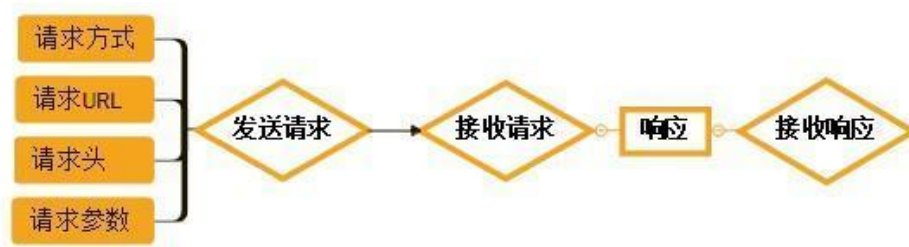


# postman 知识

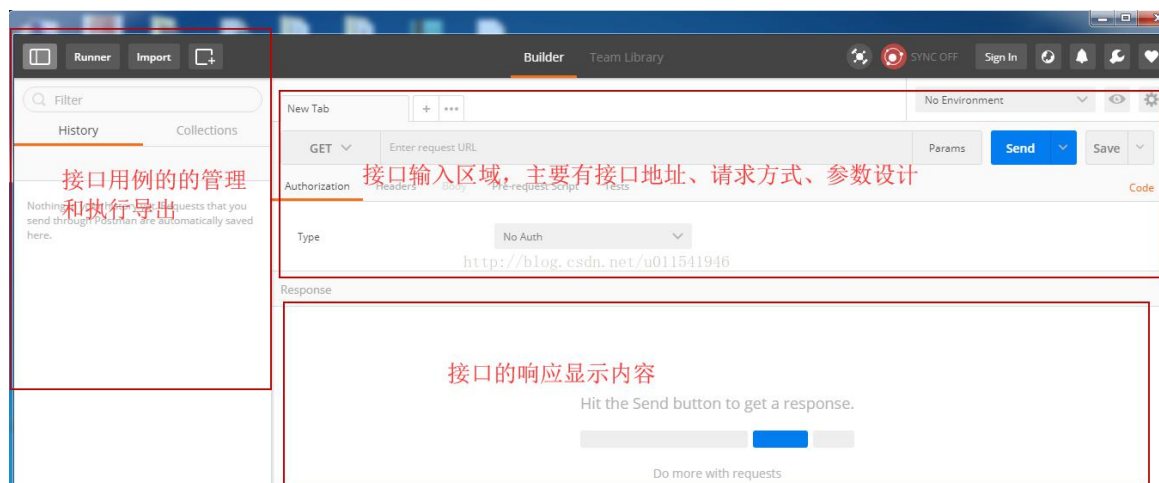
## 一. postman 基本功能

### 1. 