

# 接口文档

---

- 此服务器以开启 `cors` (跨域资源共享)
- 此服务器中所有数据都是以 `json` 文件形式保存, 并未连接数据库
- 登录状态 `默认` 在重启服务器后会消失
  - 如果需要重启服务器后登陆状态保持
  - 请打开 `src/conf/index.js` 文件
  - 修改 `keep` 字段为 `true`
- 登录状态 `默认` 会保存 `1小时`
  - 如果需要修改
  - 请打开 `src/conf/index.js` 文件
  - 修改 `tokenKeep` 字段, 单位是 秒(s)
- 本服务器对于用户名和密码有默认的验证正则
  - 用户名: `/^[a-z0-9]\w{3,11}$/`
  - 密码: `/\w{6,12}/`
  - 如果需要修改
  - 请打开 `src/conf/index.js` 文件
  - 修改 `username` 和 `password` 正则
  - 注意: 一定要修改为 正则 数据类型, 否则服务器会出现问题
- 此服务器默认端口号是 `9000`
  - 如果需要修改
  - 请打开 `src/package.json` 文件
  - 修改 `start` 字段后面的 `9000`
- 此服务器所有需要登录后使用的接口
  - 均在 请求头 内携带 `authorization` 字段, 传递 `token` 值方可
  - 登录成功后会在 响应体 内返回 `token`
- 服务器中默认账号
  - 查看 `src/db/users.json` 文件

## 请求基准地址: `http://localhost:端口号`

---

- 如果你没有修改过, 基准地址为: `http://localhost:9000`
- 如果你修改过, 把端口号位置书写成你修改后的内容即可

## 状态码

---

状态码为返回数据的标识, 相同含义的状态总是会返回相同的状态码, 部分状态码具体含义取决于不同功能

状态码	介绍	备注
1	成功	任何成功都会返回 1
0	失败	各种失败都会返回 0

# 开局测试

---

## 1 测试请求1

### 1.1 请求地址

/test/first

### 1.2 请求方式

get

### 1.3 携带参数

无

### 1.4 响应数据

哇塞, 你已经成功使用 ajax 发送给我了一个请求, 这是我给你的回应, 我是一个字符串类型 ^\_^!

## 2 测试请求2

### 2.1 请求地址

/test/second

### 2.2 请求方式

get

### 2.3 携带参数

无

### 2.4 响应数据

json 格式数据类型

```
{
  "message": "我已经接受到了你的请求，这是我给你的回应，我是一个 json 格式",
  "tips": "后端返回给前端的数据",
  "code": 1,
  "age": 18
}
```

## 3 测试请求3

### 3.1 请求地址

/test/third

### 3.2 请求方式

get

### 3.3 携带参数

参数名称	是否必填	数据类型	默认值	参数介绍
name	必填	string	-	-
age	必填	number	-	-

### 3.4 响应数据

json 格式数据

```
{
  "message": "我接收到了你的请求，你的请求方式是 GET，并且正确携带了 'name' 和 'age' 参数给我！<(*
  ──▽─*)/",
  "code": 1,
  "info": {
    "msg": "这是你带来的参数，我还返回给你，让你看看，证明你带过来了",
    "name": "asd",
    "age": "11"
  }
}
```

## 4 测试请求4

### 4.1 请求地址

/test/fourth

### 4.2 请求方式

post

## 4.3 携带参数

参数名称	是否必填	数据类型	默认值	参数介绍
name	必填	string	-	-
age	必填	number	-	-

## 4.4 响应数据

json 格式数据

```
{
  "message": "我接收到了你的请求，你的请求方式是 POST，并且正确携带了 'name' 和 'age' 参数给我！",
  "<(*~▽~*)/",
  "code": 1,
  "info": {
    "msg": "这是你带来的参数，我还返回给你，让你看看，证明你带过来了",
    "name": "133",
    "age": "22"
  }
}
```

# 仿站电商前台接口介绍

## 商品相关

### 1 获取商品列表

#### 1.1 请求地址

/goods/list

#### 1.2 请求方式

get

#### 1.3 携带参数

参数名称	是否必填	数据类型	默认值	参数介绍
current	选填	number	1	当前页
pagesize	选填	number	12	一页多少条数据
search	选填	string	-	模糊搜索关键字
filter	选填	hot,sale	-	(热销/折扣)筛选
saleType	选填	number(5 ~ 10)	10	折扣类型
sortType	选填	id    sale    price	id	排序类型
sortMethod	选填	ASC    DESC	ASC	ASC 正序, DESC 倒序
category	选填	string	-	分类筛选条件

