

数鼎科技 二手车估值 API 文档

目录

1. 全局说明	3
1.1. 接入地址	3
1.2. 网关认证	3
1.3. 全局错误码	3
2. 接口说明	3
2.1. 获取品牌接口	3
2.2. 获取车型接口	5
2.3. 获取型号接口	7
2.4. 获取省份接口	10
2.5. 获取城市接口	11
2.6. 获取颜色接口	13
2.7. 获取估值接口	15

1. 全局说明

1.1. 接入地址

- <http://cloud-api.pistonint.com>

1.2. 网关认证

- 所有接口由阿里云 api gateway 提供服务，调用前请下载 SDK 参考调用说明
- 请将阿里云 UID 设置在 header 中

1.3. 全局错误码

参数名	类型	说明
0	String	请求失败
1	String	请求成功
2	String	请求无数据

2. 接口说明

2.1. 获取品牌接口

- 接口描述：获取所有品牌

- 请求 URL: /brand/find
- 请求方式: GET

参数名	必选	类型	说明
无	无	无	无

- 返回示例:

```
{
  "code": 1,
  "msg": "执行成功",
  "data": [
    {
      "id": "bra00010",
      "name": "AC Schnitzer",
      "abbrName": "AC Schnitzer",
      "dept": null,
      "category": "豪华",
      "fullSpell": "acschnitzer",
      "valid": true,
      "logo":
"http://images.pistonint.com/bizusedcar/newcar/logo/bra00010002.png"
    },
    .....
  ],
  "took": 0.89
}
```

返回参数说明:

名称	类型	说明
code	Integer	返回编码 , 1: 成功; 0: 失败 (其它原因); 2: 无数据;
msg	String	响应消息
data	Array	品牌列表
data.id	String	品牌 id

data.name	String	品牌名称
data.abbrName	String	品牌简称
data.dept	String	品牌部门
data.category	String	类别
data.fullSpell	String	品牌全拼
data.valid	Boolean	有效
data.logo	String	品牌 logo 的 url 地址
took	Double	接口响应耗费时间（单位：秒）

2.2. 获取车型接口

- 接口描述：根据品牌获取所有车型
- 请求 URL：/model/find
- 请求方式：POST
- 数据类型：JSON (application/json)

参数名	必选	类型	说明
brandIds	是	Array	指定的品牌 ids，如： <pre>{ "brandIds": ["bra00050"] }</pre>

• 返回示例：

```
{
  "code": 1,
  "msg": "执行成功",
  "data": [
    {
      "id": "mod35660",
      "name": "Audi Concept Shanghai",
      "oemBrandId": "man03850",
      "oemBrandName": "上汽大众奥迪",
      "oemId": "oem01300",
      "oemName": "上汽大众",
      "brandId": "bra00050",
      "brandName": "奥迪",
      "fullSpell": "audiconceptshanghai",
      "property": null,
      "onSaleFlag": false,
      "isImport": "CKD",
      "pictureUrls": null
    },
    .....
  ],
  "took": 0.649
}
```

• 返回参数说明：

名称	类型	说明
code	Integer	返回编码， 1：成功； 0：失败（其它原因）； 2：无数据；
msg	String	响应消息
data	Array	车型列表
data.id	String	车型 id
data.name	String	车型名称
data.oemBrandId	String	厂商品牌关联 id

data.oemBrandName	String	厂商品牌名称
data.oemId	String	厂商 id
data.oemName	String	厂商名称
data.brandId	String	品牌 id
data.brandName	String	品牌名称
data.fullSpell	String	车型名称全拼
data.property	String	合资/自主/进口标识
data.onSaleFlag	Boolean	是否在售
data.isImport	String	是否进口
data.pictureUrls	List<String>	车型图片地址的列表
took	Double	接口响应耗费时间(单位:秒)

2.3. 获取型号接口

- 接口描述: 根据车型获取所有型号
- 请求 URL: /basicTrim/find
- 请求方式: POST
- 数据类型: JSON (application/json)

