

餐馆物资管理平台

简介：餐馆物资管理平台的所有API文档

HOST:localhost:8080

联系人：

Version:0.0.1-SNAPSHOT

接口路径： /v2/api-docs

公共接口，包括上传图片等

测试：直接获取token

接口描述:

接口地址: /common/test/getToken

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

暂无

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

添加时上传图片

接口描述:

接口地址: /common/uploadFile

请求方式: POST

consumes: ["multipart/form-data"]

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
file	图片文件	formData	false	file	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

分类数据

新增

接口描述:

接口地址: /category

请求方式: POST

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "cateDescription": "广东的特色菜",
  "cateName": "粤菜",
  "cateType": 1
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
categoryVo	categoryVo	body	true	CategoryVo	CategoryVo

schema属性说明

CategoryVo

参数名称	参数说明	in	是否必须	数据类型	schema
cateDescription	描述	body	false	string	
cateName	名称	body	false	string	
cateType	分类所属	body	false	integer(int32)	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

更新

接口描述:

接口地址: /category

请求方式: PUT

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "cateDescription": "广东的特色菜",
  "cateId": "0",
  "cateName": "粤菜",
  "cateType": 1
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
categoryVo1	categoryVo1	body	true	CategoryVo1	CategoryVo1

schema属性说明

CategoryVo1

参数名称	参数说明	in	是否必须	数据类型	schema
cateDescription	描述	body	false	string	
catelId	ID	body	false	string	
cateName	名称	body	false	string	
cateType	分类所属	body	false	integer(int32)	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

获取所有菜品分类

接口描述:

接口地址: /category/findAllForDish

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

暂无

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

分页查询分类，可指定排序方式、模糊查询关键字

接口描述:

接口地址: /category/findPage

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
categoryType	0表示查询菜品分类，1表示查询配料分类	query	true	integer	
keyword	模糊查询关键字，非必需	query	false	string	
pageId	页码	query	true	integer	
pageSize	每页记录数	query	true	integer	
sortColumn	排序字段，可选值：cate_id、cate_name、create_time	query	true	string	
sortOrder	升序还是降序，可选值：asc、desc	query	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

首页获取所有分类和菜品，包括优惠与热销与推荐

接口描述:

接口地址: /category/getAllInfo/{cId}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
cId	用户ID	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

获取推荐商品

接口描述:

接口地址: /category/getRecommendDishes/{cId}/{number}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
cId	用户ID	path	true	string	
number	推荐数量	path	true	integer	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

id查询

接口描述:

接口地址: /category/{id}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
id	id	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

删除

接口描述:

接口地址: /category/{id}

请求方式: DELETE

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
id	id	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

员工数据

新增

接口描述:

接口地址: /employee

请求方式: POST

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "createTime": "2022-03-18 17:45:00",
  "eid": "",
  "ejob": "",
  "ename": "",
  "ephone": "",
  "esalary": 0,
  "esex": 0,
  "eworktime": 0,
  "modifyTime": "2022-03-18 17:45:00",
  "rid": ""
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
employee	employee	body	true	员工	员工

schema属性说明

员工

参数名称	参数说明	in	是否必须	数据类型	schema
createTime	创建时间	body	false	string(date-time)	
eid		body	false	string	
ejob		body	false	string	
ename		body	false	string	
ephone		body	false	string	
esalary		body	false	number	
esex		body	false	integer(int32)	
eworktime		body	false	integer(int32)	
modifyTime	更新时间	body	false	string(date-time)	
rid		body	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

更新

接口描述:

接口地址: /employee

请求方式: PUT

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "createTime": "2022-03-18 17:45:00",
  "eid": "",
  "ejob": "",
  "ename": "",
  "ephone": "",
  "esalary": 0,
  "esex": 0,
  "eworktime": 0,
  "modifyTime": "2022-03-18 17:45:00",
  "rid": ""
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
employee	employee	body	true	员工	员工

schema属性说明

员工

参数名称	参数说明	in	是否必须	数据类型	schema
createTime	创建时间	body	false	string(date-time)	
eid		body	false	string	
ejob		body	false	string	
ename		body	false	string	
ephone		body	false	string	
esalary		body	false	number	
esex		body	false	integer(int32)	
eworktime		body	false	integer(int32)	
modifyTime	更新时间	body	false	string(date-time)	
rid		body	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

查询分页数据

接口描述:

接口地址: /employee/findListByPage/{rid}/{current}/{limit}

请求方式：GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
current	页码	path	false	string	
limit	每页条数	path	false	string	
rld	餐馆id	path	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

获取所有餐厅各个职位的员工数量

接口描述:

接口地址: /employee/getNumForEachPos

请求方式：GET

consumes: ``

produces: ["*/*"]

请求参数:
暂无

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

获取一个餐厅各个职位的员工数量

接口描述:

接口地址: /employee/getNumForEachPosInRestaurant/{restaurantId}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
restaurantId		path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

获取某个餐厅的员工总数

接口描述:

接口地址: /employee/getNumInRestaurant/{restaurantId}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
restaurantId		path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

获取所有餐厅的员工总数

接口描述:

接口地址: /employee/getTotalNum

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

暂无

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

id查询

接口描述:

接口地址: /employee/{id}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
id	id	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

删除

接口描述:

接口地址: /employee/{id}

请求方式: DELETE

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
id	id	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

库存数据

新增

接口描述:

接口地址: /stock

请求方式: POST

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "createTime": "2022-03-18 17:45:00",
  "mcId": "0000000000000000006",
  "modifyTime": "2022-03-18 17:45:00",
  "rid": "",
  "sid": "",
  "sstock": 0,
  "version": 0
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
stock	stock	body	true	库存	库存

schema属性说明

库存

参数名称	参数说明	in	是否必须	数据类型	schema
createTime	创建时间	body	false	string(date-time)	
mcId	材料ID	body	false	string	
modifyTime	更新时间	body	false	string(date-time)	
rid		body	false	string	
sid		body	false	string	
sstock		body	false	number	
version	版本号	body	false	integer(int32)	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

更新

接口描述:

接口地址: /stock

请求方式: PUT

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "createTime": "2022-03-18 17:45:00",
  "mcId": "000000000000000006",
  "modifyTime": "2022-03-18 17:45:00",
  "rid": "",
  "sid": "",
  "sstock": 0,
  "version": 0
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
stock	stock	body	true	库存	库存

schema属性说明

库存

参数名称	参数说明	in	是否必须	数据类型	schema
createTime	创建时间	body	false	string(date-time)	
mcId	材料ID	body	false	string	
modifyTime	更新时间	body	false	string(date-time)	
rid		body	false	string	
sid		body	false	string	
sstock		body	false	number	
version	版本号	body	false	integer(int32)	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

获取所有阈值之下的物资数据

接口描述:

接口地址: /stock/findAllBelowThreshold

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
r_id	餐馆ID	query	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

分页查询库存，可指定排序方式等，比findPage多了阈值等功能

接口描述:

接口地址: `/stock/findBelowThreshold`

请求方式: `GET`

consumes: ``

produces: `["*/*"]`

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
belowThreshold	是否在阈值之下:0, 1	query	false	integer	
catId	分类ID	query	false	string	
keyword	关键词	query	false	string	
pageId	页码	query	false	integer	
pageSize	每页条数	query	false	integer	
r_id	餐馆ID	query	false	string	
sortColumn	排序字段: s_stock, modify_time, abundant	query	false	string	
sortOrder	排序规则	query	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

