

# 零零汽 API 端接口文档

## 目录

零零汽 API 端接口文档 .....	1
<b>1 车架号查询接口 .....</b>	<b>3</b>
1.1 车架号查询 - 查询历史接口 .....	3
1.2 车架号查询 - 解析车架号接口 .....	4
1.3 车架号查询 - 车辆信息接口 .....	6
1.4 车架号查询 - 主组列表 .....	8
1.5 车架号查询 - 分组列表 .....	10
1.6 车架号查询 - 零件列表接口 .....	13
1.7 车架号查询 - 零件图片描点信息接口 .....	16
获取图片描点信息。(该接口在车型查询, 零件号查询与车架号查询中通用) .....	16
1.8 车架号查询 - 零件详情: 零件基础信息 .....	19
1.9 车架号查询 - 零件详情: 渠道价格接口 .....	21
1.10 车架号查询 - 零件详情: 零件替换件接口 .....	23
1.11 车架号查询 - 零件详情: 零件组件接口 .....	24
1.12 车架号查询 - 查询零件号/零件名所在分组 .....	26
<b>2 车型查询相关接口 .....</b>	<b>29</b>
2.1 车型查询 - 品牌列表接口 .....	29
2.2 车型查询 - 第二层条件筛选接口 .....	30
2.3 车型查询 - 第三层条件筛选接口 .....	32
2.4 车型查询 - 第四层条件筛选接口 .....	34
2.5 车型查询 - 第五层条件筛选接口 .....	36
2.6 车型查询 - 第六层条件筛选接口 .....	38
2.7 车型查询 - 第七层条件筛选接口 .....	40
2.8 车型查询 - 第八层条件筛选接口: .....	42
2.9 车型查询 - 第九层条件筛选接口(最后一层): .....	43
2.10 车型查询 - 车辆信息接口 .....	45
2.11 车型查询 - 主组列表接口 .....	47
2.12 车型查询 - 分组列表接口 .....	49
2.13 车型查询 - 零件列表 .....	51
<b>3 零件号查询相关接口 .....</b>	<b>54</b>
3.1 零件号查询 - 零件号查询历史接口 .....	54
3.2 零件号查询 - 确认零件号是否存在和确认品牌: .....	55
3.3 零件号查询 - 获取零件列表(2017-12-27 版本): .....	57
3.4 零件号查询 - 获取基础信息(包含技术信息) .....	59
3.5 零件号查询 - 渠道价格 .....	60
3.6 零件号查询 - 获取替换件 .....	62
3.7 零件号查询 - 获取适用车型 .....	63

3.8	零件号查询 – 获取组件.....	65
3.9	零件号查询 – 获取品牌件 .....	66

# 1 车架号查询接口

## 1.1 车架号查询 – 查询历史接口

### i. 描述：

用户车架号历史查询记录接口

### ii. 请求地址:

操作	车架号查询历史
协议	HTTPS
方式	POST
URL 全径	https://llq.007vin.com/search/vins

### iii. 请求参数:

序号	参数名	中文名称	必选	数据类型	备注
2	uid	用户 id	Y	String	当用户没有登录时，默认 uid 为：5232e6c0e7618e47e39db82a0eaead9a
3	brand	品牌代号	Y	String	可为空值
4	vin	车架号	Y	String	输入车架号能进行模糊匹配，可为空值

### iv. 返回参数:

序号	返回值	中文名称	数据类型	备注(同一状态码有多种情况，用顿号隔开)
1	code	状态码	Int	1：成功 2：车架号长度不正确
2	msg	返回信息	String	-
3	time	返回时间	Int	-
4	data	返回数据	list	List 里包含 dict
参数	brand	品牌	String	-
	brandname	品牌中文名	String	-

	vin	车架号	String	-
--	-----	-----	--------	---

**v. 请求 URL :**

<https://llq.007vin.com/search/vins?brand=bmw&hash=a51ea8096c4ba6549c076a1158d53b6e&hashid=9662104318c7a9ee0648344ffaad14bf&time=1507453514&uid=d592fcb187a1d63a79035f24ec8c85b7&vin=>

**vi. 返回数据实例 :**

```
{
  "msg": "获取数据成功",
  "code": 1,
  "data": [{
    "brand": "bmw",
    "brandname": "宝马 ",
    "vin": "LBVPG96059SC09483"
  }],
}
```

## 1.2 车架号查询 - 解析车架号接口

**i. 描述:**

在车架号输入时，对车架号进行解析。

**ii. 请求地址:**

操作	解析车架号
协议	HTTPS
方式	POST
URL 全径	https://llq.007vin.com/parse/vins

**iii. 请求参数 :**

序号	参数名	中文名称	必选	数据类型	备注
1	hashid	Token 值	Y	String	-

2	uid	用户 id	Y	String	当用户没有登录时，默认的 uid 为： 5232e6c0e7618e47e39db82a0eaead9a
3	vin	车架号	Y	String	-

**iv. 返回参数：**

序号	返回值	中文名称	数据类型	备注(同一状态码有多种情况，用顿号隔开)
1	code	状态码	Int	1：成功 4：车架号长度不正确
2	msg	返回信息	String	-
3	time	返回时间	Int	时间戳
4	data	返回数据	dict	-
参数	title	标题	String	由“国家” - “品牌” - “年份”构成
	brand	品牌	String	-
	brandname	品牌中文名	String	-

**v. 请求 URL 实例：**

https://llq.007vin.com/parse/vins?hash=23c0ce8dc29fe0f6dc64bb80506d3ec8&hashid=9662104318c7a9ee0648344ffaad14bf&time=1507459063&uid=d592fcb187a1d63a79035f24ec8c85b7&vin=LBVPG96059SC09483

**vi. 返回参数实例：**

```
{
  "msg": "解析成功",
  "code": 1,
  "data": {
    "brand": "bmw",
    "brandname": "宝马",
    "title": "中国 - 宝马 - 2009 年"
  }
},
```

```
"time": 1507459048
```

```
}
```

### 1.3 车架号查询 - 车辆信息接口

#### i. 描述：

通过 vin 获取车辆信息。

#### ii. 请求地址：

操作	获取车辆信息
协议	HTTPS
方式	POST
URL 全径	https://llq.007vin.com/ppyvin/searchvins

#### iii. 请求参数：

序号	参数名	中文名称	必选	数据类型	备注
1	hashid	Token 值	Y	String	-
2	uid	用户 id	Y	String	当用户没有登录时，默认的 uid 为： 5232e6c0e7618e47e39db82a0eaead9a
3	vin	车架号	Y	String	-
4	brand	品牌代号	Y	String	当没有获取品牌时，brand = "all"

#### iv. 返回参数：

序号	返回值	中文名称	数据类型	备注(同一状态码有多种情况，用顿号隔开)
1	code	状态码	Int	1：成功  4：车架号为空、无此车辆数据、暂时不支持该品牌、品牌没有找到相关信息  4001：无法获取品牌，进行弹

				框提示
2	msg	返回信息	String	-
3	time	返回时间	Int	-
4	data	返回数据	dict	-
参数	mains	主要信息	String	-
	vin	车架号	String	-
	lfc_log	左上角品牌 logo 地址	String	-
	imglogo	车型图片地址	String	-
	name	汽车品牌中文名称	String	-
	sub	次要数据	String	保留字段
	brand	汽车品牌代号	String	-
	auth	Auth 串	String	用于进行车架号的零件号/零件名/工程码/错误码等查询

#### v. 请求 URL 实例:

https://llq.007vin.com/pppyvin/searchvins?brand=bmw&hash=35641c8ca0745545f995a43b5e556dd2&hashid=9662104318c7a9ee0648344ffaad14bf&time=1507459393&uid=d592fcb187a1d63a79035f24ec8c85b7&vin=LBVPG96059SC09483

#### vi. 返回参数实例:

```
{
  "msg": "数据获取成功",
  "code": 1,
  "data": {
    "sub": "",
    "brand": "bmw",
    "mains": "车型: 宝马-X1 X20(EUR)_2012 年 排量: 2.00 系列 : X E84() 发动机  
型号:N20",
    "vin": "LBVPG96059SC09483",
    "lfc_log": "https://007vin.com/img/bmw.png",
    "imglogo":
```

```

"https://llq.007vin.com/staticimgs/bmw/cars/c102fe6c9fcf0619ea8f8a47ef42f735.jpg"
,
    "name": "宝马"
},
    "time": 1507459377
}

```

## 1.4 车架号查询 - 主组列表

### i. 描述:

获取主组列表。

### ii. 请求地址:

操作	获取主组列表
协议	HTTPS
方式	POST
URL 全径	https://llq.007vin.com/ppyvin/group

### iii. 请求参数:

序号	参数名	中文名称	必选	数据类型	备注
1	hashid	Token 值	Y	String	-
2	uid	用户 id	Y	String	当用户没有登录时, 默认的 uid 为: 5232e6c0e7618e47e39db82a0eaead9a
3	vin	车架号	Y	String	-
4	brand	品牌代号	Y	String	-

