

接口文档

以下参数是使用postman调用， 参数格式是x-www-form-urlencoded
返回都是采用以下格式：

```
{
  "code":xxx,
  "status": "xxx",
  "message": "xxx"
}
```

用户(user)的接口:

1:用户注册:

url: http://localhost:8080/design/user/register.do

method: POST

参数:

account: account
password: password
name: name
telephone: telephone

返回:

```
{
  "code": 200,
  "status": "ok",
  "message": "登录成功/账号已经存在/账号或密码包含空格"
}
```

2:用户登录:

url: http://localhost:8080/design/user/login.do

method: POST

参数:

account: account
password: password

返回:

```
{
  "code": 200,
  "status": "ok",
  "message": "登录成功/密码错误/账号不存在"
}
```

3:获取用户信息:

url: http://localhost:8080/design/user/get_person.do

method: POST

参数:

account: account

返回:

```
{
  "code": 200,
  "status": "ok",
  "message": "该用户不存在"
}
{
  "code": 200,
  "status": "ok",
  "message": {
    "id": 3,
    "account": "newAcc",
    "password": "d8mf5hssd8liitg6ucv1pcvbq52r0l1e",
    "name": "isWho",
  }
}
```

```
        "telephone": "telephone"
    }
}
```

4:更改密码:

url: http://localhost:8080/design/user/forget_password.do
method: POST

参数: account: account
oldPassword: oldPassword
newPassword: newPassword

返回:

```
{
    "code": 200,
    "status": "ok",
    "message": "密码修改成功/输入的基本信息不正确/账号不存在"
}
```

5:忘记密码:

url: http://localhost:8080/design/user/change_password.do
method: POST

参数:
account: account
password: password
name: name
telephone: telephone

返回:

```
{
    "code": 200,
    "status": "ok",
    "message": "密码修改成功/旧密码输入错误/账号不存在"
}
```

6:更改用户信息:

url: http://localhost:8080/design/user/update_person.do
method: POST

参数:
account: account
password: password
name: name
telephone: telephone

返回:

```
{
    "code": 200,
    "status": "ok",
    "message": "账号不存在/输入的基本信息不正确/基本信息修改成功"
}
```

7:大食堂订餐:

url: http://localhost:8080/design/user/update_person.do
method: POST

参数:
account: account
password: password
name: name
telephone: telephone

返回:

```
{
  "code": 200,
  "status": "ok",
  "message": {
    "id": 3,
    "account": "newAcc",
    "password": "-caou3n22ki6oveam3h5kg95889p926bs",
    "name": "isWho?",
    "telephone": "telephone"
  }
}
```

8:更改用户信息:

url: http://localhost:8080/design/user/update_person.do
method: POST

参数: account: account
password: password
name: name
telephone: telephone

返回: {
"code": 200,
"status": "ok",
"message": "账号不存在/输入的基本信息不正确/基本信息修改成功"
}

9:获取用户信息:

url: http://localhost:8080/design/user/getPerson.do
method: POST

参数: account: account
password: password
name: name
telephone: telephone

返回:
{
"code": 200,
"status": "ok",
"message": "账号不存在/输入的基本信息不正确/基本信息修改成功"
}

厨师(cooker)的接口:

1:插入菜单:

url: http://localhost:8080/design/cooker/add_menu.do
method: POST

参数:
name:name
type:type
price:price
season:season

返回:
{
"code": 200,
"status": "ok",
"message": "插入新菜品成功"
}

2:查询菜单数量:

url: http://localhost:8080/design/cooker/get_menu_number.do
method: GET

返回:

```
{
  "code": 200,
  "status": "ok",
  "message": {
    "number": "xxx"
  }
}
```

3:移除菜单:

url: http://localhost:8080/design/cooker/remove_menu.do
method: POST

参数:

id:id
name:name

返回:

```
{
  "code": 200,
  "status": "ok",
  "message": "删除菜品成功"
}
{
  "code": 200,
  "status": "ok",
  "message": "该菜品不存在"
}
```

4:更新菜单:

url: http://localhost:8080/design/cooker/update_menu.do
method: POST

参数:

id:id
name:name
type:type
price:price
season:season

返回:

```
{
  "code": 200,
  "status": "ok",
  "message": "更新菜品成功"
}
```

5:查询菜单某个菜品的具体信息:

url: http://localhost:8080/design/cooker/check_detail.do
method: POST

参数:

id:id
name:name
type:type
price:price
season:season

返回:

```
{
  "code": 200,
  "status": "ok",
  "message": {
    "id": 4,
    "name": "name4",