参数名	必选	类型	说明
modelIds	是	Array	指定的车型 ids

- 返回示例:

```
{
  "code": 1,
  "msg": "执行成功",
}
```

```

"data": [
  {
    "id": "tri49237",
    "name": "阿斯顿·马丁(进口) Rapide 2017 款 6.0L AMR",
    "abbrName": "2017 款 6.0L AMR",
    "valid": true,
    "year": 2017,
    "msrp": 3600000,
    "onSaleFlag": false,
    "isClassic": false,
    "stopTime": "20190301",
    "startTime": null,
    "bodyTypeId": "bod00030",
    "modelId": "mod00050",
    "oemId": "oem00030",
    "brandId": "bra00030",
    "oemName": "阿斯顿·马丁(进口)",
    "brandName": "阿斯顿·马丁",
    "modelName": "Rapide",
    "bodyType": "掀背车",
    "generation": null
  },
  .....
],
"took": 0.812
}

```

● 返回参数说明:

名称	类型	说明
code	Integer	返回编码 , 1: 成功; 0: 失败 (其它原因); 2: 无数据;
msg	String	响应消息
data	Array	型号列表
data.id	String	型号 id
data.name	String	型号全称
data.abbrName	String	型号简称

data.valid	Boolean	是否有效
data.year	Int	年款
data.msrp	Double	指导价
data.onSaleFlag	Boolean	是否在售标记
data.isClassic	Boolean	是否为经典款
data.stopTime	String	退市时间
data.startTime	String	上市时间
data.bodyTypeId	String	车身类型 id
data.modelId	String	车型 id
data.oemId	String	厂商 id
data.brandId	String	品牌 id
data.oemName	String	厂商名称
data.brandName	String	品牌名称
data.modelName	String	车型名称
data.bodyType	String	车身结构
data.generation	String	车型代
took	Double	接口响应耗费时间（单位：秒）

2.4. 获取省份接口

- 接口描述：获取所有省份列表
- 请求 URL：/province/find

- 请求方式：GET

参数名	必选	类型	说明
无	无	无	无

- 返回示例：

```
{
  "code": 1,
  "msg": "执行成功",
  "data": [
    {
      "id": "pro00010",
      "name": "安徽省",
      "abbrName": "安徽",
      "countryId": "CHN",
      "provinceCode": "340000",
      "fullSpell": "anhui"
    },
    .....
  ],
  "took": 0.0
}
```

- 返回参数说明：

名称	类型	说明
code	Integer	返回编码， 1：成功； 0：失败（其它原因）；
msg	String	响应消息
data	Array	省份列表
data.id	String	省份 id
data.name	String	省份全称
data.abbrName	String	省份简称

data.countryId	String	国家 id
data.provinceCode	String	省份代码
data.fullSpell	String	省份名称全拼
took	Double	接口响应耗费时间（单位：秒）

2.5. 获取城市接口

- 接口描述：根据省份 id 获取城市列表
- 请求 URL：/city/find
- 请求方式：GET

参数名	必选	类型	说明
provinceId	是	String	省份 id, 如: pro00010

- 返回示例：

```
{
  "code": 1,
  "msg": "执行成功",
  "data": [
    {
      "id": "cit00160",
      "name": "宣城市",
      "abbrName": "宣城",
      "fullSpell": "xuancheng",
      "valid": true,
      "areaCode": "0563",
      "cityCode": "341800",
      "provinceCode": "340000",
      "plateCode": "皖 P",
      "provId": "pro00010",
      "provName": null,
      "cityRank": "四线"
    },
    .....
  ],
}
```

```

    "took": 0.0
  }

```

- 返回参数说明:

名称	类型	说明
code	Integer	返回编码 , 1: 成功; 0: 失败 (其它原因);
msg	String	响应消息
data	Array	城市列表
data.id	String	城市 id
data.name	String	城市名称
data.abbrName	String	简称
data.fullSpell	String	名称全拼
data.valid	Boolean	是否有效
data.areaCode	String	区号编码
data.cityCode	String	城市代码
data.provinceCode	String	省份代码
data.plateCode	String	车牌编码
data.provId	String	省份 id
data.provName	String	省份名称
data.cityRank	String	几线城市
took	Double	接口响应耗费时间 (单位: 秒)