### iv. 返回参数:

序号	返回值	中文名称	数据类型	备注(同一状态码有多种情况, 用顿号隔开)
1	code	状态码	Int	1: 成功



				4: 数据异常
2	msg	返回信息	String	-
3	time	返回时间	Int	-
4	data	返回数据	list	List 里包含 dict
参数	groupnum	主组号	String	-
	groupname	主组名称	String	-
	auth	加密 auth 串	String	用来请求分组列表
	img	主组图片 url	String	-
	name	主组名称	String	保留字段

#### v. 请求 URL 实例:

https://llq.007vin.com/ppylvin/group?brand=bmw&hash=3bc17b58eec0cad1be7cbe2  
9ae53056b&hashid=9662104318c7a9ee0648344ffaad14bf&time=1507460143&uid=  
d592fcb187a1d63a79035f24ec8c85b7&vin=LBVPG96059SC09483

#### vi. 返回参数实例:

```
{
  "msg": "",
  "code": 1,
  "data": [{
    "groupname": "01 技术文献 ",
    "groupnum": "01",
    "auth":
      "bjd2enFwNy81NyAkJicnNzk1N3tgeDcvNTclJDc5NTdgfHE3LzU3Nzk1N3ZhbgVwNy8
      1N3YIJSw3aA%3D%3D",
    "name": "01 技术文献",
    "img":
      "https://llq.007vin.com/stcimgs/img/2e0804f99852ce93d5e2469e8a0f8edc.jpg"
  }],
}
```

```
"time": 1507460128
}
```

## 1.5 车架号查询 - 分组列表

### i. 描述：

进入主组后查询分组列表。在路虎中，存在子组(层级关系为：主组 > 分组 > 子组)，用字段 has\_subs 来判断，假如存在子组的话，在分组层中显示 has\_subs 为 True 的元素，然后使用这些元素中的 auth 再次请求分组列表接口。

### ii. 请求地址：

操作	获取分组列表
协议	HTTPS
方式	POST
URL 全径	https://llq.007vin.com/pppyin/subgroup

### iii. 请求参数：

序号	参数名	中文名称	必选	数据类型	备注
1	hashid	Token 值	Y	String	-
2	uid	用户 id	Y	String	当用户没有登录时，默认的 uid 为：5232e6c0e7618e47e39db82a0eae9a
3	vin	车架号	Y	String	-
4	brand	品牌代号	Y	String	-
5	is_filter	是否过滤	Y	String	0：否 1：是
6	auth	加密 Auth 串	Y	String	从主组列表中获取

### iv. 返回参数：

序号	返回值	中文名称	数据类型	备注(同一状态码有多种情况，用顿
----	-----	------	------	------------------

				号隔开)
1	code	状态码	Int	1：成功  4: 车架号为空、无此车辆数据、暂时不支持该品牌、品牌没有找到相关信息  4001: 无法获取品牌，进行弹框提示。让用户进行品牌选择。
2	msg	返回信息	String	-
3	time	返回时间	Int	时间戳
4	data	返回数据	list	List 里包含 dict
参数	base_brandCode	车厂代码	String	如 MINI 的车厂码为 bmw
	uid	标题 id	String	保留字段
	url	分组图片 url	String	-
	mid	分组号	String	-
	subgroupname	分组名称	String	-
	num	主组号	String	-
	is_filter	是否过滤	Int	0: 否 1：是
	auth	Auth 串	String	用来请求零件列表
	model	型号	String	显示在文字列表中
	subgroup	分组号	String	-
	colorvalue	过滤颜色 标识	Int	1. 默认黑色 2. 红色
	description	分组描述 信息	String	文字很长，显示在文字列表中
	has_subs	是否有子 组	boolean	True：有 False：无

#### V. 请求 URL 实例:

https://llq.007vin.com/ppyvin/subgroup?auth=bjd2enFwNy81NyAkJicnNzk1N3tgeDc  
vNTckljc5NTdggfHE3LzU3Nzk1N3ZhbGVwNy81N3YlJSw3aA%253D%253D&brand=bm  
w&hash=bde380791e74aa448843d524d5753110&hashid=9662104318c7a9ee06483  
44ffaad14bf&is\_filter=0&time=1507460934&uid=d592fcb187a1d63a79035f24ec8c85  
b7&vin=LBVPG96059SC09483

#### vi. 返回参数实例:

```
{
  "msg": "",
  "code": 1,
  "data": {
    "base_brandCode": "bmw",
    "uid": "5dc2901daa576579b9ee41ef8665e8d6",
    "url":
      "https://data.007vin.com/ppyimage/tmbnail?code=bmw&key=bjd2enFwNy81NyAkJi  
cnNzk1N2B8cTcvNTc3OTU3d2d0e3E3LzU3d3hiNzk1N3h8cTcvNTchISltISA3OTU3dm  
FsZXA3LzU3diUildc5NTd7YHg3LzU3JC3OTU3fGZKc3x5YXBnNy81JTtk1N2Zgd3Jnem  
BINy81NyQiSiUmJCw3aA%3D%3D&w=160&h=160",
    "mid": "17_0319",
    "subgroupname": "17_0319 机油冷却管/热交换器",
    "num": "17",
    "is_filter": 0,
    "auth":
      "bjd2enFwNy81NyAkJicnNzk1N3tgeDcvNTckljc5NTdggfHE3LzU3Nzk1N2Zgd3JnemBle  
3R4cDcvNTcklkoJiQsNUlglylmdElgl3Z3LElglCR3IkglCYiUlglnd0JDU6NUlgliVwcUlglX  
B0iUlglYyJ0lglCMjLTc5NTd8ZkpzfHlhcGc3LzU3OTU3d2d0e3E3LzU3d3hiNzk1N3h8cT  
cvNTchISltISA3OTU3dmFsZXA3LzU3diUildc5NTdmYHdyZ3pgZTcvNTcklkoJiQsN2g%  
3D",
    "model": "",
    "subgroup": "05",
```

```

        "colorvalue": 1,
        "description": ""
    }
],
    "time": 1507460918
}

```

## 1.6 车架号查询 - 零件列表接口

### i. 描述:

获取该车架号相应分组里的零件列表。

### ii. 请求地址:

操作	获取零件列表
协议	HTTPS
方式	POST
URL 全径	https://llq.007vin.com/ppyvin/parts

### iii. 请求参数:

序号	参数名	中文名称	必选	数据类型	备注
1	hashid	Token 值	Y	String	-
2	uid	用户 id	Y	String	当用户没有登录时, 默认的 uid 为: 5232e6c0e7618e47e39db82a0eae9a
3	vin	车架号	Y	String	-
4	brand	品牌代码	Y	String	-
5	is_filter	是否过滤	Y	String	0 : 否 1 : 是
6	auth	Auth 串	Y	String	从分组列表接口中获取

### iv. 返回参数:

序号	返回值	中文名称	类型	备注(同一状态码有多种情况, 用顿号隔开)
1	code	状态码	Int	0 : 网络异常 1 : 成功 2 : 没有相关数据 4 : 数据异常
2	msg	返回信息	String	-
3	time	返回时间	Int	时间戳
4	flage	是否折叠	Int	保留字段
5	uid	用户 id	String	-
6	itid	描点与零件号对应关系	String	保留字段
7	auth	Auth 串	String	-
8	label	分组名称	String	-
9	data	返回数据	list	List 里包含 dict
参数	remark	说明	String	-
	modelname	型号名称	String	保留字段
	pid	零件号	String	在零件价格、替换件使用该参数
	itid	描点位置	String	与零件序号(num)有对应关系
	detail	详情	String	-
	multi_price	是否是多价格	Int	0 : 不是 1 : 是
	label	零件名称	String	0 : 不是 1 : 是
	step	零件位置 标记	Int	保留字段
	num	零件序号	String	-
	is_filter	是否过滤	Int	0: 过滤 1: 不过滤

prices	渠道价格	String	-
isreplace	是否有替换件	Int	0：否 1：是
real_pid	真实零件号	String	在组件信息中使用该参数
model	型号	String	-
quantity	数量	String	-

#### v. 请求 URL 实例:

https://llq.007vin.com/ppyyin/parts?auth=bjd2enFwNy81NyAkJicnNzk1N3tgeDcvNTckljc5NTdggfHE3LzU3Nzk1N2Zgd3JnemBle3R4cDcvNTcklkolJicnNVBtZXR7Znx6ezVhdHt%252BNzk1N3xmSnN8eWfWZzcVNSU5NTd3Z3R7cTcvNTd3eGI3OTU3eHxxNy81NyQmLSwmLTc5NTd2YWxlcDcvNTd2JSUsNzk1N2Zgd3JnemBINy81NyQiSiUmJyc3aA%253D%253D&brand=bmw&hash=2447374cc79a1d5ff83e2b06ffe269c7&hashid=9662104318c7a9ee0648344ffaad14bf&is\_filter=0&time=1507512921&uid=d592fcb187a1d63a79035f24ec8c85b7&vin=LBVPG96059SC09483