查询分页数据

接口描述:

接口地址: /stock/findListByPage/{current}/{limit}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
current	current	path	true	integer	
limit	limit	path	true	integer	
page	页码		false	string	
pageCount	每页条数		false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

获取一个餐厅的所有库存(名字, 剩余)信息

接口描述:

接口地址: /stock/getAllInRestaurant/{restaurantId}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
restaurantId	餐厅ID	path	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

id查询

接口描述:

接口地址: /stock/{id}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
id	id	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

删除

接口描述:

接口地址: `/stock/{id}`

请求方式: `DELETE`

consumes: ``

produces: `["*/*"]`

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
id	id	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

微信支付

根据订单号请求微信支付二维码

接口描述:

接口地址: /paylog/createNative/{orderNo}/{note}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
note	备注	path	false	string	
orderNo	订单号	path	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

根据订单号查询订单支付状态，并更改订单支付状态

接口描述:

接口地址: /paylog/queryPayStatus/{orderNo}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
orderNo	订单号	path	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

投诉与建议

提交建议

接口描述:

接口地址: /advice

请求方式: POST

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "content": "这是一条建议",
  "name": "菜品问题",
  "oid": "",
  "star": 5
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
adviceVo	被提交的建议	body	true	AdviceVo	AdviceVo

schema属性说明

AdviceVo

参数名称	参数说明	in	是否必须	数据类型	schema
content	建议内容	body	false	string	
name	建议主题	body	false	string	
oid		body	false	string	
star	评价等级	body	false	integer(int32)	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

修改建议

接口描述:

接口地址: /advice

请求方式: PUT

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "content": "这是一条建议",
  "createTime": "2022-03-18 17:45:00",
  "id": "0",
  "modifyTime": "2022-03-18 17:45:00",
  "name": "菜品问题",
  "oid": "",
  "star": 5,
  "status": 0
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
advice	修改后的建议	body	true	建议	建议

schema属性说明

建议

参数名称	参数说明	in	是否必须	数据类型	schema
content	建议内容	body	false	string	
createTime	创建时间	body	false	string(date-time)	
id	ID	body	false	string	
modifyTime	更新时间	body	false	string(date-time)	
name	建议主题	body	false	string	
oid		body	false	string	
star	评价等级	body	false	integer(int32)	
status	处理状态	body	false	integer(int32)	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

分页查询，按提交时间倒序排序

接口描述:

接口地址: `/advice/findAdviceByPage/{current}/{limit}`

请求方式: `GET`

consumes: ``

produces: `["*/*"]`

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
current	当前页码	path	true	integer	
limit	每页记录数	path	true	integer	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

分页查询某个顾客提交过的建议，按提交时间倒序排序

接口描述:

接口地址: /advice/findAdviceFromCustomer/{cId}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
cId	顾客ID	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

分页查询大于等于指定星级的数量

接口描述:

接口地址: /advice/findNumberByDate

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
from	起始日期, 非必需	query	false	string	
r_id	餐馆ID	query	false	string	
star	大于等于指定星级	query	false	integer	
to	终止日期, 非必需	query	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

分页查询建议，可指定排序方式等

接口描述:

接口地址: /advice/findPage

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
from	起始日期，非必需	query	false	string	
keyword	关键词,可以按照建议内容，建议类别，顾客姓名查询	query	false	string	
pageId	页码	query	false	integer	
pageSize	每页条数	query	false	integer	
r_id	餐馆ID	query	false	string	
sortColumn	排序字段: name, star, status, o_id, create_time	query	false	string	
sortOrder	排序规则	query	false	string	
status	是否已处理，0为未处理，1为已处理，不指定则均查询	query	false	integer	
to	终止日期，非必需	query	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

通过顾客id分页查询建议

接口描述:

接口地址: /advice/findPageByCustomerId

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
customerId	顾客ID	query	false	string	
pageId	页码	query	false	integer	
pageSize	每页条数	query	false	integer	
r_id	餐馆ID	query	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

修改建议状态

接口描述:

接口地址: /advice/updateStatus/{id}/{status}

请求方式: PUT

consumes: ["application/json"]

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
id	建议ID	path	true	string	
status	状态, 0为未处理, 1为已处理	path	true	integer	

响应示例:


```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

删除建议

接口描述:

接口地址: /advice/{id}

请求方式: DELETE

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
id	建议ID	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

投诉反馈信息

查询分页数据

接口描述:

接口地址: /feedback

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
page	页码	query	false	string	
pageCount	每页条数	query	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

新增

接口描述:

接口地址: /feedback

请求方式: POST

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "adviceId": "0",
  "aid": "",
  "content": "0"
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
feedbackVo	feedbackVo	body	true	FeedbackVo	FeedbackVo

schema属性说明

FeedbackVo

参数名称	参数说明	in	是否必须	数据类型	schema
adviceId	建议id	body	false	string	
aid		body	false	string	
content	反馈内容	body	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

更新

接口描述:

接口地址: /feedback

请求方式: PUT

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "adviceId": "0",
  "aid": "",
  "content": "0",
  "createTime": "2022-03-18 17:45:00",
  "fid": "",
  "modifyTime": "2022-03-18 17:45:00"
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
feedback	feedback	body	true	反馈信息	反馈信息

schema属性说明

反馈信息

参数名称	参数说明	in	是否必须	数据类型	schema
adviceId	建议id	body	false	string	
aid		body	false	string	
content	反馈内容	body	false	string	
createTime	创建时间	body	false	string(date-time)	
fid		body	false	string	
modifyTime	更新时间	body	false	string(date-time)	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

建议id查询

接口描述:

接口地址: /feedback/findByAdviceId/{adviceId}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
adviceId	adviceId	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

id查询

接口描述:

接口地址: /feedback/{id}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
id	id	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

删除

接口描述:

接口地址: /feedback/{id}

请求方式: DELETE

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
id	id	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

权限-角色关系管理

新增

接口描述:

接口地址: /menuRole

请求方式： POST

consumes: ["application/json"]

produces: ["*/*"]

请求示例：

