

iblog API文档

说明

每个请求失败的时候会返回错误信息，在响应体中的err_message属性展示。

首页

/ 请求首页文章列表（8个）(GET)

```
DataHomepage = [
  {
    id: '0',
    title: '如何使用JavaScript如何使用JavaScript',
    date: '2019-4-19',
    abstract: '今年早些时候，奇舞团开源的 Node.js 框架 — ThinkJS ' +
      '迎来了她的 3.0 版本。 尽管今年我很少更新博客，但「每次 ThinkJS ' +
      '发布大版本，我都要更新博客程序」 的老传统还是不能丢。所以， ' +
      '你现在看到的这个博客，已经是基于 ThinkJS 3 全面重构后的新版。'
  },
  {
    id: '1',
    title: '如何使用JavaScript如何使用JavaScript',
    date: '2019-4-19',
    abstract: '今年早些时候，奇舞团开源的 Node.js 框架 — ThinkJS ' +
      '迎来了她的 3.0 版本。 尽管今年我很少更新博客，但「每次 ThinkJS ' +
      '发布大版本，我都要更新博客程序」 的老传统还是不能丢。所以， ' +
      '你现在看到的这个博客，已经是基于 ThinkJS 3 全面重构后的新版。'
  },
  {
    id: '2',
    title: '如何使用JavaScript如何使用JavaScript',
    date: '2019-4-19',
    abstract: '今年早些时候，奇舞团开源的 Node.js 框架 — ThinkJS ' +
      '迎来了她的 3.0 版本。 尽管今年我很少更新博客，但「每次 ThinkJS ' +
      '发布大版本，我都要更新博客程序」 的老传统还是不能丢。所以， ' +
      '你现在看到的这个博客，已经是基于 ThinkJS 3 全面重构后的新版。'
  },
  {
    id: '3',
    title: '如何使用JavaScript如何使用JavaScript',
    date: '2019-4-19',
    abstract: '今年早些时候，奇舞团开源的 Node.js 框架 — ThinkJS ' +
      '迎来了她的 3.0 版本。 尽管今年我很少更新博客，但「每次 ThinkJS ' +
      '发布大版本，我都要更新博客程序」 的老传统还是不能丢。所以， ' +
      '你现在看到的这个博客，已经是基于 ThinkJS 3 全面重构后的新版。'
  },
  {
    id: '4',
    title: '如何使用JavaScript如何使用JavaScript',
    date: '2019-4-19',
    abstract: '今年早些时候，奇舞团开源的 Node.js 框架 — ThinkJS ' +
      '迎来了她的 3.0 版本。 尽管今年我很少更新博客，但「每次 ThinkJS ' +
      '发布大版本，我都要更新博客程序」 的老传统还是不能丢。所以， ' +
      '你现在看到的这个博客，已经是基于 ThinkJS 3 全面重构后的新版。'
  },
  {
    id: '5',
    title: '如何使用JavaScript如何使用JavaScript',
    date: '2019-4-19',
    abstract: '今年早些时候，奇舞团开源的 Node.js 框架 — ThinkJS ' +
      '迎来了她的 3.0 版本。 尽管今年我很少更新博客，但「每次 ThinkJS ' +
      '发布大版本，我都要更新博客程序」 的老传统还是不能丢。所以， ' +
      '你现在看到的这个博客，已经是基于 ThinkJS 3 全面重构后的新版。'
  },
  {
    id: '6',
    title: '如何使用JavaScript如何使用JavaScript',
    date: '2019-4-19',
    abstract: '今年早些时候，奇舞团开源的 Node.js 框架 — ThinkJS ' +
      '迎来了她的 3.0 版本。 尽管今年我很少更新博客，但「每次 ThinkJS ' +
      '发布大版本，我都要更新博客程序」 的老传统还是不能丢。所以， ' +
      '你现在看到的这个博客，已经是基于 ThinkJS 3 全面重构后的新版。'
  },
  {
    id: '7',
    title: '如何使用JavaScript如何使用JavaScript',
    date: '2019-4-19',
    abstract: '今年早些时候，奇舞团开源的 Node.js 框架 — ThinkJS ' +
      '迎来了她的 3.0 版本。 尽管今年我很少更新博客，但「每次 ThinkJS ' +
      '发布大版本，我都要更新博客程序」 的老传统还是不能丢。所以， ' +
      '你现在看到的这个博客，已经是基于 ThinkJS 3 全面重构后的新版。'
  },
]
```

```
{
  id: '3',
  title: '如何使用JavaScript如何使用JavaScript',
  date: '2019-4-19',
  abstract: '今年早些时候，奇舞团开源的 Node.js 框架 — ThinkJS ' +
    '迎来了她的 3.0 版本。 尽管今年我很少更新博客，但「每次 ThinkJS ' +
    '发布大版本，我都要更新博客程序」 的老传统还是不能丢。所以， ' +
    '你现在看到的这个博客，已经是基于 ThinkJS 3 全面重构后的新版。'
},
{
  id: '4',
  title: '如何使用JavaScript如何使用JavaScript',
  date: '2019-4-19',
  abstract: '今年早些时候，奇舞团开源的 Node.js 框架 — ThinkJS ' +
    '迎来了她的 3.0 版本。 尽管今年我很少更新博客，但「每次 ThinkJS ' +
    '发布大版本，我都要更新博客程序」 的老传统还是不能丢。所以， ' +
    '你现在看到的这个博客，已经是基于 ThinkJS 3 全面重构后的新版。'
},
{
  id: '5',
  title: '如何使用JavaScript如何使用JavaScript',
  date: '2019-4-19',
  abstract: '今年早些时候，奇舞团开源的 Node.js 框架 — ThinkJS ' +
    '迎来了她的 3.0 版本。 尽管今年我很少更新博客，但「每次 ThinkJS ' +
    '发布大版本，我都要更新博客程序」 的老传统还是不能丢。所以， ' +
    '你现在看到的这个博客，已经是基于 ThinkJS 3 全面重构后的新版。'
},
{
  id: '6',
  title: '如何使用JavaScript如何使用JavaScript',
  date: '2019-4-19',
  abstract: '今年早些时候，奇舞团开源的 Node.js 框架 — ThinkJS ' +
    '迎来了她的 3.0 版本。 尽管今年我很少更新博客，但「每次 ThinkJS ' +
    '发布大版本，我都要更新博客程序」 的老传统还是不能丢。所以， ' +
    '你现在看到的这个博客，已经是基于 ThinkJS 3 全面重构后的新版。'
},
{
  id: '7',
  title: '如何使用JavaScript如何使用JavaScript',
  date: '2019-4-19',
  abstract: '今年早些时候，奇舞团开源的 Node.js 框架 — ThinkJS ' +
    '迎来了她的 3.0 版本。 尽管今年我很少更新博客，但「每次 ThinkJS ' +
```

```
    '发布大版本，我都要更新博客程序」 的老传统还是不能丢。所以， ' +  
    '你现在看到的这个博客，已经是基于 ThinkJS 3 全面重构后的新版。 '  
  },  
]
```

