

## 快递物流节点跟踪

### 接口信息

调用地址: [https://ali-deliver.showapi.com/showapi\\_expInfo](https://ali-deliver.showapi.com/showapi_expInfo)

请求方式: GET

返回类型: JSON

API 调用: [API 简单身份认证调用方法 \(APPCODE\)](#) [API 签名认证调用方法 \(AppKey & AppSecret\)](#)

获取认证: [AppKey & AppCode](#)

### 接口参数

请求参数 (Header)

请求参数 (Query)

字段名称	必填	字段详情
com string	Y	快递公司字母简称, 可以从接口“快递公司查询”中查到该信息。可以使用“auto”代替表示自动识别, 不推荐大面积使用。 实例值: yuantong
nu string	Y	快递单号 实例值: YT6493188734653
phone string	N	收/寄件人手机号后四位, 目前顺丰、跨越、中通为必填, 注: 隐私号需要完整后四位, 例: 13xxxxx1-234, 则需要 13xxxxx1234
callback_url string	N	本接口提供实时查询和异步查询两种方式。异步方式具体说明请参考“产品详情”下的“产品使用常见问题”。 http://www.xxxx.cn/express.do。设置该参数即为使用异步方式, 否则为实时查询。

### 请求示例

#### Java

```
public static void main(String[] args) {
    String host = "https://ali-deliver.showapi.com";
    String path = "/showapi_expInfo";
    String method = "GET";
    String appcode = "你自己的 AppCode";
    Map<String, String> headers = new HashMap<String, String>();
    //最后在 header 中的格式(中间是英文空格)为 Authorization:APPCODE
    83359fd73fe94948385f570e3c139105
    headers.put("Authorization", "APPCODE " + appcode);
    Map<String, String> querys = new HashMap<String, String>();
    querys.put("com", "zhongtong");
    querys.put("nu", "535962308717");
}
```

```

    querys.put("receiverPhone", "receiverPhone");
    querys.put("senderPhone", "senderPhone");

    try {
        /**
         * 重要提示如下:
         * HttpUtils 请从
         *
https://github.com/aliyun/api-gateway-demo-sign-java/blob/master/src/main/java/com/aliyun/api/gateway/demo/util/HttpUtils.java
         * 下载
         *
         * 相应的依赖请参照
         *
https://github.com/aliyun/api-gateway-demo-sign-java/blob/master/pom.xml
         */
        HttpResponse response = HttpUtils.doGet(host, path, method, headers,
        querys);
        System.out.println(response.toString());
        //获取 response 的 body
        //System.out.println(EntityUtils.toString(response.getEntity()));
    } catch (Exception e) {
        e.printStackTrace();
    }
}

```

## 成功响应

### 字段说明

```

{
  "update": "更新时间戳",
  "upgrade_info": "提示信息, 用于提醒用户可能出现的情况",
  "updateStr": "更新时间",
  "logo": "快递公司 logo",
  "dataSize": "数据节点的长度",
  "status": "快递状态 1 暂无记录 2 在途中 3 派送中 4 已签收 (完结状态) 5 用户拒签 6 疑难件 7 无效单 (完结状态) 8 超时单 9 签收失败 10 退回",
  "fee_num": "计费次数。例如: 0 为计费 0 次, 即不计费; 1 为计费 1 次",
  "tel": "快递公司联系方式",
  "data": "在途跟踪数据",
  "- time": "物流跟踪发生的时间",
  "- context": "物流跟踪信息",
  "expSpellName": "快递编码",
  "msg": "返回提示信息",
  "mailNo": "快递单号",
  "queryTimes": "无走件记录时被查询次数 注意: 在 24 小时内, 查询次数>10 次将会计费",

```

"ret\_code": "0 查询成功 或 提交成功。 1 输入参数错误。 2 查不到物流信息。 3 单号不符合规则。 4 快递公司编码不符合规则。 5 快递查询渠道异常。 6 auto 时未查到单号对应的快递公司,请指定快递公司编码。 7 单号与手机号不匹配 其他参数: 接口调用失败",

"flag": "true: 查询成功, 表示 ret\_code=0 且 data 的长度>0。可使用本字段做是否读取 data 列表的依据。 false: 查询失败。",

"expTextName": "快递简称",

"possibleExpList": "自动识别结果"