#### vi. 返回数据实例:

```
{
  "flage": 1,
  "code": 1,
  "uid": "2b45735d8b3e0a64a5907ee0a49efb1b",
  "itid": "",
  "auth":
    "bjd2enFwNy81NyAkJicnNzk1N3tgeDcvNTckljc5NTdggfHE3LzU3Nzk1N2Zgd3JnemBle3R4cDcvNTcklkolJicnNVBtZXR7Znx6ezVhdHt%2BNzk1N3xmSnN8eWfWZzcVNSU5NTd3Z3R7cTcvNTd3eGI3OTU3eHxxNy81NyQmLSwmLTc5NTd2YWxlcDcvNTd2JSUsNzk1N2Zgd3JnemBINy81NyQiSiUmJyc3aA%3D%3D",
  "label": "17_0322 Expansion tank",
  "time": 1507512905,
  "msg": "成功",
}
```

```
"data": [{
  "remark": "",
  "modelName": "",
  "pid": "17137607482",
  "itid": "1",
  "detail": "详情",
  "multi_price": 0,
  "label": "Expansion tank",
  "step": 1,
  "num": "1",
  "is_filter": 0,
  "prices": "¥1120.0",
  "isreplace": 1,
  "real_pid": "17137607482",
  "model": "",
  "quantity": "1"
},]
}
```

## 1.7 车架号查询 - 零件图片描点信息接口

### i. 描述:

获取图片描点信息。(该接口在车型查询, 零件号查询与车架号查询中通用)

### ii. 请求地址:

操作	获取零件图片描点信息
协议	HTTPS
方式	POST
URL 全径	https://llq.007vin.com/ppycars/subimgs



## iii. 请求参数：

序号	参数名	中文名称	必选	数据类型	备注
1	hashid	Token 值	Y	String	-
2	uid	用户 id	Y	String	当用户没有登录时，默认的 uid 为： 5232e6c0e7618e47e39db82a0eae9a
3	brand	品牌代号	Y	String	-
4	auth	Auth 串	Y	String	从分组信息中获取

## iv. 返回参数：

序号	返回值	中文名称	类型	备注
1	code	状态码	Int	1：成功 4：没有数据
2	msg	返回信息	String	-
3	time	返回时间	Int	时间戳
4	data	返回数据	Dict	-
参数	lfc_log	左上角品牌 log 图标 url	String	-
	img_url	零件图片 url	String	-
	mapdata	描点信息	Dict	-
	参数	minx	水平坐标最小值	String
		miny	垂直坐标最小值	String
		maxx	水平坐标最大值	String
		maxy	垂直坐标最大值	String
		num	坐标个数	String
	point	图片信息	Dict	-
	参数	h	图片高度	Num
		w	图片长度	Num

## v. 请求 URL 实例：

https://llq.007vin.com/ppycars/subimgs?auth=bjd2enFwNy81NyAkJicnNzk1N3tgeDcv  
NTckljc5NTdggfHE3LzU3Nzk1N2Zgd3JnemBle3R4cDcvNTcklkoJicnNVBtZXR7Znx6ezV

hdHt%252BNzk1N3xmSnN8eWFwZzcVNSU5NTd3Z3R7cTcvNTd3eGI3OTU3eHxxNy81  
 NyQmLSwmLTc5NTd2YWxlCdcvNTd2JSUsNzk1N2Zgd3JnemBINy81NyQiSiUmJyc3aA  
 %253D%253D&brand=bmw&hash=9c9bb5704d603c87d28be27cc9380fb9&hashid=9  
 662104318c7a9ee0648344ffaad14bf&time=1507512921&uid=d592fcb187a1d63a790  
 35f24ec8c85b7

**vi. 返回参数实例：**

```
{
  "msg": "",
  "code": 1,
  "data": {
    "lfc_log": "https://007vin.com/img/bmw.png",
    "mapdata": [{
      "minx": "608",
      "miny": "311",
      "maxx": "615",
      "maxy": "331",
      "num": "1"
    }],
    "imgurl":
      "http://data.007vin.com/app/imgs?brand=bmw&auth=bjd2enFwNy81NyAkIcnNzk1  

      N3h8cTcvNyQmLSwmLTdo&uid=d592fcb187a1d63a79035f24ec8c85b7&hashid=96  

      62104318c7a9ee0648344ffaad14bf",
    "point": {
      "h": 0,
      "w": 0
    }
  },
  "time": 1507512905
}
```

## 1.8 车架号查询 - 零件详情：零件基础信息

### i. 描述:

在零件详情页面中获取零件基础信息（在零件号查询，车架号查询与车型查询中，零件基础信息接口通用）。

### ii. 请求地址:

操作	获取零件基础信息
协议	HTTPS
方式	POST
URL 全径	https://llq.007vin.com/ppypart/partdetail

### iii. 请求参数：

序号	参数名	中文名称	必选	数据类型	备注
1	hashid	Token 值	Y	String	-
2	uid	用户 id	Y	String	当用户没有登录时，默认的 uid 为： 5232e6c0e7618e47e39db82a0eaead9a
3	brand	品牌代号	Y	String	-
4	pid	零件号	Y	String	-

### iv. 返回参数：

序号	返回值	中文名称	类型	备注
1	code	状态码	Int	0：网络异常 1：成功 4：无数据
2	msg	返回信息	String	-
3	time	返回时间	Int	时间错
4	data	返回数据	dict	-

参数	headermessage		零件信息	List	包含零件号以及名称
	image		零件实物图 url	String	保留字段
	headname		标签栏	List	包含该零件信息的标签，如：基础信息，替换件，组件
	basemessage		基础信息	Dict	-
	参数	list	基础信息	String	-

#### v. 请求 URL 实例：

<https://llq.007vin.com/ppypart/partdetail?brand=bmw&hash=4f8bff5962d2dc149c8caad32366782e&hashid=9662104318c7a9ee0648344ffaad14bf&pid=17137607482&time=1507513971&uid=d592fcb187a1d63a79035f24ec8c85b7>

#### vi. 返回参数实例：

```
{
  "msg": "",
  "code": 1,
  "data": {
    "headermessage": ["零件号:17137607482", "名称:Expansion tank"],
    "image": "",
    "headname": ["基础信息", "渠道价格", "替换件"],
    "basemessage": {
      "list": ["件数:1", "型号:", "备注:"]
    }
  },
  "label": "",
  "time": 1507513955
}
```

## 1.9 车架号查询 - 零件详情：渠道价格接口

### i. 详情:

在零件信息里, 点击渠道价格, 获取该零件的渠道价格的信息 (在零件号查询, 车架号查询与车型查询中, 渠道价格接口通用)。

### ii. 请求地址：

操作	获取零件渠道价格
协议	HTTPS
方式	POST
URL 全径	https://llq.007vin.com/ppypart/partprice

### iii. 请求参数：

序号	参数名	中文名称	必选	数据类型	备注
1	hashid	Token 值	Y	String	-
2	uid	用户 id	Y	String	当用户没有登录时, 默认的 uid 为: 5232e6c0e7618e47e39db82a0eaead9a
3	brand	品牌代号	Y	String	-
4	pid	零件号	Y	String	-

### iv. 返回参数：

序号	返回值	中文名称	类型	备注
1	code	状态码	Int	0：网络异常 1：成功 4：零件号为空、没有数据
2	msg	返回信息	String	-
3	time	返回时间	Int	-
4	data	返回数据	list	List 里包含 dict
参数	origin	产地	String	包含零件号以及名称

	remark	备注	String	-
	eot_price	不含税进 货价	String	-
	factory_type	产商类型	Int	1: 原厂件, 2: 品牌件, 3: 适用件
	parttype	零件类型	String	List 与基础信息对应
	pid	零件号	String	-
	prices	售价	String	-
	mill	厂家	String	-
	cost_price	含税进 货价	String	-
5	label	零件名称	String	-

**v. 请求 URL 实例：**

<https://llq.007vin.com/ppypart/partprice?brand=bmw&hash=12baf64332564bf8a18ec6bf1dee658b&hashid=9662104318c7a9ee0648344ffaad14bf&pid=17137524812&time=1507515755&uid=d592fcb187a1d63a79035f24ec8c85b7>

**vi. 返回数据实例：**

```
{
  "msg": "成功",
  "code": 1,
  "data": [{
    "origin": "--",
    "remark": "--",
    "eot_price": "--",
    "factory_type": 1,
    "parttype": "原厂件",
    "pid": "17137524812",
    "prices": "¥162.0",
    "mill": "宝马",
```

```

        "cost_price": "--"
    },
    "label": "",
    "time": 1507515739
}

```

## 1.10 车架号查询 - 零件详情：零件替换件接口

### i. 描述：

在零件详情中，点击替换件，获取替换件数据的接口（在零件号查询，车架号查询与车型查询中，零件替换件接口通用）。

### ii. 请求地址：

操作	获取零件替换件
协议	HTTPS
方式	POST
URL 全径	https://llq.007vin.com/ppypart/partsreplace

### iii. 请求参数：

序号	参数名	中文名称	必选	数据类型	备注
1	hashid	Token 值	Y	String	-
2	uid	用户 id	Y	String	当用户没有登录时，默认的 uid 为：5232e6c0e7618e47e39db82a0eaead9a
3	brand	品牌代号	Y	String	-
4	pid	零件号	Y	String	-

