**智慧社区API说明书**

服务器基础地址：https://www.codeboy.com/zhsqapi/

服务器备用地址：https://web.codeboy.com/zhsqapi/

目录

[一、用户模块 1](#_Toc123014344)

[1.1、用户注册 1](#_Toc123014345)

[1.2、用户登录 1](#_Toc123014346)

[1.3、当前登录用户档案 1](#_Toc123014347)

[1.4、用户列表 1](#_Toc123014348)

[1.5、修改当前登录用户头像 1](#_Toc123014349)

[1.6、修改当前登录用户密码 1](#_Toc123014350)

[1.7、检测邮箱是否存在 1](#_Toc123014351)

[1.8、检测手机号是否存在 1](#_Toc123014352)

[1.9、检测用户名是否存在 1](#_Toc123014353)

[1.10、退出登录 1](#_Toc123014354)

[二、首页模块 1](#_Toc123014355)

[2.1、首页数据 1](#_Toc123014356)

[三、缴费模块 1](#_Toc123014357)

[3.1、当前登录用户各项待缴费金额汇总 1](#_Toc123014358)

[3.2、获取指定类型的收费单位列表 1](#_Toc123014359)

[3.3、提交缴费记录 1](#_Toc123014360)

[3.4、当前登录用户缴费记录 1](#_Toc123014361)

[四、业主模块 1](#_Toc123014362)

[4.1、根据业主缴费编号查询该业主的基本信息 1](#_Toc123014363)

[五、智慧停车模块 1](#_Toc123014364)

[5.1、返回所有停车区坐标及当前空闲数据 1](#_Toc123014365)

[六、邻里互动模块 1](#_Toc123014366)

[6.1、上传一张互动消息中涉及到的图片 1](#_Toc123014367)

[6.2、添加一条互动消息 1](#_Toc123014368)

[6.3、分页获取互动消息列表 1](#_Toc123014369)

[七、社区商城模块 1](#_Toc123014370)

[7.1、商城首页数据 1](#_Toc123014371)

[7.2、返回指定类型的商品列表 1](#_Toc123014372)

[7.3、返回指定编号的商品详情 1](#_Toc123014373)

[八、管理员模块 1](#_Toc123014374)

[8.1、管理员登录 1](#_Toc123014375)

[8.2、获取当前登录管理员的基础信息及权限信息 1](#_Toc123014376)

[8.3、修改当前登录的管理员密码 1](#_Toc123014377)

[8.4、分页获取居民列表 1](#_Toc123014378)

[8.5、查询特定居民详情 1](#_Toc123014379)

[8.6、删除特定居民记录 1](#_Toc123014380)

[8.7、修改特定居民记录 1](#_Toc123014381)

# 一、用户模块

## 1.1、用户注册

接口地址：user/register

请求方式：POST

请求主体格式：application/json

请求数据说明：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 必填 | 类型 | 说明 |
| phone | 是 | string | 手机号(有格式验证和重复验证) |
| pwd | 是 | string | 密码 |

响应主体格式：json object；属性说明：

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 说明 |
| code | int | 返回码  2000-注册成功  4001-手机号为空  4002-密码为空  4003-手机号格式错误  4004-手机号已被占用 |
| msg | string | 原因说明 |

响应主体示例：

{ "code":2000, "msg":"register succ" }

## 1.2、用户登录

接口地址：user/login

请求方式：POST

请求主体格式：application/json

请求数据说明：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 必填 | 类型 | 说明 |
| phone | 是 | string | 手机号 |
| pwd | 是 | string | 密码 |

响应主体格式：json object，属性：

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 说明 |
| code | int | 返回码  2000-登录成功  4001-手机号为空  4002-密码为空  4003-手机号或密码错误 |
| msg | string | 原因说明 |
| token | string | 此次登录获得的身份令牌 |

响应主体示例：

{

    "code": 2000,

    "msg": "login succ",

    "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXV....."