接口请求流程

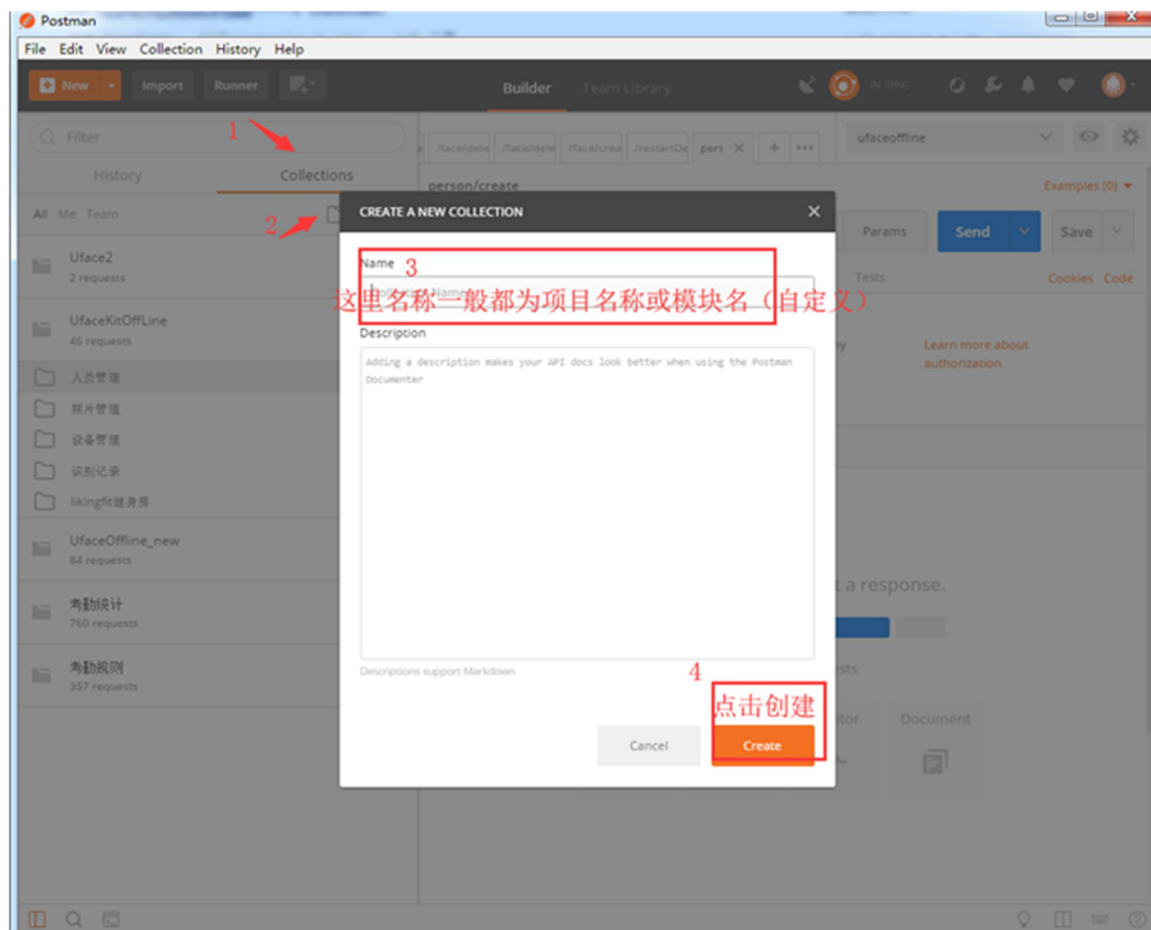


简书 · 搬渡

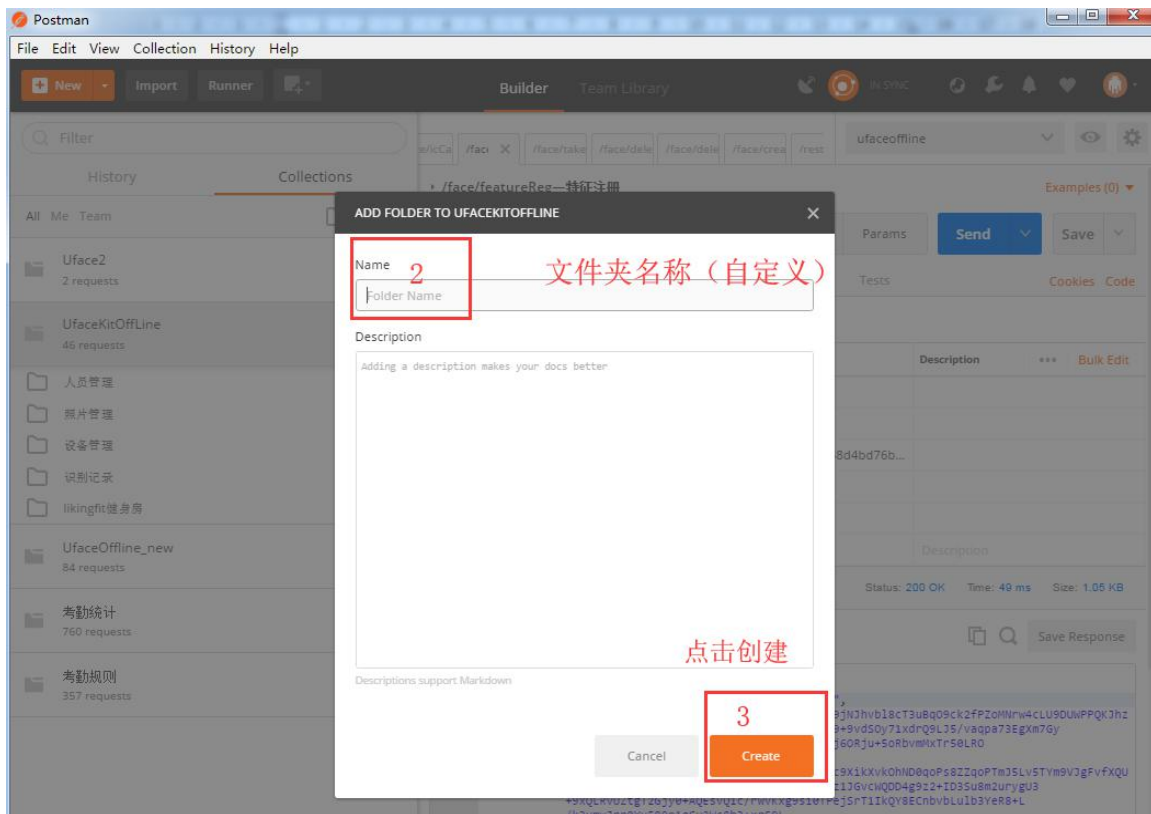
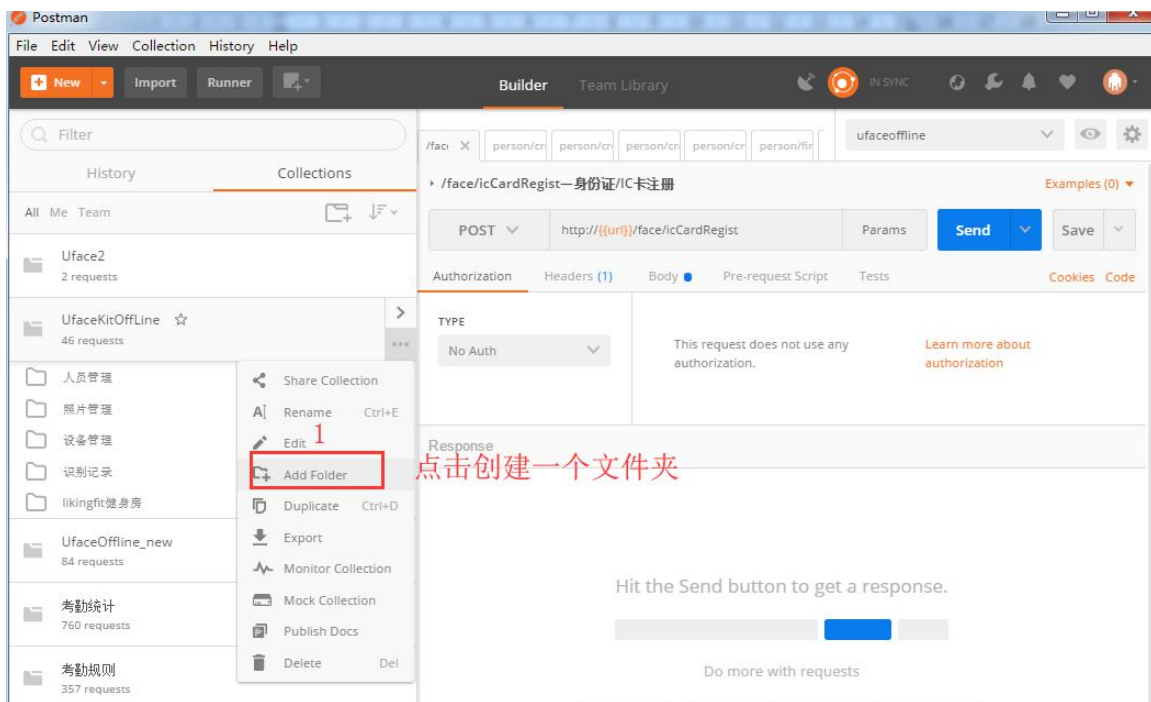
### 2. postman 区域划分及其作用