### iv. 返回参数：

序号	返回值	中文名称	类型	备注
1	code	状态码	Int	1：成功 4：车架号为空、无此车辆数据、暂时不支

				持该品牌、品牌没有找到相关信息 4001: 无法获取品牌, 进行弹框提示
2	msg	返回信息	String	-
3	time	返回时间	Int	时间戳
4	data	零件号列表	list	-
5	label	零件名称	String	-

**v. 请求 URL 实例：**

<https://llq.007vin.com/ppypart/partsreplace?brand=bmw&hash=b55107dc8e3d89d9a17a85d6875a2449&hashid=9662104318c7a9ee0648344ffaad14bf&pid=17137524812&time=1507515756&uid=d592fcb187a1d63a79035f24ec8c85b7>

**vi. 返回数据实例：**

```
{
  "msg": "成功",
  "code": 1,
  "data": ["17130409622", "17110152373"],
  "label": "",
  "time": 1507515740
}
```

## 1.11 车架号查询 – 零件详情：零件组件接口

**i. 描述：**

在零件详情中，点击替组件，获取组件数据的接口（零件号查询，车架号查询与车型查询中，零件基础信息接口通用）。

**ii. 请求地址：**

操作	获取零件替换件
协议	HTTPS



方式	POST
URL 全径	https://llq.007vin.com/ppypart/partscompt

### iii. 请求参数：

序号	参数名	中文名称	必选	数据类型	备注
1	hashid	Token 值	Y	String	-
2	uid	用户 id	Y	String	当用户没有登录时，默认的 uid 为： 5232e6c0e7618e47e39db82a0eae9a
3	brand	品牌代号	Y	String	-
4	pid	零件号	Y	String	-

### iv. 返回参数：

序号	返回值	中文名称	类型	备注
1	code	状态码	Int	0：网络异常 1：成功 4：车架号为空、无此车辆数据、品牌没有找到相关信息
2	msg	返回信息	String	-
3	time	返回时间	Int	时间戳
4	label	零件名称	String	-
5	data	返回数据	List	List 里包含 dict
参数	remark	备注	String	-
	pid	零件号	String	-
	label	名称	String	-
	num	件数	String	-
	model	型号	String	-
	id	编号	String	-

### v. 请求 URL 实例：

<https://llq.007vin.com/ppypart/partsreplace?brand=bmw&hash=b55107dc8e3d89d9>

[a17a85d6875a2449&hashid=9662104318c7a9ee0648344ffaad14bf&pid=17137524812&time=1507515756&uid=d592fcb187a1d63a79035f24ec8c85b7](http://a17a85d6875a2449&hashid=9662104318c7a9ee0648344ffaad14bf&pid=17137524812&time=1507515756&uid=d592fcb187a1d63a79035f24ec8c85b7)

**vi. 返回数据实例：**

```
{
  "msg": "成功",
  "code": 1,
  "data": [{
    "remark": "",
    "pid": "11002420297",
    "label": "保修 传动机构",
    "num": "1",
    "model": "N20B20A",
    "id": ""
  }],
  "label": "",
  "time": 1507865396
}
```

## 1.12 车架号查询 – 查询零件号/零件名所在分组

**i. 描述：**

在车架号下面(只支持车架号零件号/零件名查询)，获取该车架号下特定零件号或者零件名称所在的分组的列表,零件号可不输全。

**ii. 请求地址：**

操作	查询零件号/零件名所在分组
协议	HTTPS
方式	POST
URL 全径	https://llq.007vin.com/engine/parts_search

## iii. 请求参数：

序号	参数名	中文名称	必选	数据类型	备注
1	hashid	Token 值	Y	String	-
2	uid	用户 id	Y	String	当用户没有登录时，默认的 uid 为： 5232e6c0e7618e47e39db82a0eae9a
3	brand	品牌代号	Y	String	-
4	vin	车架号	Y	String	-
5	pid	零件号 / 零件名 / 工程码	Y	String	-
6	auth	Auth 串	Y	String	从 ppyvin/searchvins 中获取
7	searchtype	查询类型	Y	String	pid: 零件号 label: 零件名 engpid: 工厂码
8	filter	是否过滤	Y	String	填写 1 即可

## iv. 返回参数：

序号	返回值	中文名称	类型	备注
1	code	状态码	Int	0：网络异常 1：成功
2	msg	返回信息	String	-
3	time	返回时间	Int	时间戳
4	total_page	总页数	String	-
5	total	数据数量	String	-
6	page	当前页码	String	-
7	data	返回数据	List	List 里包含 dict
参数	base_brandCode	品牌	String	-
	smark	保留字段	String	-

	url	分组图片 url	String	-
	pid	零件号	String	-
	mid	分组号	String	-
	auth	Auth 串	String	用于请求该分组的零件列表
	subgroupname	分组名称	String	-
	model	保留字段	String	-
	smodel	保留字段	String	-
	pid_list	模糊搜索对应零件号列表	dict	Key 为零件号 Value 全部为 true
	maingroup	主组号	String	-

## 2 车型查询相关接口

### 2.1 车型查询 – 品牌列表接口

i. 描述:

获取所有当前支持品牌查询的品牌列表。

ii. 请求地址：

操作	品牌列表获取
协议	HTTPS
方式	POST
URL 全径	https://llq.007vin.com/ppycars/brand

iii. 请求参数：

此接口中，只用传通用参数

iv. 返回参数：

序号	返回值	中文名称	类型	备注
1	code	状态码	Int	1：成功 2：查询失败
2	msg	返回信息	String	-
3	time	返回时间	Int	时间戳
4	data	返回数据	List	List 里包含 dict
参数	brandEn	品牌英文名	String	-
	brand	品牌代号	String	-
	brandCn	品牌中文名	String	-
	brandImg	品牌 log 图片 url	String	-

v. 请求 URL 实例：

<https://llq.007vin.com/ppycars/brand?hash=d900809a5f56a798570c51f896ad4ef3&ti>

[me=1507517277](#)

vi. 返回数据实例：

```
{
  "msg": "success",
  "code": 1,
  "data": [{
    "brandEn": "LAND_ROVER",
    "brand": "land_rover",
    "brandCn": "路虎",
    "brandImg": "https://llq.007vin.com/pppy/img/brand/land_rover.png"
  }],
  "time": 1507517261
}
```

## 2.2 车型查询 - 第二层条件筛选接口

i. 描述:

选择品牌之后，进行第二层条件选择。

ii. 请求地址:

操作	进行第二层条件选择
协议	HTTPS
方式	POST
URL 全径	https://llq.007vin.com/cars/show

iii. 请求参数：

序号	参数名	中文名称	必选	数据类型	备注
1	hashid	Token 值	Y	String	-

2	brand	品牌	Y	String	-
3	uid	用户 id	Y	String	-

**iv. 返回参数：**

序号	返回值	中文名称	类型	备注
1	code	状态码	Int	1：成功 2：无数据
2	msg	返回信息	String	-
3	time	返回时间	Int	时间戳
4	title	条件类型	String	-
5	data	返回数据	List	List 里包含 dict
参数	has_next	是否拥有下一层条件	String	1：是 0：否
	code	品牌代号	String	-
	last	是否是最后一层	Boolean	Flase：否 True：是
	name	条件名称	String	-
	auth	Auth 串	String	-

**v. 请求 URL 实例：**

<https://llq.007vin.com/cars/show?brand=audi&hash=6d42fe4733e894aa1812056246147a57&hashid=9662104318c7a9ee0648344ffaad14bf&time=1507517812&uid=d592fcb187a1d63a79035f24ec8c85b7>

**vi. 返回数据实例：**

```
{
  "msg": "",
  "title": "选择车厂",
  "code": 1,
  "data": [{
    "has_next": "1",
```

```

"code": "audi",

"last": false,

"name": "一汽-大众",

"auth":

"bjd7dHhwNy81N0lglXAIJUIgl3YicThJYCAsjyJJYCFzJCI3OTU3YHxxNy81NyQkJCQ3aA

%3D%3D"

}

],

"time": 1507517796

}

```

## 2.3 车型查询 - 第三层条件筛选接口

### i. 描述:

经过两层条件筛选后，进行第三层条件选择。

### ii. 请求地址:

操作	第三次条件选择
协议	HTTPS
方式	POST
URL 全径	https://llq.007vin.com/cars/code

### iii. 请求参数 :

序号	参数名	中文名称	必选	数据类型	备注
1	hashid	Token 值	Y	String	-
2	brand	品牌代号	Y	String	-
3	uid	用户 id	Y	String	-
4	auth	Auth 串	Y	String	从第二层筛选条件中获取



**iv. 返回参数：**

序号	返回值	中文名称	类型	备注
1	code	状态码	Int	0：数据异常 1：成功
2	msg	返回信息	String	-
3	time	返回时间	Int	时间戳
4	title	条件类型名称	String	-
5	data	返回数据	List	List 里包含 dict
参数	code	品牌代号	String	-
	last	是否是最后一层	Boolean	True：是 False：否
	name	条件名称	String	-
	auth	Auth 串	String	-
	keys	链接参数	String	保留字段

