

# 个人项目

Base URLs:

- 开发环境: <http://dev-cn.your-api-server.com>

## Authentication

## MusicApp/用户模块

### POST 用户登录

POST /localhost:8080/user/login

Body 请求参数

```
username: root
password: "123456"
```

### 请求参数

名称	位置	类型	必选	说明
body	body	object	否	none
» username	body	string	否	none
» password	body	string	否	none

返回示例

成功

```
{
  "status": "success",
  "message": "登录成功",
  "data": {
    "userId": "12345",
    "username": "example_user",
    "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9..."
  }
}
```

```
{
  "status": "error",
  "message": "用户名或密码错误"
}
```

401 Response

```
{}
```

## 返回结果

状态码	状态码含义	说明	数据模型
200	<a href="#">OK</a>	成功	Inline
401	<a href="#">Unauthorized</a>	没有权限	Inline

## 返回数据结构

## POST 用户注册

POST /localhost:8080/user/register

Body 请求参数

```
username: root
password: "123456"
repassword: "123456"
```

## 请求参数

名称	位置	类型	必选	说明
body	body	object	否	none
» username	body	string	否	none
» password	body	string	否	none
» repassword	body	string	否	确认密码

返回示例

成功

```
{
  "status": "success",
  "message": "注册成功"
}
```

```
{
  "status": "error",
  "message": "注册失败，用户名已存在"
}
```

400 Response

```
{}
```

## 返回结果

状态码	状态码含义	说明	数据模型
200	<a href="#">OK</a>	成功	Inline
400	<a href="#">Bad Request</a>	请求有误	Inline

## 返回数据结构

## POST 用户注销

POST /localhost:8080/user/logout

返回示例

成功

```
{
  "status": "success",
  "message": "注销成功"
}
```

```
{
  "status": "error",
  "message": "注销失败"
}
```

401 Response

```
{}
```

## 返回结果

状态码	状态码含义	说明	数据模型
200	<a href="#">OK</a>	成功	Inline
401	<a href="#">Unauthorized</a>	没有权限	Inline

## 返回数据结构

## PUT 上传用户头像

PUT /localhost:8080/user/updateAvatar

## 请求参数

名称	位置	类型	必选	说明
avatar	query	string	否	图片url

返回示例

200 Response

```
{}
```

返回结果

状态码	状态码含义	说明	数据模型
200	<a href="#">OK</a>	成功	Inline

返回数据结构

PUT 修改密码

PUT /localhost:8080/user/updatePassword

请求参数

名称	位置	类型	必选	说明
password	query	string	否	none

返回示例

200 Response

```
{}
```

返回结果

状态码	状态码含义	说明	数据模型
200	<a href="#">OK</a>	成功	Inline

返回数据结构

MusicApp/音乐操作模块

GET 音乐搜索（条件搜索歌名，歌手）

GET /localhost:8080/song/search

Body 请求参数

**keyword**: 晴天

## 请求参数

名称	位置	类型	必选	说明
body	body	object	否	none
» keyword	body	string	否	none

返回示例

200 Response

<pre>{   "code": "string",   "date[]": "string" }</pre>
---

## 返回结果

状态码	状态码含义	说明	数据模型
200	<a href="#">OK</a>	成功	Inline
404	<a href="#">Not Found</a>	记录不存在	Inline

## 返回数据结构

状态码 200

名称	类型	必选	约束	中文名	说明
» code	string	true	none		none
» date[]	string	true	none		none

## POST 音乐收藏

POST /localhost:8080/song/favorite

## 请求参数

名称	位置	类型	必选	说明
songID	query	string	否	none

返回示例

成功

```
{
  "status": "success",
  "message": "收藏成功"
}
```

```
{
  "status": "error",
  "message": "收藏失败，歌曲不存在或已收藏"
}
```

## 返回结果

状态码	状态码含义	说明	数据模型
200	<a href="#">OK</a>	成功	Inline

## 返回数据结构

## POST 取消收藏

POST /localhost:8080/song/cancelfavorite

## 请求参数

名称	位置	类型	必选	说明
songID	query	string	否	none

返回示例

成功

```
{
  "status": "success",
  "message": "取消成功"
}
```

```
{
  "status": "error",
  "message": "取消收藏失败"
}
```