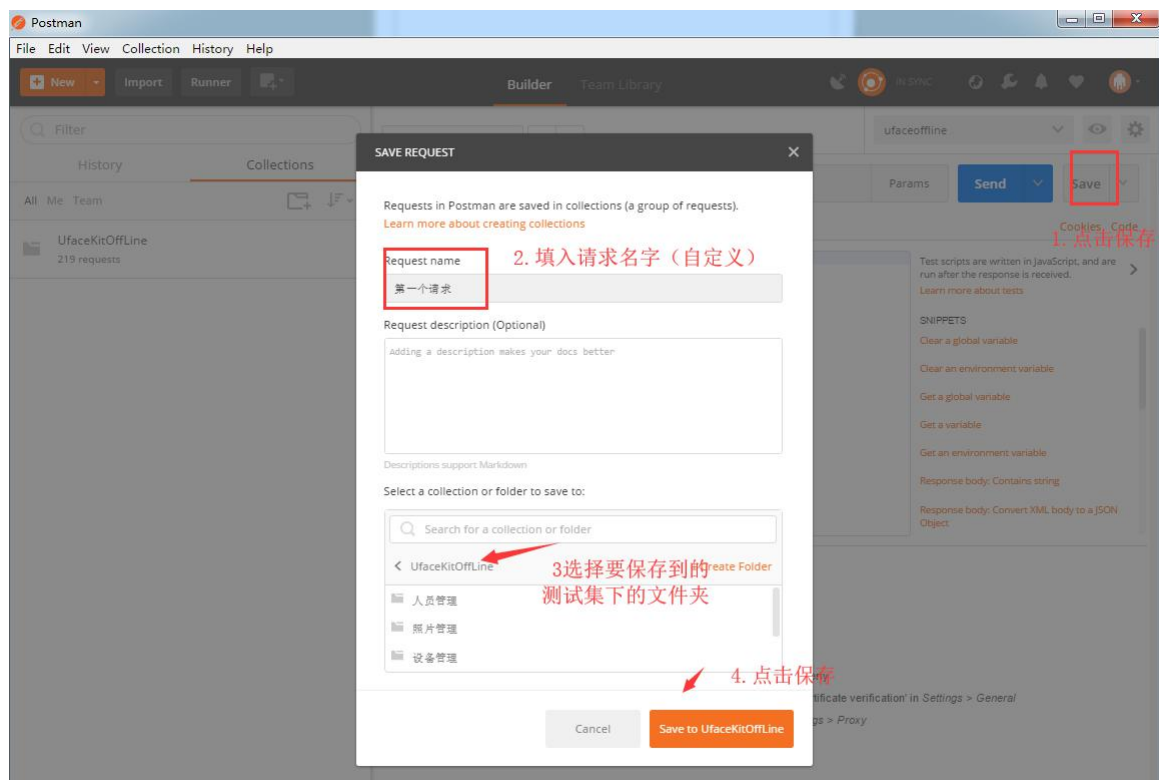
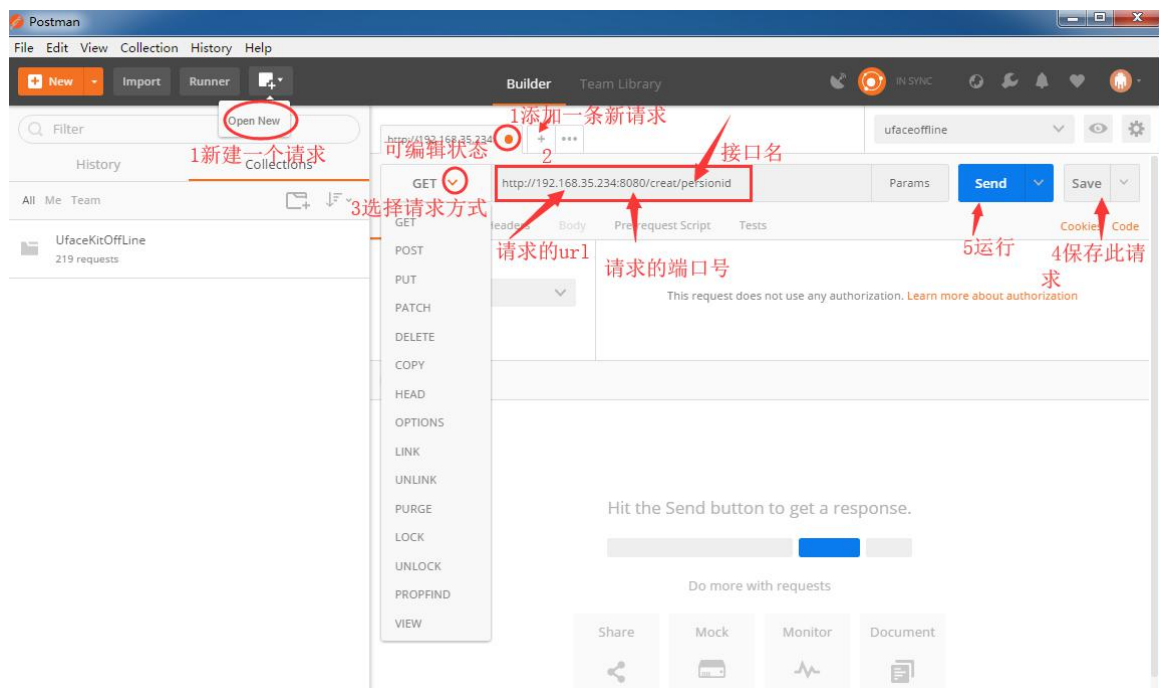
### 3. 打开 postman，建立一个 collections 的测试集