`/page/:num` 请求对应页数的文章列表，`num` 为指定页数(`num => 2`)，返回对应页数的8个文章标题，返回数据格式同上。

归档页面

`/archive` (GET) 返回数据格式

```
DataArchive = [  
  {  
    year: '2019',  
    titleList: [  
      {  
        id: '001',  
        title: '这是一篇文章',  
        date: '2019-04-19'  
      },  
      {  
        id: '002',  
        title: '这是一篇文章',  
        date: '2019-04-19'  
      },  
      {  
        id: '003',  
        title: '这是一篇文章',  
        date: '2019-04-19'  
      },  
    ],  
  },  
  {  
    year: '2018',  
    titleList: [  
      {  
        id: '004',  
        title: '这是一篇文章',  
        date: '2019-04-19'  
      },  
    ],  
  },  
]
```

```

        id: '005',
        title: '这是一篇文章',
        date: '2019-04-19'
    },
    {
        id: '006',
        title: '这是一篇文章',
        date: '2019-04-19'
    },
]
},
{
    year: '2017',
    titleList: [
        {
            id: '007',
            title: '这是一篇文章',
            date: '2019-04-19'
        },
        {
            id: '008',
            title: '这是一篇文章',
            date: '2019-04-19'
        },
        {
            id: '009',
            title: '这是一篇文章',
            date: '2019-04-19'
        },
    ]
},
]

