

第一章 接口操作说明

1.1 接口说明

1.1.1 接口规范

接口根地址：http://设备 ip 地址:8090/

接口形式：通过 HTTP 请求的方式对外提供服务。

接口安全：初次调用接口需要先**设置设备密码**（pass），后续调用任何接口都需要传入设备密码（pass）作为接口安全校验秘钥。

1.1.2 接口返回

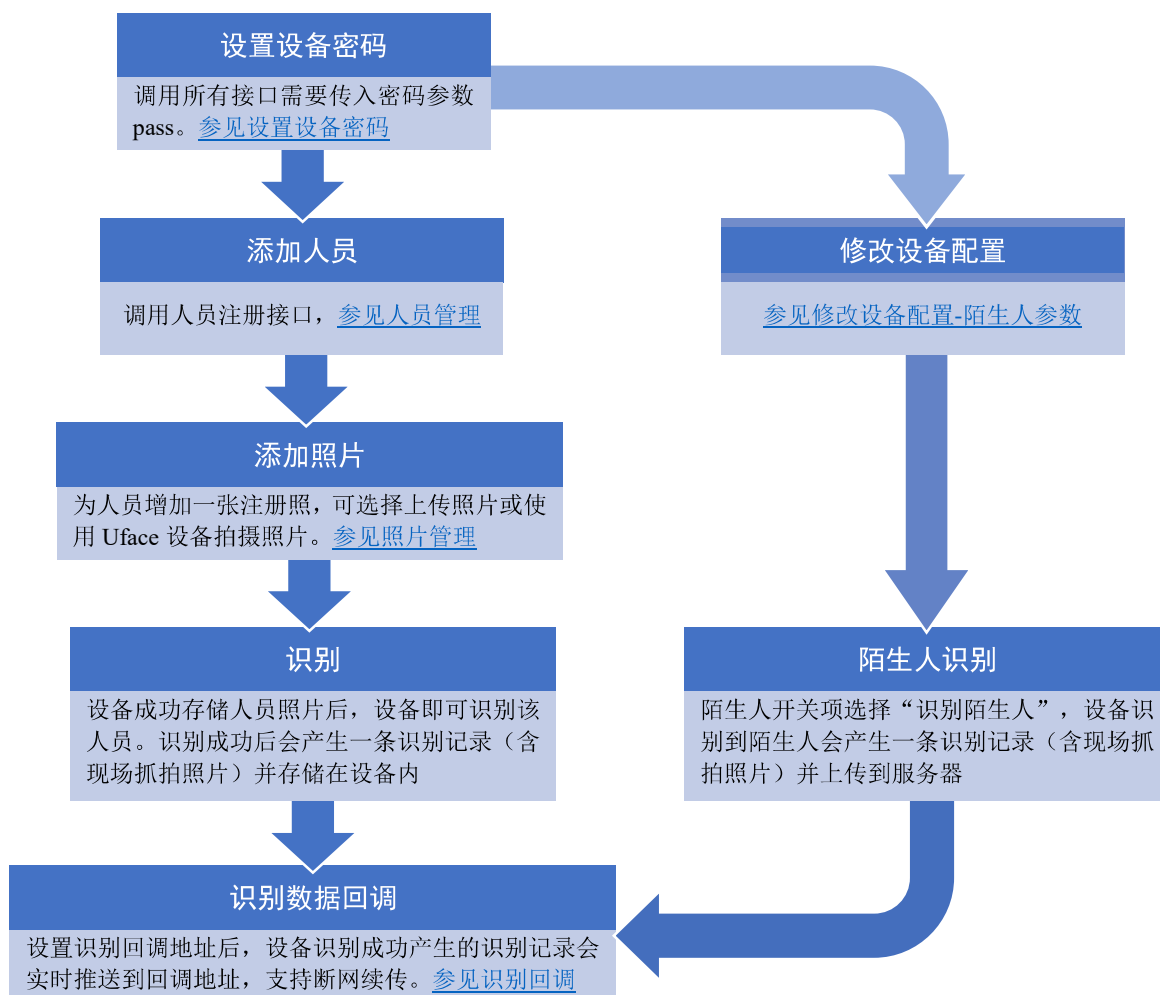
接口通用返回说明：

```
public class ResultInfo<T> {  
    private Int result;//表示接口是否调通，1 成功，0 失败，通常只要设备服务器能响应，  
    该值均为 1  
    private Boolean success;//此次操作是否成功，成功为 true，失败为 false  
    private T data;//接口返回的业务数据，类型可为数值、字符串或集合等  
    private String msg;//接口返回的信息，通常是错误类型码的原因信息  
}
```