}

示例

{

"showapi\_res\_error": "",

"showapi\_fee\_num": 1,

"showapi\_res\_code": 0,

"showapi\_res\_id": "628ed5cc0de3769f067c7806",

"showapi\_res\_body": {

"update": 1653528013043,

"upgrade\_info": "",

"updateStr": "2022-05-26 09:20:13",

"logo": "http://static.showapi.com/app2/img/expImg/yuantong.jpg",

"dataSize": 11,

"status": 4,

"fee\_num": 1,

"tel": "021-69777888/95554",

"data": [

{

"time": "2022-05-19 20:25:29",

"context": "客户签收人: 已签收, 签收人凭取货码签收。 已签收 感谢使用圆通速递, 期待再次为您服务 如有疑问请联系: 13\*\*\*\*\*11, 投诉电话: 13\*\*\*\*\*11 疫情期间圆通每天对网点多次消毒, 快递小哥新冠疫苗已接种, 每天测量体温, 佩戴口罩"

},

{

"time": "2022-05-19 10:11:11",

"context": "包裹已经到达附近街区, 安排派送中, 如有疑问请联系 13\*\*\*\*\*11"

},

{

"time": "2022-05-19 09:35:46",

"context": "【云南省昆明市新迎凉亭分部公司】 派件中 派件人: 张胜华 电话 13\*\*\*\*\*11 圆通快递小哥新冠疫苗已接种, 每天已测体温, 请放心收寄快递 如有疑问, 请联系: 13\*\*\*\*\*11"

},

{

"time": "2022-05-19 04:36:19",

"context": "【云南省昆明市新迎公司】 已收入"

},

{

"time": "2022-05-18 17:06:56",

"context": "【昆明转运中心】 已发出 下一站 【云南省昆明市新迎公司】"

},

```
{
  "time": "2022-05-18 17:01:48",
  "context": "【昆明转运中心公司】 已收入"
},
{
  "time": "2022-05-18 01:08:52",
  "context": "【成都转运中心】 已发出 下一站 【昆明转运中心公司】"
},
{
  "time": "2022-05-18 01:08:45",
  "context": "【成都转运中心公司】 已收入"
},
{
  "time": "2022-05-17 22:10:46",
  "context": "【四川省成都市大丰】 已发出 下一站 【成都转运中心公司】"
},
{
  "time": "2022-05-17 21:26:31",
  "context": "【四川省成都市大丰公司】 已打包"
},
{
  "time": "2022-05-17 21:25:56",
  "context": "【四川省成都市大丰公司】 已揽收 取件人:xxx (13*****11)"
}
],
"expSpellName": "yuantong",
"msg": "查询成功",
"mailNo": "YT6493188734653",
"queryTimes": 1,
"ret_code": 0,
"flag": true,
"expTextName": "圆通速递",
"possibleExpList": []
}
}
```

## 快递公司查询

### 接口信息

调用地址: [https://ali-deliver.showapi.com/showapi\\_expressList](https://ali-deliver.showapi.com/showapi_expressList)

请求方式: GET

返回类型: JSON

API 调用: [API 简单身份认证调用方法 \(APPCODE\)](#) [API 签名认证调用方法 \(AppKey & AppSecret\)](#)

获取认证: [AppKey & AppCode](#)

## 接口参数

请求参数 (Header)

请求参数 (Query)

字段名称	必填	字段详情
expName string	N	快递/物流公司名称, 比如传入 顺丰 实例值: 风
maxSize long	N	分页时, 每页返回的最大记录数 实例值: 5
page long	N	分页的页数 实例值: 1

请求示例

Java

```
public static void main(String[] args) {
    String host = "https://ali-deliver.showapi.com";
    String path = "/showapi_expressList";
    String method = "GET";
    String appcode = "你自己的AppCode";
    Map<String, String> headers = new HashMap<String, String>();
    //最后在 header 中的格式(中间是英文空格)为 Authorization:APPCODE
    83359fd73fe94948385f570e3c139105
    headers.put("Authorization", "APPCODE " + appcode);
    Map<String, String> querys = new HashMap<String, String>();
    querys.put("expName", "风");
    querys.put("maxSize", "5");
    querys.put("page", "1");
```

```
try {
    /**
     * 重要提示如下:
     * HttpUtils 请从
     *
```