}

## 1.3、当前登录用户档案

接口地址：user/profile

请求方式：GET

请求主体：无

请求头部：token

请求数据说明：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 必填 | 类型 | 说明 |
| token | 是 | string | 请求头部中必须携带登录成功后得到的身份令牌，且必须在有效期内 |

响应主体格式：json object，属性：

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 说明 |
| code | int | 返回码  2000-修改成功  4100-令牌缺失  4101-令牌无效  4102-令牌已过期 |
| msg | string | 原因说明 |

成功的响应主体示例：

**{**

**"uid"**: 3**,**

**"uname"**: "doudou"**,**

**"email"**: "dou@qq.com"**,**

**"phone"**: "13501234569"**,**

**"avatar"**: "img/avatar/default.png"**,**

**"user\_name"**: "窦志强"**,**

**"gender"**: 1

**}**

## 1.4、用户列表

注意：此接口仅做测试使用，正式上线后不可用

接口地址：user/list

请求方式：GET

请求主体：无

请求示例：user/list?pageNum=1&pageSize=10

请求参数说明：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 必填 | 类型 | 说明 |
| pageNum | 否 | int | 页号，默认值为1 |
| pageSize | 否 | int | 页面大小，默认值为10 |

响应主体格式：json array

响应数据示例：

[

    {

        "uid": 3,

        "phone": "13501234569",

        "uname": "dou123",

        "email": "dou@tedu.cn",

        "avatar": "img/default.png",

        "gender": 1

    },

...

]

## 1.5、修改当前登录用户头像

接口地址：user/update/avatar

请求方式：POST

请求主体格式：multipart/form-data

请求头部：token

请求数据说明：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 必填 | 类型 | 说明 |
| token | 是 | string | 请求头部中必须携带登录成功后得到的身份令牌，且必须在有效期内 |
| avatar | 是 | file | 新的用户头像文件 |

响应主体格式：json object，属性：

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 说明 |
| code | int | 返回码  2000-修改成功  4100-令牌缺失  4101-令牌无效  4102-令牌已过期 |
| msg | string | 原因说明 |

成功的响应主体示例：

{ "code":2000, "msg":"update succ" }

## 1.6、修改当前登录用户密码

接口地址：user/update/pwd

请求方式：POST

请求主体格式：application/json

请求头部：token

请求数据说明：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 必填 | 类型 | 说明 |
| token | 是 | string | 请求头部中必须携带登录成功后得到的身份令牌，且必须在有效期内 |
| oldPwd | 是 | string | 原有密码 |
| newPwd | 是 | string | 新密码 |

响应主体格式：json object，属性：

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 说明 |
| code | int | 返回码  2000-修改成功  2000-修改成功  4001-旧密码为空  4002-新密码为空  4003-旧密码错误  4004-新旧密码一样  4100-令牌缺失  4101-令牌无效  4102-令牌已过期 |
| msg | string | 原因说明 |

成功的响应主体示例：

{ "code":2000, "msg":"update succ" }

## 1.7、检测邮箱是否存在

接口地址：user/check/email

请求方式：GET

请求主体：无

请求示例：user/check/email?email=ya@qq.com

请求参数说明：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 必填 | 类型 | 说明 |
| email | 是 | string | 用户邮箱 |

响应主体格式：json object，属性：

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 说明 |
| code | int | 返回码  2000-不存在  2001-已存在 |
| msg | string | 原因说明 |

响应主体示例：

{ "code":2000, "msg":"non exists" }

## 1.8、检测手机号是否存在

接口地址：user/check/phone

请求方式：GET

请求主体：无

请求示例：user/check/phone?phone=13501234567

请求参数说明：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 必填 | 类型 | 说明 |
| phone | 是 | string | 手机号码 |

响应主体格式：json object，属性：

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 说明 |
| code | int | 返回码  2000-不存在  2001-已存在 |
| msg | string | 原因说明 |

响应主体示例：

{ "code":2000, "msg":"non exists" }

## 1.9、检测用户名是否存在

接口地址：user/check/uname

请求方式：GET

请求主体：无

请求示例：user/check/uname?uname=yaya

请求参数说明：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 必填 | 类型 | 说明 |
| uname | 是 | string | 用户名 |

响应主体格式：json object，属性：

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 说明 |
| code | int | 返回码  2000-不存在  2001-已存在 |
| msg | string | 原因说明 |