文档中所涉及到的接口返回示例，个别接口的返回数据会有略微调整，须以真实的返回结果为准。

1.1.3 接口调用流程

首次使用设备：设备开机后进入识别主界面。设备未调通过接口，设备无初始密码。操作流程见流程图。

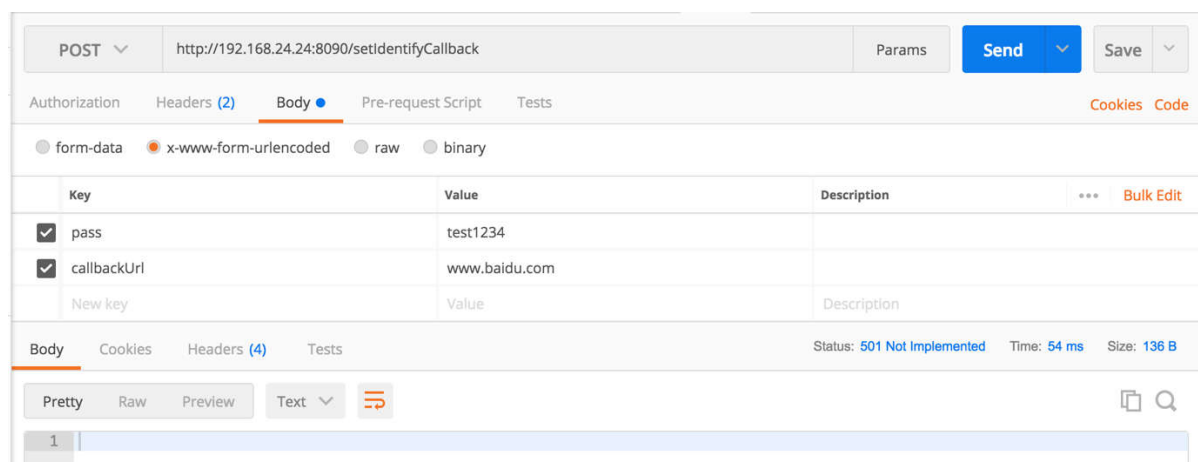


- 设备初始无密码，调用所有接口都需要传入密码参数；在设置设备密码接口，newPass 和 oldPass 传入一样的值，即成为设备密码
- 照片注册成功后，该注册照片会保存在设备内；可通过照片查询接口获得人员注册照片 url，访问 url 的照片时，设备需连网在线，且客户端与设备处于同一网段
- 人脸识别成功后，会显示注册人员的姓名；若填写了卡号，可直接刷对应卡号的卡进行识别，识别成功后会显示与该卡号对应的注册人员姓名

- 若设置了回调地址，人员识别成功后会向回调地址 POST 字段 ip、personId、time（识别时间）、deviceKey 和 type: face/card_0/1；若设置了公网回调地址，识别时设备离线或设备连局域网，则设备连公网后识别记录会上传到公网回调地址

1.1.4 注意事项

- 1、调用设备接口时，请勿同时在其他客户端服务器上调用同一台设备的接口。
- 2、若调用接口返回提示“参数异常”，请按以下步骤检查：
 - a) 检查参数名称是否拼错、参数名称中是否带有空格或回车
 - b) 检查参数值是否符合规范，如 Int 值未定义、id 含有数字字母以外的非法字符等
 - c) Json 中是否含有空格（直接使用接口返回的 Json 字符串需要去空格），是否多了或少了"，请仔细核对每个接口的 postman 示例
- 3、若调用接口返回为空，表示 url 错误（IP 有误、拼写错误、字段遗漏等），如图所示。下图中 url 大小写有误。



- 4、若调用接口返回 Could not get any response，可能原因有：IP 地址错误、端口号错误或遗漏、传参形式错误（POST 请求参数放在 body 里，格式为 x-www-urlencoded）等。