<https://github.com/aliyun/api-gateway-demo-sign-java/blob/master/src/main/java/com/aliyun/api/gateway/demo/util/HttpUtils.java>

```
* 下载
*
* 相应的依赖请参照
```

```

    *
https://github.com/aliyun/api-gateway-demo-sign-java/blob/master/pom.xml
    */
    HttpResponse response = HttpUtils.doGet(host, path, method, headers,
querys);
    System.out.println(response.toString());
    //获取 response 的 body
    //System.out.println(EntityUtils.toString(response.getEntity()));
} catch (Exception e) {
    e.printStackTrace();
}
}
}

```

成功响应

```

{
  "showapi_res_code": 0,
  "showapi_res_error": "",
  "showapi_res_body": {
    "ret_code": 0,
    "flag": true,
    "expressList": [ //快递公司列表数据
      {
        "imgUrl": "http://app2.showapi.com/img/expImg/rufeng.jpg", //快递公司
的图片/logo //快递公司图标
        "simpleName": "rufeng", //拼音或英文简称
        "phone": "400-010-6660", //官方电话
        "expName": "如风达快递", //快递名称
        "url": "http://www.rufengda.com", //官方网址
        "note": "" //备注信息
      },
      {
        "imgUrl": "http://app2.showapi.com/img/expImg/yafeng.jpg",
        "simpleName": "yafeng",
        "phone": "400-628-0018",
        "expName": "亚风快递",
        "url": "http://www.broad-asia.net",
        "note": ""
      },
      {
        "imgUrl": "http://app2.showapi.com/img/expImg/spring.jpg",
        "simpleName": "spring",
        "phone": "021-62968358",
        "expName": "春风物流",
        "url": "http://www.spring56.com/",
        "note": ""
      }
    ]
  }
}

```

```
}  
}
```

## 单号查快递公司

### 接口信息

调用地址: <https://ali-deliver.showapi.com/fetchCom>

请求方式: GET

返回类型: JSON

API 调用: [API 简单身份认证调用方法 \(APPCODE\)](#) [API 签名认证调用方法 \(AppKey & AppSecret\)](#)

获取认证: [AppKey & AppCode](#)

### 接口参数

请求参数 (Header)

请求参数 (Query)

字段名称	必填	字段详情
nu string	N	快递单号

### 请求示例

#### Java

```
public static void main(String[] args) {  
    String host = "https://ali-deliver.showapi.com";  
    String path = "/fetchCom";  
    String method = "GET";  
    String appcode = "你自己的 AppCode";  
    Map<String, String> headers = new HashMap<String, String>();  
    //最后在 header 中的格式(中间是英文空格)为 Authorization:APPCODE  
83359fd73fe94948385f570e3c139105  
    headers.put("Authorization", "APPCODE " + appcode);  
    Map<String, String> querys = new HashMap<String, String>();  
    querys.put("nu", "nu");
```

```
try {  
    /**  
    * 重要提示如下:  
    * HttpUtils 请从  
    *
```

<https://github.com/aliyun/api-gateway-demo-sign-java/blob/master/src/main/java/com/aliyun/api/gateway/demo/util/HttpUtils.java>

\* 下载

```

    *
    * 相应的依赖请参照
    *
https://github.com/aliyun/api-gateway-demo-sign-java/blob/master/pom.xml
    */
    HttpResponse response = HttpUtils.doGet(host, path, method, headers,
querys);
    System.out.println(response.toString());
    //获取 response 的 body
    //System.out.println(EntityUtils.toString(response.getEntity()));
} catch (Exception e) {
    e.printStackTrace();
}
}
}

```

#### 成功响应

```

{
  "showapi_res_code": 0,
  "showapi_res_error": "",
  "showapi_res_body": {
    "ret_code": 0,
    "data": [
      {
        "simpleName": "zhongtong", //快递公司字母简称,可以从"快递公司列表"或"快递公司查询" 接口中查到该信息 如不知道快递公司名,可以使用"auto"代替,此时将自动识别快递单号
        "expName": "中通快递" //快递名
      }
    ],
    "msg": "操作成功!"
  }
}

```

#### 失败响应

```

{
  "showapi_res_code": 0,
  "showapi_res_error": "",
  "showapi_res_body": {
    "ret_code": -1,
    "msg": "操作失败!"
  }
}

```