# Jeebbs RESTFul API 文档

SERVER IP: 222.201.145.145  
BASE URL: /jeebbs  
API VERSION: 1.0.0  
@Author [ztwnag](mailto:531946934@qq.com)  
Exampl Site: [here](http://222.201.145.145:8091/jeebbs/swagger-ui.html#/)

## news-controller： 财经新闻模块接口详情

### 获取新闻列表

#### 简要描述：

* 使用分页查找方式获取新闻列表

#### 请求URL：

* /api/news/findAll

#### 请求方式：

* GET

#### 请求参数

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | Required | 类型 | 说明 | 示例 |
| pageNum | 否 | integer | 当前页码，从1开始；默认值为1 | 1 |
| pageSize | 否 | integer | 每页显示的记录数；默认值为10 | 10 |
| orderBy | 否 | string | 记录排序方式，默认值按时间降序(stmp desc) | stmp desc |

#### 返回示例

* 200 - successful
* {  
   "endRow": 0,  
   "firstPage": 0,  
   "hasNextPage": true,  
   "hasPreviousPage": true,  
   "isFirstPage": true,  
   "isLastPage": true,  
   "lastPage": 0,  
   # 新闻列表  
   "list": [  
   {}  
   ],  
   "navigatePages": 0,  
   "navigatepageNums": [  
  0  
   ],  
   "nextPage": 0,  
   "orderBy": "string",  
   "pageNum": 0,  
   "pageSize": 0,  
   "pages": 0,  
   "prePage": 0,  
   "size": 0,  
   "startRow": 0,  
   "total": 0  
  }
* 204 - empty content
* {}
* 400~500 - some error happended
* {  
   "status": "int",  
   "error": "string",  
   "path": "string",  
   "exception": "string",  
   "message": "String",  
   "errors": "String",  
   "trace": "String",  
   "timestamp": "yyyy-MM-dd HH:mm:ss"  
  }

### 根据日期获取新闻列表

#### 简要描述：

* 返回从某个日期开始（包含），到某个日期结束（包含）的新闻列表

#### 请求URL：

* /api/news/findByDate

#### 请求方式：

* GET

#### 请求参数

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | Required | 类型 | 说明 | 示例 |
| from | 否 | string | 起始日期 | 2017-1-1 |
| to | 否 | string | 结束日期 | 2017-12-31 |
| pageNum | 否 | integer | 当前页码，从1开始；默认值为1 | 1 |
| pageSize | 否 | integer | 每页显示的记录数；默认值为10 | 10 |
| orderBy | 否 | string | 记录排序方式，默认值按时间降序(stmp desc) | stmp desc |

#### 返回示例

* 200 - successful
* {  
   "endRow": 0,  
   "firstPage": 0,  
   "hasNextPage": true,  
   "hasPreviousPage": true,  
   "isFirstPage": true,  
   "isLastPage": true,  
   "lastPage": 0,  
   # 新闻列表  
   "list": [  
   {}  
   ],  
   "navigatePages": 0,  
   "navigatepageNums": [  
  0  
   ],  
   "nextPage": 0,  
   "orderBy": "string",  
   "pageNum": 0,  
   "pageSize": 0,  
   "pages": 0,  
   "prePage": 0,  
   "size": 0,  
   "startRow": 0,  
   "total": 0  
  }
* 204 - empty content
* {}
* 400~500 - some error happended
* {  
   "status": "int",  
   "error": "string",  
   "path": "string",  
   "exception": "string",  
   "message": "String",  
   "errors": "String",  
   "trace": "String",  
   "timestamp": "yyyy-MM-dd HH:mm:ss"  
  }

### 根据新闻链接查找新闻

#### 简要描述：

* 根据新闻链接查找新闻，采用精确匹配方式

#### 请求URL：

* /api/news/findByHref

#### 请求方式：

* GET

#### 请求参数

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | Required | 类型 | 说明 | 示例 |
| href | 否 | string | 新闻链接，默认返回最新的新闻 | http://example.com |

#### 返回示例

* 200 - successful
* {  
   "id": "long",  
   "source": "string",  
   "title": "string",  
   "href": "string",  
   "profile": "string",  
   "stmp": "yyyy-MM-dd HH:mm:ss"  
  }
* 204 - empty content
* {}
* 400~500 - some error happended
* {  
   "status": "int",  
   "error": "string",  
   "path": "string",  
   "exception": "string",  
   "message": "String",  
   "errors": "String",  
   "trace": "String",  
   "timestamp": "yyyy-MM-dd HH:mm:ss"  
  }