第二章 接口操作

2.1 设备管理类接口

2.1.1 设置设备密码

请求数据

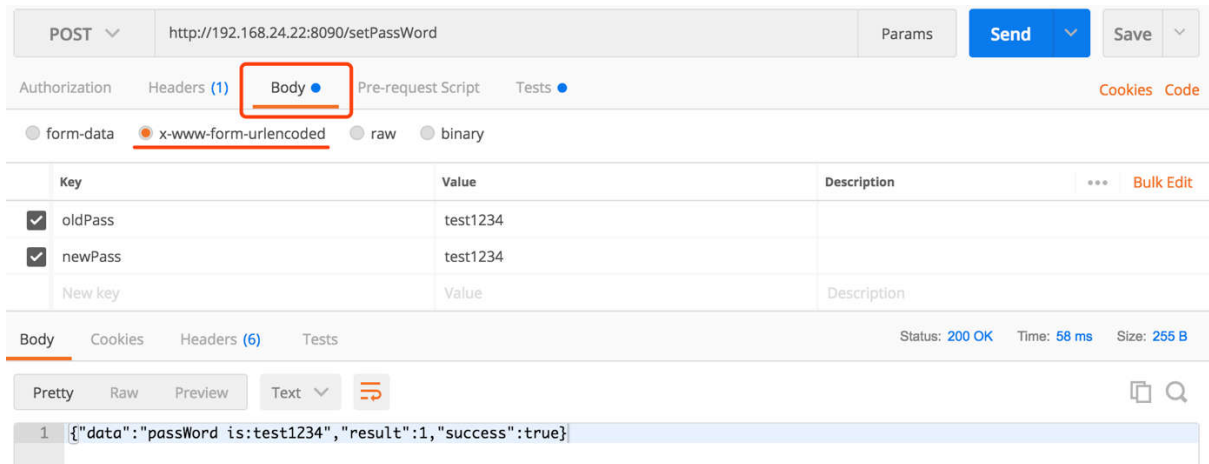
Method	URL			
POST	http://设备 IP:8090/setPassWord			
参数名	描述	类型	必传	附加说明
oldPass	旧密码	String	Y	<ul style="list-style-type: none">• 新设备或重置后（恢复初始化）的设备，调用其他接口前，需要先进行初始密码设置，oldPass 和 newPass 传入一样的值即可• 修改密码时，分别传入新、旧密码• 此接口调用不需要传入 pass 密码不可为空或空格
newPass	新密码	String	Y	

设置初始密码示例

 注意：a) POST 请求参数放在 body 里

b) ContentType: application/x-www-form-urlencoded

c) 本章所有示例均使用 postman




返回示例

```
{
  "data": "passWord is: 12345", //设备密码，也称接口调用密码，请妥善保管。若忘记密码，则需
    重置设备，设备会清空所有数据
  "result": 1, //接口调通
  "success": true //设备密码设置成功
}
```

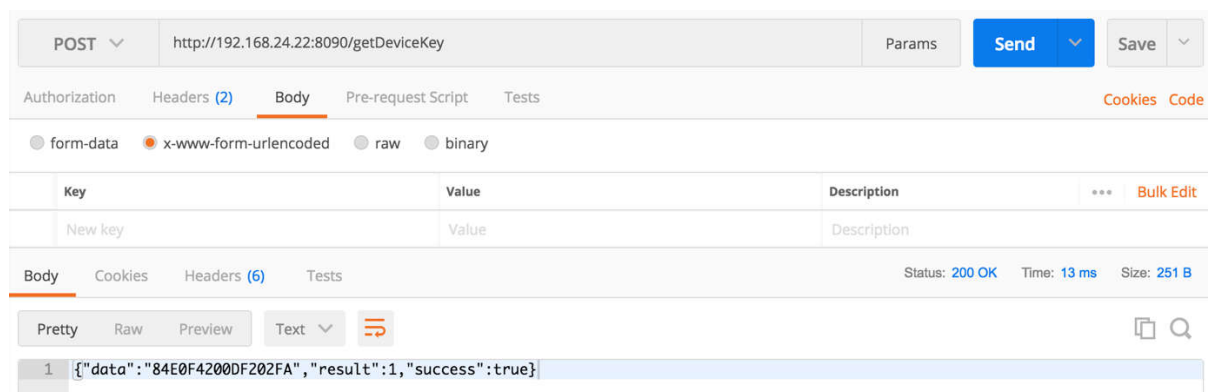
2.1.2 设备序列号获取

请求数据

Method	URL
POST	http://设备 IP:8090/getDeviceKey

 注意：调用该接口不需要传入参数，不需要传入 pass

postman 示例



返回示例

```
{
  "data": "84E0F4200DF202FA", //设备序列号，设备出厂唯一标识码
  "result": 1, //接口调通
  "success": true //成功获取设备序列号
}
```