```

专题

`/category` (GET) 返回数据格式

```
categoryData = [
```

```
{
  categoryTitle: 'HTTP相关',
  categoryResume: '博客写了这么多年，数量一直没上去。' +
    '大部分时候遇上有意思的东西， 研究明白之后只' +
    '是多了几篇收藏，或者是 Evernote 里多了几段' +
    '零散的记录',
  titleList: [
    {
      id: '1',
      title: '这是一篇文章这是一篇文章',
      date: '2019-04-12'
    },
    {
      id: '2',
      title: '这是一篇文章这是一篇文章',
      date: '2019-04-12'
    },
    {
      id: '3',
      title: '这是一篇文章这是一篇文章',
      date: '2019-04-12'
    },
    {
      id: '4',
      title: '这是一篇文章这是一篇文章',
      date: '2019-04-12'
    },
  ],
}
{
  categoryTitle: 'JAVA',
  categoryResume: '博客写了这么多年，数量一直没上去。' +
    '大部分时候遇上有意思的东西， 研究明白之后只' +
    '是多了几篇收藏，或者是 Evernote 里多了几段' +
    '零散的记录',
  titleList: [
    {
      id: '11',
      title: '这是一篇文章这是一篇文章',
      date: '2019-04-12'
    },
    {
```

```
    id: '12',
    title: '这是一篇文章这是一篇文章',
    date: '2019-04-12'
  },
  {
    id: '13',
    title: '这是一篇文章这是一篇文章',
    date: '2019-04-12'
  },
  {
    id: '14',
    title: '这是一篇文章这是一篇文章',
    date: '2019-04-12'
  },
]
},
{
  categoryTitle: '数据结构',
  categoryResume: '博客写了这么多年，数量一直没上去。' +
    '大部分时候遇上有意思的东西， 研究明白之后只' +
    '是多了几篇收藏，或者是 Evernote 里多了几段' +
    '零散的记录',
  titleList: [
    {
      id: '21',
      title: '这是一篇文章这是一篇文章',
      date: '2019-04-12'
    },
    {
      id: '22',
      title: '这是一篇文章这是一篇文章',
      date: '2019-04-12'
    },
    {
      id: '23',
      title: '这是一篇文章这是一篇文章',
      date: '2019-04-12'
    },
    {
      id: '24',
      title: '这是一篇文章这是一篇文章',
      date: '2019-04-12'
    }
  ]
}
```

```
    },
  ]
},
{
  categoryTitle: '前端框架',
  categoryResume: '博客写了这么多年，数量一直没上去。' +
    '大部分时候遇上有意思的东西， 研究明白之后只' +
    '是多了几篇收藏，或者是 Evernote 里多了几段' +
    '零散的记录',
  titleList: [
    {
      id: '31',
      title: '这是一篇文章这是一篇文章',
      date: '2019-04-12'
    },
    {
      id: '32',
      title: '这是一篇文章这是一篇文章',
      date: '2019-04-12'
    },
    {
      id: '33',
      title: '这是一篇文章这是一篇文章',
      date: '2019-04-12'
    },
    {
      id: '34',
      title: '这是一篇文章这是一篇文章',
      date: '2019-04-12'
    },
  ]
},
]
```

关于本站

`/about` (GET) 返回数据格式

```
aboutData = [
  {
    id: '100',
    title: '个人介绍',
    content: '博客写了这么多年，数量一直没上去。' +
      '大部分时候遇上有意思的东西， 研究明白之后只是' +
      '多了几篇收藏，或者是 Evernote 里多了几段零散' +
      '的记录， '
  },
  {
    id: '101',
    title: '版权声明',
    content: '博客写了这么多年，数量一直没上去。' +
      '大部分时候遇上有意思的东西， 研究明白之后只是' +
      '多了几篇收藏，或者是 Evernote 里多了几段零散' +
      '的记录， '
  },
  {
    id: '102',
    title: '支持本站',
    content: '博客写了这么多年，数量一直没上去。' +
      '大部分时候遇上有意思的东西， 研究明白之后只是' +
      '多了几篇收藏，或者是 Evernote 里多了几段零散' +
      '的记录， '
  },
  {
    id: '103',
    title: '关于本站',
    content: '博客写了这么多年，数量一直没上去。' +
      '大部分时候遇上有意思的东西， 研究明白之后只是' +
      '多了几篇收藏，或者是 Evernote 里多了几段零散' +
      '的记录， '
  },
]
```