### 根据新闻标识符(id)查找新闻

#### 简要描述：

* 根据新闻标识符(id)查找新闻，采用精确匹配方式

#### 请求URL：

* /api/news/findById

#### 请求方式：

* GET

#### 请求参数

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | Required | 类型 | 说明 | 示例 |
| id | 否 | long | id应该满足1≤id; id如果为空或id=-1, 默认返回最新采集的财经新闻; id<1(id!=-1)或id不存在, 返回为空(204) | 1 |

#### 返回示例

* 200 - successful
* {  
   "id": "long",  
   "source": "string",  
   "title": "string",  
   "href": "string",  
   "profile": "string",  
   "stmp": "yyyy-MM-dd HH:mm:ss"  
  }
* 204 - empty content
* {}
* 400~500 - some error happended
* {  
   "status": "int",  
   "error": "string",  
   "path": "string",  
   "exception": "string",  
   "message": "String",  
   "errors": "String",  
   "trace": "String",  
   "timestamp": "yyyy-MM-dd HH:mm:ss"  
  }

### 根据新闻标题查找新闻

#### 简要描述：

* 根据新标题接查找新闻，采用精确匹配方式

#### 请求URL：

* /api/news/findByTitle

#### 请求方式：

* GET

#### 请求参数

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | Required | 类型 | 说明 | 示例 |
| title | 否 | string | 新闻标题，默认返回最新的新闻 | example title |

#### 返回示例

* 200 - successful
* {  
   "id": "long",  
   "source": "string",  
   "title": "string",  
   "href": "string",  
   "profile": "string",  
   "stmp": "yyyy-MM-dd HH:mm:ss"  
  }
* 204 - empty content
* {}
* 400~500 - some error happended
* {  
   "status": "int",  
   "error": "string",  
   "path": "string",  
   "exception": "string",  
   "message": "String",  
   "errors": "String",  
   "trace": "String",  
   "timestamp": "yyyy-MM-dd HH:mm:ss"  
  }

### 获取当天的新闻列表

#### 简要描述：

* 使用分页查找方式获取当天的新闻列表 > 有副作用，如果在一天最早一次数据库更新之前使用，会返回空（还未有当天新闻产生） 建议: 只作为查询使用，而非展示页面

#### 请求URL：

* /api/news/findToday

#### 请求方式：

* GET

#### 请求参数

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | Required | 类型 | 说明 | 示例 |
| pageNum | 否 | integer | 当前页码，从1开始；默认值为1 | 1 |
| pageSize | 否 | integer | 每页显示的记录数；默认值为10 | 10 |
| orderBy | 否 | string | 记录排序方式，默认值按时间降序(stmp desc) | stmp desc |

#### 返回示例

* 200 - successful
* {  
   "endRow": 0,  
   "firstPage": 0,  
   "hasNextPage": true,  
   "hasPreviousPage": true,  
   "isFirstPage": true,  
   "isLastPage": true,  
   "lastPage": 0,  
   # 新闻列表  
   "list": [  
   {}  
   ],  
   "navigatePages": 0,  
   "navigatepageNums": [  
  0  
   ],  
   "nextPage": 0,  
   "orderBy": "string",  
   "pageNum": 0,  
   "pageSize": 0,  
   "pages": 0,  
   "prePage": 0,  
   "size": 0,  
   "startRow": 0,  
   "total": 0  
  }
* 204 - empty content
* {}
* 400~500 - some error happended
* {  
   "status": "int",  
   "error": "string",  
   "path": "string",  
   "exception": "string",  
   "message": "String",  
   "errors": "String",  
   "trace": "String",  
   "timestamp": "yyyy-MM-dd HH:mm:ss"  
  }

### 根据多参数查询新闻

#### 简要描述：

* 使用 id, source, title, href, profile, stmp 参数查询新闻，参数个数可选，采用模糊匹配方式，如果不存在参数，则返回全部新闻

#### 请求URL：

* /api/news/search

#### 请求方式：

* GET

#### 请求参数

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | Required | 类型 | 说明 | 示例 |
| id | 否 | long | 新闻标识符 | 1 |
| source | 否 | string | 新闻来源 | 搜狐新闻 |
| title | 否 | string | 新闻标题 | example title |
| href | 否 | string | 新闻链接 | http://example.com |
| profile | 否 | string | 新闻摘要 | ...profile something more... |
| stmp | 否 | string | 新闻发布时间 | 2017-1-1 |
| pageNum | 否 | integer | 当前页码，从1开始；默认值为1 | 1 |
| pageSize | 否 | integer | 每页显示的记录数；默认值为10 | 10 |
| orderBy | 否 | string | 记录排序方式，默认值按时间降序(stmp desc) | stmp desc |