响应主体示例：

{ "code":2000, "msg":"non exists" }

## 1.10、退出登录

接口地址：user/logout

请求方式：GET

请求头部：token

请求数据说明：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 必填 | 类型 | 说明 |
| token | 是 | string | 请求头部中必须携带登录成功后得到的身份令牌，且必须在有效期内 |

响应主体示例：

{ "code":2000, "msg":"logout succ" }

# 二、首页模块

## 2.1、首页数据

接口地址：index/data

请求方式：GET

请求主体：无

请求参数：无

请求头部：token - 用户登录后保存在客户端的身份凭证

响应主体格式：json object；属性说明：

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 说明 |
| carousels | array | 轮播广告数据列表 |
| menuItems | array | 当前用户定制的功能菜单列表 |
| activities | array | 社区活动数据列表 |

响应主体示例：

{

    "carousels": [

        {

            "cid": 1,

            "pic": "img/ad/1516731234475.jpg",

            "title": "天通苑广场",

            "href": "/pages/ad/ad?aid=1"

        },

......

    ],

    "menuItems": [

        {

            "iid": 2,

            "title": "物业缴费",

            "icon": "img/rmb.png",

            "href": "/pages/property-fee/property-fee"

        },

  ......

    ],

    "activities": [

        {

            "aid": 3,

            "title": "天通苑社区2023年春节庆祝活动",

            "content": "一年之计在于春，而春节亦是中国最盛大、最热闹、最重要的......",

            "startTime": 1640966433344,

            "pic": "img/activity/1516731276128.png"

        },

        ......

    ]

}

# 三、缴费模块

## 3.1、当前登录用户各项待缴费金额汇总

接口地址：fee/list

请求方式：GET

请求主体：无

请求参数：无

请求头部：token - 用户登录后保存在客户端的身份凭证

响应主体格式：json object；

响应主体示例：

{

    "shui": -828.53,

    "dian": -1399.67,

    "ranqi": -545.11,

    "wuye": -398.11,

    "tingche": -273.88,

    "kuandai": 318.82

}

## 3.2、获取指定类型的收费单位列表

接口地址：fee/collector

请求方式：GET

请求主体：无

请求示例：fee/collector?type=2

请求数据说明：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 必填 | 类型 | 说明 |
| type | 是 | number | 收费类型， 1-水费、2-电费、3-燃气费、4-物业费、5-停车费、6-宽带费 |

响应主体格式：json array；响应主体示例：

[

   {

        "cid": 101,

        "cname": "北京市自来水有限公司",

        "addr": "北京市东城区民旺园101号",

        "phone": "010-69181821"

    },

......

]

## 3.3、提交缴费记录

接口地址：fee/add

请求方式：POST

请求主体格式：application/json

请求头部：token - 用户登录后保存在客户端的身份凭证

请求数据说明：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 必填 | 类型 | 说明 |
| token | 是 | string | 请求头部中必须携带登录成功后得到的身份令牌，且必须在有效期内 |
| type | 是 | number | 1-水费、2-电费、3-燃气费、4-物业费、5-停车费、6-宽带费 |
| collectorId | 是 | number | 收费单位编号 |
| householdId | 是 | string | 业主用户编号 |
| amount | 是 | float(10,2) | 缴费金额 |

响应主体格式：json object，属性：

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 说明 |
| code | int | 返回码  2000-修改成功  4100-令牌缺失  4101-令牌无效  4102-令牌已过期  4001-收费单位编号缺失  4002-业主用户编号缺失  4005-缴费金额缺失  4006-缴费类型缺失 |
| msg | string | 原因说明 |

成功的响应主体示例：

{ "code":2000, "msg":"add succ" }

## 3.4、当前登录用户缴费记录

接口地址：fee/record

请求方式：GET

请求主体：无

请求参数：type - 费用的类型；

startTime - 缴费开始时间（时间戳数字）；

endTime - 缴费结束时间（时间戳数字）；

请求头部：token - 用户登录后保存在客户端的身份凭证

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 必填 | 类型 | 说明 |
| type | 可选 | number | 收费类型， 1-水费、2-电费、3-燃气费、4-物业费、5-停车费、6-宽带费  如果未指定，默认显示全部缴费记录 |
| startTime | 可选 | number | 缴费开始时间（时间戳数字） |
| endTime | 可选 | number | 缴费结束时间（时间戳数字） |

响应主体格式：json array；

响应主体示例：

[

    {

        "fid": 36,

        "type": 4,

        "collectorName": "天通苑物管有限公司",

        "householdId": "W60011",

        "householdName": "秦小雅",

        "householdAddr": "天通苑社区A18-1单元1205",

        "paidTime": 1783615514141,

        "amount": 20

    },

......