**v. 请求 URL 实例：**

https://llq.007vin.com/cars/code?auth=bjd7dHhwNy81N0lgIXAIJUlgl3YicThJYCAsjJJ  
YCFzJCI3OTU3YHxxNy81NyQkJCQ3aA%253D%253D&brand=audi&hash=dc74e2c8c1  
2666f3f7cb43a712194171&hashid=9662104318c7a9ee0648344ffaad14bf&time=150  
7519583&uid=d592fcb187a1d63a79035f24ec8c85b7

**vi. 返回数据实例：**

```
{
  "msg": "",
  "title": "数据获取成功",
  "code": 1,
  "data": [{
    "msg": "",
    "title": "选择车型",
```

```

"code": 1,

"data": [{

  "code": "audi",

  "last": false,

  "name": "TTRS Audi TTRS Coupe/Roadster 2018>> 1 奥迪 TTRS Coupe(双
门) Ro.(敞篷跑车) 2018 款 ",

  "keys": "",

  "auth":

"bjd7dHhwNy81N0lglXAIJlgl3YicThJYcAsJyJJYCFzJCI3OTU3dnp7cXxhfHp7Ny81N0F
BR0Y1VGBxfDVBQUdGNVZ6YGVwOkd6dHFmYXBnNTUnJSQtKys1JDVJYHNzJHQ1SW
AgLCMgSWAtc3B0NUFBR0Y1VnpgZXA9SWAgJnZ2SWAsIHAtPDVHejs9SWAjlCBwSW
Aid3MiSWAtcXEkSWAtcyMjPDUnJSQtSWAgcClhSWAjdYzWn2g%3D"

}],

"time": 1507517796

}

```

## 2.4 车型查询 - 第四层条件筛选接口

### i. 描述:

经过前三层条件筛选后，进行第四层条件选择

### ii. 请求地址:

操作	第四次条件选择接口
协议	HTTPS
方式	POST
URL 全径	https://llq.007vin.com/cars/model

### iii. 请求参数 :

序号	参数名	中文名称	必选	数据类型	备注

1	hashid	Token 值	Y	String	-
2	brand	品牌代号	Y	String	-
3	uid	用户 id	Y	String	-
4	auth	Auth 串	Y	String	从第三层筛选条件接口中获取

#### iv. 返回参数：

序号	返回值	中文名称	类型	备注
1	code	状态码	Int	0：数据异常 1：成功
2	msg	返回信息	String	-
3	time	返回时间	Int	时间戳
4	title	条件类型名称	String	-
5	data	数据	List	List 里包含 dict
参数	code	品牌代号	String	-
	last	是否是最后一层	Boolean	True：是 False：否
	name	条件名称	String	-
	auth	Auth 串	String	-
	keys	链接参数	String	保留参数

#### v. 请求 URL 实例：

https://llq.007vin.com/cars/model?  
auth=bjd7dHhwNy81N0lgInMtcElgICNzcTc5NTd2entxfGF8ens3LzU3SWAjl3MhSWAjl3MnSWAil3BwSWAgcyAgN2g%253D&brand=porsches&hash=fe6050e8025160f384de42f5bb9d9d68&hashid=fdf7430a0a7166c5ecb9bb1b1f11c46b&time=1507532790&uid=d592fcb187a1d63a79035f24ec8c85b7

#### vi. 返回数据实例：

```
{
  "msg": "",
```

```

"title": "选择车型",
"code": 1,
"data": [{
  "code": "porsches",
  "last": false,
  "name": "ATK",
  "keys": "",
  "auth":
    "bjd5fHh8YTcvNTdUQV43OTU3e3R4cDcvNTdJYCJzLXBJYCAjc3E3OTU3dnp7cXxhfHp
    7Ny81N0lglyNzlUlglyYjJ0lgliNwcElgIHMglDdo"
}],
"time": 1507532577
}

```

## 2.5 车型查询 - 第五层条件筛选接口

### i. 描述:

经过前四层条件筛选后，进行第五层条件选择

### ii. 请求地址:

操作	第五次条件选择接口
协议	HTTPS
方式	POST
URL 全径	https://llq.007vin.com/cars/litm

### iii. 请求参数:

序号	参数名	中文名称	必选	数据类型	备注
5	hashid	Token 值	Y	String	-
7	brand	品牌	Y	String	-

8	uid	用户 id	Y	String	-
9	auth	Auth 串	Y	String	从第四层筛选条件中获取

**iv. 返回参数:**

序号	返回值	中文名称	类型	备注
1	code	状态码	Int	0：数据异常 1：成功
2	msg	返回信息	String	-
3	time	返回时间	Int	时间戳
4	title	条件类型名称	String	-
5	data	数据	List	List 里包含 dict
参数	code	品牌代号	String	-
	last	是否是最后一层	Boolean	True：是 False：否
	name	条件	String	-
	auth	Auth 串	String	-
	keys	参数连接	String	保留参数

**v. 请求 URL 实例：**

https://llq.007vin.com/cars/litm?auth=bjd5fHh8YTcvNTdTISM1Umd0ezVBemBncGc9JyUklTgnJSQiPDc5NTd7dHhwNy81NydJYCJ2c3c3OTU3dnp7cXxhfHp7Ny81Nyckl3E3aA%253D%253D&brand=bmw&hash=cb522d2f357e1a7d1f761f9398b5e1d7&hashid=df7430a0a7166c5ecb9bb1b1f11c46b&time=1507533431&uid=d592fcb187a1d63a79035f24ec8c85b7

**vi. 返回数据实例：**

```
{
  "msg": "",
  "title": "选择车身",
```

```

"code": 1,
"data": [{
  "code": "bmw",
  "last": false,
  "name": "Gran Tourer",
  "keys": "",
  "auth":
    "bjd5fHh8YTcvNTdTISM1Umd0ezVBemBncGc9JyUklTgnJSQiPDc5NTd7dHhwNy81Ny
    dJYCJ2c3c3OTU3dnp7cXxhfHp7Ny81Nyckl3E3OTU3eXx4fGEnNy81N1JndHs1QXpgZ
    3BnN2g%3D"
}],
"time": 1507533218
}

```

## 2.6 车型查询 - 第六层条件筛选接口

### i. 描述:

经过前五层条件筛选后，进行第六层条件选择。

### ii. 请求地址:

操作	第六次条件选择接口
协议	HTTPS
方式	POST
URL 全径	https://llq.007vin.com/cars/litn

### iii. 请求参数：

序号	参数名	中文名称	必选	数据类型	备注
1	hashid	Token 值	Y	String	-
2	brand	品牌	Y	String	-

3	uid	用户 id	Y	String	-
4	auth	Auth 串	Y	String	从第五层筛选条件中获取

**iv. 返回参数：**

序号	返回值	中文名称	类型	备注
1	code	状态码	Int	0：数据异常 1：成功
2	msg	返回信息	String	-
3	time	返回时间	Int	时间戳
4	title	条件类型名称	String	-
5	data	返回数据	list	List 里包含 dict
参数	code	品牌代号	String	-
	last	是否是最后一层	Boolean	True：是 False：否
	name	条件名称	String	-
	auth	Auth 串	String	-
	keys	链接参数	String	保留参数

**v. 请求 URL 实例：**

https://llq.007vin.com/cars/litn?auth=bjd5fHh8YTcvNTdTISM1Umd0ezVBemBncGc9Jy  
 UkITgnJSQiPDc5NTd7dHhwNy81NydJYCJ2c3c3OTU3dnp7cXxfHp7Ny81Nyckl3E3OT  
 U3eXx4fGEnNy81N1JndHs1QXpgZ3BnN2g%253D&brand=bmw&hash=5f88bf037ce2  
 82ff3157bfae08462333&hashid=fdf7430a0a7166c5ecb9bb1b1f11c46b&time=150753  
 3618&uid=d592fcb187a1d63a79035f24ec8c85b7

**vi. 返回数据实例：**

```
{
  {
    "msg": "",
    "title": "选择市场",
```

```

"code": 1,
"data": [{
  "code": "bmw",
  "last": false,
  "name": "ECE",
  "keys": "",
  "auth":
    "bjd5fHh8YSY3LzU3UFZQNzk1N3l8eHxhNy81N1MhlzVSZ3R7NUF6YGdwZz0nJSQhO
    CclJCI8Nzk1N3t0eHA3LzU3J0lglnZzdzc5NTd2entxfGF8ens3LzU3JyQjcTc5NTd5fHh8Y
    Sc3LzU3Umd0ezVBemBncGc3aA%3D%3D"
}],
"time": 1507533406
}

```

## 2.7 车型查询 - 第七层条件筛选接口

### i. 描述:

经过前六层条件筛选后，进行第七层条件选择。

### ii. 请求地址:

操作	第六次条件选择接口
协议	HTTPS
方式	POST
URL 全径	https://llq.007vin.com/cars/litf

### iii. 请求参数：

序号	参数名	中文名称	必选	数据类型	备注
1	hashid	Token 值	Y	String	-
2	brand	品牌	Y	String	-



3	uid	用户 id	Y	String	-
4	auth	Auth 串	Y	String	从第六层筛选条件中获取

**iv. 返回参数：**

序号	返回值	中文名称	类型	备注
1	code	状态码	Int	0：数据异常 1：成功
2	msg	返回信息	String	-
3	time	返回时间	Int	时间戳
4	title	条件类型名称	String	-
5	data	返回数据	List	List 里包含 dict
参数	code	品牌代号	String	-
	last	是否是最后一层	Boolean	True：是 False：否
	name	条件名称	String	-
	auth	Auth 串	String	-
	keys	链接参数	String	保留参数