#### 返回示例

* 200 - successful
* {  
   "endRow": 0,  
   "firstPage": 0,  
   "hasNextPage": true,  
   "hasPreviousPage": true,  
   "isFirstPage": true,  
   "isLastPage": true,  
   "lastPage": 0,  
   # 新闻列表  
   "list": [  
   {}  
   ],  
   "navigatePages": 0,  
   "navigatepageNums": [  
  0  
   ],  
   "nextPage": 0,  
   "orderBy": "string",  
   "pageNum": 0,  
   "pageSize": 0,  
   "pages": 0,  
   "prePage": 0,  
   "size": 0,  
   "startRow": 0,  
   "total": 0  
  }
* 204 - empty content
* {}
* 400~500 - some error happended
* {  
   "status": "int",  
   "error": "string",  
   "path": "string",  
   "exception": "string",  
   "message": "String",  
   "errors": "String",  
   "trace": "String",  
   "timestamp": "yyyy-MM-dd HH:mm:ss"  
  }

## stock-controller: 股票查询模块接口详情

### 获取股票列表

#### 简要描述：

* 使用分页查找方式获取某日的深沪股市的股票数据，若是当天的数据则为实时数据，否则返回历史数据

#### 请求URL：

* /api/stock/findAll

#### 请求方式：

* GET

#### 请求参数

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | Required | 类型 | 说明 | 示例 |
| date | 否 | string | 日期；默认值为离闭市的最近一天(比如在周六，返回周五闭市的数据) | 2017-1-1 |
| pageNum | 否 | integer | 当前页码，从1开始；默认值为1 | 1 |
| pageSize | 否 | integer | 每页显示的记录数；默认值为10 | 10 |
| orderBy | 否 | string | 记录排序方式，默认值按时间降序(stmp desc) | stmp desc |

#### 返回示例

* 200 - successful
* {  
   "endRow": 0,  
   "firstPage": 0,  
   "hasNextPage": true,  
   "hasPreviousPage": true,  
   "isFirstPage": true,  
   "isLastPage": true,  
   "lastPage": 0,  
   # 股票数据列表  
   "list": [  
   {}  
   ],  
   "navigatePages": 0,  
   "navigatepageNums": [  
  0  
   ],  
   "nextPage": 0,  
   "orderBy": "string",  
   "pageNum": 0,  
   "pageSize": 0,  
   "pages": 0,  
   "prePage": 0,  
   "size": 0,  
   "startRow": 0,  
   "total": 0  
  }
* 204 - empty content
* {}
* 400~500 - some error happended
* {  
   "status": "int",  
   "error": "string",  
   "path": "string",  
   "exception": "string",  
   "message": "String",  
   "errors": "String",  
   "trace": "String",  
   "timestamp": "yyyy-MM-dd HH:mm:ss"  
  }

### 根据股票代码查找股票数据

#### 简要描述：

* 依据股票代码查找股票数据，可以指定日期，来返回历史数据

#### 请求URL：

* /api/stock/findByCode

#### 请求方式：

* GET

#### 请求参数

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | Required | 类型 | 说明 | 示例 |
| code | 是 | string | 股票代码 | sh000001 |
| from | 否 | string | 起始日期 | 2017-1-1 |
| to | 否 | string | 结束日期 | 2017-12-31 |
| pageNum | 否 | integer | 当前页码，从1开始；默认值为1 | 1 |
| pageSize | 否 | integer | 每页显示的记录数；默认值为10 | 10 |
| orderBy | 否 | string | 记录排序方式，默认值按时间降序(stmp desc) | stmp desc |

#### 返回示例

* 200 - successful
* {  
   "endRow": 0,  
   "firstPage": 0,  
   "hasNextPage": true,  
   "hasPreviousPage": true,  
   "isFirstPage": true,  
   "isLastPage": true,  
   "lastPage": 0,  
   # 股票数据列表  
   "list": [  
   {}  
   ],  
   "navigatePages": 0,  
   "navigatepageNums": [  
  0  
   ],  
   "nextPage": 0,  
   "orderBy": "string",  
   "pageNum": 0,  
   "pageSize": 0,  
   "pages": 0,  
   "prePage": 0,  
   "size": 0,  
   "startRow": 0,  
   "total": 0  
  }
* 204 - empty content
* {}
* 400~500 - some error happended
* {  
   "status": "int",  
   "error": "string",  
   "path": "string",  
   "exception": "string",  
   "message": "String",  
   "errors": "String",  
   "trace": "String",  
   "timestamp": "yyyy-MM-dd HH:mm:ss"  
  }