]

# 四、业主模块

## 4.1、根据业主缴费编号查询该业主的基本信息

接口地址：household/query

请求方式：GET

请求主体：无

请求示例：household/query?hhid=S60011

注意：前端对此接口的请求需要防抖，时间间隔在2s以上

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 必填 | 类型 | 说明 |
| hhid | 必需 | string | 待查询的业主缴费编号 |

响应主体格式：json object，属性：

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 说明 |
| code | int | 返回码  2000-查询成功  4001-业主缴费编号缺失  4002-业主缴费编号不存在 |
| msg | string | 原因说明，或查询到的数据 |

响应主体示例：

{

    "code": 2000,

    "msg": {

        "hid": 4,

        "householdId": "W60011",

        "householdName": "秦\*雅",

        "householdAddr": "天通苑社区A18-1单元1205"

    }

}

# 五、智慧停车模块

## 5.1、返回所有停车区坐标及当前空闲数据

接口地址：parking/zone/list

请求方式：GET

请求主体：无

请求示例：parking/zone/list

响应主体格式：json array，

响应主体示例：

[

    {

        "zid": 1,

        "zname": "天通苑一区",

        "longitude": 116.417721,

        "latitude": 40.063224,

        "unoccupied": 96

    },

......

]

# 六、邻里互动模块

## 6.1、上传一张互动消息中涉及到的图片

接口地址：chat/pic/upload

请求方式：POST

请求类型：multipart/form-data

请求数据：pic，其值上传的图片内容

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 必填 | 类型 | 说明 |
| pic | 必需 | blob | 待上传的图片 |

响应主体格式：json object，属性：

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 说明 |
| code | int | 返回码  2000-上传成功  4000-上传失败 |
| msg | string | 原因说明，或查询到的数据 |
| fileName | string | 保存在服务器上的文件名 |

响应主体示例：

{

    "code": 2000,

    "msg": "upload succ",

    "fileName": "img/chat/181610451646028.png"

}

## 6.2、添加一条互动消息

接口地址：chat/add

请求方式：POST

请求类型：application/json

请求头部：token - 用户登录后保存在客户端的身份凭证

请求数据：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 必填 | 类型 | 说明 |
| location | 必需 | string | 当前用户所在地理位置 |
| content | 必需 | string | 互动消息的内容 |
| picList | 必需 | array | 空数组：[ ]  或者：[{fileName: '...'}, {fileName:'...'}] |

响应主体格式：json object，属性：

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 说明 |
| code | int | 返回码  2000-消息添加成功 |

响应主体示例：

{

    "code": 2000,

    "msg": "msg inserted succ"

}

## 6.3、分页获取互动消息列表

接口地址：chat/query

请求方式：GET

请求参数：pageNum

请求数据：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 必填 | 类型 | 说明 |
| pageNum | 非必需 | number | 默认值为1；要查询的互动消息页号 |

响应主体格式：json object，属性：

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 说明 |
| code | int | 返回码  2000-消息添加成功 |

响应主体示例：

{

    "pageSize": 9,

    "recordCount": 43,

    "pageCount": 5,

    "pageNum": 1,

    "data": [

        {

            "mid": 42,

            "uid": 3,

            "location": "北京市天通苑三区",

            "content": "42投票：什么样的社区服务才是您心目中完美的社区服务呢？",

            "pubTime": 1790126345163,

            "likeCount": 0,

            "replyCount": 5,

            "uname": "dou123",

            "email": "dou@tedu.cn",

            "avatar": "img/user1.png",

            "gender": 1,

            "picList": [

                {

                    "pid": 1,

                    "url": "img/chat/16704373811634187.jpg"

                },

                {

                    "pid": 2,

                    "url": "img/chat/16715253811634318.jpg"

                },

                {

                    "pid": 3,

                    "url": "img/chat/16125973811637190.jpg"

                }

            ]

        },

......