#### 4. 在测试集内创建一个装请求的文件夹

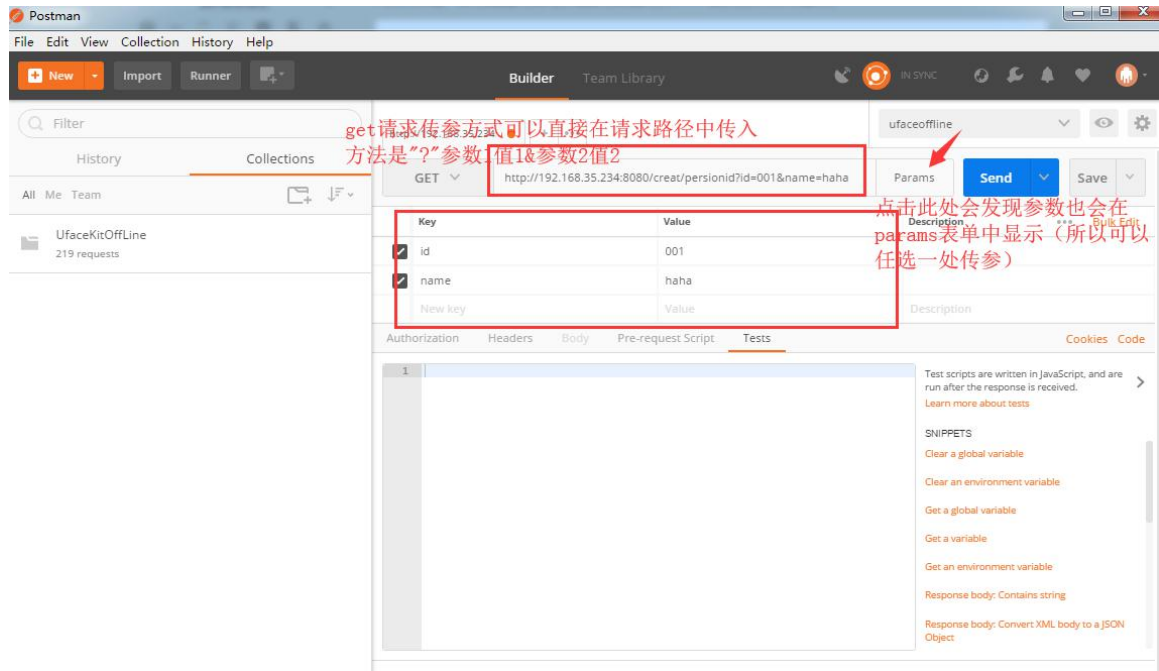


5. 新建一条 postman 接口请求（导航栏中的文件图标与接口输入区最上方的+号图标都可以新建）并保存



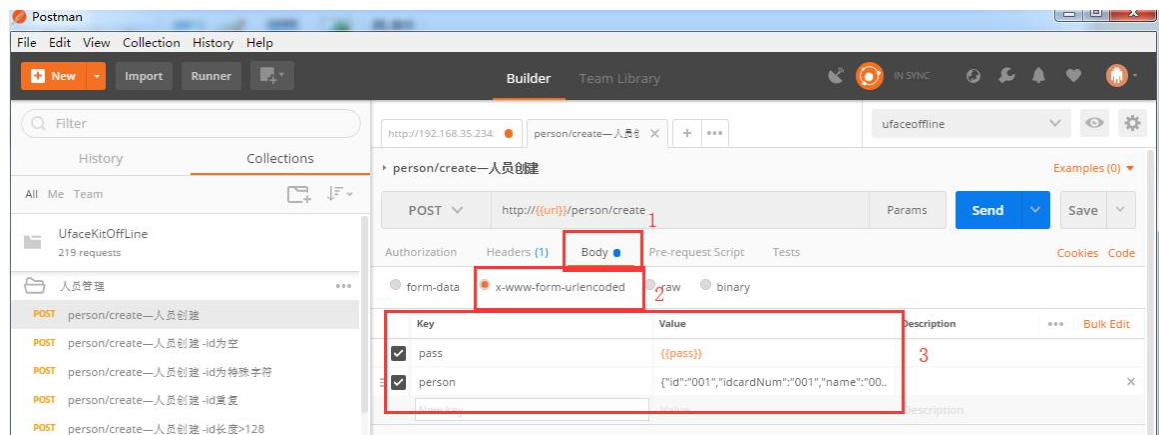
## 6. 几种主要请求的传参方式

### A. get 请求

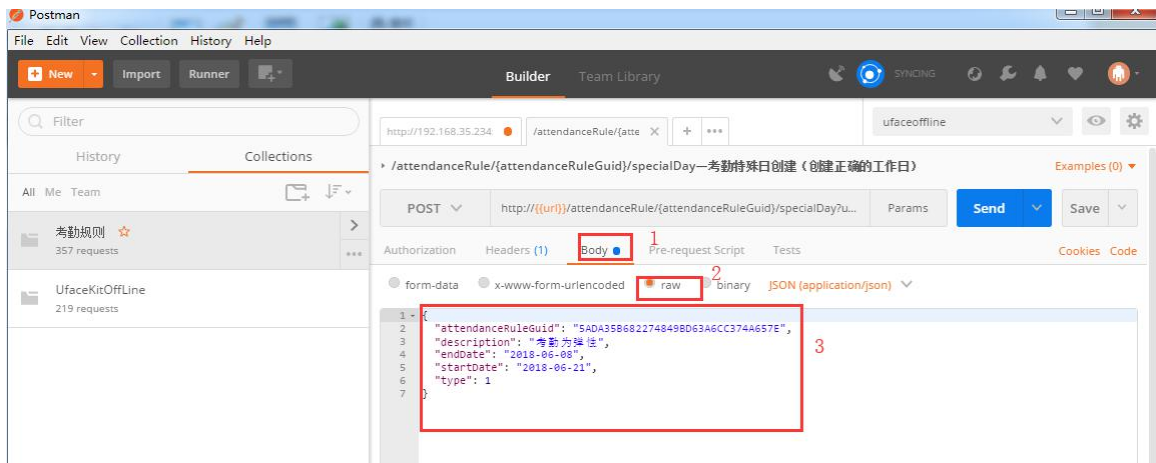


### B. post 请求

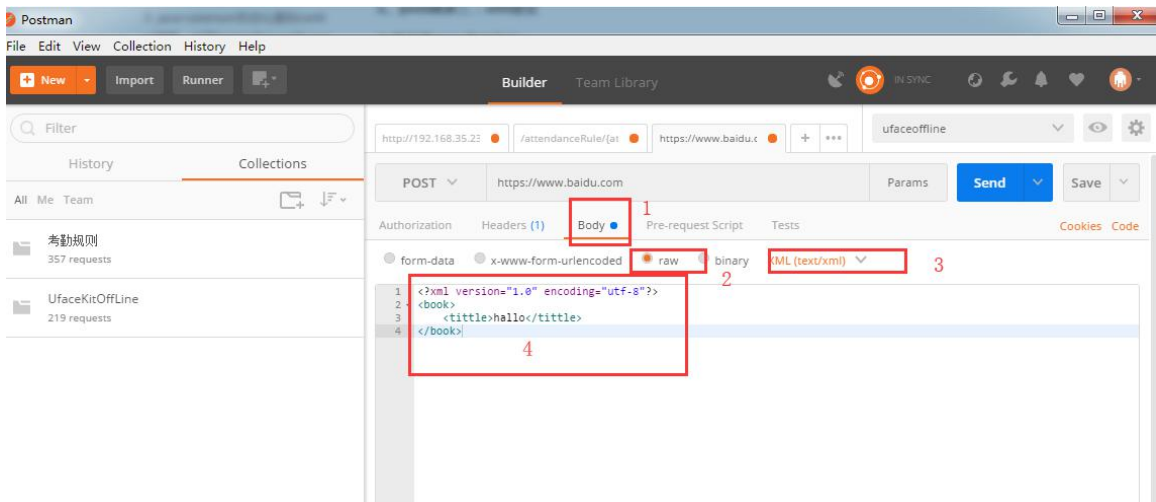