2.1.3 识别回调

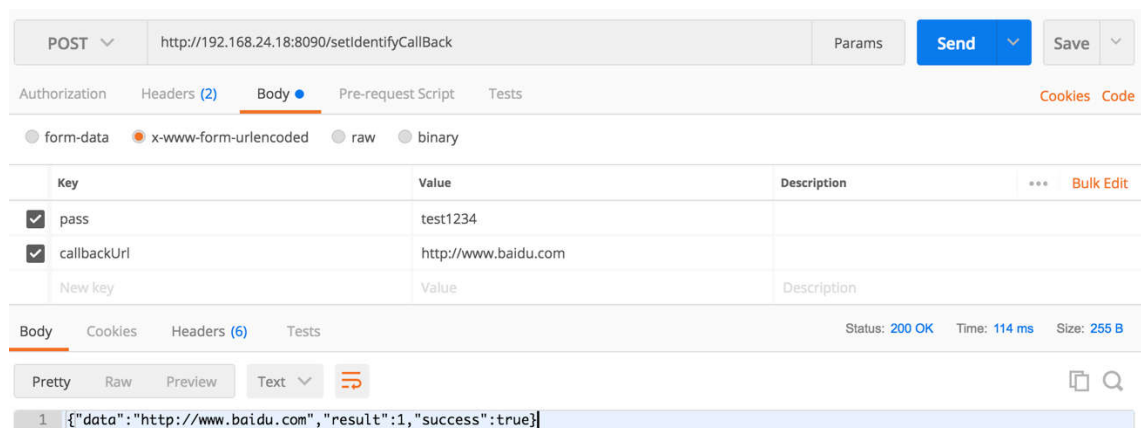
请求数据

Method	URL			
POST	http://设备 IP:8090/setIdentifyCallBack			
参数名	描述	类型	必传	附加说明
pass	设备密码	String	Y	
callbackUrl	回调地址	String	Y	<ul style="list-style-type: none"> 设备成功识别人员后，会向该地址 POST 字段 ip、personId、time（识别时间）、deviceKey、type: face/card_0/1/2 和 path，参见回调字段说明 传入内容为空可以清空回调地址，清空后识别成功将不再进行回调 回调 URL 需要符合正则表达式: String check = <code>"((http ftp https):/)/([a-zA-Z0-9\\._-]+\\.[a-zA-Z]{2,6}) ([0-9]{1,3}\\.[0-9]{1,3}\\.[0-9]{1,3}\\.[0-9]{1,3}) ([0-9]{1,4})*/([a-zA-Z0-9\\&%_\\./~])*?"</code>

