

账号相关

用token直接登录

- 无参数
- 成功/失败

请求头携带本地token访问/api/account/login接口，接口会验证token的正确性，若正确则创建一个新的token返回，之后用这个新的token来进行后续操作
这样对于三天内打开过应用的人，就可以一直免登录

设置头像

- 参数 “profilePic”
- 返回：成功/失败

[/api/account/set-profile](#)

参数为“profile”，可使用form-data上传multipartfile文件

获取头像

- 参数无
- 返回 “profilePic”

可使用/api/account/info接口获取用户信息，其中包含头像的url，如“17030888015481.jpg”

此时访问<http://43.138.75.58:8080/static/17030888015481.jpg>即可获取到该图片

步数相关

步数（不是接口）

后端：每天00:00 重置每个人的今日步数为0

数据库是每天每个用户记录一个步数，所以会“自动”清零



获取步数数据 (by date/ by range-date)

参数:

"date": "2023-01-01"

"accountId": (非必须)

参数:

"startDate": "2023-12-12",

"endDate": "2023-12-13"

"accountId": (非必须)

[/api/sports/steps](#)

没有任何参数时获取当前用户当天的步数

可选填accountId参数获取某个用户步数

可选填"date"获取某天的步数

"startDate"和endDate同理

一天只会会有一个数据

返回:

```
{
  "data": [
    {"date": "2023-12-12",
     "steps": 200},
    {"date": "2023-12-13",
     "steps": 0 },
  ]
}
```

运动相关



获取运动数据 (by date)

参数:

“date”: “2023-12-12”

“accountId”: (非必须)

和下一个需求都可以访问
</api/sports/exercise/list>
参数要求同步数的获取

返回: (数组)

```
{  
  "data": [  
    {"start": "2023-12-12 22:13",  
      "end": "2023-12-12 23:32",  
      "duration": "100",  
      "type": int类型,  
      "feelings": int类型,  
      "remark": string类型}, {}, {}, ...  
  ]  
}
```

关于duration怎么计算后面会说到，不只是根据start end而已，中间会有暂停



获取运动数据 (by date range)

参数:

"startDate": "2023-12-12",

"endDate": "2023-12-13"

"accountId": (非必须)

返回: (数组)

```
{  
  "data": [  
    {"start": "2023-12-12 22:13",  
      "end": "2023-12-12 23:32",  
      "duration": "100",  
      "type": int类型,  
      "feelings": int类型,  
      "remark": string类型}, {}, {}, ...  
  ]  
}
```




开始运动

参数:

```
{  
  "type": int类型  
  "time": "2023-01-01 02:09"  
  "accountId": (非必须)  
}
```

返回:

成功/失败

后端:

添加开始的时间

Time = ["2023-01-01 02:09",];

可以用一个数组来表示。

isExercising = true;

看第二页注意



开始运动

假设现在是19号，如果要在18号开始运动，是不行的
即：保证只有今天会添加数据

也保证同一时候只会有一个会进行。





暂停运动

参数:

"time": ", "2023-01-01 02:30";

"accountId": (非必须)

返回:

成功/失败

后端:

添加暂停的时间

Time = ["2023-01-01 02:09", "2023-01-01 02:30"];

isPausing = true;



继续运动

参数:

time: "2023-01-01 02:31"

"accountId": (非必须)

返回:

成功/失败

后端:

现在的时间即为继续的时间

Time = ["2023-01-01 02:09", "2023-01-01 02:30",
"2023-01-01 02:31"];

isPausing = false;



修改活动 🏆

2023年12月19日 (二) 📅

今日步数 🏃 2460 步

运动中 🏃

开始时间: 14:42:55

运动类型: 排球

暂停运动

结束运动

开始时间: 2023-12-12 22:13

结束时间: 2023-12-12

总时长: 100分钟

运动类型: 羽毛球

感觉: 较差

运动感受与备注

运动感受:

开心

还好

不好

备注:

取消

开始

结束运动

参数:

time: "2023-01-01 03:01"

"feelings": 1

"remark": 没有则没这参数

"accountId": (非必须)

返回:

成功/失败

后端:

现在的时间即为结束的时间

Time = ["2023-01-01 02:09", "2023-01-01 02:30",
"2023-01-01 02:31", "2023-01-01 03:01"];

isExercising = false;

计算duration (见下一页)

如何计算duration(分钟)

没有暂停情况:

Time = ["2023-01-01 02:09", "2023-01-01 03:01"];

duration = "2023-01-01 03:01" - "2023-01-01 02:09" = xx (单位: 分钟)

有暂停情况:

Time = ["2023-01-01 02:09", "2023-01-01 02:30", "2023-01-01 02:31", "2023-01-01 03:01"];

a

b

c

d

duration = (b-a) + (d-c) = xx 分钟

有多次暂停情况:

duration = (b-a) + (d-c) + (f-e) + (h-g) ... = xx 分钟



获取当前有没有在运动

参数:

“date”: “2023-01-01”

“accountId”: (非必须)

如果只能创建今天的运动那date参数就没必要了?
没有运动时data为null

运动时:

没有运动时:

返回: {
“isExercising”: 1,
“isPausing”: 0,
“startTime”: “2023-01-01 02:09”,
“type”: int类型,
}

返回: {
“isExercising”: 0,
}

“isPausing”: 0表示在暂停状态, 1在运行状态



获取今日已运动时长总和

参数:

“accountId”: (非必须)

返回: {
“duration”: 132,
}

分钟

“isPausing”: 0表示在暂停状态, 1在运行状态

运动+步数

统计并入了筛选过滤的接口

统计

参数：
“startDate”: “2023-12-12”,
“endDate”: “2023-12-13”
“accountId”: (非必须)

[0,500)	3
[500,1000)	2
[1000,3000)	5
>=3000	0

[0,30)	3
[30,60)	1
[60, 120)	6
>=120	4

返回：
{
 “data”: [
 {“name”: “steps”,
 “less”: 3,
 “fewer”:2
 “medium”:5
 “more”: 0},

 {“name”: “exercise”,
 “less”: 3,
 “fewer”:1
 “medium”:6
 “more”: 4},
]
}

参数:

“startDate”: “2023-12-12”,

“endDate”: “2023-12-31”

“startTime”: “09:02”

“endTime”: “12:12”

“minDuration”: 0

“maxDuration”: 240

“feelings”: “0110” //和血压那些的一样原理

“remark”: “01” //和血压那些的一样原理

筛选过滤

返回:

“data”:

{

“exerciseList”: [

{“start”: “2023-12-12 22:13”,

“end”: “2023-12-12 23:32”,

“duration”: “100”,

“type”: int类型,

“feelings”: int类型,

“remark”: string类型},

{}, {}, ...

],

“countedDataList”: [

{“name”: “steps”,

“less”: 3,

“fewer”:2

“medium”:5

“more”: 0},

{“name”: “exercise”,

“less”: 3,

“fewer”:1

“medium”:6

“more”: 4},

]