#### 表单提交



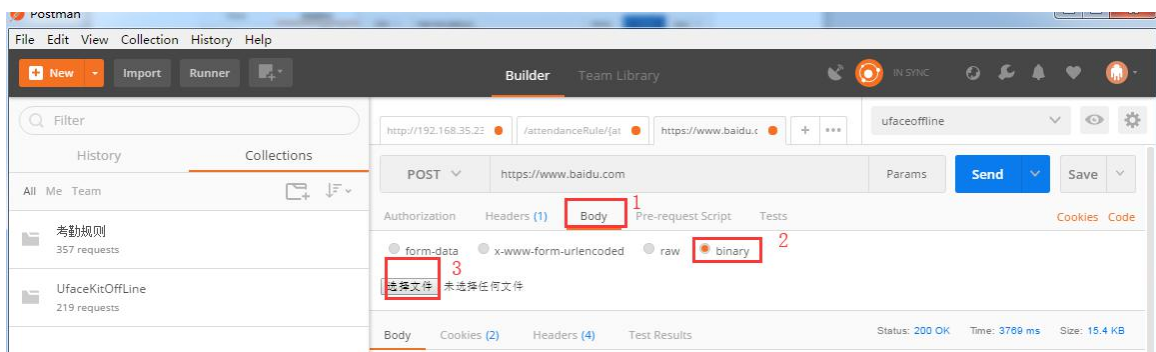
#### json 提交



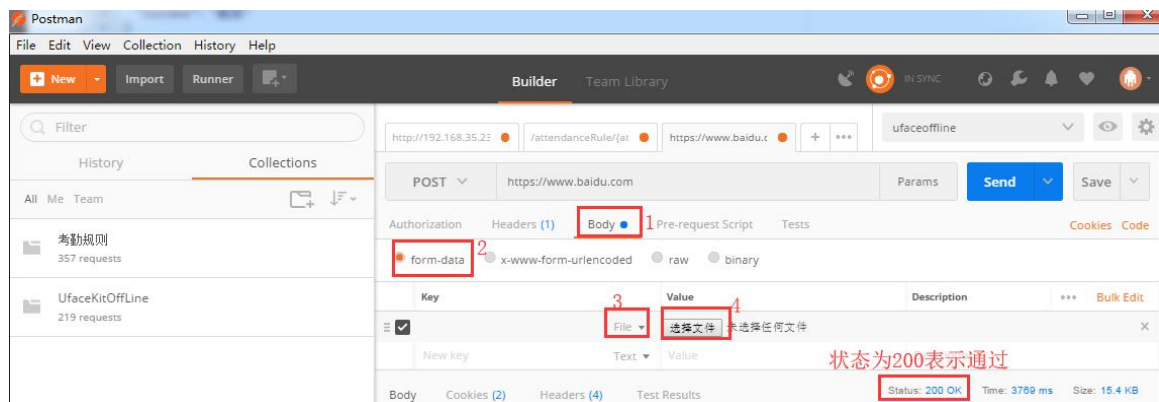
## xml 提交



## 二进制文件提交：可以发送音频、视频、文件等文件

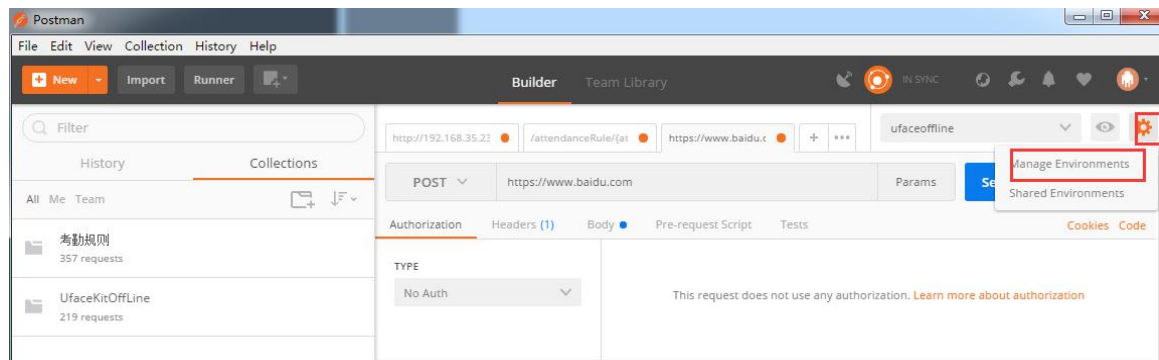


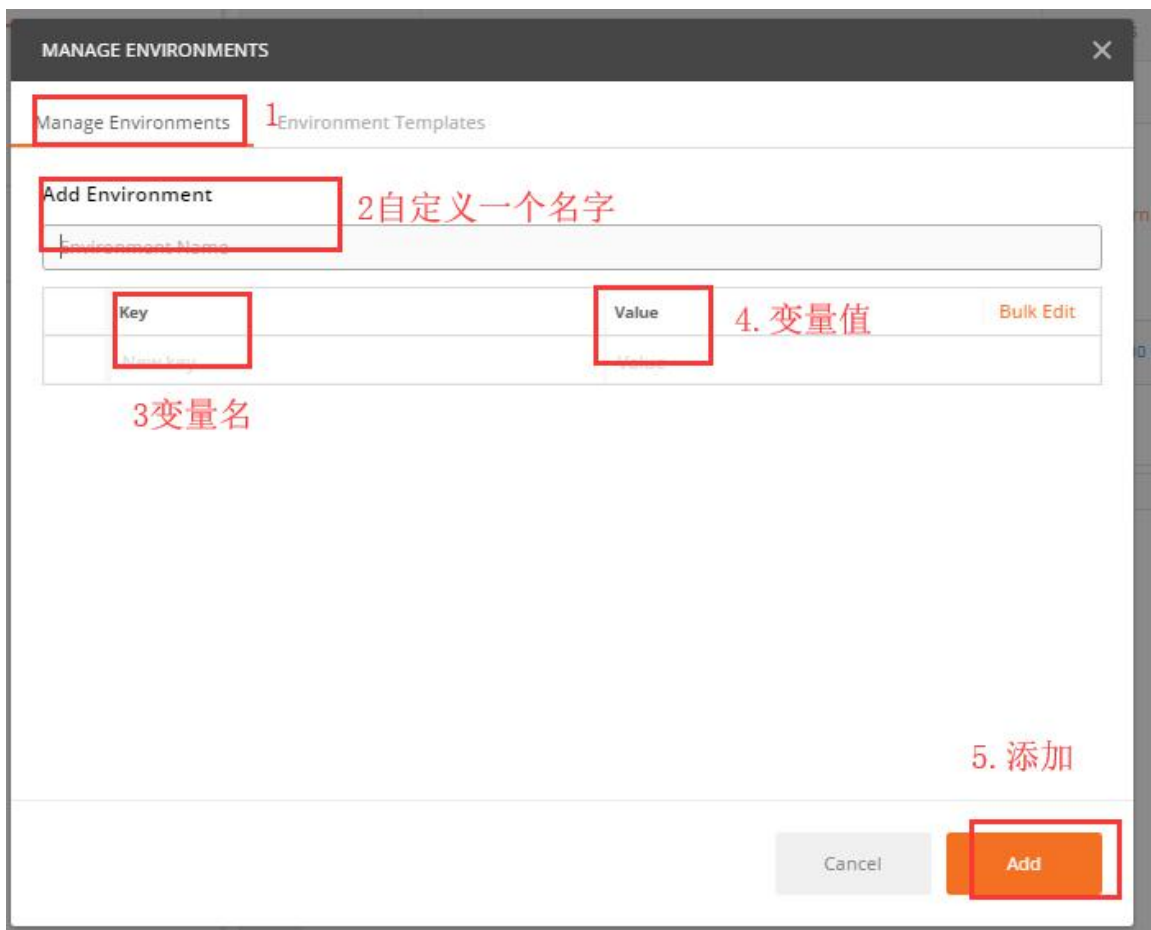
通过某个接口上传文件：选择 body 下的 form-data 选项修改 text 为 file，填入参数名，选择本地文件（但该文件不会作为历史保存，只能在每次需要发送请求的时候，重新添加文件）。