i 注意：a) 设备配置接口设置 saveIdentifyTime（识别记录时间窗）后，在该时间间隔内，同一个人多次识别只会向回调地址 POST 一条识别记录

b) URL 端口说明：若使用页面作为回调地址，端口号须为 8090；若使用 API 作为回调地址，端口号最大为 9999，且路由器对该端口号开放。

postman 示例



2.3 人员管理类接口

2.3.1 人员注册

请求数据

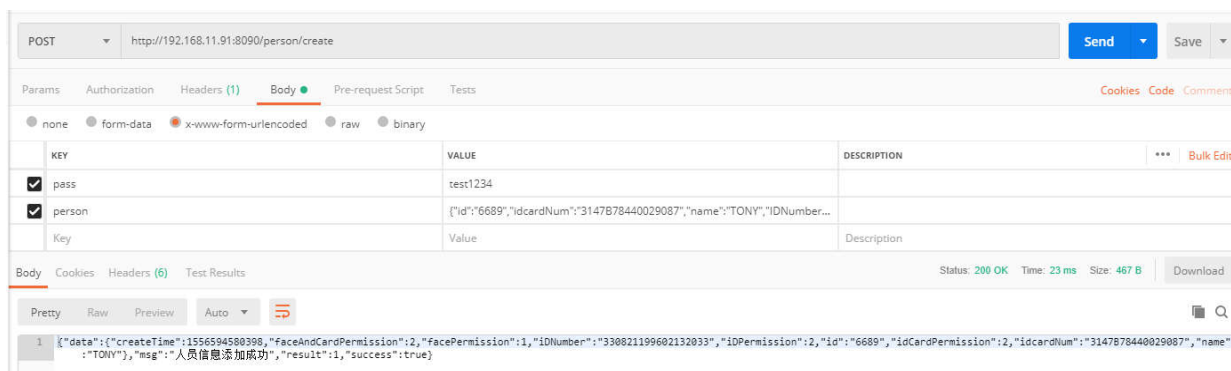
Method	URL			
POST	http://设备 IP:8090/person/create			
参数名	描述	类型	必传	附加说明
pass	设备密码	String	Y	
person	人员信息集合	Json	Y	<ul style="list-style-type: none"> Json 示例: <pre>{ "id": "6689", "idcardNum": "3147B78440029087", "name": "TONY", "IDNumber": "330821199602132033", "facePermission": "1", "idCardPermission": "2", "faceAndCardPermission": "2", "IDPermission": "2" }</pre> id、idcardNum、name、facePermission、idCardPermission、faceAndCardPermission、IDPermission 内容可传空。 若传入 id 值, id 只允许数字和英文字母, 区分大小写, 位数限制为 0-255 位。

postman 示例

 注意: a) POST 请求参数放在 body 里


b) ContentType: application/x-www-form-urlencoded

c) 本章所有示例均使用 postman



返回示例

```
{
  "data": {
    "createTime": "1532398934472", //String, 人员创建时的时间戳（毫秒级），以设备时间为准
    "id": "abc", //String, 人员 id, id 为人员标识，限制 255 位。注册人员时，若 id 不填，则系统会自动生成 32 位 id 并返回；若填了 id，系统会以此 id 为主键存入本地数据库中；若 id 重复，会报错
    "idcardNum": "410822199908221428", //String, 卡号，注册时可以不填，长度无限制。若注册人员时填写了卡号，可直接刷对应卡号的卡进行识别，屏幕也会显示与该卡号对应的人员的名字
    "name": "郝雨凡", //String, 姓名，不可为空。识别成功后会在屏幕上显示该名字
    "IDNumber": "33082119920811782", //String, 身份证编号。
    "facePermission": 2, //刷脸权限，默认为 2 开。
    "idCardPermission": 2, //刷卡权限，默认为 2 开。
    "faceAndCardPermission": 2, //人卡权限，默认为 2 开。
    "IDPermission": 2, //人证比对权限 默认为 2 开。
  },
  "msg": "人员信息添加成功",
  "result": 1, //接口调通
  "success": true //人员注册成功
}
```

 注意：姓名、卡号、id 注册成功，实际上是人员信息写入设备数据库。若使用设备读取卡号进行注册，实际上是先读取卡号，再将卡号写入数据库，因此与注册人员时直接填写卡号效果一样。