## 返回结果

状态码	状态码含义	说明	数据模型
200	<a href="#">OK</a>	成功	Inline

## 返回数据结构

## GET 查看收藏音乐

GET /localhost:8080/song/listfavorite

- 返回示例
- 成功

```
{
  "status": "success",
  "data": [
    {
      "id": 1,
      "title": "Song Title 1",
      "artist": "Artist Name 1",
      "album": "Album Name 1",
      "duration": "3:45",
      "genre": "Pop"
    },
    {
      "id": 2,
      "title": "Song Title 2",
      "artist": "Artist Name 2",
      "album": "Album Name 2",
      "duration": "4:02",
      "genre": "Rock"
    }
  ]
}
```

## 返回结果

状态码	状态码含义	说明	数据模型
200	<a href="#">OK</a>	成功	Inline

## 返回数据结构

## GET 查询音乐详情

GET /localhost:8080/song/findsongbyID

## 请求参数

名称	位置	类型	必选	说明
songID	query	string	否	none

- 返回示例
- 成功

```
{
  "status": "success",
  "data": {
    "songId": "1",
    "songName": "Example Song",
    "artist": "Example Artist",
    "album": "Example Album",
    "imageUrl": "http://example.com/song/image.jpg",
    "audioUrl": "http://example.com/song/audio.mp3"
  }
}
```

返回结果

状态码	状态码含义	说明	数据模型
200	<a href="#">OK</a>	成功	Inline

返回数据结构

# MusicApp/歌手模块

## GET 查询歌手列表

GET /localhost:8080/singer/listSinger

返回示例

成功

```
{
  "status": "success",
  "data": [
    {
      "artistId": "1",
      "name": "Artist1",
      "description": "Description of Artist1"
    },
    {
      "artistId": "2",
      "name": "Artist2",
      "description": "Description of Artist2"
    }
  ]
}
```

返回结果

状态码	状态码含义	说明	数据模型
200	<a href="#">OK</a>	成功	Inline



## 返回数据结构

## GET 查询歌手详细信息

GET /localhost:8080/singer

### 请求参数

名称	位置	类型	必选	说明
id	query	string	否	歌手id

返回示例

成功

```
{
  "status": "success",
  "data": {
    "artistId": "1",
    "name": "Artist1",
    "description": "Description of Artist1"
  }
}
```

## 返回结果

状态码	状态码含义	说明	数据模型
200	<a href="#">OK</a>	成功	Inline

## 返回数据结构

## MusicApp/后台管理模块

## GET 后台首页数据总数

GET /localhost:8080/admin/count

返回示例

成功

```
{
  "status": "success",
  "data": {
    "userCount": 100,
    "artistCount": 50,
    "musicCount": 500,
    "mvCount": 200,
    "musicByArtistChart": {
      "artist1": 30,
      "artist2": 20
    }
  }
}
```

```
    },
    "musicTypePieChart": {
      "pop": 40,
      "rock": 30
    }
  }
}
```

返回结果

状态码	状态码含义	说明	数据模型
200	<a href="#">OK</a>	成功	Inline

返回数据结构

GET 查看用户详情

GET /localhost:8080/admin/listuser

- 返回示例
- 成功

```
{
  "status": "success",
  "data": [
    {
      "userId": "1",
      "username": "user1",
      "email": "user1@example.com"
    },
    {
      "userId": "2",
      "username": "user2",
      "email": "user2@example.com"
    }
  ]
}
```

返回结果

状态码	状态码含义	说明	数据模型
200	<a href="#">OK</a>	成功	Inline

# MusicApp/后台管理模块/歌手管理

## POST 添加歌手

POST /localhost:8080/admin/addSinger

Body 请求参数

```
name: 周杰伦
sex: "1"
birth: 2004-02-10
pic: string
location: 江苏
introduction: 牛逼
```

### 请求参数

名称	位置	类型	必选	说明
body	body	object	否	none
» name	body	string	否	none
» sex	body	integer	否	性别
» birth	body	string	否	生日
» pic	body	string	否	图片url
» location	body	string	否	家乡
» introduction	body	string	否	歌手介绍