搜索界面

`/search` (GET) 返回搜索热词，返回字符串数组

`/search/: content` (GET) content为搜索内容，返回数据格式(标题搜索)


```
SeaechData = [
  {
    id: '0',
    title: '如何使用JavaScript如何使用JavaScript',
    date: '2019-4-19',
    abstract: '今年早些时候，奇舞团开源的 Node.js 框架 — ThinkJS ' +
      '迎来了她的 3.0 版本。 尽管今年我很少更新博客，但「每次 ThinkJS ' +
      '发布大版本，我都要更新博客程序」 的老传统还是不能丢。所以， ' +
      '你现在看到的这个博客，已经是基于 ThinkJS 3 全面重构后的新版。'
  },
  {
    id: '1',
    title: '如何使用JavaScript如何使用JavaScript',
    date: '2019-4-19',
    abstract: '今年早些时候，奇舞团开源的 Node.js 框架 — ThinkJS ' +
      '迎来了她的 3.0 版本。 尽管今年我很少更新博客，但「每次 ThinkJS ' +
      '发布大版本，我都要更新博客程序」 的老传统还是不能丢。所以， ' +
      '你现在看到的这个博客，已经是基于 ThinkJS 3 全面重构后的新版。'
  },
  {
    id: '2',
    title: '如何使用JavaScript如何使用JavaScript',
    date: '2019-4-19',
    abstract: '今年早些时候，奇舞团开源的 Node.js 框架 — ThinkJS ' +
      '迎来了她的 3.0 版本。 尽管今年我很少更新博客，但「每次 ThinkJS ' +
      '发布大版本，我都要更新博客程序」 的老传统还是不能丢。所以， ' +
      '你现在看到的这个博客，已经是基于 ThinkJS 3 全面重构后的新版。'
  },
  {
    id: '3',
    title: '如何使用JavaScript如何使用JavaScript',
    date: '2019-4-19',
    abstract: '今年早些时候，奇舞团开源的 Node.js 框架 — ThinkJS ' +
      '迎来了她的 3.0 版本。 尽管今年我很少更新博客，但「每次 ThinkJS ' +
      '发布大版本，我都要更新博客程序」 的老传统还是不能丢。所以， ' +
      '你现在看到的这个博客，已经是基于 ThinkJS 3 全面重构后的新版。'
  },
  {
    id: '4',
    title: '如何使用JavaScript如何使用JavaScript',
    date: '2019-4-19',
    abstract: '今年早些时候，奇舞团开源的 Node.js 框架 — ThinkJS ' +
```

```
    '迎来了她的 3.0 版本。 尽管今年我很少更新博客，但「每次 ThinkJS ' +  
    '发布大版本，我都要更新博客程序」 的老传统还是不能丢。所以， ' +  
    '你现在看到的这个博客，已经是基于 ThinkJS 3 全面重构后的新版。'  
  },  
  {  
    id: '5',  
    title: '如何使用JavaScript如何使用JavaScript',  
    date: '2019-4-19',  
    abstract: '今年早些时候，奇舞团开源的 Node.js 框架 — ThinkJS ' +  
      '迎来了她的 3.0 版本。 尽管今年我很少更新博客，但「每次 ThinkJS ' +  
      '发布大版本，我都要更新博客程序」 的老传统还是不能丢。所以， ' +  
      '你现在看到的这个博客，已经是基于 ThinkJS 3 全面重构后的新版。'  
  },  
  {  
    id: '6',  
    title: '如何使用JavaScript如何使用JavaScript',  
    date: '2019-4-19',  
    abstract: '今年早些时候，奇舞团开源的 Node.js 框架 — ThinkJS ' +  
      '迎来了她的 3.0 版本。 尽管今年我很少更新博客，但「每次 ThinkJS ' +  
      '发布大版本，我都要更新博客程序」 的老传统还是不能丢。所以， ' +  
      '你现在看到的这个博客，已经是基于 ThinkJS 3 全面重构后的新版。'  
  },  
  {  
    id: '7',  
    title: '如何使用JavaScript如何使用JavaScript',  
    date: '2019-4-19',  
    abstract: '今年早些时候，奇舞团开源的 Node.js 框架 — ThinkJS ' +  
      '迎来了她的 3.0 版本。 尽管今年我很少更新博客，但「每次 ThinkJS ' +  
      '发布大版本，我都要更新博客程序」 的老传统还是不能丢。所以， ' +  
      '你现在看到的这个博客，已经是基于 ThinkJS 3 全面重构后的新版。'  
  },  
]  
]
```