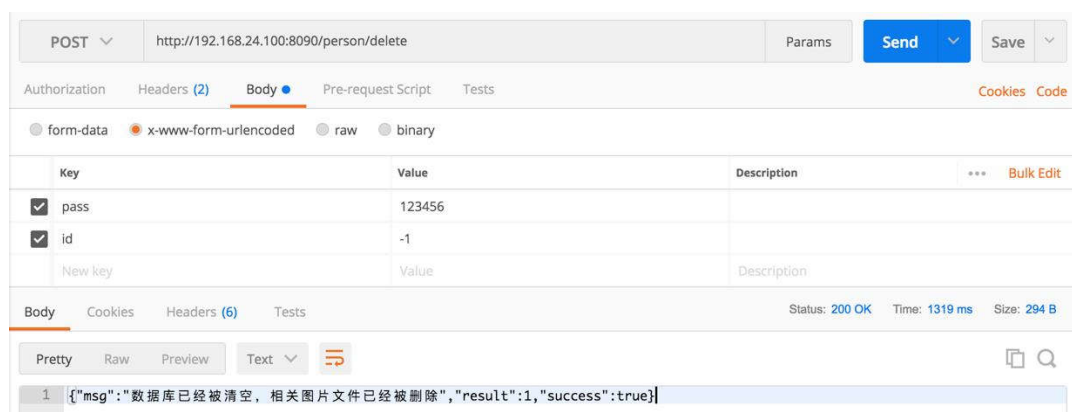
2.3.2 人员删除（批量）

请求数据

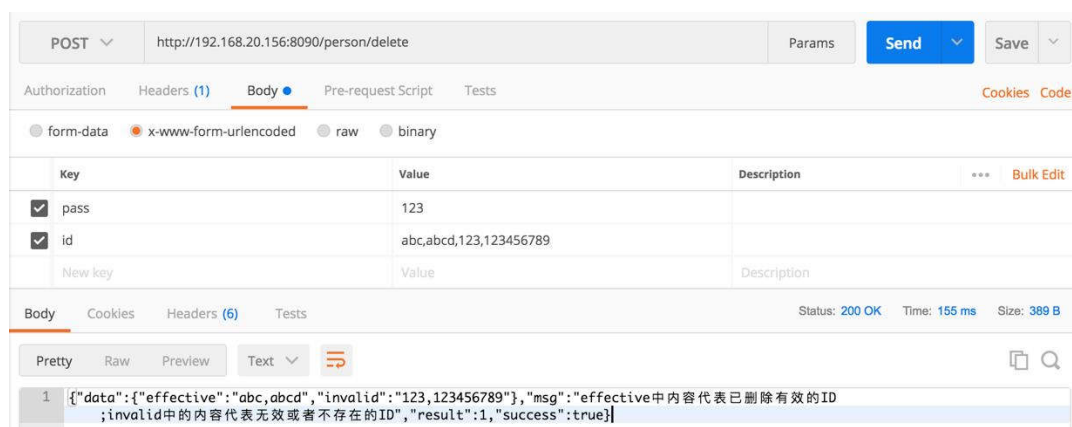
Method	URL			
POST	http://设备 IP:8090/person/delete			
参数名	描述	类型	必传	附加说明
pass	设备密码	String	Y	
id	人员 id	String	Y	<ul style="list-style-type: none"> 删除多个人员，personId 用英文逗号拼接 传入-1 则删除所有人员 以下数据会被删除，本地将不再做存储： <ul style="list-style-type: none"> 该人员 id、人员信息 该人员对应的识别记录、现场抓拍照片 该人员对应的注册照片

postman 示例

(1) id 传入-1，删除设备上的所有人员，数据不可恢复



(2) id 传入多个人员，用英文逗号拼接；返回结果区分已删除的 id 和无效 id。



返回示例


```
{
  "data": {
    "effective": "abc,abcd",
    "invalid": "123,123456789"
  },
  "msg": "effective 中内容代表已删除有效的 ID;invalid 中的内容代表无效或者不存在的 ID",
  "result": 1,//接口调通
  "success": true//已删除有效的人员 id
}
```