```

```

        "type": "type2",
        "price": 123,
        "season": "sea"
    }
}
{
    "code": 200,
    "status": "ok",
    "message": "该菜品不存在"
}

```

6:分页查询菜单:

url: http://localhost:8080/design/cooker/select_menu.do
method: POST

参数:

start:start
end:end

返回:

```

{
    "code": 200,
    "status": "ok",
    "message": [
        {
            "id": 1,
            "name": "name",
            "type": "type?",
            "price": 133,
            "season": "Sps"
        },
        {
            "id": 2,
            "name": "name",
            "type": "type",
            "price": 13,
            "season": "Sp"
        }
    ]
}

```

7:按照name模糊分页查询菜单:

url: http://localhost:8080/design/cooker/select_menu_by_name.do
method: POST

参数:

name:name
start:start
end:end

返回:

```

{
    "code": 200,
    "status": "ok",
    "message": [
        {
            "id": 1,
            "name": "name",
            "type": "type?",
            "price": 133,
            "season": "Sps"
        },
        {
            "id": 5,

```

```
        "name": "name1",
        "type": "type1",
        "price": 12,
        "season": "sea"
    }
]
}
```

8:按照season模糊分页查询菜单:

url: http://localhost:8080/design/cooker/select_menu_by_season.do
method: POST

参数:

season:season
start:start
end:end

返回:

```
{
  "code": 200,
  "status": "ok",
  "message": [
    {
      "id": 4,
      "name": "name4",
      "type": "type2",
      "price": 123,
      "season": "sea"
    },
    {
      "id": 5,
      "name": "name1",
      "type": "type1",
      "price": 12,
      "season": "sea"
    }
  ]
}
```

9:按照type模糊分页查询菜单:

url: http://localhost:8080/design/cooker/select_menu_by_type.do
method: POST

参数:

type:type
start:start
end:end

返回:

```
{
  "code": 200,
  "status": "ok",
  "message": [
    {
      "id": 2,
      "name": "name",
      "type": "type",
      "price": 13,
      "season": "Sp"
    },
    {
      "id": 3,
      "name": "third",
      "type": "type",

```

```
        "price": 13,  
        "season": "Sp"  
    }  
]  
}
```

10:查询大食堂菜单数量:

url: http://localhost:8080/design/cooker/get_public_number.do

method: GET

返回:

```
{  
  "code": 200,  
  "status": "ok",  
  "message": {  
    "number": "xxx"  
  }  
}
```

11:在大菜单中选取食堂菜单:

url: http://localhost:8080/design/cooker/select_menu_as_public.do

method: POST

参数:

id:id

返回:

```
{  
  "code": 200,  
  "status": "ok",  
  "message": "选择成功"  
}
```

12:移除食堂菜单:

url: http://localhost:8080/design/cooker/remove_menu_public.do

method: POST

参数:

id:id

返回:

```
{  
  "code": 200,  
  "status": "ok",  
  "message": "移出成功"  
}
```

13:分页查询食堂菜单:

url: http://localhost:8080/design/cooker/select_public_menus.do

method: POST

参数:

start:start

end:end

返回:

```
{  
  "code": 200,  
  "status": "ok",  
  "message": [  
    {  
      "id": 1,  
      "menuId": 1,  
      "name": "name",  
      "type": "type?",  
      "price": 133,  
      "season": "Sp",  
      "category": "category",  
      "description": "description"  
    }  
  ]  
}
```

```
        "season": "Sps"
    }
  ]
}
```

14:按照季节分页查询食堂菜单:

url: http://localhost:8080/design/cooker/select_public_menu_by_season.do
method: POST

参数:

season:season
start:start
end:end

返回:

```
{
  "code": 200,
  "status": "ok",
  "message": [
    {
      "id": 3,
      "menuId": 4,
      "name": "name4",
      "type": "type2",
      "price": 123,
      "season": "sea"
    }
  ]
}
```

15:按照名称分页查询食堂菜单:

url: http://localhost:8080/design/cooker/select_public_menu_by_name.do
method: POST

参数:

name:name
start:start
end:end

返回:

```
{
  "code": 200,
  "status": "ok",
  "message": [
    {
      "id": 1,
      "menuId": 1,
      "name": "name",
      "type": "type?",
      "price": 133,
      "season": "Sps"
    }
  ]
}
```

16:按照type分页查询食堂菜单:

url: http://localhost:8080/design/cooker/select_public_menu_by_type.do
method: POST