**v. 请求 URL 实例：**

https://llq.007vin.com/cars/litf?auth=bjd5fHh8YSY3LzU3UFZQNzk1N3l8eHxhNy81N1MhlzVSZ3R7NUF6YGdwZz0nJSQhOCclJCI8Nzk1N3t0eHA3LzU3J0lglNzdzC5NTd2entxfGF8ens3LzU3JyQjcTc5NTd5fHh8YSc3LzU3Umd0ezVBemBncGc3aA%253D%253D&brand=bmw&hash=e1e8c795e287d36e11804bf2d58c6924&hashid=fd7430a0a7166c5ecb9bb1b1f11c46b&time=1507533762&uid=d592fcb187a1d63a79035f24ec8c85b7

**vi. 返回数据实例：**

```
{
  "msg": "",
  "title": "选择驾驶",
  "code": 1,
```

```

"data": [{
  "code": "bmw",
  "last": false,
  "name": "左驾驶",
  "keys": "",
  "auth":
    "bjd2enFwNy81NyAiJyYnNzk1N2B8cTcvNTc4JiEkLSliJCAAnICQslyAmJyAglzdo"
}, {
  "code": "bmw",
  "last": false,
  "name": "右驾驶 ",
  "keys": "",
  "auth":
    "bjd2enFwNy81NyAiJyYnNzk1N2B8cTcvNTc4JCUgliUgJiwhliMhJiwjly0sLDdo"
}],
"time": 1507533550
}

```

## 2.8 车型查询 - 第八层条件筛选接口：

### i. 描述:

经过前七层条件筛选后，进行第八层条件选择。

### ii. 请求地址:

操作	第六次条件选择接口
协议	HTTPS
方式	POST
URL 全径	https://qpds.007vin.com/cars/litfi

### iii. 请求参数：

序	参数名	中文名称	必选	数据类型	备注
---	-----	------	----	------	----

号					
1	hashid	Token 值	Y	String	-
2	brand	品牌	Y	String	-
3	uid	用户 id	Y	String	-
4	auth	Auth 串	Y	String	从第七层筛选条件中获取

#### iv. 返回参数：

序号	返回值	中文名称	类型	备注
1	code	状态码	Int	0：数据异常 1：成功
2	msg	返回信息	String	-
3	time	返回时间	Int	时间戳
4	title	条件类型名称	String	-
5	data	返回数据	list	List 里包含 dict
参数	code	品牌代号	String	-
	last	是否是最后一层	Boolean	True：是 False：否
	name	条件名称	String	-
	auth	Auth 串	String	-
	keys	链接参数	String	保留参数

## 2.9 车型查询 - 第九层条件筛选接口(最后一层)：

### i. 描述：

经过前八层条件筛选后，进行第九层条件选择。

### ii. 请求地址：

操作	第六次条件选择接口
----	-----------

协议	HTTPS
方式	POST
URL 全径	https://qpds.007vin.com/cars/litsx

### iii. 请求参数：

序号	参数名	中文名称	必选	数据类型	备注
1	hashid	Token 值	Y	String	-
2	brand	品牌	Y	String	-
3	uid	用户 id	Y	String	-
4	auth	Auth 串	Y	String	从第八层筛选条件中获取

### iv. 返回参数：

序号	返回值	中文名称	类型	备注
1	code	状态码	Int	0：数据异常 1：成功
2	msg	返回信息	String	-
3	time	返回时间	Int	时间戳
4	title	条件类型名称	String	-
5	data	返回数据	list	List 里包含 dict
参数	code	品牌代号	String	-
	last	是否是最后一层	Boolean	True：是 False：否
	name	条件名称	String	-
	auth	Auth 串	String	-
	keys	链接参数	String	保留参数

## 2.10 车型查询 - 车辆信息接口

### i. 描述:

车型查询中获取车辆信息。

### ii. 请求地址:

操作	获取车辆信息
协议	HTTPS
方式	POST
URL 全径	https://llq.007vin.com/ppycars/carsinfos

### iii. 请求参数 :

序号	参数名	中文名称	必选	数据类型	备注
1	hashid	Token 值	Y	String	-
2	brand	品牌代号	Y	String	-
3	uid	用户 id	Y	String	-
4	auth	Auth 串	Y	String	从最后一层筛选条件中获取

### iv. 返回参数 :

序号	返回值	中文名称	类型	备注
1	code	状态码	Int	0 : 网络异常 1 : 成功 2 : 无数据 4 : 数据异常
2	msg	返回信息	String	-
3	time	返回时间	Int	时间戳
4	data	返回数据	Dict	-
参数	brand	品牌代号	String	-

	mains	这两信息	String	-
	name	中文名称	String	-
	lfc_log	左上角品牌 图标 url	String	-
	imglogo	车型图片 url	String	-
	code	保留参数(忽 视)		

**v. 请求 URL 实例：**

<https://llq.007vin.com/ppycars/carsinfos?auth=bjd2enFwNy81NyAiJyYnNzk1N2B8cTcvNTc4JiEkLSliJCAnlCQslyAmJyAglzdo&brand=bmw&hash=093e93f25db7e4450e9ccc7de873ff97&hashid=fd7430a0a7166c5ecb9bb1b1f11c46b&time=1507534200&uid=d592fcb187a1d63a79035f24ec8c85b7>

**vi. 返回数据实例：**

```
{
  "msg": "",
  "code": 1,
  "data": {
    "code": 1,
    "name": "2\u7cfb",
    "brand": "bmw",
    "mains": "品牌：宝马车系：2 系 \n 车型：216d\n 序列号：F46 Gran
    Tourer(2014-2017)\n 车身：Gran Tourer\n 市场：ECE\n 驾驶：左驾驶\n",
    "lfc_log": "https://007vin.com/img/bmw.png",
    "imglogo":
    "https://llq.007vin.com/staticimgs/bmw/cars/9b7aa03bfe67a8b122a42ee74c5e5090.j
    pg"
  },
  "time": 1507533988
}
```

## 2.11 车型查询 - 主组列表接口

### i. 描述:

车型查询中获取主组列表。

### ii. 请求地址:

操作	获取主组列表
协议	HTTPS
方式	POST
URL 全径	https://llq.007vin.com/ppycars/group

### iii. 请求参数 :

序号	参数名	中文名称	必选	数据类型	备注
1	hashid	Token 值	Y	String	-
2	brand	品牌代号	Y	String	-
3	uid	用户 id	Y	String	-
4	auth	Auth 串	Y	String	从最后一层筛选条件中获取

### iv. 返回参数 :

序号	返回值	中文名称	数据类型	备注
1	code	状态码	Int	0 : 网络异常 1 : 成功 2 : 无数据
2	msg	返回信息	String	-
3	time	返回时间	Int	时间戳
4	data	返回数据	List	List 里包含 dict
参数	brand	品牌代号	String	-
	img	主组图片 url	String	-

	name	主组名称	String	保留字段
	groupname	主组名称	String	-
	groupnum	主组号	String	-
	auth	Auth 串	String	

**v. 请求 URL 实例：**

https://llq.007vin.com/ppycars/group?auth=bjd2enFwNy81NyAiJyYnNzk1N2B8cTcvNTc4JiEkLSliJCAnlCQslyAmJyAglzdo&brand=bmw&hash=d05f4e500b4d3b97c653e47cea0028a5&hashid=fdf7430a0a7166c5ecb9bb1b1f11c46b&time=1507534670&uid=d592fcb187a1d63a79035f24ec8c85b7

**vi. 返回数据实例：**

```
{
  "msg": "",
  "code": 1,
  "data": [{
    "name": "01 技术文献",
    "img":
      "https://data.007vin.com/stcimgs/img/2e0804f99852ce93d5e2469e8a0f8edc.jpg",
    "brand": "bmw",
    "auth":
      "bjd2enFwNy81NyAiJyYnNzk1N3tgeDcvNTcJJdc5NTdggfHE3LzU3OCYhJC0iliQgJyAkL
      CMgJicglCM3OTU3dmFsZXA3LzU3diUILDdo",
    "groupname": "01 技术文献",
    "groupnum": "01"
  }],
  "time": 1507534457
}
```



## 2.12 车型查询 - 分组列表接口

### i. 描述:

车型查询中获取分组信息

### ii. 请求地址:

操作	获取分组列表
协议	HTTPS
方式	POST
URL 全径	https://llq.007vin.com/ppycars/subgroup

### iii. 请求参数 :

序号	参数名	中文名称	必选	数据类型	备注
1	hashid	Token 值	Y	String	-
2	brand	品牌	Y	String	-
3	uid	用户 id	Y	String	-
4	auth	Auth 串	Y	String	从分组列表中获取

