

接口9_10

updated 2016/9/9 23:34:11

目前已实现接口

login

```
-> success-User;cherry-aaa-叶沁媛-1-310104199709091234-13141344286-中国-Adult-null-1110
or success-Administrator;cherry-979988-叶沁媛-13141344286-Boss

-> xpasswd
-> xuser
```

register

```
<- register
<- cherry-aaa-叶沁媛-Male-中国-310104199709091234-13141344286-Adult-清华大学,2015011469

-> success
-> username
-> xID
-> server unexpected error!
```

查询已购车票

```
<- ticketbought-UserName
->
```

```
8;(traininfo1);(traininfo2);..
traininfo还是: 车次名-始发站-终点站-始发时间(yyyy/mm/dd hh:mm)-到终点时间-TrainType-座位-票价-TicketType
```

按条件查询车次。

```
<- searchtrain
始发站-终点站-始发时间(yyyy/mm/dd hh:mm)-终点时间
->
```

```
8;(traininfo1);(traininfo2);..
traininfo:车次名-始发站-终点站-始发时间(yyyy/mm/dd hh:mm)-到终点时间-TrainType-price-剩余票数-0-0 (1就是过)
```

查询一辆列车的中间站

```
<- midstation-G20-2016/08/20 15:00
->
```

```
8;(traininfo1);(traininfo2);...
traininfo:车次名-始发站-终点站-始发时间(yyyy/mm/dd hh:mm)-到终点时间-stationid
```

买票

```
<- buyticket
<- cherry-G4-starttime-origin-destination-tickettype

-> success
-> xticket
-> server unexpected error!
```

退票

```
<- returnticket
```

```
<- cherry-G22-2016/09/09 17:00
```

```
-> success
```

```
-> xticket
```

修改用户信息

```
<- updateinfo
```

```
<- cherry979988-password-叶沁媛-Male-其他国家-31010419970909242X-13141344286-Adult
```

```
-> success
```

```
-> unexpected error
```

目前bug

null

未完成接口约定

查询可购座位：

//这个貌似不做了对吧OAO

```
<- seatbought
```

```
<- 车次名-始发时间(yyyy/mm/dd hh:mm)-到终点时间-始发站stationid-终点站stationid-stationNum(这个是指这趟车一共几段,stationid0~5就是6)-cartype
```

这里需要说明下始发站stationid-终点站stationid是什么东西：

如果是A->B->C->D，

A->C 就是0-1

B->C 就是1-1

A->D 就是0-2

有可能要你提供这些有点难...不行再说，关键我这里算这些东西总觉得不太靠谱...

由于我这里买票的数据少，我就返回已经卖出的票好了

```
->3-1-60-100
```

第一个数是已卖出的票数，后面三个分别是已卖出的座位

用户登出

```
<- logout-cherry
```

```
-> success
```

```
-> unexpected error
```

已经实现，没测试过。

添加列车信息

```
<- addtrain-TrainId-StartTime-Endtime-origin-destination-cartype-length-stationid-0（0是永久添加，1是临时添加）-days
```

务必看过来！！这个有变

最后一个days在永久添加时没有作用，但是临时添加时可以决定加几天(如果你发0，那就相当于没有加哦)

永久添加时你随便发个过来就行（要发哦，否则我这里[]out of range）。

都是从我发的那个starttime(yyyy/MM/dd hh:mm)开始添加。

如果有中间站，再发一次请求，你把stationid改了就行

麻烦你了orz要帮我挡掉非法添加哟

已经实现。做了一点测试：

相应速度明显变慢，原因很明显，数据库添加操作是最慢的。

所以建议临时添加天数不要超过10，否则就是很慢

永久添加大概响应需要1.5s左右。（在两张表里加数据嘛，慢慢慢）

-> success
-> exists(这车已经存在了，不能添加)
-> unexpected error

修改列车信息

其实跟修改用户信息一样

<- updatetraininfo-TrainId-StartTime-Endtime-origin-destination-cartype-length-stationid-0(0是永久修改，1是临时修改)-days

合理性也需要你来挡住OAO，我只管改数据

如果是永久的，会从这个starttime之后开始修改信息，有个程序默认天数，不是根据你给的days。

如果是临时，连续天大于存在的天，我会停止修改，然后返回成功（就哪怕只有一天的车，你修改两天，我只会修改一天，返回成功）

最后也加上days,原因同上。

修改列车时，starttime先别修改（没实现），因为这个改了怎么定位？？看来什么能改什么不能改你还要想想。

已实现。做了点测试，大概也要1.5s左右QAQ

-> success
-> success but xpermanent（这个只是说明下临时车不能永久修改，你可以不处理，也可以弹个窗口之类的）
-> unexpected error

删除列车

<- deletetrain-trainid-starttime-0-days（0是永久删除，1是临时删除）

最后一个days跟上面一样，临时删才起作用

如果你需要删除整辆车，比方说有6段，你需要发6次(有问题再说吧...)

-> success
-> success but xpermanent
-> unexpected error

修改管理员信息

<- updateadmininfo

<- cherry-979988-叶沁媛-13141344286-Boss

-> success
-> unexpected error

注意事项

- 现在添加列车和修改列车，每一步的相应都非常慢，我觉得会出bug，你做做测试。
- 开服务器有可能会长达半分钟，不是卡，就是在初始化数据（当然不大可能，除非很长时间不开了）

最后任务？

你当然先把管理员界面写完。。。中间遇到bug反馈给我。

我不打算修小bug了，想想怎么写报告了。