## 1.4 响应数据

根据你的参数给出的商品列表

```
{
  "message": "获取商品列表成功",
  "code": 1,
  "yourParams": {
    "msg": "这里是传递过来的参数，带给你看看 ^_^ ",
    ...
  },
  "list": [ ... ]
}
```

# 2 获取商品详细信息

## 2.1 请求地址

/goods/item

## 2.2 请求方式

get

## 2.3 携带参数

支持 restful 风格 /goods/item/:id

参数名称	是否必填	数据类型	默认值	参数介绍
id	必填	number	-	商品 id

## 2.4 响应数据

如果该商品存在, 即为该商品的详细信息

```
{
  "message": "获取商品详细信息成功",
  "code": 1,
  "yourParams": {
    "msg": "这里是传递过来的参数, 带给你看看 ^_^ ",
    "id": "1"
  },
  "info": {
    ...
  }
}
```

## 3 获取分类列表

### 3.1 请求地址

/goods/category

### 3.2 请求方式

get

### 3.3.3 携带参数

无

### 3.3.4 相应数据

所有商品分类内容

```
{
  "message": "获取分类列表成功",
  "code": 1,
  "list": [ ... ]
}
```

## 用户相关

### 1 用户登录

#### 1.1 请求地址

/users/login

#### 1.2 请求方式

post

### 1.3 携带参数

参数名称	是否必填	数据类型	默认值	参数介绍
username	必填	string	-	用户名
password	必填	string	-	密码

### 1.4 响应数据

根据你的用户名和密码返回登录状态

```
{
  "message": "恭喜登录成功",
  "code": 1,
  "user": {...},
  "token": "..."
```

## 2 注销登录

### 2.1 请求地址

/users/logout

### 2.2 请求方式

get

### 2.3 携带参数

参数名称	是否必填	数据类型	默认值	参数介绍
id	必填	number	-	注销登录的用户 id

### 2.4 响应数据

```
{
  "message": "注销成功",
  "code": 1
}
```

## 3 注册新用户

### 3.1 请求地址

/users/register

### 3.2 请求方式

post

### 3.3 携带参数

参数名称	是否必填	数据类型	默认值	参数介绍
username	必填	string	-	用户名
password	必填	string	-	密码
rpassword	必填	string	-	重复密码(必须和密码一致)
nickname	必填	string	-	用户昵称

### 3.4 响应数据

```
{
  "message": "恭喜注册成功!",
  "code": 1
}
```

## 4 获取用户详细信息

- 注意: 登录后方可查看

### 4.1 请求地址

/users/info

### 4.2 请求方式

get

### 4.3 携带参数

参数名称	是否必填	数据类型	默认值	参数介绍
id	必填	number	-	查看信息的用户 id

### 4.4 响应数据



```
{
  "message": "获取用户信息成功",
  "code": 1,
  "user": { ... }
}
```

## 5 修改个人资料

- 注意: 登录后方可修改

### 5.1 请求地址

/users/update

### 5.2 请求方式

post

### 5.3 携带参数

参数名称	是否必填	数据类型	默认值	参数介绍
id	必填	number	-	要修改的用户 id
age	选填(如需修改)	number	-	修改年龄信息
gender	选填(如需修改)	男    女	-	修改性别信息
nickname	选填(如需修改)	string	-	修改昵称信息

### 5.4 响应数据

```
{
  "message": "修改用户信息成功 ! 0(n_n)0哈哈~",
  "code": 1,
  "user": {...}
}
```

## 6 修改密码

- 注意: 登录后方可修改

### 6.1 请求地址

/users/rpwd

### 6.2 请求方式

post

6.3 携带参数

参数名称	是否必填	数据类型	默认值	参数介绍
id	必填	number	-	要修改的用户 id
oldPassword	必填	number	-	原始密码
newPassword	必填	number	-	新密码
rNewPassword	必填	number	-	重复新密码(必须和新密码一致)

6.4 响应数据

修改密码成功后, 会自动注销当前登录状态, 需要重新登录

```
{
  "message": "修改密码成功，已经注销登录状态，请重新登录！^_^",
  "code": 1
}
```

获取轮播图列表

请求地址

carousel/list

请求方式

get

携带参数

无

响应数据

```
{
  code: 1,
  message: '获取轮播图列表成功',
  list: [...] // {id: 轮播图唯一表示,name:轮播图名称}
}
```