返回示例

成功

```
{
  "status": "success",
  "message": "歌手添加成功"
}
```

### 返回结果

状态码	状态码含义	说明	数据模型
200	<a href="#">OK</a>	成功	Inline

## 返回数据结构

## PUT 修改歌手信息

PUT /localhost:8080/admin/updatesinger

Body 请求参数

```
singerID: "0001"
name: 周杰伦
description: 牛逼
```

## 请求参数

名称	位置	类型	必选	说明
body	body	object	否	none
» singerID	body	string	否	none
» name	body	string	否	none
» description	body	string	否	none

返回示例

成功

```
{
  "status": "success",
  "message": "歌手信息更新成功"
}
```

## 返回结果

状态码	状态码含义	说明	数据模型
200	<a href="#">OK</a>	成功	Inline

## 返回数据结构

## DELETE 删除歌手

DELETE /localhost:8080/admin/deletesinger

## 请求参数

名称	位置	类型	必选	说明
singerID	query	string	否	none

返回示例

成功

```
{
  "status": "success",
  "message": "歌手删除成功"
}
```

返回结果

状态码	状态码含义	说明	数据模型
200	<a href="#">OK</a>	成功	Inline

返回数据结构

GET 查找歌手

GET /localhost:8080/admin/searchsinger

请求参数

名称	位置	类型	必选	说明
query	query	string	否	关键词

返回示例

成功

```
{
  "status": "success",
  "data": [
    {
      "artistId": "1",
      "name": "Artist1",
      "description": "Description of Artist1"
    },
    {
      "artistId": "2",
      "name": "Artist2",
      "description": "Description of Artist2"
    }
  ]
}
```

返回结果

状态码	状态码含义	说明	数据模型
200	<a href="#">OK</a>	成功	Inline

返回数据结构

MusicApp/后台管理模块/音乐歌单管理

POST 新增歌曲

POST /localhost:8080/admin/song/add

Body 请求参数

```
name: string
singer_name: string
pic: string
url: string
```

请求参数

名称	位置	类型	必选	说明
body	body	object	否	none
» name	body	string	是	(string): 歌单名
» singer_name	body	string	是	(string): 歌单描述
» pic	body	string	是	图片路径
» url	body	string	是	歌曲路径

返回示例

成功

```
{
  "status": "success",
  "message": "音乐歌单添加成功"
}
```

返回结果

状态码	状态码含义	说明	数据模型
200	<a href="#">OK</a>	成功	Inline

## 返回数据结构

## PUT 修改歌曲

PUT /localhost:8080/admin/song/update

Body 请求参数

```
name: string
singer_name: string
pic: string
url: string
id: 0
```

## 请求参数

名称	位置	类型	必选	说明
body	body	object	否	none
» name	body	string	是	(string): 歌单名
» singer_name	body	string	是	(string): 歌单描述
» pic	body	string	是	图片路径
» url	body	string	是	歌曲路径
» id	body	integer	否	歌曲id

返回示例

成功

```
{
  "status": "success",
  "message": "音乐歌单添加成功"
}
```

## 返回结果

状态码	状态码含义	说明	数据模型
200	<a href="#">OK</a>	成功	Inline

## 返回数据结构

## DELETE 删除歌曲

DELETE /localhost:8080/admin/playlist/delete

## 请求参数

名称	位置	类型	必选	说明
id	query	integer	是	歌单ID

返回示例

成功

```
{
  "status": "success",
  "message": "歌单删除成功"
}
```

## 返回结果

状态码	状态码含义	说明	数据模型
200	<a href="#">OK</a>	成功	Inline

## 返回数据结构

### GET 查询歌曲内容

GET /localhost:8080/admin/playlist

## 请求参数

名称	位置	类型	必选	说明
id	query	integer	是	歌曲id

返回示例

成功

```
{
  "status": "success",
  "data": [
    {}
  ]
}
```

## 返回结果

状态码	状态码含义	说明	数据模型
200	<a href="#">OK</a>	成功	Inline



返回数据结构

## 数据模型

---