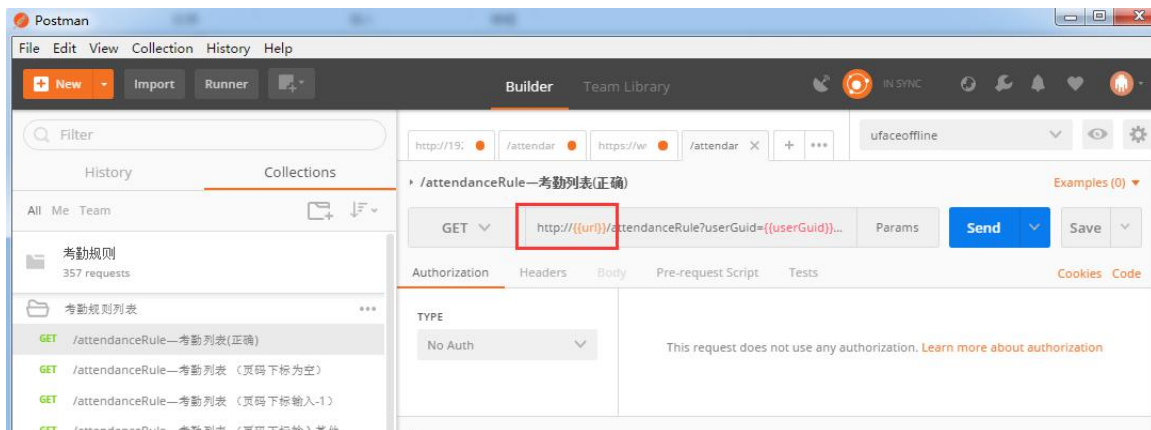
## 7. 环境变量配置

点击环境管理按钮添加环境



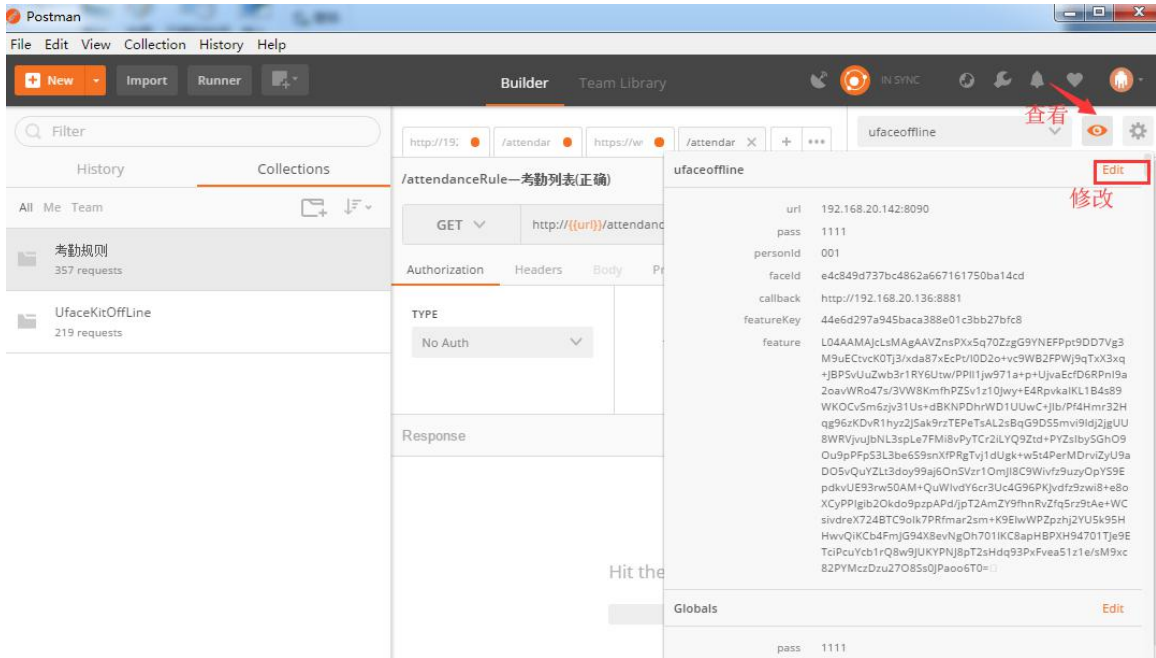


环境变量的用法（比如添加一个请求路径的环境变量直接在请求中用“{{变量名}}”调用一下）如下图：