```
{
  "createTime": "2022-03-18 17:45:00",
  "menuId": "0",
  "modifyTime": "2022-03-18 17:45:00",
  "roleId": "0"
}
```

请求参数：

参数名称	参数说明	in	是否必须	数据类型	schema
menuRole	权限-角色关系	body	true	权限-角色关系	权限-角色关系

schema属性说明

权限-角色关系

参数名称	参数说明	in	是否必须	数据类型	schema
createTime		body	false	string(date-time)	
menuId		body	false	string	
modifyTime		body	false	string(date-time)	
roleId		body	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

删除

接口描述:

接口地址: /menuRole/{menuId}/{roleId}

请求方式: DELETE

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
menuId	权限编号	path	true	string	
roleId	角色编号	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

权限管理

新增

接口描述:

接口地址: /menu

请求方式: POST

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "createTime": "2022-03-18 17:45:00",
  "description": "-",
  "id": "0",
  "key": "modify",
  "modifyTime": "2022-03-18 17:45:00",
  "name": "菜品信息修改",
  "typeId": "0"
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
menu	权限	body	true	权限	权限

schema属性说明

权限

参数名称	参数说明	in	是否必须	数据类型	schema
createTime		body	false	string(date-time)	
description		body	false	string	
id		body	false	string	
key		body	false	string	
modifyTime		body	false	string(date-time)	
name		body	false	string	
typeId		body	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

更新

接口描述:

接口地址: /menu

请求方式: PUT

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "createTime": "2022-03-18 17:45:00",
  "description": "-",
  "id": "0",
  "key": "modify",
  "modifyTime": "2022-03-18 17:45:00",
  "name": "菜品信息修改",
  "typeId": "0"
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
menu	权限	body	true	权限	权限

schema属性说明

权限

参数名称	参数说明	in	是否必须	数据类型	schema
createTime		body	false	string(date-time)	
description		body	false	string	
id		body	false	string	
key		body	false	string	
modifyTime		body	false	string(date-time)	
name		body	false	string	
typeld		body	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

查询角色拥有的所有权限

接口描述:

接口地址: /menu/findForRole/{roleId}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
roleId	角色编号	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

分页查询

接口描述:

接口地址: /menu/page

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
keyword	查询关键字	query	false	string	
pageId	页号	query	true	integer	
pageSize	每页记录数	query	true	integer	
typeId	类型编号	query	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

删除

接口描述:

接口地址: /menu/{id}

请求方式: DELETE

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
id	权限编号	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

权限类型管理

获取所有权限类别信息

接口描述:

接口地址: /menuType

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

暂无

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

新增

接口描述:

接口地址: /menuType

请求方式: POST

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "createTime": "2022-03-18 17:45:00",
  "description": "",
  "id": "",
  "key": "",
  "modifyTime": "2022-03-18 17:45:00",
  "name": ""
}
```

请求参数：

参数名称	参数说明	in	是否必须	数据类型	schema
menuType	权限类型	body	true	权限类型	权限类型

schema属性说明

权限类型

参数名称	参数说明	in	是否必须	数据类型	schema
createTime		body	false	string(date-time)	
description	描述	body	false	string	
id	编号	body	false	string	
key	键	body	false	string	
modifyTime		body	false	string(date-time)	
name	名称	body	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

更新

接口描述:

接口地址: /menuType

请求方式: PUT

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "createTime": "2022-03-18 17:45:00",
  "description": "",
  "id": "",
  "key": "",
  "modifyTime": "2022-03-18 17:45:00",
  "name": ""
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
menuType	权限类型	body	true	权限类型	权限类型

schema属性说明

权限类型

参数名称	参数说明	in	是否必须	数据类型	schema
createTime		body	false	string(date-time)	
description	描述	body	false	string	
id	编号	body	false	string	
key	键	body	false	string	
modifyTime		body	false	string(date-time)	
name	名称	body	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

ID查询

接口描述:

接口地址: /menuType/{id}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
id	权限类型ID	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

删除

接口描述:

接口地址: /menuType/{id}

请求方式: DELETE

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
id	权限类型编号	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

材料-进货商关系

查询分页数据

接口描述:

接口地址: /mc-buyer

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
page	页码	query	false	string	
pageCount	每页条数	query	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

新增

接口描述:

接口地址: /mc-buyer

请求方式: POST

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "bid": "",
  "id": "0",
  "mcId": "0"
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
mcBuyer	mcBuyer	body	true	材料-进货商关系	材料-进货商关系

schema属性说明

材料-进货商关系

参数名称	参数说明	in	是否必须	数据类型	schema
bid		body	false	string	
id	ID	body	false	string	
mcId	材料ID	body	false	string	

响应示例:

--

响应参数:

暂无

响应状态:

状态码	说明	schema
200	OK	

更新

接口描述:

接口地址: /mc-buyer

请求方式: PUT

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

<pre>{ "bid": "", "id": "0", "mcId": "0" }</pre>
--

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
mcBuyer	mcBuyer	body	true	材料-进货商关系	材料-进货商关系

schema属性说明

材料-进货商关系

参数名称	参数说明	in	是否必须	数据类型	schema
bid		body	false	string	
id	ID	body	false	string	
mcId	材料ID	body	false	string	

响应示例:

--

响应参数:

暂无

响应状态:

状态码	说明	schema
200	OK	

id查询

接口描述:

接口地址: /mc-buyer/{id}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
id	id	path	true	integer	

响应示例:

<pre>{ "bid": "", "id": "0", "mcId": "0" }</pre>
--

响应参数:

参数名称	参数说明	类型	schema
bid		string	
id	ID	string	
mcId	材料ID	string	

响应状态:

状态码	说明	schema
200	OK	材料-进货商关系

删除

接口描述:

接口地址: /mc-buyer/{id}

请求方式: DELETE

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
id	id	path	true	integer	

响应示例:

--

响应参数:

暂无

响应状态:

状态码	说明	schema
200	OK	

材料数据

新增

接口描述:

接口地址: /material-code

请求方式: POST

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "cateId": "0",
  "createTime": "2022-03-18 17:45:00",
  "mcDescription": "这是一段描述",
  "mcId": "0",
  "mcImgurl": "https://img.szu.com/12",
  "mcName": "白菜",
  "mcPriceperunit": 3,
  "mcThreshold": 100,
  "mcUnit": "kg",
  "modifyTime": "2022-03-18 17:45:00"
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
materialCode	materialCode	body	true	材料	材料

schema属性说明

材料

参数名称	参数说明	in	是否必须	数据类型	schema
cateld	菜品分类ID	body	false	string	
createTime	创建时间	body	false	string(date-time)	
mcDescription	描述	body	false	string	
mcId	ID	body	false	string	
mcImgurl	图片URL	body	false	string	
mcName	名称	body	false	string	
mcPriceperunit	单价	body	false	number	
mcThreshold	阈值，当材料的库存低于阈值时，则需要进行采购	body	false	number	
mcUnit	单位	body	false	string	
modifyTime	更新时间	body	false	string(date-time)	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

更新

接口描述:

接口地址: /material-code

请求方式: PUT

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "cateId": "0",
  "createTime": "2022-03-18 17:45:00",
  "mcDescription": "这是一段描述",
  "mcId": "0",
  "mcImgurl": "https://img.szu.com/12",
  "mcName": "白菜",
  "mcPriceperunit": 3,
  "mcThreshold": 100,
  "mcUnit": "kg",
  "modifyTime": "2022-03-18 17:45:00"
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
materialCode	materialCode	body	true	材料	材料

schema属性说明

材料

参数名称	参数说明	in	是否必须	数据类型	schema
cateld	菜品分类ID	body	false	string	
createTime	创建时间	body	false	string(date-time)	
mcDescription	描述	body	false	string	
mcId	ID	body	false	string	
mcImgurl	图片URL	body	false	string	
mcName	名称	body	false	string	
mcPriceperunit	单价	body	false	number	
mcThreshold	阈值，当材料的库存低于阈值时，则需要进行采购	body	false	number	
mcUnit	单位	body	false	string	
modifyTime	更新时间	body	false	string(date-time)	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

获取所有的配料

接口描述:

接口地址: /material-code/findAll

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

暂无

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

获取所有的配料名字和单位

接口描述:

接口地址: /material-code/findAllNameAndUnit

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:
暂无

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

查询分页数据

接口描述:

接口地址: /material-code/findListByPage/{current}/{limit}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
current	current	path	true	integer	
limit	limit	path	true	integer	
page	页码		false	string	
pageCount	每页条数		false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

分页查询物资，可指定分类、排序方式、模糊查询关键字

接口描述:

