

按次计费查询物流

接口信息

调用地址: <https://postv2.market.alicloudapi.com/express2/queryNu>

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“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【auto 是基于以往单号的特征进行人工智能分析, 会有一定的误差, 请谨慎使用】
nu string	Y	快递单号
phone string	N	收/寄件人手机尾号后四位, 顺丰快递必须填写本字段【寄件人手机号或者收件人手机号】 注: 隐私号需要完整后四位, 例: 13xxxxx1-234, 则需要传入的后四位为 1234
delivery_address string	N	最终收件地址, 可用于反生成最终经纬度和预测收件时间, 添加时最少包含省市区
shipping_address string	N	寄件地址, 用于预测收件时间, 添加时最少包含省市区

请求示例

——Java

```
public static void main(String[] args) {
    String host = "https://postv2.market.alicloudapi.com";
    String path = "/express2/queryNu";
    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", "com");
    querys.put("nu", "nu");
}
```

```
querys.put("phone", "phone");
querys.put("delivery_address", "delivery_address");
querys.put("shipping_address", "shipping_address");
```

```
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

参数说明

nu: 快递单号,
logo: 快递公司 logo,
ret_code: 调用说明;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;;101 揽件;102 在途中;103 派送中;104 已签收 (完结状态);105 用户拒签;106 疑难件;107 无效单 (完结状态) ;108 超时单;109 签收失败;110 退回;111 转投;112 待签;113 邮件回收,
com: 快递编码,
update_time: 更新时间,
tel: 快递公司联系方式,
showapi_inner_fee_num: 计费次数,
query_num: 查询次数,
data: 物流跟踪信息列表, 即在途数据,
- time: 发生的时间,
- context: 跟踪信息,
- status: 状态信息;101 揽件;102 在途中;103 派送中;104 已签收 (完结状态);105 用户拒签;106 疑难件;107 无效单 (完结状态) ;108 超时单;109 签收失败;110 退回;111 转投;112 待签,
- location: 位置对应的坐标信息, 坐标系是: GCJ-02,
- address: 位置信息,
com_name: 快递公司名字,

possible_exp_list: 猜测列表。仅当 **com** 参数值为 **auto**，且查询失败时才有返回值。 ,
msg: 返回说明,
delivery_time: 预计到达时间（格式: **yyyy-MM-dd HH:mm**）。该信息是基于历史数据分析的结果，不代表实际情况，仅供参考。

示例:

```
{
  "showapi_res_error": "",
  "showapi_fee_num": 1,
  "showapi_res_code": 0,
  "showapi_res_id": "6541e3940de3763a6c8b7d2e",
  "showapi_res_body": {
    "delivery_time": "2023-04-20 16:00",
    "nu": "772020145776676",
    "logo": "http://static.showapi.com/app2/img/expImg/shentong.jpg",
    "original_com": "shentong",
    "com": "shentong",
    "tel": "400-889-5543",
    "data": [
      {
        "time": "2023-04-20 15:53:17",
        "status": 104,
        "location": "",
        "address": "广东鹤山公司",
        "context": "快件已由【厂里代收点代收】签收，如有疑问请联系派件员：15815758884，您的快递已经妥投，感谢使用申通快递，期待再次为您服务！投诉电话：0750-8833253"
      },
      .....
      {
        "time": "2023-04-18 15:10:05",
        "status": 101,
        "location": "114.304569,30.593354",
        "address": "湖北省-武汉市",
        "context": "[武汉市]【KA 湖北好药师武汉仓项目】(13122713776)的赵勇刚(13367246665)已揽收"
      }
    ],
    "msg": "查询成功",
    "possible_exp_list": [],
    "ret_code": 104,
    "update_time": "2023-11-01 13:35:16",
    "showapi_inner_fee_num": 1,
    "query_num": 1,
    "com_name": "申通快递",
    "delivery_address_details": {
      "address": "广东省-江门市",
      "location": "113.081548,22.578948"
    }
  }
}
```