参数:

type:type
start:start
end:end

返回:

```
{
```



```
"code": 200,
"status": "ok",
"message": [
  {
    "id": 1,
    "menuId": 1,
    "name": "name",
    "type": "type?",
    "price": 133,
    "season": "Sps"
  }
]
```

17:查询餐厅菜单的数量:

url: http://localhost:8080/design/cooker/get_private_number.do

method: GET

返回:

```
{
  "code": 200,
  "status": "ok",
  "message": {
    "number": "2"
  }
}
```

18:在大菜单中选取餐厅菜单:

url: http://localhost:8080/design/cooker/select_menu_as_private.do

method: POST

参数:

id:id

返回:

```
{
  "code": 200,
  "status": "ok",
  "message": "选择成功"
}
```

19:移除餐厅菜单:

url: http://localhost:8080/design/cooker/remove_menu_private.do

method: POST

参数:

id:id

返回:

```
{
  "code": 200,
  "status": "ok",
  "message": "移出成功"
}
```

20:分页查询餐厅菜单:

url: http://localhost:8080/design/cooker/select_private_menus.do

method: POST

参数:

start:start

end:end

返回:

```
{
  "code": 200,
  "status": "ok",

```

```
"message": [  
  {  
    "id": 1,  
    "menuId": 1,  
    "name": "name",  
    "type": "type?",  
    "price": "133",  
    "season": "Sps"  
  }  
]
```

21:分页查询餐厅菜单:

url: http://localhost:8080/design/cooker/select_private_menu_by_season.do

method: POST

参数:

season:season
start:start
end:end

返回:

```
{  
  "code": 200,  
  "status": "ok",  
  "message": [  
    {  
      "id": 3,  
      "menuId": 4,  
      "name": "name4",  
      "type": "type2",  
      "price": "123",  
      "season": "sea"  
    }  
  ]  
}
```

22:通过name分页查询餐厅菜单:

url: http://localhost:8080/design/cooker/select_private_menu_by_name.do

method: POST

参数:

name:name
start:start
end:end

返回:

```
{  
  "code": 200,  
  "status": "ok",  
  "message": [  
    {  
      "id": 1,  
      "menuId": 1,  
      "name": "name",  
      "type": "type?",  
      "price": "133",  
      "season": "Sps"  
    }  
  ]  
}
```

23:通过type分页查询餐厅菜单:

url: http://localhost:8080/design/cooker/select_private_menu_by_type.do

method: POST

参数:

type:type
start:start
end:end

返回:

```
{
  "code": 200,
  "status": "ok",
  "message": [
    {
      "id": 1,
      "menuId": 1,
      "name": "name",
      "type": "type?",
      "price": "133",
      "season": "Sps"
    }
  ]
}
```

24:分页查询餐厅已点的菜单 (type, useTime) :

url: http://localhost:8080/design/cooker/select_ordered_private_menus.do

method: POST

参数:

type:type
useTime:useTime
start:start
end:end

返回:

```
{
  "code": 200,
  "status": "ok",
  "message": [
    {
      "id": 1,
      "menuId": 1,
      "account": "acca",
      "name": "name",
      "type": "breakfast",
      "price": 133,
      "season": "Sps",
      "orderTime": "2019-02-04 12:22:45.0",
      "useTime": "2019-02-05",
      "number": 1
    }
  ]
}
```

25:分页查询食堂已点的菜单 (type, useTime) :

url: http://localhost:8080/design/cooker/select_ordered_public_menus.do

method: POST

参数:

type:type
useTime:useTime
start:start
end:end

返回:

```
{
  "code": 200,
```

```
    "status": "ok",
    "message": [
      {
        "id": 1,
        "menuId": 1,
        "account": "acca",
        "name": "name",
        "type": "breakfast",
        "price": 133,
        "season": "Sps",
        "orderTime": "2019-02-04 12:22:45.0",
        "useTime": "2019-02-05",
        "number": 1
      }
    ]
  }
}
```

26:分页查询食堂菜单:

url: http://localhost:8080/design/cooker/select_ordered_public_menu_numbers.do
method: POST

参数:

type:type
useTime:useTime
start:start
end:end

返回:

```
{
  "code": 200,
  "status": "ok",
  "message": [
    {
      "id": 0,
      "menuId": 0,
      "account": "acca",
      "name": "name",
      "type": "breakfast",
      "price": 133,
      "season": "Sps",
      "orderTime": "2019-02-06 17:55:08.0",
      "useTime": "2019-02-05",
      "number": 1
    }
  ]
}
```