注意: 在浏览器中直接通过 `http://127.0.0.1:9000/轮播图名称` 访问该轮播图

仿站电商后台接口介绍

1 后台登录

1.1 请求地址

/users/backend/login

## 1.2 请求方式

post

## 1.3 携带参数

参数名称	是否必填	数据类型	默认值	参数介绍
username	必填	string	-	用户名
password	必填	string	-	密码

## 1.4 响应数据

```
{
  "message": "恭喜登录成功",
  "code": 1,
  "user": {...},
  "token": "..."
```

后台登录注意: 用户名只能是: admin 密码为: 123456 只有admin用户才具有后台管理权限

# 2 后台获取验证码

## 2.1 请求地址

/users/backend/code

## 2.2 请求方式

post

## 2.3 携带参数

无

## 2.4 响应数据

```
{
  "message": "获取验证码成功",
  "code": 1,
  "data": {...} // {text:验证码,data:验证码的svg图片}
}
```

# 3 后台退出注销

- 注意: 登录后方可退出

### 3.1 请求地址

/users/backend/logout

### 3.2 请求方式

get

### 3.3 携带参数

参数名称	是否必填	数据类型	默认值	参数介绍
id	必填	string	-	用户id

### 3.4 响应数据

```
{
  "message": "注销成功",
  "code": 1,
}
```

## 4 获取用户列表

- 注意: 登录后方可获取

### 4.1 请求地址

/users/backend/list

### 4.2 请求方式

get

### 4.3 携带参数

参数名称	是否必填	数据类型	默认值	参数介绍
id	必填	string	-	后台管理员id

### 4.4 响应数据

根据你的用户名和密码返回登录状态

```
{
  "code": 1,
  "list": [...], // 所有的用户信息,不包含管理员admin的
}
```

## 5 获取搜索用户列表

- 注意: **登录后方可获取**

### 5.1 请求地址

/users/backend/search

### 5.2 请求方式

get

### 5.3 携带参数

参数名称	是否必填	数据类型	默认值	参数介绍
id	必填	string	-	后台管理员id
searchStr	非必填	string	"	搜索关键字

### 5.4 响应数据

根据你的用户名和密码返回登录状态

```
{
  "code": 1,
  "list": [...], // 所有的用户信息,不包含管理员admin的
}
```

## 6 重置用户密码

- 注意: **登录后方可重置**

### 6.1 请求地址

/users/backend/reset

### 6.2 请求方式

post

### 6.3 携带参数

参数名称	是否必填	数据类型	默认值	参数介绍
id	必填	string	-	后台管理员id
uid	必填	string	-	重置用户的id

## 6.4 响应数据

根据你的用户名和密码返回登录状态

```
{
  "code": 1,
  "message": '重置密码成功'
}
```

# 7 用户禁用/解禁

- 注意: *登录后方可操作*

## 7.1 请求地址

/users/backend/flag

## 7.2 请求方式

post

## 7.3 携带参数

参数名称	是否必填	数据类型	默认值	参数介绍
id	必填	string	-	后台管理员id
uid	必填	string	-	操作用户的id

## 7.4 响应数据

根据你的用户名和密码返回登录状态

```
{
  "code": 1,
  "message": '操作成功'
}
```

注意: 被禁用的用户在前台不可登录

## 8 删除用户

- 注意: 登录后方可操作

### 8.1 请求地址

/users/backend/del

### 8.2 请求方式

post

### 8.3 携带参数

参数名称	是否必填	数据类型	默认值	参数介绍
id	必填	string	-	后台管理员id
uid	必填	string	-	删除用户的id

### 8.4 响应数据

```
{
  "code": 1,
  "message": '操作成功'
}
```

注意: 被删除的用户在前台不可登录

## 9 获取轮播图列表

### 9.1 请求地址

carousel/list

### 9.2 请求方式

get

### 9.3 携带参数

无

### 9.4 响应数据

```
{
  code: 1,
  message: '获取轮播图列表成功',
  list: [...] // {id: 轮播图唯一表示,name:轮播图名称}
}
```

注意: 在浏览器中直接通过 `http://127.0.0.1:9000/轮播图名称` 访问该轮播图

## 10 删除轮播图

- 注意: 登录后方可操作

### 10.1 请求地址

`carousel/backend/remove`

### 10.2 请求方式

`post`

### 10.3 携带参数

参数名称	是否必填	数据类型	默认值	参数介绍
id	必填	string	-	后台管理员id
imgId	必填	string	-	要删除的轮播图的id标识

### 10.4 响应数据

```
{
  code: 1,
  message: '删除成功',
}
```

## 11 添加轮播图

- 注意: 登录后方可操作

### 11.1 请求地址

`carousel/backend/add`

### 11.2 请求方式

`post`

### 11.3 携带参数



参数名称	是否必填	数据类型	默认值	参数介绍
id	必填	string	-	后台管理员id
carousel	必填	文件	-	上传的文件

## 11.4 响应数据

```
{
  code: 1,
  message: '上传成功',
  data: {...} // {id: 轮播图唯一表示,name:上传的轮播图文件名}
}
```