

boot lanuch 使用swagger 建API文

# Table of Contents

- 概 ..... 1
  - 版本信息 ..... 1
  - 系方式 ..... 1
  - URI scheme ..... 1
  - ..... 1
  - 源..... 1
  - Ajax-response-controller ..... 1
  - User-controller ..... 7
- 定 ..... 13
  - BooksInfo ..... 13
  - ReadInfo ..... 14
  - ResponseBase ..... 14
  - TUser..... 14

# 概

## 版本信息

版本 : 1.0.0

## 系方式

名字 : 郭晨

## URI scheme

域名 : localhost:1111

基 路径 : /

- ajax-response-controller : Ajax Response Controller
- user-controller : User Controller

# 源

## Ajax-response-controller

Ajax Response Controller

加 籍信息

POST /book

参数

型	名称	明	型
Body	booksInfo 必填	booksInfo	<a href="#">BooksInfo</a>

HTTP代	明	型
200	OK	<a href="#">ResponseBase</a>
201	Created	无内容
401	Unauthorized	无内容
403	Forbidden	无内容
404	Not Found	无内容

消耗

- `application/json`

生成

- `/`

HTTP 求示例

求 `path`

`/book`

求 `body`

```
{
  "auto" : "string",
  "date" : "string",
  "id" : 0,
  "money" : "string",
  "read" : {
    "age" : "string",
    "name" : "string"
  },
  "remark" : "string"
}
```

HTTP 示例

200

```
{
  "code" : "string",
  "data" : "object",
  "isok" : true,
  "message" : "string"
}
```

## 看 籍信息

GET /book

HTTP代	明	型
200	OK	<a href="#">ResponseBase</a>
401	Unauthorized	无内容
403	Forbidden	无内容
404	Not Found	无内容

生成

• /

HTTP 求示例

求 path

/book

HTTP 示例

200

```
{
  "code" : "string",
  "data" : "object",
  "isok" : true,
  "message" : "string"
}
```

更新 籍信息

PUT /book

参数

型	名称	明	型
Body	<b>booksInfo</b> 必填	booksInfo	<a href="#">BooksInfo</a>

HTTP代	明	型
200	OK	<a href="#">ResponseBase</a>
201	Created	无内容
401	Unauthorized	无内容
403	Forbidden	无内容
404	Not Found	无内容

消耗

- [application/json](#)

生成

- [/](#)

HTTP 求示例

求 path

/book

求 body

```
{
  "auto" : "string",
  "date" : "string",
  "id" : 0,
  "money" : "string",
  "read" : {
    "age" : "string",
    "name" : "string"
  },
  "remark" : "string"
}
```

## HTTP 示例

200

```
{
  "code" : "string",
  "data" : "object",
  "isok" : true,
  "message" : "string"
}
```

## 除 籍信息

DELETE /book

## 参数

型	名称	明	型
Query	id 必填	id	integer (int32)

HTTP代	明	型
200	OK	<a href="#">ResponseBase</a>
204	No Content	无内容
401	Unauthorized	无内容

HTTP代	明	型
403	Forbidden	无内容

生成

• /

HTTP 求示例

求 path

/book

求 query

```
{
  "id" : 0
}
```

HTTP 示例

200

```
{
  "code" : "string",
  "data" : "object",
  "isok" : true,
  "message" : "string"
}
```

根据id 看 籍信息

GET /book/{id}

参数

型	名称	明	型
Path	id 必填	id	integer (int32)



HTTP代	明	型
200	OK	<a href="#">ResponseBase</a>
401	Unauthorized	无内容
403	Forbidden	无内容
404	Not Found	无内容

生成

• /

HTTP 求示例

求 path

/book/0

HTTP 示例

200

```
{
  "code" : "string",
  "data" : "object",
  "isok" : true,
  "message" : "string"
}
```

# User-controller

User Controller

添加用

POST /user

参数

型	名称	明	型
Body	<b>user</b> 必填	user	<a href="#">TUser</a>

HTTP代	明	型
200	OK	<a href="#">ResponseBase</a>
201	Created	无内容
401	Unauthorized	无内容
403	Forbidden	无内容
404	Not Found	无内容

消耗

- [application/json](#)

生成

- [/](#)

HTTP 求示例

求 path

/user

求 body

```
{
  "email" : "string",
  "id" : 0,
  "password" : "string",
  "phone" : "string",
  "sex" : "string",
  "username" : "string"
}
```

HTTP 示例

```
{
  "code" : "string",
  "data" : "object",
  "isok" : true,
  "message" : "string"
}
```

用

```
GET /user
```

HTTP代	明	型
200	OK	<a href="#">ResponseBase</a>
401	Unauthorized	无内容
403	Forbidden	无内容
404	Not Found	无内容

生成

• /

HTTP 求示例

求 path

```
/user
```

HTTP 示例

```
{
  "code" : "string",
  "data" : "object",
  "isok" : true,
  "message" : "string"
}
```

## 更新用

PUT /user

### 参数

型	名称	明	型
Body	<b>user</b> 必填	user	TUser

HTTP代	明	型
200	OK	ResponseBase
201	Created	无内容
401	Unauthorized	无内容
403	Forbidden	无内容
404	Not Found	无内容

### 消耗

- application/json

### 生成

- /

### HTTP 求示例

求 path

/user

求 body

```
{
  "email" : "string",
  "id" : 0,
  "password" : "string",
  "phone" : "string",
  "sex" : "string",
  "username" : "string"
}
```

HTTP 示例

200

```
{
  "code" : "string",
  "data" : "object",
  "isok" : true,
  "message" : "string"
}
```

除用

DELETE /user

参数

型	名称	明	型
Query	id 必填	id	integer (int32)

HTTP代	明	型
200	OK	<a href="#">ResponseBase</a>
204	No Content	无内容
401	Unauthorized	无内容
403	Forbidden	无内容

生成

• /

HTTP 求示例

求 path

```
/user
```

求 query

```
{
  "id" : 0
}
```

HTTP 示例

200

```
{
  "code" : "string",
  "data" : "object",
  "isok" : true,
  "message" : "string"
}
```

根据id 用

```
GET /user/{id}
```

参数

型	名称	明	型
Path	id 必填	id	integer (int32)

  

HTTP代	明	型
200	OK	<a href="#">ResponseBase</a>

HTTP代	明	型
401	Unauthorized	无内容
403	Forbidden	无内容
404	Not Found	无内容

生成

• /

HTTP 求示例

求 path

```
/user/0
```

HTTP 示例

200

```
{
  "code" : "string",
  "data" : "object",
  "isok" : true,
  "message" : "string"
}
```

定

# BooksInfo

名称	明	型
auto 可	例 : "string"	string
date 可	例 : "string"	string (date-time)
id 可	例 : 0	integer (int32)

名称	明	型
<b>money</b> 可	例: "string"	string
<b>read</b> 可	例: "ReadInfo"	ReadInfo
<b>remark</b> 可	例: "string"	string

## ReadInfo

名称	明	型
<b>age</b> 可	例: "string"	string
<b>name</b> 可	例: "string"	string

## ResponseBase

名称	明	型
<b>code</b> 可	例: "string"	string
<b>data</b> 可	例: "object"	object
<b>isok</b> 可	例: true	boolean
<b>message</b> 可	例: "string"	string

## TUser

名称	明	型
<b>email</b> 可	例: "string"	string
<b>id</b> 可	例: 0	integer (int32)
<b>password</b> 可	例: "string"	string



名称	明	型
<b>phone</b> 可	例: "string"	string
<b>sex</b> 可	例: "string"	string
<b>username</b> 可	例: "string"	string