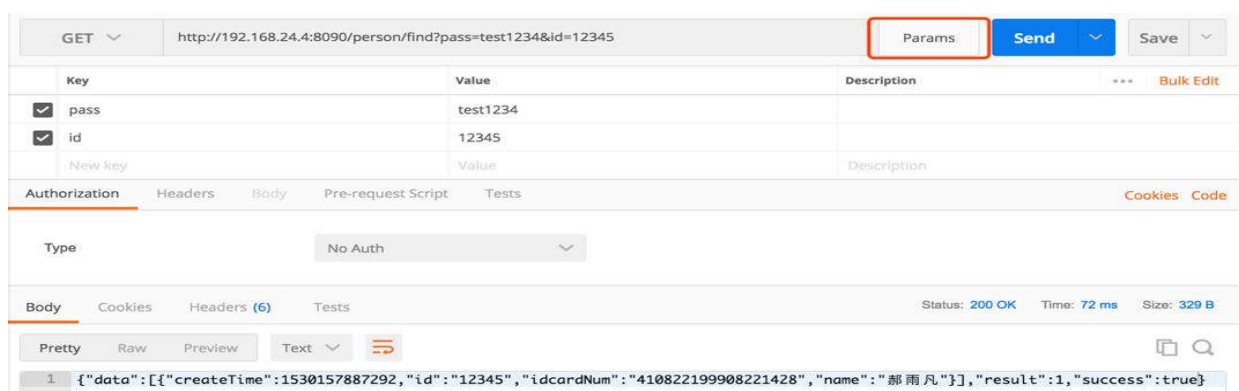
2.3.3 人员信息查询

请求数据

Method	URL			
GET	http://设备 IP:8090/person/find			
参数名	描述	类型	必传	附加说明
pass	设备密码	String	Y	
id	人员 id	String	Y	<ul style="list-style-type: none"> 查询指定 id 的人员信息 id 传入-1 为查询设备上所有人员信息

postman 示例

 注意：此接口与人员分页查询接口功能相似，建议使用[人员分页查询接口](#)，避免设




备注注册人数较多时查询导致设备崩溃。

2.4 照片管理类接口

2.4.1 照片注册（base64）

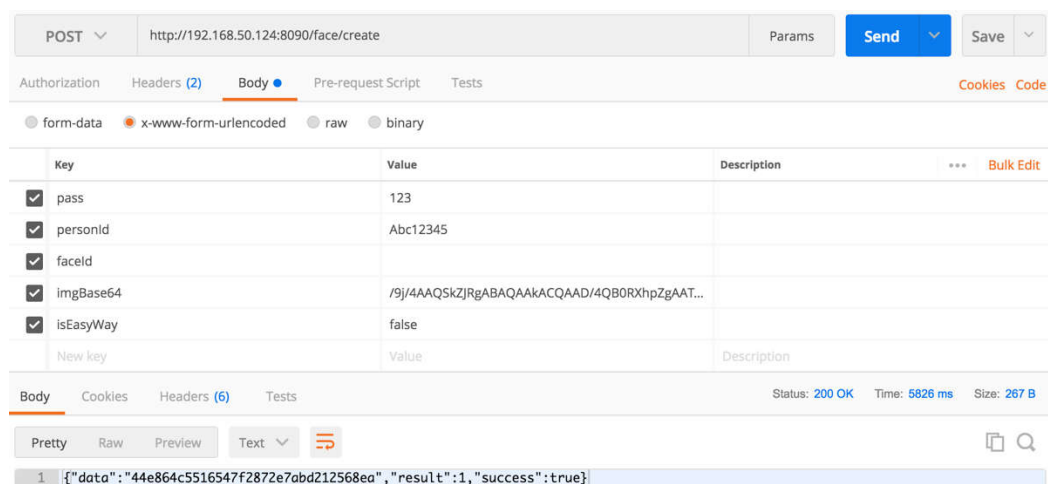
请求数据

Method	URL			
POST	http://设备 IP:8090/face/create			
参数名	描述	类型	必传	附加说明
pass	设备密码	String	Y	
personId	人员 id	String	Y	<ul style="list-style-type: none"> 用于标识该照片属于某个人 id 必须先注册人员，才能添加照片
faceId	照片 id	String	Y	若 faceId 传入内容为空，则系统会自动生成一个 32 位的 faceId 并在照片注册成功后返回
imgBase64	照片的 base64 编码字符串	String	Y	<ul style="list-style-type: none"> 不加头部，如：data:image/jpg;base64, 图片格式支持 png、jpg、jpeg
isEasyWay	选择宽松 or 严格照片注册方式	Boolean	N	非必传，默认 false：严格检测照片质量；true：宽松检测照片质量

 注意：a) 照片注册成功后，设备会将该照片保存到设备中，方便外部后续对注册照进行查找和其他用途。

b) 若设置了注册照片回调地址，注册成功后设备会将注册照信息通过 POST 请求方式回调给预先设置的回调地址，[参见注册照片回调](#)。

postman 示例



返回示例

```
{
  "data": "d7fe98be463c4d06a70605d4d58b5f33", //照片 id，每张注册照都有一个 id，限制
  32 位。若 faceId 不传，由系统分配的 faceId 为 32 位
  "result": 1, //接口调通
  "success": true //照片注册成功
}
```