### iv. 返回参数 :

序号	返回值	中文名称	数据类型	备注
1	code	状态码	int	0 : 网络异常 1 : 成功 2 : 无数据
2	msg	返回信息	String	-
3	time	返回时间	Int	时间戳
4	data	返回数据	List	List 里包含 dict
参数	base_brandCode	厂商代码	String	如 MINI 的厂商代码为 bmw
	uid	标题 id	String	保留字段
	url	分组图片信息	String	-
	mid	分组号	String	-

subgroupname	分组名称	String	-
num	主组号	String	-
auth	Aut 串	String	-
has_subs	是否有子组	Boolean	只有路虎拥有子组
model	型号	String	-
subgroup	分组号	String	-
colorvalue	过滤颜色标识	String	此接口中，colorvalue = 1
description	描述		-

#### v. 请求 URL 实例：

https://llq.007vin.com/ppycars/subgroup?auth=bjd2enFwNy81NyAiJyYnNzk1N3tgeD  
cvNTclJDc5NTdggfHE3LzU3OCYhJC0iliQgJyAkLCMgJicgICM3OTU3dmFsZXA3LzU3diUI  
LDdo&brand=bmw&hash=9797b71217ea2dfdf42f4bd20034b898&hashid=fd7430a  
0a7166c5ecb9bb1b1f11c46b&time=1507535091&uid=d592fcb187a1d63a79035f24e  
c8c85b7

#### vi. 返回数据实例：

```
{
  "msg": "",
  "code": 1,
  "data": [{"base_brandCode": "bmw",
    "uid": "63ff0848b39be1204e8aea4f03f40b7c",
    "url":
      "https://data.007vin.com/app/tmbnail?code=bmw&key=bjd2enFwNy81NyAiJyYnNzk1N3tgeDcvNTclJDc5NTdggfHE3LzU3OCYhJC0iliQgJyAkLCMgJicgICM3OTU3ZmB3cmd6YGV7dHhwNy81NyUkSiQILSVJYCMmcSdJYCwtliBJYC13dy1JYCAmCHNJYCMgLSJJYC  
MiJ3ZXWEI3OTU3fGZKc3x5YXBnNy81JTk1N3dndHtxNy81N3d4Yjc5NTdmYHdyZ3pg  
ZTcvNTclJEokJS0lNzk1N3h8cTcvNTchlictSA3OTU3dmFsZXA3LzU3diUILDc5NTd8Zm  
Zgd2Y3LzUlaA%3D%3D&w=160&h=160",
    "mid": "01_1080",
    "subgroupname": "01_1080 插页 许可文本 BMW",
```

```

    "num": "01",

    "auth":

    "bjd2enFwNy81NyAiJyYnNzk1N3tgeDcvNTclJDC5NTdggfHE3LzU3OCYhJC0iliQgJyAkL
    CMgJicglCM3OTU3ZmB3cmd6YGV7dHhwNy81NyUkSiQILSVJYCMmcSdJYCwtliBJYC1
    3dy1JYCAmCHNJYCMgLSJJYCMiJ3ZXWEI3OTU3fGZKc3x5YXBnNy81JTk1N3dndHtxNy
    81N3d4Yjc5NTdmYHdyZ3pgZTcvNTclJEokJS0INzk1N3h8cTcvNTchlictISA3OTU3dmFs
    ZXA3LzU3diUldc5NTd8ZmZgd2Y3LzUlaA%3D%3D",

    "has_subs": false,

    "model": "",

    "subgroup": "05",

    "colorvalue": "1",

    "description": ""

  },

],

  "time": 1507534457

}

```

## 2.13 车型查询 - 零件列表

### i. 描述:

在车型查询中，获取特定分组下面的零件列表，不存在过滤。

### ii. 请求地址:

操作	车型查询中获取零件列表
协议	HTTPS
方式	POST
URL 全径	https://llq.007vin.com/ppycars/parts

### iii. 请求参数：

序号	参数名	中文名称	必	数据类型	备注
----	-----	------	---	------	----

			选		
1	hashid	Token 值	Y	String	-
2	uid	用户 id	Y	String	当用户没有登录时，默认的 uid 为： 5232e6c0e7618e47e39db82a0eaead9a
3	brand	品牌代号	Y	String	-
4	auth	Auth 串	Y	String	从分组列表接口中获取

#### iv. 返回参数：

序号	返回值	中文名称	类型	备注
1	code	状态码	Int	1：成功 2：无数据
2	msg	返回信息	String	-
3	time	返回时间	Timestamp	-
4	auth	Auth 串	Int	-
5	label	分组详情 页面标题	String	-
6	data	返回数据	List	List 中包含 dict
参数	remark	说明	String	-
	modelName	型号名称	String	保留字段
	pid	零件号	String	在零件价格、替换件使用该参数
	itid	描点位置	String	与零件序号(num)有对应关系
	detail	详情	String	-
	multi_price	是否是多 价格	Int	0：不是 1：是
	label	零件名称	String	-
	step	零件位置 标记	Int	保留字段
	num	零件序号	String	-
	is_filter	是否过滤	Int	0: 未过滤

				1: 过滤
	price	渠道价格	String	-
	isreplace	是否有替换件	Int	0 : 否 1 : 是
	real_pid	真实零件号	String	在组件信息中使用该参数
	model	型号	String	-
	quantity	数量	String	-

### 3 零件号查询相关接口

#### 3.1 零件号查询 - 零件号查询历史接口

##### i. 描述:

零件号查询历史获取

##### ii. 请求地址:

操作	零件号查询历史
协议	HTTPS
方式	POST
URL 全径	https://llq.007vin.com/search/parts

##### iii. 请求参数 :

序号	参数名	中文名称	必选	数据类型	备注
1	hashid	Token 值	Y	String	-
2	uid	用户 id	Y	String	-
3	part	零件号	Y	String	进行模糊匹配, 可为空

##### iv. 返回参数 :

序号	返回值	中文名称	类型	备注
1	code	状态码	Int	1: 成功
2	msg	返回信息	String	-
3	time	返回时间	Int	-
4	data	返回数据	List	List 中包含 dict
参数	part	零件号	String	-
	brandname	品牌名称	String	-

##### v. 请求 URL 实例 :

https://llq.007vin.com/search/parts?hash=00863ed4b7ba704cbc1ac71dd92acc8c&hashid=fdf7430a0a7166c5ecb9bb1b1f11c46b&part=&time=1507536584&uid=d592fcb

187a1d63a79035f24ec8c85b7

#### vi. 返回数据实例：

```
{
  "msg": "获取数据成功",
  "code": 1,
  "data": [{
    "part": "11537521049",
    "brandname": "宝马"
  }]
  "time": 1507533218
}
```

### 3.2 零件号查询 – 确认零件号是否存在和确认品牌：

#### i. 描述：

查询零件号时，只有当没有选择品牌的时候才请求该接口，否则直接请求零件列表接口。该接口返回结果存在以下两种结果：

- (a) 不存在此零件。
- (b) 存在此零件，无法确认品牌。需进行品牌选择。

#### ii. 请求地址：

操作	获取零件列表
协议	HTTPS
方式	POST
URL 全径	https://qpds.007vin.com/ppypart/parts_brand_determination

#### iii. 请求参数：

序号	参数名	中文名称	必选	数据类型	备注

1	hashid	Token 值	Y	String	-
2	uid	用户 id	Y	String	-
3	pid	零件号	Y	String	-

**iv. 返回参数(情景 1) – 无法确认品牌，code 为 6 时：**

序号	返回值	中文名称	类型	备注
1	code	状态码	Int	0：找不到该零件号信息 1：成功获取该零件号的品牌等信息 6：无法确认该零件品牌
2	msg	返回信息	String	-
3	time	返回时间	Timestamp	-
4	brand	品牌号	String	-
5	img	品牌图片 url	String	无法确认品牌时为空
7	data	返回数据	list	List 中包含 dict
参数	brand	品牌代号	String	-
	img	品牌图片 url	String	-
	name	品牌中文名称	String	-

**v. 返回参数(情景 2) – 无法确认品牌，code 为 0 和 1 时：**

序号	返回值	中文名称	类型	备注
1	code	状态码	Int	0：找不到该零件号信息 1：成功获取该零件号的品牌等信息 6：无法确认该零件品牌
2	msg	返回信息	String	-
3	time	返回时间	Timestamp	-
6	data	返回数据	list	List 中包含 dict
参数	brand	品牌代号	String	-



has_compt	是否有试用 车型(此参数 暂时没用)	String	1：有 0：没有
has_replace	是否有替换 件(此参数暂 时没用)	String	1：有 0：没有
label	零件名称	String	-
pid	零件号	String	-
price	价格(此参数 暂时没用)	String	-
remark	备注(此参数 暂时没用)	String	-
s_pid	零件号(参数 暂时没用)	String	-
status	状态值	String	1：获取数据成功 0：没有该零件