    ]

}

# 七、社区商城模块

## 7.1、商城首页数据

接口地址：mall/index

请求方式：GET

请求参数：无

请求头部：token - 用户登录后保存在客户端的身份凭证

响应主体格式：json object，属性：

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 说明 |
| userInfo | object | 当前登录用户信息 |
| carousels | array | 商城首页轮播广告列表 |
| goodsTypes | array | 商品种类列表 |

响应主体示例：

{

    "userInfo": {

        "uname": "秦小雅",

        "avatar": "img/user0.png",

        "email": "ya@tedu.cn",

        "gender": 0

    },

    "carousels": [

        {

            "cid": 4,

            "pic": "img/ad/1816513463342.jpg",

            "title": "免费天天领",

            "href": "mdetails",

            "type": 2

        },

        ......

    ],

    "goodsTypes": [

        {

            "tid": 1,

            "tname": "折扣"

        },

        ......

    ]

}

## 7.2、返回指定类型的商品列表

接口地址：mall/goods/list

请求方式：GET

查询参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 必填 | 类型 | 说明 |
| tid | 可选 | int | 待查询的商品类型  1-折扣商品，默认值  2-新年精选  3-品牌精选  ... |
| order | 可选 | string | 查询结果排序方法，可选值：  soldcount：销量，默认值  price-asc：价格升序排列  price-desc：价格降序排列 |

响应主体格式：json array，

响应主体示例：

[

    {

        "gid": 26,

        "mainPic": "img/goods/main/d0948013322282e0.jpg",

        "goodsName": "GUO LIAN国联 国产大虾 净重1.8kg 90-108只 盒装活冻 海鲜烧烤家庭食材",

        "originalPrice": 119,

        "discount": 0.95,

        "soldCount": 87

    },

......

]

## 7.3、返回指定编号的商品详情

接口地址：mall/goods/details

请求方式：GET

查询参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 必填 | 类型 | 说明 |
| gid | 必需 | int | 待查询的商品编号 |

响应主体格式：json object，查询失败的响应消息说明：

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 说明 |
| code | int | 返回码  4001-商品编号缺失  4002-商品编号不存在 |

查询成功的响应主体示例：

{

    "gid": 2,

    "mainPic": "img/goods/main/328921366c9cae0e.jpg",

    "shortTitle": "小台农芒果",

    "goodsName": "海南小台农芒果新鲜小台芒9斤应当季水果现摘甜心小芒果忙果整箱批发 9斤 精选大果",

    "descriptions": "新鲜小台芒，香甜好吃，10斤批发价特惠装",

    "originalPrice": 38.8,

    "discount": 0.9,

    "soldCount": 318,

    "details": "<div class="gdetails"><img src="img/goods/details/b82a8f47c68b43d1.jpg">...</div>",

    "isNewYearSelection": 0,

    "isBrandSelection": 1

}

# 八、管理员模块

## 8.1、管理员登录

接口地址：admin/login

请求方式：POST

请求主体格式：application/json

请求数据说明：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 必填 | 类型 | 说明 |
| aname | 是 | string | 管理员登录名 |
| apwd | 是 | string | 管理员登录密码 |

响应主体格式：json object，属性：

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 说明 |
| code | int | 返回码  2000-登录成功  4001-管理员名为空  4002-密码为空  4003-管理员名或密码错误 |
| msg | string | 原因说明 |
| token | string | 此次登录获得的身份令牌 |

响应主体示例：

{

    "code": 2000,

    "msg": "login succ",

    "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXV....."

}

## 8.2、获取当前登录管理员的基础信息及权限信息

接口地址：admin/info

请求方式：GET

请求主体：无

请求参数：无

请求头部：token - 管理员登录后保存在客户端的身份凭证

响应主体格式：json object；

响应主体示例：

{

    "aname": "aimin",

    "avatar": "img/user.png",

    "userName": "毛爱民",

    "phone": "13901234567",

    "notes": "毛爱民",

    "rname": "物业人员",

    "privileges": [

        {

            "pid": 1,

            "pname": "首页",

            "path": "/admin/index",

            "icon": "home",

            "parentId": 0,

            "children": []

        },

        {

            "pid": 2,

            "pname": "智慧物业",

            "path": "/admin/property",

            "icon": "building",

            "parentId": 0,

            "children": [

                {

                    "pid": 8,

                    "pname": "物业缴费",

                    "path": "/admin/property/fee/pay",

                    "icon": null,

                    "parentId": 2,

                    "children": []

                },

                ......