接口地址: /material-code/findPage

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
categoryId	分类ID	query	false	string	
keyword	模糊查询关键字，非必需	query	false	string	
pageId	页码	query	true	integer	
pageSize	每页记录数	query	true	integer	
sortColumn	排序字段，可选值： mc_priceperunit、create_time	query	false	string	
sortOrder	升序还是降序，可选值：asc、 desc	query	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

id查询

接口描述:

接口地址: /material-code/{id}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
id	id	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

删除

接口描述:

接口地址: /material-code/{id}

请求方式: DELETE

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
id	id	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

管理员-角色关系管理

新增

接口描述:

接口地址: /adminRole

请求方式: POST

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "adminId": "0",
  "createTime": "2022-03-18 17:45:00",
  "modifyTime": "2022-03-18 17:45:00",
  "roleId": "0"
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
adminRole	管理员-角色关系	body	true	管理员-角色关系	管理员-角色关系

schema属性说明

管理员-角色关系

参数名称	参数说明	in	是否必须	数据类型	schema
adminId		body	false	string	
createTime		body	false	string(date-time)	
modifyTime		body	false	string(date-time)	
roleId		body	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

删除

接口描述:

接口地址: /adminRole/{adminId}/{roleId}

请求方式: DELETE

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
adminId	管理员编号	path	true	string	
roleId	角色编号	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

管理员数据

新增

接口描述:

接口地址: /admin

请求方式: POST

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "aaccount": "",
  "aid": "",
  "aname": "",
  "anickname": "",
  "apassword": "",
  "apermission": 0,
  "avatar": "",
  "createTime": "2022-03-18 17:45:00",
  "modifyTime": "2022-03-18 17:45:00",
  "rid": ""
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
admin	admin	body	true	管理员	管理员

schema属性说明

管理员

参数名称	参数说明	in	是否必须	数据类型	schema
aaccount		body	false	string	
aid		body	false	string	
aname		body	false	string	
anickname		body	false	string	
apassword		body	false	string	
apermission		body	false	integer(int32)	
avatar	头像链接	body	false	string	
createTime	创建时间	body	false	string(date-time)	
modifyTime	更新时间	body	false	string(date-time)	
rid		body	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

更新

接口描述:

接口地址: /admin

请求方式：PUT

consumes: ["application/json"]

produces: ["*/*"]

请求示例：

```
{
  "aaccount": "",
  "aid": "",
  "aname": "",
  "anickname": "",
  "apassword": "",
  "apermission": 0,
  "avatar": "",
  "createTime": "2022-03-18 17:45:00",
  "modifyTime": "2022-03-18 17:45:00",
  "rid": ""
}
```

请求参数：

参数名称	参数说明	in	是否必须	数据类型	schema
admin	admin	body	true	管理员	管理员

schema属性说明

管理员

参数名称	参数说明	in	是否必须	数据类型	schema
aaccount		body	false	string	
aid		body	false	string	
aname		body	false	string	
anickname		body	false	string	
apassword		body	false	string	
apermission		body	false	integer(int32)	
avatar	头像链接	body	false	string	
createTime	创建时间	body	false	string(date-time)	
modifyTime	更新时间	body	false	string(date-time)	
rid		body	false	string	

响应示例：

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

查询分页数据

接口描述:

接口地址: /admin/findListByPage/{current}/{limit}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
current	current	path	true	integer	
limit	limit	path	true	integer	
page	页码		false	string	
pageCount	每页条数		false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

后台获取登录信息

接口描述:

接口地址: /admin/info

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
token	登录成功后返回的token信息	header	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

后台登录

接口描述:

接口地址: /admin/login

请求方式: POST

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "password": "",
  "username": ""
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
authentication	登录的账号密码	body	true	Authentication	Authentication

schema属性说明

Authentication

参数名称	参数说明	in	是否必须	数据类型	schema
password		body	false	string	
username		body	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

分页查询

接口描述:

接口地址: /admin/page

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
keyword	查询关键字	query	false	string	
pageId	页号	query	true	integer	
pageSize	每页记录数	query	true	integer	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

验证旧密码是否正确

接口描述:

接口地址: /admin/validatePassword

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
ald	管理员id	query	true	string	
password	管理员密码	query	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

id查询

接口描述:

接口地址: /admin/{id}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
id	id	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

删除

接口描述:

接口地址: /admin/{id}

请求方式: DELETE

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
id	id	path	true	string	

响应示例:


```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

菜品-材料关系数据

获取一个菜品所含有的材料

接口描述:

接口地址: /dish-mc

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
d_id	菜品ID	query	false	string	

响应示例:

```
[
  {
    "createTime": "2022-03-18 17:45:00",
    "did": "",
    "id": "0",
    "mcId": "0",
    "modifyTime": "2022-03-18 17:45:00",
    "number": 5
  }
]
```

响应参数:

参数名称	参数说明	类型	schema
createTime	创建时间	string(date-time)	string(date-time)
did		string	
id	ID	string	
mcId	材料ID	string	
modifyTime	更新时间	string(date-time)	string(date-time)
number	材料用量	number	

响应状态:

状态码	说明	schema
200	OK	菜品-材料关系

新增

接口描述:

接口地址: /dish-mc

请求方式: POST

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "createTime": "2022-03-18 17:45:00",
  "did": "",
  "id": "0",
  "mcId": "0",
  "modifyTime": "2022-03-18 17:45:00",
  "number": 5
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
dishMc	dishMc	body	true	菜品-材料关系	菜品-材料关系

schema属性说明

菜品-材料关系

参数名称	参数说明	in	是否必须	数据类型	schema
createTime	创建时间	body	false	string(date-time)	
did		body	false	string	
id	ID	body	false	string	
mcId	材料ID	body	false	string	
modifyTime	更新时间	body	false	string(date-time)	
number	材料用量	body	false	number	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

更新

接口描述:

接口地址: /dish-mc

请求方式: PUT

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "createTime": "2022-03-18 17:45:00",
  "did": "",
  "id": "0",
  "mcId": "0",
  "modifyTime": "2022-03-18 17:45:00",
  "number": 5
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
dishMc	dishMc	body	true	菜品-材料关系	菜品-材料关系

schema属性说明

菜品-材料关系

参数名称	参数说明	in	是否必须	数据类型	schema
createTime	创建时间	body	false	string(date-time)	
did		body	false	string	
id	ID	body	false	string	
mcId	材料ID	body	false	string	
modifyTime	更新时间	body	false	string(date-time)	
number	材料用量	body	false	number	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

获取所有菜品所含有的材料

接口描述:

接口地址: /dish-mc/findAllForDish/{r_id}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
r_id	餐馆ID	path	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

查询单个

接口描述:

接口地址: /dish-mc/{mcId}/{dId}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
dId	菜品ID	path	false	string	
mcId	材料ID	path	false	string	

响应示例:

```
{
  "createTime": "2022-03-18 17:45:00",
  "did": "",
  "id": "0",
  "mcId": "0",
  "modifyTime": "2022-03-18 17:45:00",
  "number": 5
}
```

响应参数:

参数名称	参数说明	类型	schema
createTime	创建时间	string(date-time)	string(date-time)
did		string	
id	ID	string	
mcId	材料ID	string	
modifyTime	更新时间	string(date-time)	string(date-time)
number	材料用量	number	

响应状态:

状态码	说明	schema
200	OK	菜品-材料关系

删除

接口描述:

接口地址: /dish-mc/{mcId}/{dId}

请求方式: DELETE

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
dId	菜品ID	path	false	string	
mcId	材料ID	path	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

菜品数据

分页查询所有数据

接口描述:

接口地址: /dish

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
current	页码	query	false	string	
limit	每页条数	query	false	string	

响应示例:


```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

新增

接口描述:

接口地址: /dish

请求方式: POST

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "cateId": "0",
  "ddescribe": "",
  "dingurl": "",
  "dishMcVos": [
    {
      "mcId": 0,
      "number": 5
    }
  ],
  "dname": "",
  "dprice": 0,
  "rid": ""
}
```

请求参数：

参数名称	参数说明	in	是否必须	数据类型	schema
dishVo	dishVo	body	true	DishVo	DishVo

schema属性说明

DishVo

参数名称	参数说明	in	是否必须	数据类型	schema
cateld	菜品分类ID	body	false	string	
ddescribe		body	false	string	
dimgurl		body	false	string	
dishMcVos	配料列表	body	false	array	DishMcVo
dname		body	false	string	
dprice		body	false	number	
rid		body	false	string	

DishMcVo

参数名称	参数说明	in	是否必须	数据类型	schema
mcld	材料ID	body	false	string	
number	材料用量	body	false	number	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

更新

接口描述:

接口地址: /dish

请求方式: PUT

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "cateId": "0",
  "createTime": "2022-03-18 17:45:00",
  "ddescribe": "",
  "did": "",
  "dingurl": "",
  "dishMcVos": [
    {
      "mcId": 0,
      "number": 5
    }
  ],
  "dname": "",
  "dprice": 0,
  "dsales": 0,
  "modifyTime": "2022-03-18 17:45:00",
  "rid": ""
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
dishVo	dishVo	body	true	GetDishVo	GetDishVo

schema属性说明

GetDishVo

参数名称	参数说明	in	是否必须	数据类型	schema
cateld	菜品分类ID	body	false	string	
createTime	创建时间	body	false	string(date-time)	
ddescribe		body	false	string	
did		body	false	string	
dimgurl		body	false	string	
dishMcVos	配料列表	body	false	array	DishMcVo
dname		body	false	string	
dprice		body	false	number	
dsales		body	false	integer(int32)	
modifyTime	更新时间	body	false	string(date-time)	
rid		body	false	string	

DishMcVo

参数名称	参数说明	in	是否必须	数据类型	schema
mcld	材料ID	body	false	string	
number	材料用量	body	false	number	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

分页显示某种分类下的菜品

接口描述:

接口地址: /dish/findByCategory/{category_id}/{current}/{limit}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
category_id	菜品分类编号	path	false	string	
current	页码	path	false	integer	
limit	每页条数	path	false	integer	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

分页查询菜品，可指定餐厅、菜品分类、排序方式、模糊查询关键字

接口描述:

接口地址: /dish/findPage

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
categoryId	菜品分类ID	query	true	string	
keyword	模糊查询关键字，非必需	query	false	string	
pageId	页码	query	true	integer	
pageSize	每页记录数	query	true	integer	
restaurantId	餐厅ID	query	true	string	
sortColumn	排序字段，可选值：d_id、d_name、d_price、create_time、d_sales	query	true	string	
sortOrder	升序还是降序，可选值：asc、desc	query	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

获取一个餐厅销量排行前N的菜品

接口描述:

接口地址: /dish/getSalesTopN/{restaurantId}/{n}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
n	指定获取销量排行前几的菜品	path	true	integer	
restaurantId	餐厅ID	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

增加销量

接口描述:

接口地址: /dish/increaseSales/{dishId}

请求方式: POST

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "increment": 1
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
dishId	dishId	path	true	string	
increment	increment	body	true	DishSalesIncrement	DishSalesIncrement

schema属性说明

DishSalesIncrement

参数名称	参数说明	in	是否必须	数据类型	schema
increment		body	false	integer(int32)	

响应示例:


```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

id查询

接口描述:

接口地址: /dish/{id}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
id	id	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

删除

接口描述:

接口地址: /dish/{id}

请求方式: DELETE

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
id	id	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

角色管理

查询所有角色

接口描述:

接口地址: /role

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
keyword	查询关键字	query	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

新增

接口描述:

接口地址: /role

请求方式: POST

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "createTime": "2022-03-18 17:45:00",
  "description": "-",
  "id": "0",
  "key": "modify",
  "modifyTime": "2022-03-18 17:45:00",
  "name": "菜品信息修改"
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
role	角色	body	true	角色	角色

schema属性说明

角色

参数名称	参数说明	in	是否必须	数据类型	schema
createTime		body	false	string(date-time)	
description		body	false	string	
id		body	false	string	
key		body	false	string	
modifyTime		body	false	string(date-time)	
name		body	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

更新

接口描述:

接口地址: /role

请求方式: PUT

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "createTime": "2022-03-18 17:45:00",
  "description": "-",
  "id": "0",
  "key": "modify",
  "modifyTime": "2022-03-18 17:45:00",
  "name": "菜品信息修改"
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
role	角色	body	true	角色	角色

schema属性说明

角色

参数名称	参数说明	in	是否必须	数据类型	schema
createTime		body	false	string(date-time)	
description		body	false	string	
id		body	false	string	
key		body	false	string	
modifyTime		body	false	string(date-time)	
name		body	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

查询管理员拥有的所有角色

接口描述:

接口地址: /role/findForAdmin/{adminId}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
adminId	管理员编号	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

删除

接口描述:

接口地址: `/role/{id}`

请求方式: `DELETE`

consumes: ``

produces: `["*/*"]`

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
id	角色编号	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

订单-菜品关系数据

新增

接口描述:

接口地址: /menu-order

请求方式: POST

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "createTime": "2022-03-18 17:45:00",
  "did": "",
  "modifyTime": "2022-03-18 17:45:00",
  "number": 0,
  "oid": ""
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
menuOrder	menuOrder	body	true	订单-菜品关系	订单-菜品关系

schema属性说明

订单-菜品关系

参数名称	参数说明	in	是否必须	数据类型	schema
createTime	创建时间	body	false	string(date-time)	
did		body	false	string	
modifyTime	更新时间	body	false	string(date-time)	
number	菜品数量	body	false	integer(int32)	
oid		body	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

更新

接口描述:

接口地址: /menu-order

请求方式: PUT

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "createTime": "2022-03-18 17:45:00",
  "did": "",
  "modifyTime": "2022-03-18 17:45:00",
  "number": 0,
  "oid": ""
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
menuOrder	menuOrder	body	true	订单-菜品关系	订单-菜品关系

schema属性说明

订单-菜品关系

参数名称	参数说明	in	是否必须	数据类型	schema
createTime	创建时间	body	false	string(date-time)	
did		body	false	string	
modifyTime	更新时间	body	false	string(date-time)	
number	菜品数量	body	false	integer(int32)	
oid		body	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

获取一个订单所包含的所有菜品

接口描述:

接口地址: /menu-order/findForOrder/{oId}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
old	old	path	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

查找单个

接口描述:

接口地址: /menu-order/{oId}/{dId}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
dId	菜品ID	path	false	string	
old	订单ID	path	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

删除

接口描述:

接口地址: /menu-order/{oId}/{dId}

请求方式: DELETE

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
dId	菜品ID	path	false	string	
old	订单ID	path	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

订单数据

创建订单

接口描述:

接口地址: /order/createOrder

请求方式: POST

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "allPrice": 0,
  "cid": "",
  "dinnerNum": 0,
  "note": "",
  "orderMenuVoList": [
    {
      "did": "",
      "number": 0
    }
  ],
  "rid": ""
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
orderVo	orderVo	body	true	OrderVo	OrderVo

schema属性说明

OrderVo

参数名称	参数说明	in	是否必须	数据类型	schema
allPrice		body	false	number	
cid		body	false	string	
dinnerNum		body	false	integer(int32)	
note		body	false	string	
orderMenuVoList		body	false	array	OrderMenuVo
rid		body	false	string	

OrderMenuVo

参数名称	参数说明	in	是否必须	数据类型	schema
did		body	false	string	
number		body	false	integer(int32)	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

获取某个顾客在某个餐馆的订单，按时间降序排序

接口描述:

接口地

址: /order/findCustomerOrderByRestaurantId/{customer_id}/{r_id}/{current}/{limit}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
current	页码	path	false	integer	
customer_id	顾客ID	path	false	string	
limit	每页条数	path	false	integer	
r_id	餐馆ID	path	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```


响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

获取某个顾客的订单，按时间降序排序

接口描述:

接口地址: /order/findForCustomer/{customer_id}/

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
customer_id	顾客ID	path	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

查询某个餐馆的订单分页数据

接口描述:

接口地址: /order/findListByPage/{rId}/{current}/{limit}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
current	页码	path	false	string	
limit	每页条数	path	false	string	
rId	餐馆id	path	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

分页查询订单，可指定排序方式、搜索关键词、日期等

接口描述:

接口地址: `/order/findPage`

请求方式: `GET`

consumes: ``

produces: `["*/*"]`

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
from	起始日期	query	false	string	
keyword	模糊查询关键字，非必需	query	false	string	
pageId	页码	query	true	integer	
pageSize	每页记录数	query	true	integer	
r_id	餐厅ID	query	true	string	
sortColumn	排序字段，可选值：o_price、modify_time	query	false	string	
sortOrder	升序还是降序，可选值：asc、desc	query	false	string	
to	终止日期，非必需	query	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

获取一个餐厅在某个日期范围内总下单数量

接口描述:

接口地址: /order/getNumInRestaurantWithinDateRange/{restaurantId}/{from}/{to}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
from	起始日期	path	true	string	
restaurantId	餐厅ID	path	true	string	
to	终止日期	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

获取某个日期范围内的所有餐厅的总下单数量

接口描述:

接口地址: /order/getNumWithinDateRange/{from}/{to}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
from	起始日期	path	true	string	
to	终止日期	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

获取所有餐厅的订单总数

接口描述:

接口地址: /order/getTotal

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

暂无

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

获取某个餐厅的订单总数

接口描述:

接口地址: /order/getTotalInRestaurant/{restaurantId}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
restaurantId	餐厅ID	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

删除订单，且把此次订单修改的库存等数据全部恢复

接口描述:

接口地址: `/order/rollbackOrder/{oId}`

请求方式: `DELETE`

consumes: ``

produces: `["*/*"]`

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
old	old	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

id查询

接口描述:

接口地址: `/order/{id}`

请求方式: `GET`

consumes: ``

produces: `["*/*"]`

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
id	id	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

账号相关

管理员退出登录

接口描述:

接口地址: /auth/adminLogout

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

暂无

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

验证token有效性

接口描述:返回数据含有一个isValid字段，如果token有效，则isValid字段为true，否则为false

接口地址: /auth/checkToken

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
token	被验证的token	header	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

获取token

接口描述:请求体包含一个code字段，其值为code的内容
如果获取token成功，则响应体包含一个token字段，其值为token的内容

接口地址: /auth/token

请求方式: POST

consumes: ["application/json"]

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
reqBody	临时登录凭证code	body	true	object	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

进货商数据

新增

接口描述:

接口地址: /buyer

请求方式: POST

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "baddress": "",
  "bid": "",
  "bname": "",
  "bphone": "",
  "createTime": "2022-03-18 17:45:00",
  "description": "猪肉进货商",
  "modifyTime": "2022-03-18 17:45:00"
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
buyer	buyer	body	true	进货商	进货商

schema属性说明

进货商

参数名称	参数说明	in	是否必须	数据类型	schema
baddress		body	false	string	
bid		body	false	string	
bname		body	false	string	
bphone		body	false	string	
createTime	创建时间	body	false	string(date-time)	
description	进货商描述	body	false	string	
modifyTime	更新时间	body	false	string(date-time)	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

更新

接口描述:

接口地址: /buyer

请求方式: PUT

consumes: ["application/json"]

produces: ["*/*"]

请求示例：

```
{
  "baddress": "",
  "bid": "",
  "bname": "",
  "bphone": "",
  "createTime": "2022-03-18 17:45:00",
  "description": "猪肉进货商",
  "modifyTime": "2022-03-18 17:45:00"
}
```

请求参数：

参数名称	参数说明	in	是否必须	数据类型	schema
buyer	buyer	body	true	进货商	进货商

schema属性说明

进货商

参数名称	参数说明	in	是否必须	数据类型	schema
baddress		body	false	string	
bid		body	false	string	
bname		body	false	string	
bphone		body	false	string	
createTime	创建时间	body	false	string(date-time)	
description	进货商描述	body	false	string	
modifyTime	更新时间	body	false	string(date-time)	

响应示例：

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数：

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

查询分页数据

接口描述:

接口地址: /buyer/findListByPage/{current}/{limit}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
current	current	path	true	integer	
limit	limit	path	true	integer	
page	页码		false	string	
pageCount	每页条数		false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

id查询

接口描述:

接口地址: /buyer/{id}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
id	id	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

删除

接口描述:

接口地址: /buyer/{id}

请求方式: DELETE

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
id	id	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

采购清单数据

新增

接口描述:

接口地址: /purchase-list

请求方式: POST

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "actualNumber": 500,
  "actualPrice": 1000,
  "bid": "",
  "createTime": "2022-03-18 17:45:00",
  "expectedNumber": 500,
  "expectedPrice": 1000,
  "id": "0",
  "mcId": "0",
  "modifyTime": "2022-03-18 17:45:00",
  "prId": "0"
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
purchaseList	purchaseList	body	true	采购清单	采购清单

schema属性说明

采购清单

参数名称	参数说明	in	是否必须	数据类型	schema
actualNumber	实际采购量	body	false	number	
actualPrice	实际单价	body	false	number	
bid		body	false	string	
createTime	创建（采购）时间	body	false	string(date-time)	
expectedNumber	预期采购量	body	false	number	
expectedPrice	预期单价	body	false	number	
id	ID	body	false	string	
mcId	材料ID	body	false	string	
modifyTime	更新时间	body	false	string(date-time)	
prId	采购记录ID	body	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

更新

接口描述:

接口地址: /purchase-list

请求方式: PUT

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "actualNumber": 500,
  "actualPrice": 1000,
  "bid": "",
  "createTime": "2022-03-18 17:45:00",
  "expectedNumber": 500,
  "expectedPrice": 1000,
  "id": "0",
  "mcId": "0",
  "modifyTime": "2022-03-18 17:45:00",
  "prId": "0"
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
purchaseList	purchaseList	body	true	采购清单	采购清单

schema属性说明

采购清单

参数名称	参数说明	in	是否必须	数据类型	schema
actualNumber	实际采购量	body	false	number	
actualPrice	实际单价	body	false	number	
bid		body	false	string	
createTime	创建（采购）时间	body	false	string(date-time)	
expectedNumber	预期采购量	body	false	number	
expectedPrice	预期单价	body	false	number	
id	ID	body	false	string	
mcId	材料ID	body	false	string	
modifyTime	更新时间	body	false	string(date-time)	
prId	采购记录ID	body	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