```
}  
}  
}
```

失败响应

```
{  
  "showapi_res_error": "",  
  "showapi_fee_num": 0,  
  "showapi_res_code": 0,  
  "showapi_res_id": "6541e3bb0de3763a6ca6a660",  
  "showapi_res_body": {  
    "delivery_time": "",  
    "nu": "772020145776676",  
    "logo": "",  
    "original_com": "shenton",  
    "com": "shenton",  
    "tel": "",  
    "data": [],  
    "msg": "当前快递编码未收录或者已失效",  
    "possible_exp_list": [],  
    "ret_code": 4,  
    "update_time": "2023-11-01 13:35:55",  
    "showapi_inner_fee_num": 0,  
    "query_num": 3,  
    "com_name": "",  
    "delivery_address_details": []  
  }  
}
```

•

自动识别

接口信息

调用地址: <https://postv2.market.alicloudapi.com/express2/autoNu>

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	Y	快递单号
addOther string	N	存在时返回 logo、官网、电话

请求示例

——Java

```
public static void main(String[] args) {
    String host = "https://postv2.market.alicloudapi.com";
    String path = "/express2/autoNu";
    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");
    querys.put("addOther", "addOther");

    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

参数

ret_code:0: 查询成功; 1: 参数错误

data:

- imgUrl:快递公司 logo
- phone:快递公司电话
- com:快递公司编码
- com_name:快递公司名称
- url:快递公司官网

msg:返回说明

示例

```
{
  "showapi_res_error": "",
  "showapi_fee_num": 1,
  "showapi_res_code": 0,
  "showapi_res_id": "6541f5230de3763b6cf7bf7e",
  "showapi_res_body": {
    "ret_code": 0,
    "data": [
      {
        "comCode": "zhongtong",
        "com": "zhongtong",
        "com_name": "中通快递"
      }
    ],
    "msg": "操作成功!"
  }
}
```

失败响应

```
{
  "showapi_res_error": "must input nu field",
  "showapi_fee_num": 0,
  "showapi_res_code": -1,
  "showapi_res_id": "6541f52c0de3763a6caae188",
  "showapi_res_body": {}
}
```

快递公司列表查询

接口信息

调用地址: <https://postv2.market.alicloudapi.com/express2/queryCom>

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
exp_name string	N	快递公司名

请求示例

——Java

```
public static void main(String[] args) {
    String host = "https://postv2.market.alicloudapi.com";
    String path = "/express2/queryCom";
    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("exp_name", "exp_name");

    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/aliyu
         n/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();
    }
}
```

```
    } catch (Exception e) {  
        e.printStackTrace();  
    }  
}
```

成功响应

参数

flag: -,
express_list: -,
- img_url: logo,
- com: 快递编码,
- phone: 联系方式,
- exp_name: 快递名称,
- url: 快递官网,
- note: -,
msg: -

示例

```
{  
  "showapi_res_error": "",  
  "showapi_res_code": 0,  
  "showapi_res_id": "60ebb1770de3761e66738645",  
  "showapi_res_body": {  
    "ret_code": 0,  
    "express_list": [  
      {  
        "img_url": "http://static.showapi.com/app2/img/expImg/shunfeng.jpg",  
        "phone": "95338",  
        "exp_name": "顺丰速运",  
        "com": "shunfeng",  
        "url": "http://www.sf-express.com",  
        "note": ""  
      }  
    ],  
    "msg": "查询成功!"  
  }  
}
```

失败响应

```
{  
  "showapi_res_error": "",  
  "showapi_fee_num": 0,  
  "showapi_res_code": 0,  
  "showapi_res_id": "6541e3bb0de3763a6ca6a660",  
  "showapi_res_body": {  
    "delivery_time": "",  
  }  
}
```

```
"nu": "772020145776676",
"logo": "",
"original_com": "shenton",
"com": "shenton",
"tel": "",
"data": [],
"msg": "当前快递编码未收录或者已失效",
"possible_exp_list": [],
"ret_code": 4,
"update_time": "2023-11-01 13:35:55",
"showapi_inner_fee_num": 0,
"query_num": 3,
"com_name": "",
"delivery_address_details": []
```

```
}
```

```
}
```