2.6. 获取颜色接口

- 接口描述：获取所有颜色列表
- 请求 URL：/color/find
- 请求方式：GET

参数名	必选	类型	说明
无	无	无	无

- 返回示例：

```
{
  "code": 1,
  "msg": "执行成功",
  "data": [
    {
      "id": "Co109",
      "enName": "Black",
      "cnName": "黑色",
      "seqNO": 1,
      "showFlag": true,
      "comment": null,
      "tenantId": null,
      "fullSpell": null,
      "delFlag": 0,
      "firstLetter": null,
      "number": null,
      "name": "黑色"
    },
    .....
  ],
  "took": 0.0
}
```

- 返回参数说明：

名称	类型	说明
----	----	----

code	Integer	返回编码 ， 1：成功； 0：失败（其它原因）；
msg	String	响应消息
data	Array	颜色列表
data. id	String	颜色 id
data. enName	String	英文名称
data. cnName	String	中文名称
data. seqNO	Integer	序列号
data. showFlag	Boolean	是否显示
data. comment	String	备注
data. tenantId	Integer	租户 ID
data. fullSpell	String	全拼
data. delFlag	Integer	逻辑删除值
data. firstLetter	Character	全拼大写首字母
data. number	Integer	编号
data. name	String	颜色名称
took	Double	接口响应耗费时间（单位：秒）

2.7. 获取估值接口

- 接口描述：二手车精准估值接口
- 请求 URL：/valuate

- 请求方式：GET

参数名	必选	类型	说明
trimId	是	String	型号 id
cityId	是	String	城市 id
regTime	是	String	上牌时间（例如：20180225）
mileage	是	Double	行驶里程（单位：万公里，不超过 50）
colorId	否	String	颜色 id
is4SMaint	否	Boolean	是否 4S 店保养，默认是 true

- 返回示例：

```
{
  "code": 1,
  "msg": "执行成功",
  "data": {
    "msrp": 3600000.0,
    "sellValuates": [
      {
        "condGrade": "A",
        "values": [
          1176000.0
        ],
        "valuePcts": [
          0.33
        ]
      },
      {
        "condGrade": "B",
        "values": [
          1093700.0
        ],
        "valuePcts": [
          0.3
        ]
      },
      {
        "condGrade": "C",
        "values": [
          1017100.0
        ],
        "valuePcts": [
          0.28
        ]
      }
    ],
    "buyValuates": [
```

```

    {
      "condGrade": "A",
      "values": [
        1070800.0
      ],
      "valuePcts": [
        0.3
      ]
    },
    {
      "condGrade": "B",
      "values": [
        995800.0
      ],
      "valuePcts": [
        0.28
      ]
    },
    {
      "condGrade": "C",
      "values": [
        926100.0
      ],
      "valuePcts": [
        0.26
      ]
    }
  ],
  "took": 0.317
}

```

• 返回参数说明:

名称	类型	说明
code	Integer	返回编码， 1: 成功; 0: 失败（其它原因）; 101: 型号匹配失败; 102: 城市匹配失败; 103: 颜色匹配失败;
msg	String	响应消息
data	Object	返回数据

data.msrp	Double	指导价
data.sellValuates[]	Array	卖车价估值/残值预测
data.sellValuates[].condGrade	String	车况等级 (A—优秀车况; B—良好车况; C—一般车况)
data.sellValuates[].values[]	List<Double>	估值 (单位: 元)
data.sellValuates[].valuePcts[]	List<Double>	保值率 (公式: 估值/指导价)
data.buyValuates[]	Array	买车价估值/残值预测
data.buyValuates[].condGrade	String	车况等级 (A—优秀车况; B—良好车况; C—一般车况)
data.buyValuates[].values[]	List<Double>	估值 (单位: 元)
data.buyValuates[].valuePcts[]	List<Double>	保值率 (公式: 估值/指导价)
took	Double	接口响应耗费时间(单位: 秒)