查询指定采购记录下的采购清单

接口描述:

接口地址: /purchase-list/findByPurchaseRecordId

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
purchaseRecordId	采购记录id	query	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

查询分页数据

接口描述:

接口地址: /purchase-list/findListByPage/{current}/{limit}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
current	current	path	true	integer	
limit	limit	path	true	integer	
page	页码		false	string	
pageCount	每页条数		false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

分页查询采购清单，可指定排序方式等

接口描述:

接口地址: /purchase-list/findPage

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数：

参数名称	参数说明	in	是否必须	数据类型	schema
keyword	关键词，可以按照进货商或材料名查询，只可选其一	query	false	string	
pageId	页码	query	false	integer	
pageSize	每页条数	query	false	integer	
sortColumn	排序字段 actual_price, modify_time	query	false	string	
sortOrder	排序规则	query	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

修改指定采购清单的价格和数量

接口描述:

接口地址: /purchase-list/updatePriceAndNumber

请求方式: POST

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
[
  {
    "mcActualNumber": 500,
    "mcActualPrice": 1000,
    "purchaseListId": "0"
  }
]
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
purchaseListVos	purchaseListVos	body	true	array	PurchaseListVos

schema属性说明

PurchaseListVos

参数名称	参数说明	in	是否必须	数据类型	schema
mcActualNumber	实际采购量	body	false	number	
mcActualPrice	实际单价	body	false	number	
purchaseListId	ID	body	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

id查询

接口描述:

接口地址: /purchase-list/{id}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
id	id	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

删除

接口描述:

接口地址: /purchase-list/{id}

请求方式: DELETE

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
id	id	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

采购记录数据

更新

接口描述:

接口地址: /purchase-record

请求方式: PUT

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "actualTotalPrice": 1000,
  "aid": "",
  "createTime": "2022-03-18 17:45:00",
  "expectedTotalPrice": 1000,
  "id": "0",
  "modifyTime": "2022-03-18 17:45:00",
  "note": "这是一次采购",
  "rid": "",
  "status": 0
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
PurchaseRecord	PurchaseRecord	body	true	采购记录	采购记录

schema属性说明

采购记录

参数名称	参数说明	in	是否必须	数据类型	schema
actualTotalPrice	实际总价	body	false	number	
aid		body	false	string	
createTime	创建（采购）时间	body	false	string(date-time)	
expectedTotalPrice	理论总价	body	false	number	
id	ID	body	false	string	
modifyTime	更新时间	body	false	string(date-time)	
note	备注	body	false	string	
rid		body	false	string	
status	采购是否完成	body	false	integer(int32)	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

创建采购记录

接口描述:

接口地址: /purchase-record/createPurchaseRecord

请求方式: POST

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "aid": "",
  "purchaseListVos": [
    {
      "bid": "",
      "expectedNumber": 0,
      "expectedPrice": 0,
      "mcId": ""
    }
  ],
  "rid": ""
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
purchaseRecordVo	purchaseRecordVo	body	true	PurchaseRecordVo	PurchaseRecordVo

schema属性说明

PurchaseRecordVo

参数名称	参数说明	in	是否必须	数据类型	schema
aid		body	false	string	
purchaseListVos		body	false	array	PurchaseListVo
rid		body	false	string	

PurchaseListVo

参数名称	参数说明	in	是否必须	数据类型	schema
bid		body	false	string	
expectedNumber		body	false	number	
expectedPrice		body	false	number	
mcId		body	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

查询分页数据

接口描述:

接口地址: /purchase-record/findListByPage/{current}/{limit}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
current	current	path	true	integer	
limit	limit	path	true	integer	
page	页码		false	string	
pageCount	每页条数		false	string	