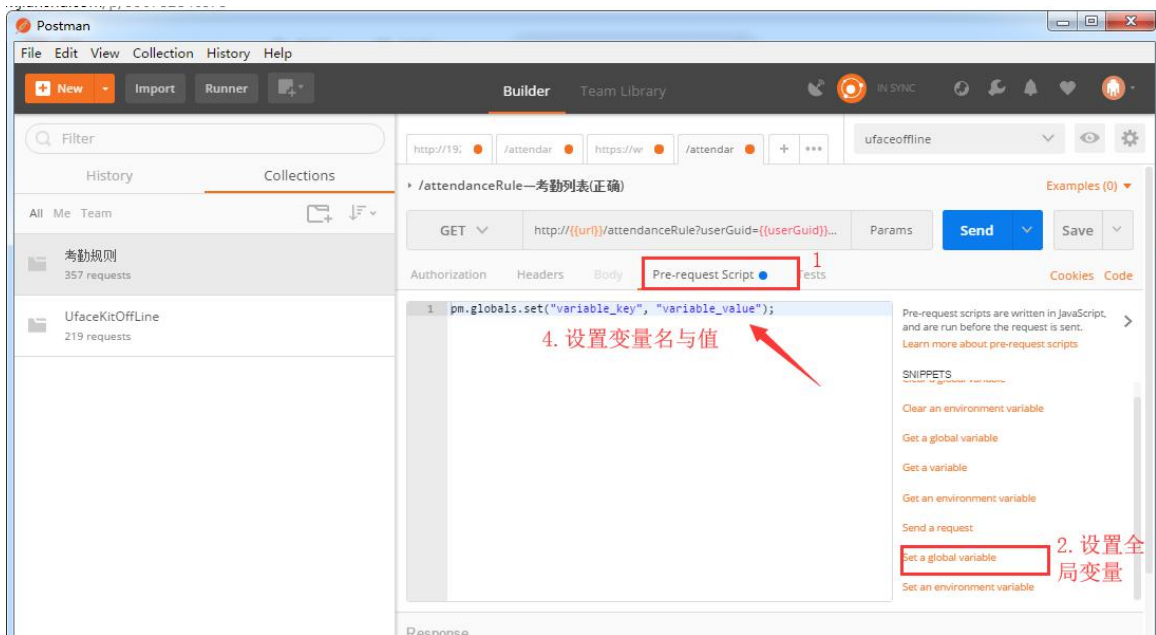
环境变量更改与查看



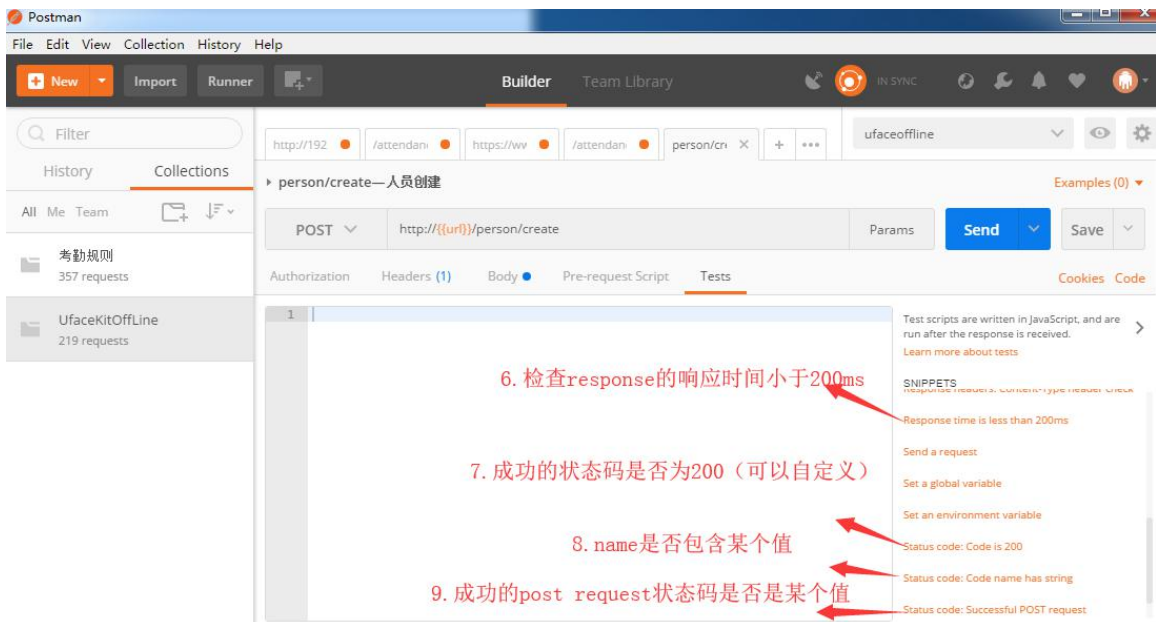
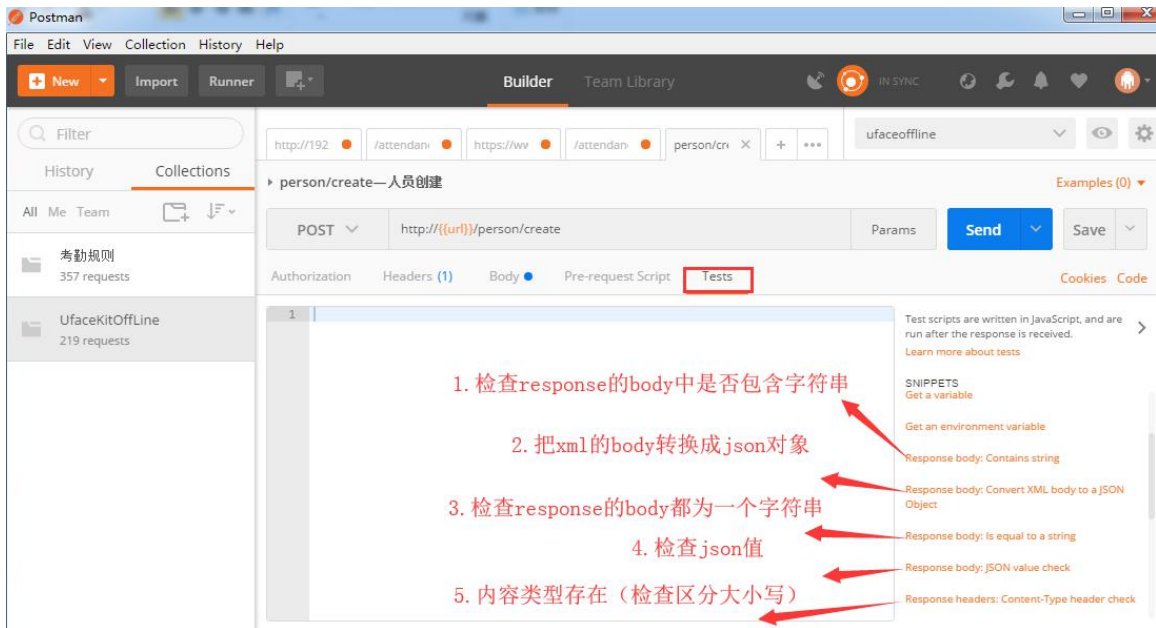