### 根据股票名称查找股票数据

#### 简要描述：

* 依据股票名称查找股票数据（模糊匹配），可以指定日期，来返回历史数据

#### 请求URL：

* /api/stock/findByName

#### 请求方式：

* GET

#### 请求参数

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | Required | 类型 | 说明 | 示例 |
| name | 是 | string | 股票名称 | 万科A |
| from | 否 | string | 起始日期 | 2017-1-1 |
| to | 否 | string | 结束日期 | 2017-12-31 |
| pageNum | 否 | integer | 当前页码，从1开始；默认值为1 | 1 |
| pageSize | 否 | integer | 每页显示的记录数；默认值为10 | 10 |
| orderBy | 否 | string | 记录排序方式，默认值按时间降序(stmp desc) | stmp desc |

#### 返回示例

* 200 - successful
* {  
   "endRow": 0,  
   "firstPage": 0,  
   "hasNextPage": true,  
   "hasPreviousPage": true,  
   "isFirstPage": true,  
   "isLastPage": true,  
   "lastPage": 0,  
   # 股票数据列表  
   "list": [  
   {}  
   ],  
   "navigatePages": 0,  
   "navigatepageNums": [  
  0  
   ],  
   "nextPage": 0,  
   "orderBy": "string",  
   "pageNum": 0,  
   "pageSize": 0,  
   "pages": 0,  
   "prePage": 0,  
   "size": 0,  
   "startRow": 0,  
   "total": 0  
  }
* 204 - empty content
* {}
* 400~500 - some error happended
* {  
   "status": "int",  
   "error": "string",  
   "path": "string",  
   "exception": "string",  
   "message": "String",  
   "errors": "String",  
   "trace": "String",  
   "timestamp": "yyyy-MM-dd HH:mm:ss"  
  }

### 预测走势上扬的股票

#### 简要描述：

* 预测当日走势上扬的股票（包括直线上升趋势和梯形上升趋势）

#### 请求URL：

* /api/stock/predict

#### 请求方式：

* GET

#### 请求参数

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | Required | 类型 | 说明 | 示例 |
| trend | 是 | string | 上升趋势，**只包含两个值:** LT(直线趋势)或TT(梯形趋势) | LT |
| pageNum | 否 | integer | 当前页码，从1开始；默认值为1 | 1 |
| pageSize | 否 | integer | 每页显示的记录数；默认值为10 | 10 |
| orderBy | 否 | string | 记录排序方式，默认值按时间降序(stmp desc) | stmp desc |

#### 返回示例

* 200 - successful
* {  
   "endRow": 0,  
   "firstPage": 0,  
   "hasNextPage": true,  
   "hasPreviousPage": true,  
   "isFirstPage": true,  
   "isLastPage": true,  
   "lastPage": 0,  
   # 股票数据列表  
   "list": [  
   {}  
   ],  
   "navigatePages": 0,  
   "navigatepageNums": [  
  0  
   ],  
   "nextPage": 0,  
   "orderBy": "string",  
   "pageNum": 0,  
   "pageSize": 0,  
   "pages": 0,  
   "prePage": 0,  
   "size": 0,  
   "startRow": 0,  
   "total": 0  
  }
* 204 - empty content
* {}
* 400~500 - some error happended
* {  
   "status": "int",  
   "error": "string",  
   "path": "string",  
   "exception": "string",  
   "message": "String",  
   "errors": "String",  
   "trace": "String",  
   "timestamp": "yyyy-MM-dd HH:mm:ss"  
  }

### 获取股票基本面

#### 简要描述：

* 获取所有深沪股票的基本面，或依据code, name参数查找某一股票的基本面数据

#### 请求URL：

* /api/stock/fundamentals

#### 请求方式：

* GET

#### 请求参数

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | Required | 类型 | 说明 | 示例 |
| code | 否 | string | 股票代码 | sh000001 |
| name | 否 | string | 股票名称 | 万科A |
| pageNum | 否 | integer | 当前页码，从1开始；默认值为1 | 1 |
| pageSize | 否 | integer | 每页显示的记录数；默认值为10 | 10 |
| orderBy | 否 | string | 记录排序方式，默认值按时间降序(stmp desc) | stmp desc |