文章详情

/post/:id (GET) 响应数据格式

```
postData = {
  id: '', // 文章id
  postTitle: '', // 文章标题
  category: '', // 文章分类
  abstract: '', // 文章简介
  postContent: '' // 文章内容
}
```

登录

`/login` (POST)

- 请求数据格式

```
loginData = {
  username: '',
  password: '',
  token: ''
}
```

- 响应数据格式

```
// 成功
{
  status: 200,
  message: success
}
// 失败
{
  err_message: ''
}
```

写作

`admin/writing` (POST)

- 请求数据格式

```
writingData = {  
  postTitle: '', // 文章标题  
  category: '', // 文章专题  
  abstract: '', // 文章简介  
  postContent: '', // 文章内容  
  token: ''  
}
```

- 响应的数据格式

```
// 成功  
{  
  status: 200,  
  message: 'success'  
}  
  
// 失败  
{  
  err_message: '' // 失败原因  
}
```

编辑

`admin/edit` (GET) 请求数据格式

```
editData = {  
  category: [  
    {  
      id: '',  
      title: '', // 标题名  
      resume: '' // 标题简介  
    },  
    {  
      id: '',  
      title: '',  
      resume: ''  
    }  
  ]  
}
```

```
    },
    {
      id: '',
      title: '',
      resume: ''
    },
  ],
  postList: [
    {
      id: '',
      title: '',
      date: ''
    },
    {
      id: '',
      title: '',
      date: ''
    },
    {
      id: '',
      title: '',
      date: ''
    },
  ],
]
}
```

`admin/edit/category` (POST) 功能：添加专题标签

- 请求数据格式如下

```
category = {
  categoryList:[
    {
      title: '',
      resume: ''
    },
    {
      title: '',
      resume: ''
    },
    {

```

```
        title: '',
        resume: ''
    }
],
token: ''
}
```

- 响应数据格式

```
// 成功
{
    status: 200
    message: 'success'
}
// 失败
{
    err_message: ''
}
```

`admin/edit/post_list` (DELETE) 删除指定的文章

- 请求数据格式

```
deleteItem = {
    id: '',
    token: ''
}
```

- 响应数据格式

```
// 成功
{
    status: 200
    message: 'success'
}
// 失败
{
    err_message: ''
}
```

/admin/edit_item/:id

- GET 获取当前id对应文章的相关信息

```
postData = {
  id: '', // 文章id
  postTitle: '', // 文章标题
  category: '', // 文章分类
  abstract: '', // 文章简介
  postContent: '' // 文章内容
}
```

- POST 编辑后的文章信息

```
// 请求
postData = {
  id: '', // 文章id
  postTitle: '', // 文章标题
  category: '', // 文章分类
  abstract: '', // 文章简介
  postContent: '', // 文章内容
  token: '' //token
}

// 响应
// 成功
{
  status: 200
  message: 'success'
}
// 失败
{
  err_message: ''
}
```

图片

/admin/img_upload 上传图片，服务端返回图片url，（POST）

```
// 请求
imgData = {
  img: xxx.png/xxx.jpg/...
  token: ''
}

// 响应
// 成功
{
  status: 200,
  message: 'success',
  url: ''
}
// 失败
{
  err_message: ''
}
```