### 3.3 零件号查询 – 获取零件列表(2017-12-27 版本)：

#### i. 描述：

获取该零件号的查询列表

#### ii. 请求地址：

操作	获取零件列表
协议	HTTPS
方式	POST
URL 全径	https://qpds.007vin.com/ppypart/parts_list

#### iii. 请求参数：

序	参数名	中文名称	必选	数据类型	备注
---	-----	------	----	------	----

号					
1	hashid	Token 值	Y	String	-
2	uid	用户 id	Y	String	-
3	pid	零件号	Y	String	-
4	brand	品牌代号	Y	String	-

## iv. 返回参数：

序号	返回值	中文名称	类型	备注
1	code	状态码	Int	0：找不到该零件号信息 1：成功获取该零件号的 品牌等信息 6：无法确认该零件品牌
2	msg	返回信息	String	-
3	time	返回时间	Timestamp	-
4	data	返回数据	list	List 中包含 dict
参数	brand	品牌代号	String	-
	has_compt	是否有试用 车型(此参数 暂时没用)	String	1：有 0：没有
	has_replace	是否有替换 件(此参数暂 时没用)	String	1：有 0：没有
	label	零件名称	String	-
	pid	零件号	String	-
	price	价格(此参数 暂时没用)	String	-
	remark	备注(此参数 暂时没用)	String	-
	s_pid	输入的零件 号	String	-
	status	状态值	String	1：获取数据成功

				0: 没有该零件
--	--	--	--	----------

### 3.4 零件号查询 – 获取基础信息(包含技术信息)

#### i. 描述：

获取零件基础信息，包括该零件包含哪些其他信息，如是否含有渠道价格，替换件等信息

#### ii. 请求地址:

操作	获取零件列表
协议	HTTPS
方式	POST
URL 全径	https://qpds.007vin.com/ppypart/parts_baseinfo

#### iii. 请求参数：

序号	参数名	中文名称	必选	数据类型	备注
1	hashid	Token 值	Y	String	-
2	uid	用户 id	Y	String	-
3	pid	零件号	Y	String	-
4	brand	品牌代号	Y	String	-

#### iv. 返回参数：

序号	返回值	中文名称	类型	备注
1	code	状态码	Int	0：找不到该零件号信息 1：成功获取该零件号的品牌等信息 6：无法确认该零件品牌
2	msg	返回信息	String	-
3	time	返回时间	Timestamp	-
4	headname	零件包含信息	List	此数组里的为该零件所含有数

				据的标题
5	img	品牌图片 url	list	-
6	pid	零件号	String	
7	data	保留字段	list	List 中包含 dict
参数	model	型号	String	-
	num	数量	String	-
	label	零件名称	String	-
	pid	零件号	String	-
	prices	价格	String	-
	remark	备注	String	-
8	partdetail	零件详情	List	里面包含 dict
	key	标题	String	-
	value	数据	string	-
9	showmessage	技术信息内容	List	-

### 3.5 零件号查询 – 渠道价格

#### i. 描述:

获取零件渠道价格

#### ii. 请求地址:

操作	获取零件列表
协议	HTTPS
方式	POST
URL 全径	https://qpds.007vin.com/ppypart/parts_price

#### iii. 请求参数 :

序号	参数名	中文名称	必选	数据类型	备注
----	-----	------	----	------	----

1	hashid	Token 值	Y	String	-
2	uid	用户 id	Y	String	-
3	pid	零件号	Y	String	-
4	brand	品牌代号	Y	String	-

**iv. 返回参数：**

序号	返回值	中文名称	类型	备注
1	code	状态码	Int	0：找不到该零件号信息 1：成功获取该零件号的品牌等信息 6：无法确认该零件品牌
2	msg	返回信息	String	-
3	time	返回时间	Timestamp	-
7	data	数据	list	List 中包含 dict
参数	data	子数据	list	List 中包含 dict
	amount	库存	String	-
	channel	来源	String	-
	cost_price	保留字段	String	-
	eot_price	不含税进价	String	-
	factory_type	厂家类型	String	1：原厂件
	location	地区	String	-
	mill	厂商	string	-
	origin	保留字段	String	-
	parttype	零件类型	String	-
	pid	零件号	String	-
	prices	销售价	String	-
	remark	说明	String	-
	supplier	服务商	String	-
	is_original_part	是否是原厂	String	1：是

		件		0：不是
	title	零件所属类型	String	如：原厂件、拆车件等

### 3.6 零件号查询 – 获取替换件

#### i. 描述:

获取零件替换件信息

#### ii. 请求地址:

操作	获取零件列表
协议	HTTPS
方式	POST
URL 全径	https://qpds.007vin.com/ppypart/parts_replacement

#### iii. 请求参数：

序号	参数名	中文名称	必选	数据类型	备注
1	hashid	Token 值	Y	String	-
2	uid	用户 id	Y	String	-
3	pid	零件号	Y	String	-
4	brand	品牌代号	Y	String	-

#### iv. 返回参数：

序号	返回值	中文名称	类型	备注
1	code	状态码	Int	0：找不到该零件号信息 1：成功获取该零件号的品牌等信息 6：无法确认该零件品牌
2	msg	返回信息	String	-

3	time	返回时间	Timestamp	-
4	data	数据	list	List 中包含 dict
参数	advise	保留字段	String	-
	brandcn	品牌中文名称	String	-
	counts	件数	String	-
	haschild	是否有下层结构	String	1：是 0：不是
	imgs	图片 url	list	-
	islast	是否是最后一个零件	String	1：是 0：不是
	lable	零件名称	String	-
	level	层级	String	-
	num	当前 num	String	-
	parentnum	父类 num	String	-
	pid	零件号	String	-
	prices	参考价格	String	-
	ptype	型号	String	-
	ridbrandCode	厂商品牌代码	String	-

### 3.7 零件号查询 – 获取适用车型

#### i. 描述:

获取零件的适用车型

#### ii. 请求地址:

操作	获取零件列表
协议	HTTPS
方式	POST
URL 全径	https://qpds.007vin.com/ppypart/parts_compatible_vehicle

## iii. 请求参数：

序号	参数名	中文名称	必选	数据类型	备注
1	hashid	Token 值	Y	String	-
2	uid	用户 id	Y	String	-
3	pid	零件号	Y	String	-
4	brand	品牌代号	Y	String	-
5	page	页码	Y	String	从第 0 页开始

## iv. 返回参数：

序号	返回值	中文名称	类型	备注
1	code	状态码	Int	0：找不到该零件号信息 1：成功获取该零件号的牌等信息 6：无法确认该零件品牌
2	msg	返回信息	String	-
3	time	返回时间	Timestamp	-
4	last	是否是最 后一页	String	1：是 0：不是
5	data	数据	list	List 中包含 dict
参 数	cars_model	车辆型号	String	-
	group_name	零件组	String	-
	main_group_name	主组名称	String	-
	market	市场	String	-
	uri_param	请求结构 图的参数	dict	-
	auth	Auth 串	String	-
	code	品牌代号	String	-
	p	零件号	String	-



	url	保留字段	String	-
	year	年份	String	-

### 3.8 零件号查询 – 获取组件

#### i. 描述:

获取组件信息

#### ii. 请求地址:

操作	获取零件列表
协议	HTTPS
方式	POST
URL 全径	https://qpds.007vin.com/ppypart/parts_component

#### iii. 请求参数 :

序号	参数名	中文名称	必选	数据类型	备注
1	hashid	Token 值	Y	String	-
2	uid	用户 id	Y	String	-
3	pid	零件号	Y	String	-
4	brand	品牌代号	Y	String	-

#### iv. 返回参数 :

序号	返回值	中文名称	类型	备注
1	code	状态码	Int	0 : 找不到该零件号信息 1 : 成功获取该零件号的品牌等信息 6 : 无法确认该零件品牌
2	msg	返回信息	String	-
3	time	返回时间	Timestamp	-

4	data	数据	list	List 中包含 dict
参数	id	位置	String	-
	label	组件名称	String	-
	model	备注	String	-
	num	件数	String	-
	pid	零件号	String	-
	remark	型号	String	-

### 3.9 零件号查询 – 获取品牌件

#### i. 描述:

获取品牌件信息

#### ii. 请求地址:

操作	获取零件列表
协议	HTTPS
方式	POST
URL 全径	https://qpds.007vin.com/ppypart/parts_article

#### iii. 请求参数 :

序号	参数名	中文名称	必选	数据类型	备注
1	hashid	Token 值	Y	String	-
2	uid	用户 id	Y	String	-
3	pid	零件号	Y	String	-
4	brand	品牌代号	Y	String	-

#### iv. 返回参数 :

序号	返回值	中文名称	类型	备注
----	-----	------	----	----

1	code	状态码	Int	0：找不到该零件号信息 1：成功获取该零件号的品牌等信息 6：无法确认该零件品牌
2	msg	返回信息	String	-
3	time	返回时间	Timestamp	-
4	data	数据	list	List 中包含 dict
参数	advise	注释	String	-
	brandcn	品牌名称	String	-
	counts	件数	String	-
	imgs	图片 url	list	取第一个
	lable	零件名称	String	-
	num	当前 num	String	-
	pid	品牌编号	String	-
	prices	参考价格	String	-
	ptype	型号	String	-
	supplier_img_url	品牌图片 url	String	-