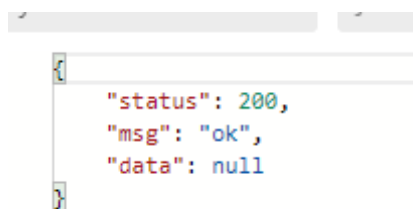
	KEY	VALUE
<input checked="" type="checkbox"/>	Content-Type	application/json
	Key	Value

请求示例:

参数	类型	备注
wxid	String	微信id
op_time	String	拉黑时间（转换成时间戳）
reason	String	拉黑原因



返回示例:



其他示例:



十、获取所有被拉黑的用户（一页十条）

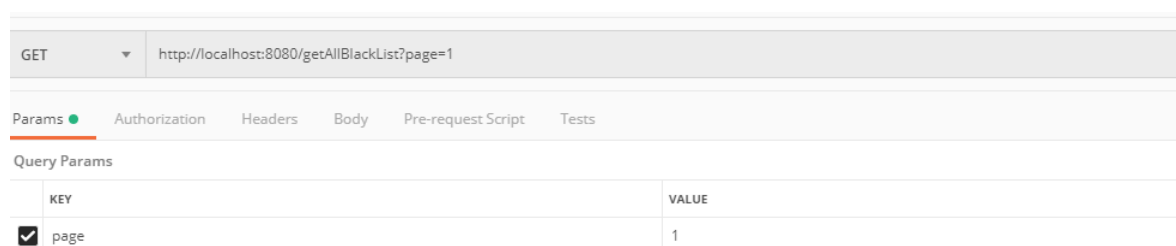
请求Url: <http://ip:8080/addAllBlackList>

请求方式: GET

请求参数:

参数	类型	备注
page	String	页数

请求示例:



返回示例:

```
1 {
2   "status": 200,
3   "msg": "ok",
4   "data": [
5     {
6       "id": 1,
7       "wxid": "1234",
8       "reason": "111",
9       "opTime": 1568044800
10    },
11    {
12      "id": 3,
13      "wxid": "5678",
14      "reason": "111",
15      "opTime": 1568044800
16    }
17  ]
18 }
```

十一、取消拉黑

请求Url: <http://ip:8080/deleteBlackList>

请求方式: POST

请求参数:

参数	类型	备注
id	String	用户id, 不是wxid

请求示例:

POST

▼

http://localhost:8080/deleteBlackList?id=2

Params

Authorization

Headers

Body

Pre-request Script

Tests

Query Params

	KEY	VALUE
<input checked="" type="checkbox"/>	id	2
	Key	Value

返回示例:

```
1 {
2   "status": 200,
3   "msg": "ok",
4   "data": null
5 }
```

其他示例:

```
Pretty Raw Preview JSON
1 {
2   "status": 400,
3   "msg": "该用户不存在",
4   "data": null
5 }
```

十二、查询某个拉黑用户

请求Url: <http://ip:8080/getBlackList>

请求方式: GET

请求参数:

参数	类型	备注
wxid	String	微信id

请求示例:

GET http://localhost:8080/getBlackList?wxid=1234

Params Authorization Headers (8) Body Pre-request Script Tests

Query Params

	KEY	VALUE
<input checked="" type="checkbox"/>	wxid	1234
	Key	Value

返回示例:

Body Cookies (1) Headers (3) Test Results

Pretty Raw Preview JSON

```
1 {
2   "status": 200,
3   "msg": "ok",
4   "data": {
5     "id": 1,
6     "wxid": "1234",
7     "reason": "111",
8     "opTime": 1568044800
9   }
10 }
```

其他示例:

```
1 {  
2   "status": 200,  
3   "msg": "无该用户记录",  
4   "data": null  
5 }
```