27:分页查询餐厅菜单:

url: http://localhost:8080/design/cooker/select_ordered_private_menu_numbers.do
method: POST

参数:

type:type
useTime:useTime
start:start
end:end

返回:

```
{
  "code": 200,
  "status": "ok",
  "message": [
    {
      "id": 0,
```

```

        "menuId": 0,
        "account": "acca",
        "name": "name",
        "type": "breakfast",
        "price": 133,
        "season": "Sps",
        "orderTime": "2019-02-04 12:22:45.0",
        "useTime": "2019-02-05",
        "number": 1
    }
  ]
}

```

客户(customer)的接口:

1:用户食堂点餐:

url: http://localhost:8080/design/customer/select_ordered_as_public.do
method: POST

参数:

```

menuId: menuId
account: account
type: type
useTime: useTime
number: number

```

返回:

```

{
  "code": 200,
  "status": "ok",
  "message": "订餐成功"
}

```

2:用户取消食堂的订餐:

url: http://localhost:8080/design/customer/remove_ordered_public.do
method: POST

参数:

```

menuId: menuId
account: account
type: type
useTime: useTime

```

返回:

```

{
  "code": 200,
  "status": "ok",
  "message": "取消订餐成功"
}

```

3:用户查询食堂的订餐:

url: http://localhost:8080/design/customer/select_ordered_public_by_customer.do
method: POST

参数:

```

account: account
password: password
name: name
telephone: telephone

```

返回:

```

{
  "code": 200,
  "status": "ok",

```

```
    "message": [
      {
        "id": 2,
        "menuId": 1,
        "account": "acc",
        "name": "name",
        "type": "breakfast",
        "price": 133,
        "season": "Sps",
        "orderTime": "2019-03-21 19:19:35.0",
        "useTime": "2019-03-21",
        "number": 1
      }
    ]
  }
}
```

4:用户餐厅点餐:

url: http://localhost:8080/design/customer/select_ordered_as_private.do

method: POST

参数:

menuId: menuId
account: account
type: type
useTime: useTime

返回:

```
{
  "code": 200,
  "status": "ok",
  "message": "订餐成功"
}
```

5:用户取消餐厅的订餐:

url: http://localhost:8080/design/customer/remove_ordered_private.do

method: POST

参数:

menuId: menuId
account: account
type: type
useTime: useTime

返回:

```
{
  "code": 200,
  "status": "ok",
  "message": "取消订餐成功"
}
```

6:用户查询餐厅的订餐:

url: http://localhost:8080/design/customer/select_ordered_private_by_customer.do

method: POST

参数:

account: account
password: password
name: name
telephone: telephone

返回:

```
{
  "code": 200,
  "status": "ok",
  "message": [
```

```

        {
            "id": 2,
            "menuId": 1,
            "account": "acc",
            "name": "name",
            "type": "breakfast",
            "price": 133,
            "season": "Sps",
            "orderTime": "2019-03-21 19:19:35.0",
            "useTime": "2019-03-21",
            "number": 1
        }
    ]
}

```

流水单以及日常收入支出(manager)的接口:

1:插入日常的账单:

url: http://localhost:8080/design/manager/insert_account.do
method: POST

参数:

date: date
name: name
message: message
money: money
paidFlag: paidFlag (0代表未付款)

返回:

```

{
    "code": 200,
    "status": "ok",
    "message": "插入账单成功"
}

```

2:修改日常的账单:

url: http://localhost:8080/design/manager/update_account.do
method: POST

参数:

id: id
date: date
name: name
message: message
money: money
paidFlag: paidFlag (0代表未付款)

返回:

```

{
    "code": 200,
    "status": "ok",
    "message": "更新账单成功"
}

```

3:删除日常的账单:

url: http://localhost:8080/design/manager/delete_account.do
method: POST

参数:

id: id

返回:

```

{
    "code": 200,
    "status": "ok",
}

```

```
        "message": "删除账单成功"
    }
}
```

4:支付日常的账单:

url: http://localhost:8080/design/manager/account_paid.do

method: POST

参数:

id: id

返回:

```
{
  "code": 200,
  "status": "ok",
  "message": "账单支付成功"
}
```

5:按照时间分页查找日常的账单:

url: http://localhost:8080/design/manager/select_account_by_date.do

method: POST

参数:

dateBegin: dateBegin

dateEnd: dateEnd

start: start

end: end

返回:

```
{
  "code": 200,
  "status": "ok",
  "message": [
    {
      "id": 1,
      "date": "2019-02-06",
      "name": "name",
      "message": "new ",
      "money": 10,
      "paidFlag": 0
    }
  ]
}
```