#### 返回示例

* 200 - successful
* {  
   "endRow": 0,  
   "firstPage": 0,  
   "hasNextPage": true,  
   "hasPreviousPage": true,  
   "isFirstPage": true,  
   "isLastPage": true,  
   "lastPage": 0,  
   # 股票数据列表  
   "list": [  
   {}  
   ],  
   "navigatePages": 0,  
   "navigatepageNums": [  
  0  
   ],  
   "nextPage": 0,  
   "orderBy": "string",  
   "pageNum": 0,  
   "pageSize": 0,  
   "pages": 0,  
   "prePage": 0,  
   "size": 0,  
   "startRow": 0,  
   "total": 0  
  }
* 204 - empty content
* {}
* 400~500 - some error happended
* {  
   "status": "int",  
   "error": "string",  
   "path": "string",  
   "exception": "string",  
   "message": "String",  
   "errors": "String",  
   "trace": "String",  
   "timestamp": "yyyy-MM-dd HH:mm:ss"  
  }

## finance-controller: 宏观金融数据分析模块详情

### 获取股票强弱比

#### 简要描述：

* 获取从xx日开始到xx日的股票强弱比

#### 请求URL：

* /api/finance/intensityRate

#### 请求方式：

* GET

#### 请求参数

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | Required | 类型 | 说明 | 示例 |
| from | 否 | string | 起始日期 | 2017-1-1 |
| to | 否 | string | 结束日期 | 2017-12-31 |

#### 返回示例

* 200 - successful
* [  
   {"date": yyyy-MM-dd, "rate": float}  
  ]
* 204 - empty content
* {}
* 400~500 - some error happended
* {  
   "status": "int",  
   "error": "string",  
   "path": "string",  
   "exception": "string",  
   "message": "String",  
   "errors": "String",  
   "trace": "String",  
   "timestamp": "yyyy-MM-dd HH:mm:ss"  
  }

### 获取股票涨跌比

#### 简要描述：

* 获取从xx日开始到xx日的股票涨跌比

#### 请求URL：

* /api/finance/upsDownsRate

#### 请求方式：

* GET

#### 请求参数

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | Required | 类型 | 说明 | 示例 |
| from | 否 | string | 起始日期 | 2017-1-1 |
| to | 否 | string | 结束日期 | 2017-12-31 |

#### 返回示例

* 200 - successful
* [  
   {"date": yyyy-MM-dd, "rate": float}  
  ]
* 204 - empty content
* {}
* 400~500 - some error happended
* {  
   "status": "int",  
   "error": "string",  
   "path": "string",  
   "exception": "string",  
   "message": "String",  
   "errors": "String",  
   "trace": "String",  
   "timestamp": "yyyy-MM-dd HH:mm:ss"  
  }

### 获取期指领先指数

#### 简要描述：

* 获取从xx日开始到xx日的期指领先指数

#### 请求URL：

* /api/finance/FutureIndex

#### 请求方式：

* GET

#### 请求参数

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | Required | 类型 | 说明 | 示例 |
| from | 否 | string | 起始日期 | 2017-1-1 |
| to | 否 | string | 结束日期 | 2017-12-31 |

#### 返回示例

* 200 - successful
* [  
   {"date": yyyy-MM-dd, "rate": float}  
  ]
* 204 - empty content
* {}
* 400~500 - some error happended
* {  
   "status": "int",  
   "error": "string",  
   "path": "string",  
   "exception": "string",  
   "message": "String",  
   "errors": "String",  
   "trace": "String",  
   "timestamp": "yyyy-MM-dd HH:mm:ss"  
  }

### 获取融资融券领先指数

#### 简要描述：

* 获取从xx日开始到xx日的融资融券领先指数

#### 请求URL：

* /api/finance/MarginIndex

#### 请求方式：

* GET

#### 请求参数

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | Required | 类型 | 说明 | 示例 |
| from | 否 | string | 起始日期 | 2017-1-1 |
| to | 否 | string | 结束日期 | 2017-12-31 |

#### 返回示例

* 200 - successful
* [  
   {"date": yyyy-MM-dd, "rate": float}  
  ]
* 204 - empty content
* {}
* 400~500 - some error happended
* {  
   "status": "int",  
   "error": "string",  
   "path": "string",  
   "exception": "string",  
   "message": "String",  
   "errors": "String",  
   "trace": "String",  
   "timestamp": "yyyy-MM-dd HH:mm:ss"  
  }