响应示例:


```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

分页查询采购记录，可指定排序方式等

接口描述:

接口地址: /purchase-record/findPage

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
from	起始日期，非必需	query	false	string	
keyword	关键词，可以按照处理人姓名查询	query	false	string	
pageId	页码	query	false	integer	
pageSize	每页条数	query	false	integer	
r_id	餐馆ID	query	false	string	
sortColumn	排序字段 status, expected_total_price, actual_total_price, modify_time	query	false	string	
sortOrder	排序规则	query	false	string	
to	终止日期，非必需	query	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

完成指定采购记录

接口描述:

接口地址: /purchase-record/updateStatusByPurchaseRecordId

请求方式： POST

consumes: ["application/json"]

produces: ["*/*"]

请求参数：

参数名称	参数说明	in	是否必须	数据类型	schema
ald	处理人（管理员）id	query	false	string	
note	采购记录备注	query	false	string	
purchaseRecordId	采购记录id	query	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

id查询

接口描述:

接口地址: /purchase-record/{id}

请求方式： GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
id	id	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

删除

接口描述:

接口地址: /purchase-record/{id}

请求方式: DELETE

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
id	id	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

销量统计

计算一个餐厅在某个时间段内的销售额

接口描述:

接口地址: /sales/getInRestaurantWithinDateRange/{restaurantId}/{from}/{to}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
from	起始日期	path	true	string	
restaurantId	餐厅ID	path	true	string	
to	终止日期	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

计算所有餐厅的总销售额

接口描述:

接口地址: `/sales/getTotal`

请求方式: `GET`

consumes: ``

produces: `["*/*"]`

请求参数:

暂无

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

计算一个餐厅的总销售额

接口描述:

接口地址: /sales/getTotalInRestaurant/{restaurantId}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
restaurantId	餐厅ID	path	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

计算所有餐厅在某个时间段内的总销售额

接口描述:

接口地址: `/sales/getWithinDateRange/{from}/{to}`

请求方式: `GET`

consumes: ``

produces: `["*/*"]`

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
from	起始日期	path	true	string	
to	终止日期	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

顾客数据

新增

接口描述:

接口地址: /customer

请求方式: POST

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "caccount": "",
  "cavatarUrl": "",
  "cid": "",
  "cname": "",
  "cpassword": "",
  "cphone": "",
  "createTime": "2022-03-18 17:45:00",
  "csex": 0,
  "fid": "",
  "id": 0,
  "modifyTime": "2022-03-18 17:45:00"
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
customer	customer	body	true	顾客	顾客

schema属性说明

顾客

参数名称	参数说明	in	是否必须	数据类型	schema
caccount		body	false	string	
cavatarUrl		body	false	string	
cid		body	false	string	
cname		body	false	string	
cpassword		body	false	string	
cphone		body	false	string	
createTime	创建时间	body	false	string(date-time)	
csex		body	false	integer(int32)	
fid		body	false	string	
id	数字id	body	false	integer(int32)	
modifyTime	更新时间	body	false	string(date-time)	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

更新

接口描述:

接口地址: /customer

请求方式: PUT

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "cavatarUrl": "",
  "cid": "",
  "cname": "",
  "cpassword": "",
  "cphone": "",
  "csex": 0,
  "fid": ""
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
customerVO	customerVO	body	true	UserUpdateCusVO	UserUpdateCusVO

schema属性说明

UserUpdateCusVO

参数名称	参数说明	in	是否必须	数据类型	schema
cavatarUrl		body	false	string	
cid		body	false	string	
cname		body	false	string	
cpassword		body	false	string	
cphone		body	false	string	
csex		body	false	integer(int32)	
fid		body	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

查询分页数据

接口描述:

接口地址: /customer/findListByPage/{current}/{limit}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
current	current	path	true	integer	
limit	limit	path	true	integer	
page	页码		false	string	
pageCount	每页条数		false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

微信授权登录后，根据登录信息返回顾客id

接口描述:

接口地址: /customer/getCustomerId

请求方式: POST

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "cavaterUrl": "",
  "cname": ""
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
customerVo	customerVo	body	true	CustomerVo	CustomerVo
token	token	header	true	string	

schema属性说明

CustomerVo

参数名称	参数说明	in	是否必须	数据类型	schema
cavatarUrl		body	false	string	
cname		body	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

管理员专用更新接口

接口描述:

接口地址: /customer/updateByAdmin

请求方式: PUT

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "caccount": "",
  "cavatarUrl": "",
  "cid": "",
  "cname": "",
  "cpassword": "",
  "cphone": "",
  "createTime": "2022-03-18 17:45:00",
  "csex": 0,
  "fid": "",
  "id": 0,
  "modifyTime": "2022-03-18 17:45:00"
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
customer	customer	body	true	顾客	顾客

schema属性说明

顾客

参数名称	参数说明	in	是否必须	数据类型	schema
caccount		body	false	string	
cavatarUrl		body	false	string	
cid		body	false	string	
cname		body	false	string	
cpassword		body	false	string	
cphone		body	false	string	
createTime	创建时间	body	false	string(date-time)	
csex		body	false	integer(int32)	
fid		body	false	string	
id	数字id	body	false	integer(int32)	
modifyTime	更新时间	body	false	string(date-time)	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

id查询

接口描述:

接口地址: /customer/{id}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
id	id	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```


响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

删除

接口描述:

接口地址: /customer/{id}

请求方式: DELETE

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
id	id	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

餐厅数据

新增

接口描述:

接口地址: /restaurant

请求方式: POST

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "createTime": "2022-03-18 17:45:00",
  "modifyTime": "2022-03-18 17:45:00",
  "raddress": "",
  "rallsales": 0,
  "rdescribe": "",
  "rid": "",
  "rimgurl": "",
  "rname": ""
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
restaurant	restaurant	body	true	餐厅	餐厅

schema属性说明

餐厅

参数名称	参数说明	in	是否必须	数据类型	schema
createTime	创建时间	body	false	string(date-time)	
modifyTime	更新时间	body	false	string(date-time)	
raddress		body	false	string	
rallsales		body	false	number	
rdescribe		body	false	string	
rid		body	false	string	
ringurl		body	false	string	
rname		body	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

更新

接口描述:

接口地址: /restaurant

请求方式: PUT

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "createTime": "2022-03-18 17:45:00",
  "modifyTime": "2022-03-18 17:45:00",
  "raddress": "",
  "rallsales": 0,
  "rdescribe": "",
  "rid": "",
  "ringurl": "",
  "rname": ""
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
restaurant	restaurant	body	true	餐厅	餐厅

schema属性说明

餐厅

参数名称	参数说明	in	是否必须	数据类型	schema
createTime	创建时间	body	false	string(date-time)	
modifyTime	更新时间	body	false	string(date-time)	
raddress		body	false	string	
rallsales		body	false	number	
rdescribe		body	false	string	
rid		body	false	string	
ringurl		body	false	string	
rname		body	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

查询所有数据

接口描述:

接口地址: /restaurant/findAll

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

暂无

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

获取不同地区的餐厅分布数量

接口描述:

接口地址: /restaurant/getNumInEachArea

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

暂无

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

获取餐厅的总数量

接口描述:

接口地址: /restaurant/getTotalNum

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

暂无

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

id查询

接口描述:

接口地址: /restaurant/{id}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
id	id	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

删除

接口描述:

接口地址: /restaurant/{id}

请求方式: DELETE

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
id	id	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

餐桌数据

新增

接口描述:

接口地址: /foodtable

请求方式: POST

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "createTime": "2022-03-18 17:45:00",
  "fid": "",
  "fmaxnum": 0,
  "frealnum": 0,
  "fstate": 0,
  "modifyTime": "2022-03-18 17:45:00",
  "rid": ""
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
foodtable	foodtable	body	true	餐桌	餐桌

schema属性说明

餐桌

参数名称	参数说明	in	是否必须	数据类型	schema
createTime	创建时间	body	false	string(date-time)	
fid		body	false	string	
fmaxnum		body	false	integer(int32)	
frealnum		body	false	integer(int32)	
fstate		body	false	integer(int32)	
modifyTime	更新时间	body	false	string(date-time)	
rid		body	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

更新（用户专用）

接口描述:

接口地址: /foodtable

请求方式: PUT

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "fid": "",
  "frealnum": 0,
  "fstate": 0
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
tableVO	更新数据	body	true	UserUpdateTableVO	UserUpdateTableVO

schema属性说明

UserUpdateTableVO

参数名称	参数说明	in	是否必须	数据类型	schema
fid		body	false	string	
frealnum		body	false	integer(int32)	
fstate		body	false	integer(int32)	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

查询分页数据

接口描述:

接口地址: /foodtable/findListByPage/{restaurantId}/{current}/{limit}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
current	页码	path	true	integer	
limit	每页记录数	path	true	integer	
restaurantId	餐厅ID	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

获取处于不同状态的餐桌数量，状态0-未使用，1-使用中

接口描述:

接口地址: /foodtable/getNumForEachState/{restaurantId}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
restaurantId	餐厅ID	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

获取某个餐厅的餐桌总数

接口描述:

接口地址: /foodtable/getNumInRestaurant/{restaurantId}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
restaurantId	餐厅ID	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

更新（管理员专用）

接口描述:

接口地址: /foodtable/updateByAdmin

请求方式: PUT

consumes: ["application/json"]

produces: ["*/*"]

请求示例:

```
{
  "createTime": "2022-03-18 17:45:00",
  "fid": "",
  "fmaxnum": 0,
  "frealnum": 0,
  "fstate": 0,
  "modifyTime": "2022-03-18 17:45:00",
  "rid": ""
}
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
foodtable	foodtable	body	true	餐桌	餐桌

schema属性说明

餐桌

参数名称	参数说明	in	是否必须	数据类型	schema
createTime	创建时间	body	false	string(date-time)	
fid		body	false	string	
fmaxnum		body	false	integer(int32)	
frealnum		body	false	integer(int32)	
fstate		body	false	integer(int32)	
modifyTime	更新时间	body	false	string(date-time)	
rid		body	false	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

id查询

接口描述:

接口地址: /foodtable/{id}

请求方式: GET

consumes: ``

produces: ["*/*"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
id	id	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result

删除

接口描述:

接口地址: /foodtable/{id}

请求方式: DELETE

consumes:``

produces:["*/"]

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
id	id	path	true	string	

响应示例:

```
{
  "code": 0,
  "data": {},
  "message": "",
  "success": true
}
```

响应参数:

参数名称	参数说明	类型	schema
code	返回码	integer(int32)	integer(int32)
data	返回数据	object	
message	返回消息	string	
success	是否成功	boolean	

响应状态:

状态码	说明	schema
200	OK	Result