6:插入日常的支出:

url: http://localhost:8080/design/manager/insert_expand.do

method: POST

参数:

date: date

type: type (in/out)

message: message

money: money

返回:

```
{
  "code": 200,
  "status": "ok",
  "message": "插入成功"
}
```

7:删除日常的支出:

url: http://localhost:8080/design/manager/delete_expand.do

method: POST

参数:

id: id

返回:

```
{
  "code": 200,
  "status": "ok",
  "message": "删除成功"
}
```

8:按照时间分页查询日常的支出:

url: http://localhost:8080/design/manager/select_expand_by_date.do

method: POST

参数:

```
dateBegin: dateBegin
dateEnd: dateEnd
start: start
end: end
```

返回:

```
{
  "code": 200,
  "status": "ok",
  "message": [
    {
      "id": 4,
      "date": "2019-03-21",
      "type": "in",
      "money": 0,
      "message": "?????"
    }
  ]
}
```

9:按照类型查询日常的支出:

url: http://localhost:8080/design/manager/select_expand_by_type.do

method: POST

参数:

```
type: type
start: start
end: end
```

返回:

```
{
  "code": 200,
  "status": "ok",
  "message": [
    {
      "id": 2,
      "date": "2019-02-06",
      "type": "in",
      "money": 0,
      "message": "mdcd"
    },
    {
      "id": 4,
      "date": "2019-03-21",
      "type": "in",
      "money": 0,
      "message": "?????"
    }
  ]
}
```

10:分页查找日常的支出:

url: http://localhost:8080/design/manager/select_account_by_date.do

method: POST

参数:

dateBegin: dateBegin
dateEnd: dateEnd
type: type
start: start
end: end

返回:

```
{
  "code": 200,
  "status": "ok",
  "message": [
    {
      "id": 4,
      "date": "2019-03-21",
      "type": "in",
      "money": 0,
      "message": "?????"
    }
  ]
}
```

仓库管理(storeman)的接口:

1:加入食物储备:

url: http://localhost:8080/design/storeman/insert.do

method: POST

参数:

name: name
date: date
security: security
foodType: foodType
number: number
measure: measure(计量单位)
usedFlag: usedFlag(0代表未使用完毕)

返回:

```
{
  "code": 200,
  "status": "ok",
  "message": "插入食材成功"
}
```

2:查询食物储备数量:

url: http://localhost:8080/design/storeman/sum_of_foods.do

method: GET

返回:

```
{
  "code": 200,
  "status": "ok",
  "message": 6
}
```

3:移除食物储备:

url: http://localhost:8080/design/storeman/remove.do

method: POST

参数:

id: id

返回:

```
{
  "code": 200,
  "status": "ok",
  "message": "移除食材成功"
}
```

4:加入食物储备:

url: http://localhost:8080/design/storeman/used.do

method: POST

参数:

id: id

返回:

```
{
  "code": 200,
  "status": "ok",
  "message": "食材已经用尽"
}
```

5:更新食物储备:

url: http://localhost:8080/design/storeman/update.do

method: POST

参数:

id: id
name: name
date: date
security: security
foodType: foodType
number: number
measure: measure(计量单位)

返回:

```
{
  "code": 200,
  "status": "ok",
  "message": "食物更新成功"
}
```

6:分页查询食物:

url: http://localhost:8080/design/storeman/select.do

method: POST

参数:

start: start
end: end

返回:

```
{
  "code": 200,
  "status": "ok",
  "message": [
    {
      "id": 1,
      "name": "name ?",
      "date": "2019-02-05",
      "security": 10,
      "foodType": "weA",
      "number": 13,
      "measure": "ge",
      "usedFlag": 0
    }
  ]
}
```

```
]
}
```

7:按照时间分页查询食物:

url: http://localhost:8080/design/storeman/total.do

method: POST

参数:

startTime: startTime

endTime: endTime

start: start

end: end

返回:

```
{
  "code": 200,
  "status": "ok",
  "message": [
    {
      "id": 2,
      "name": "name,swwwwwj",
      "date": "2019-04-07",
      "security": 10,
      "foodType": "en?",
      "number": 1,
      "measure": "?",
      "usedFlag": 0
    }
  ]
}
```