            ]

        },

        ......

    ]

}

## 8.3、修改当前登录的管理员密码

接口地址：admin/update/pwd

请求方式：POST

请求主体格式：application/json

请求头部：token

请求数据说明：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 必填 | 类型 | 说明 |
| token | 是 | string | 请求头部中必须携带登录成功后得到的身份令牌，且必须在有效期内 |
| oldPwd | 是 | string | 原有密码 |
| newPwd | 是 | string | 新密码 |

响应主体格式：json object，属性：

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 说明 |
| code | int | 返回码  2000-修改成功  2000-修改成功  4001-旧密码为空  4002-新密码为空  4003-旧密码错误  4004-新旧密码一样  4100-令牌缺失  4101-令牌无效  4102-令牌已过期 |
| msg | string | 原因说明 |

成功的响应主体示例：

{ "code":2000, "msg":"update succ" }

## 8.4、分页获取居民列表

接口地址：admin/household/list

请求方式：GET

请求头部：token - 管理员登录后保存在客户端的身份凭证

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 必填 | 类型 | 说明 |
| pageNum | 非必需 | number | 默认值为1；要查询的页号 |
| kw | 非必需 | string | 待查询的居民姓名关键字 |

响应主体格式：json object，属性：

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 说明 |
| code | int | 返回码  2000-消息添加成功 |

响应主体示例：

{

    "pageSize": 6,

    "recordCount": 38,

    "pageCount": 5,

    "pageNum": 1,

    "data": [

        {

            "hid": "1",

            "householdId": "W70381",

            "householdName": "韩仲媛",

            "gender": 0,

            "householdAddr": "天通苑社区D03-3单元1508",

            "phone": "16872084877",

            "idNum": "320102199003075467"

        },

......

    ]

}

## 8.5、查询特定居民详情

接口地址：admin/household/details

请求方式：GET

请求头部：token - 管理员登录后保存在客户端的身份凭证

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 必填 | 类型 | 说明 |
| hid | 必需 | number | 待查询的居民编号 |

响应主体格式：json object，属性：

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 说明 |
| code | int | 返回码  4000-未提交要查询的居民编号  4001-待查询的居民编号不存在 |

响应主体示例：

{

    "hid": 22,

    "householdId": "D70016",

    "householdName": "贺兴国",

    "gender": 0,

    "householdAddr": "天通苑社区B13-2单元1206",

    "phone": "16337404274",

    "idNum": "320102199003074261"

}

## 8.6、删除特定居民记录

接口地址：admin/household/delete

请求方式：GET

请求头部：token - 管理员登录后保存在客户端的身份凭证

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 必填 | 类型 | 说明 |
| hid | 必需 | number | 待删除的居民编号 |

响应主体格式：json object，属性：

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 说明 |
| code | int | 返回码  4000-未提交要删除的居民编号  4001-待删除的居民编号不存在 |

响应主体示例：

{

    "code": 2000,

    "msg": "delete succ"

}

## 8.7、修改特定居民记录

接口地址：admin/household/udpate

请求方式：POST

请求头部：token - 管理员登录后保存在客户端的身份凭证

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 必填 | 类型 | 说明 |
| hid | 必需 | number | 待修改的居民编号 |
| householdId | 可选 | string | 待修改的居民户号 |
| householdName | 可选 | string | 待修改的居民姓名 |
| householdAddr | 可选 | string | 待修改的居民住址 |
| gender | 可选 | 0或1 | 待修改的性别 |
| phone | 可选 | string | 待修改的电话 |
| idNum | 可选 | string | 带修改的身份证号 |

响应主体格式：json object，属性：

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 说明 |
| code | int | 返回码  4000-未提交要修改的居民编号  4001-待修改的居民编号不存在 |

响应主体示例：

{

    "code": 2000,

    "msg": "update succ"

}