# 变量命名用小驼峰

### 留言板模块

1·获取留言板

request

|  |  |  |
| --- | --- | --- |
| url | /myboard  /messageBoard | For myself  For visitor |
| 方法 | Get |  |
| Content-type | application/json |  |
| 参数 |  |  |

Response

|  |  |
| --- | --- |
| 参数 | 渲染页面返回 |

2·添加留言

request

|  |  |  |
| --- | --- | --- |
| url | /api/messageBoard/addMessage |  |
| 方法 | Post |  |
| Content-type | application/json |  |
| 参数 | body:{  message:String;  } |  |

Response

|  |  |
| --- | --- |
| 参数 | {  code: 200  } |

3·删除留言

request

|  |  |
| --- | --- |
| url | /api/messageBoard/delMessage |
| 方法 | Post |
| Content-type | application/json |
| 参数 | Body:{  Id : integer //数据库中唯一的id  } |
| 备注 | 删除操作不会彻底删除数据，只是设置deleteAt属性 |

Response

|  |  |
| --- | --- |
| 参数 | {  code: 200  } |

4·根据ID获取留言

request

|  |  |
| --- | --- |
| url | /api/messageBoard/getMessageById |
| 方法 | Get |
| Content-type | application/json | |
| 参数 | Body:{  Id : integer //数据库中唯一的id  } |

Response

|  |  |
| --- | --- |
| 参数 | {  code:200,  id:integer,  message:integer,  createdAt:string,  } |

5·获取留言List

request

|  |  |  |
| --- | --- | --- |
| url | /api/messageBoard/getMessageList | |
| 方法 | Get | |
| Content-type | application/json | | |
| 参数 | Body:{  pageIndex:integer //页码  } |  |

Response

|  |  |
| --- | --- |
| 参数 | {  code:200,  id:integer,  message:integer,  createdAt:string,  } |

6·获取留言评论

request

|  |  |
| --- | --- |
| url | /api/messageBoard/getMessageCommentById |
| 方法 | Get |
| Content-type | application/json | |
| 参数 | Body:{  Id : integer //数据库中唯一的id  } |

Response

|  |  |
| --- | --- |
| 参数 | {  code:200,  commentNum:integer //留言个数  commentList[ //留言数组  {comment:”lalalala” , createdAt:”2017-22-12”},  {comment:”lalalala” , createdAt:”2017-22-12”},  ....  ], |

7·获取新闻列表

request

|  |  |
| --- | --- |
| url | /api/news/getNewsList |
| 方法 | Get |
| Content-type | application/json | |
| 参数 | Body:{  Id : integer //数据库中唯一的id  } |

Response

|  |  |
| --- | --- |
| 参数 | {  code:200,  newsList[  { title: “” , createdAt:”” , zhaiyao:”” , img\_url},  {}  ]....  ], |

8·根据类型获取新闻列表

request

|  |  |
| --- | --- |
| url | /api/news/getNewsListByCategory |
| 方法 | Get |
| Content-type | application/json | |
| 参数 | Body:{  Category : String //新闻类型  } |

Response

|  |  |
| --- | --- |
| 参数 | {  code:200,  reasen:” 成功返回”，  data[  { title: “” , createdAt:”” , zhaiyao:”” , img\_url},  {}  ]....  ], |

9·根据类型，页数，加载更多等获取新闻列表

request

|  |  |
| --- | --- |
| url | /api/news/getNewsListByType |
| 方法 | Get |
| Content-type | application/json | |
| 参数 | Body:{  type : String //新闻类型  page : integer //每个页最多15条新闻  more : boolean // 如果more是true就获取更多，这里吧所有新闻列表都返回  } |

Response

|  |  |
| --- | --- |
| 参数 | {  code:200,  num:integer //新闻总数  data[  { title , ctime , url , picUrl , type //都是string类型},  { .....}, ....  ]  ], |

10.添加博客

request

|  |  |
| --- | --- |
| url | /api/news/addBlog |
| 方法 | Get |
| Content-type | application/json | |
| 参数 | Body:{  Title : string  url : string  author : string  } |

Response

|  |  |
| --- | --- |
| 参数 | {  code:200,  } |

11.获取博客列表

request

|  |  |
| --- | --- |
| url | /api/news/getBlogList |
| 方法 | Get |
| Content-type | application/json | |
| 参数 | Body:{  } |

Response

|  |  |
| --- | --- |
| 参数 | {  code:200,  num:integr //博客的数量  data:[  { title , url , author , createdAt , id}  ]  } |

12.删除博客

request

|  |  |
| --- | --- |
| url | /api/news/delBlog |
| 方法 | Get |
| Content-type | application/json | |
| 参数 | Body:{  id  } |

Response

|  |  |
| --- | --- |
| 参数 | {  code:200,  } |