## 8. 执行前脚本: Pre-request Script (一般用来构建请求参数)

### 设置全局变量

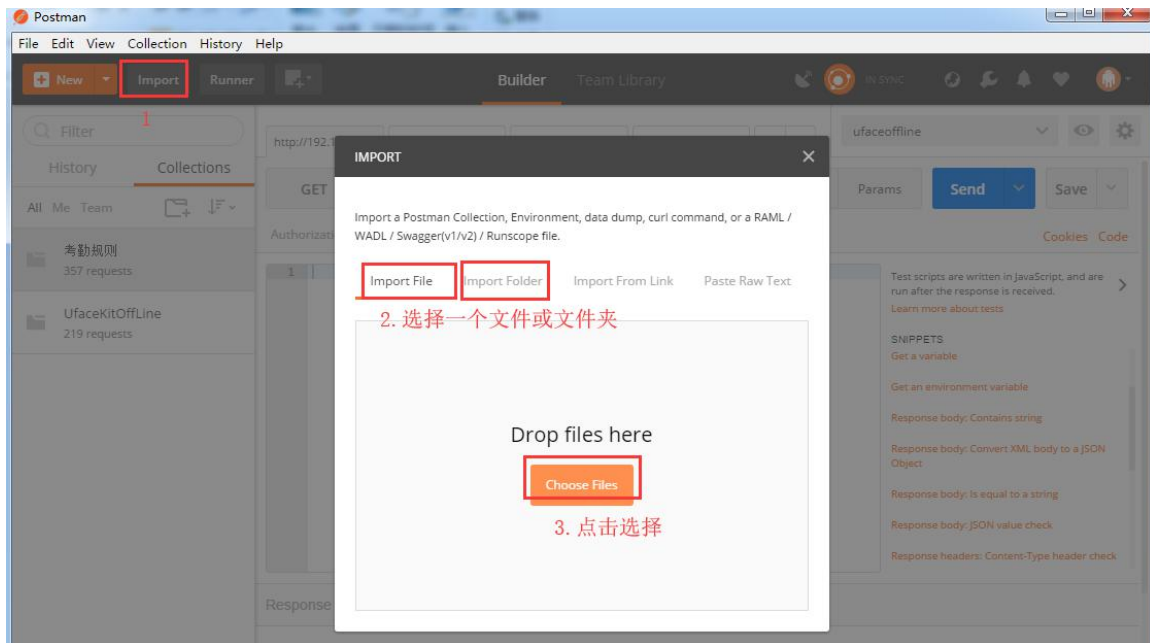


## 9. 测试完成后执行的脚本: test (在获取相应之后执行的脚本, 一般用来做测试)

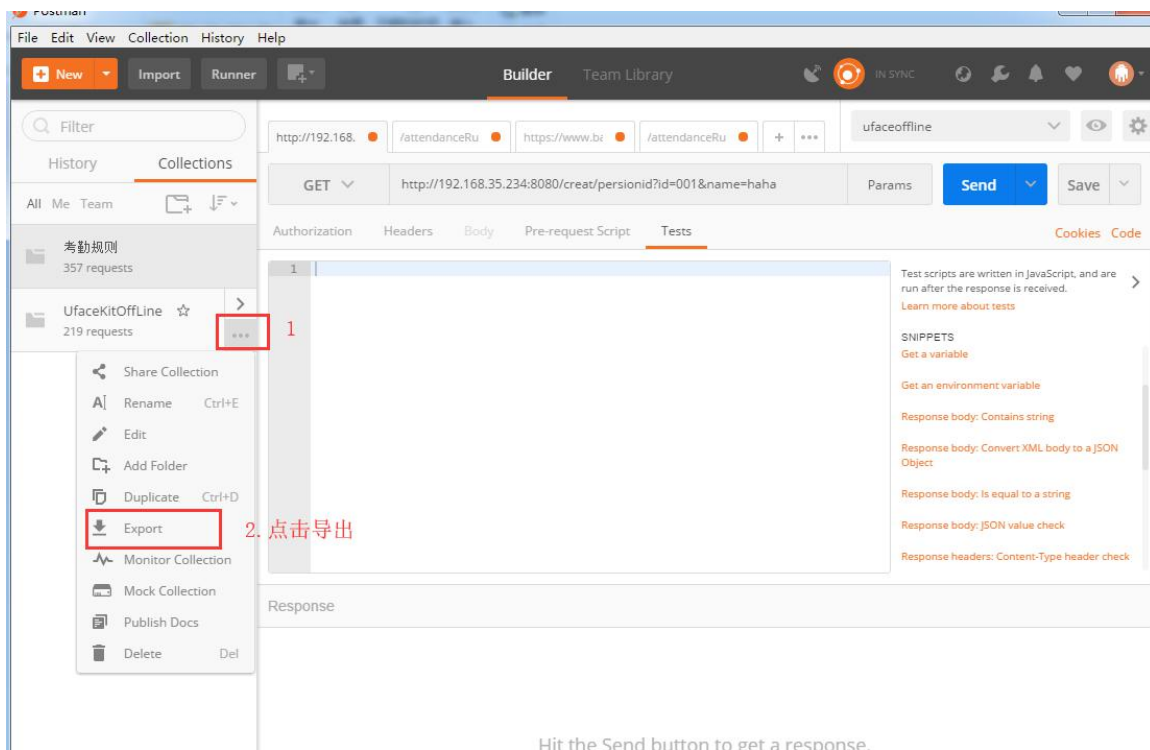


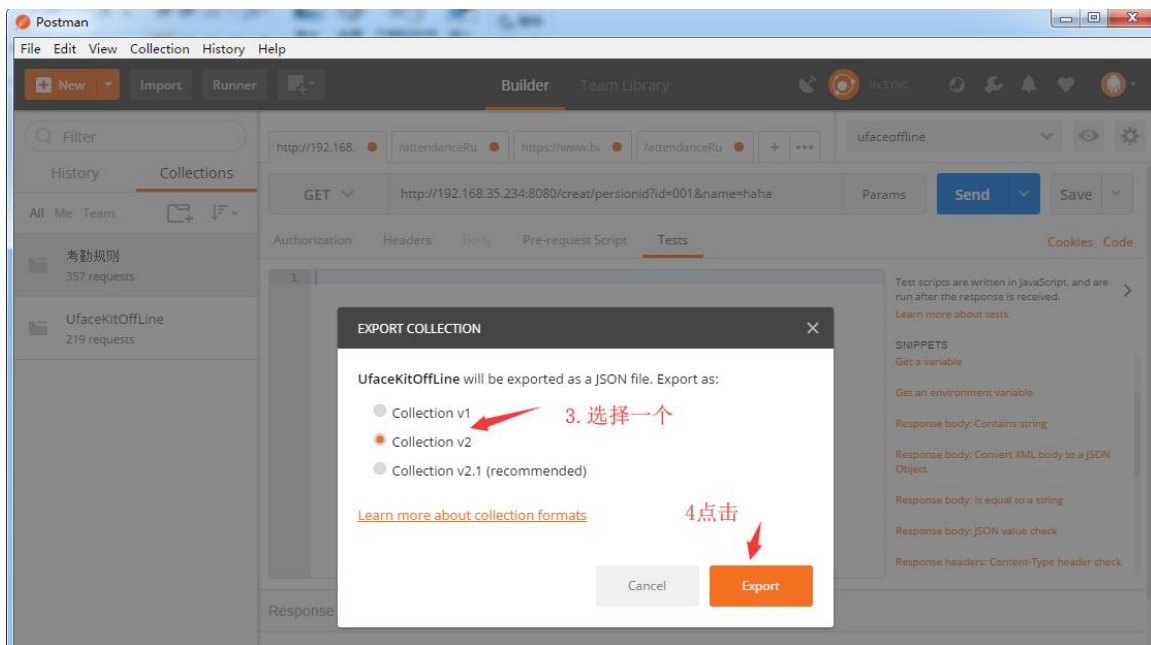
## 10. 测试集的导入导出

## 导入



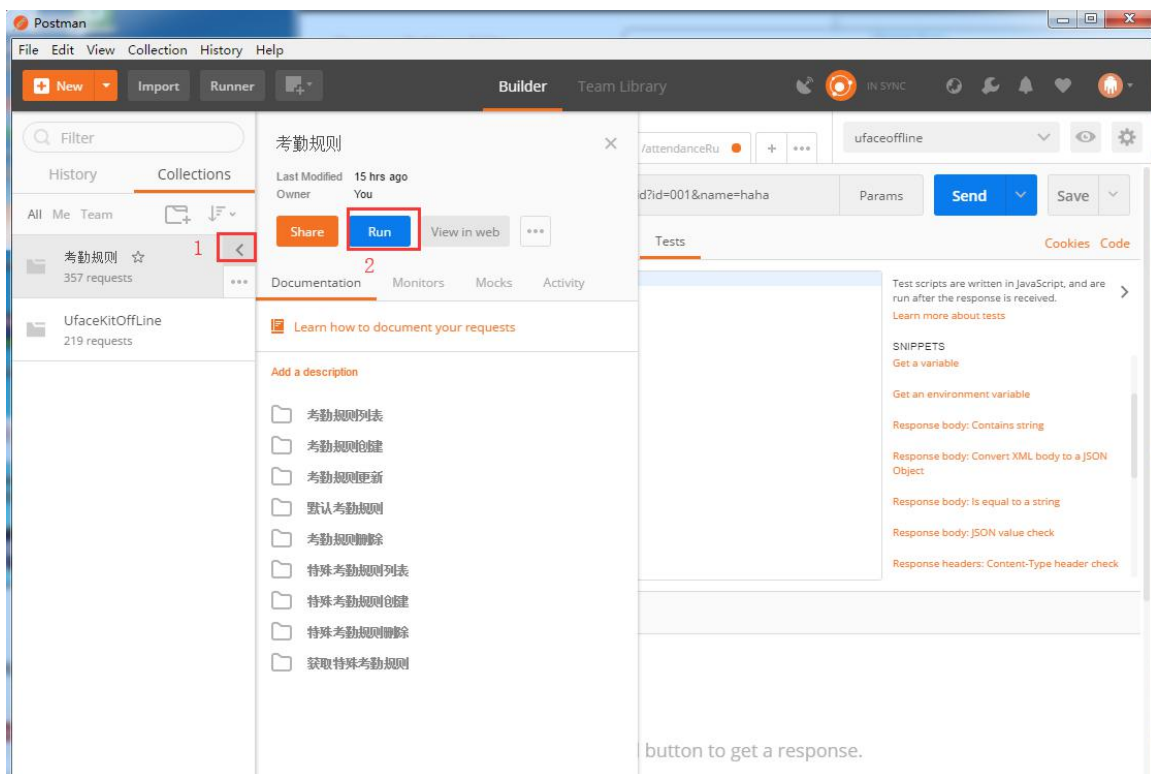
## 导出



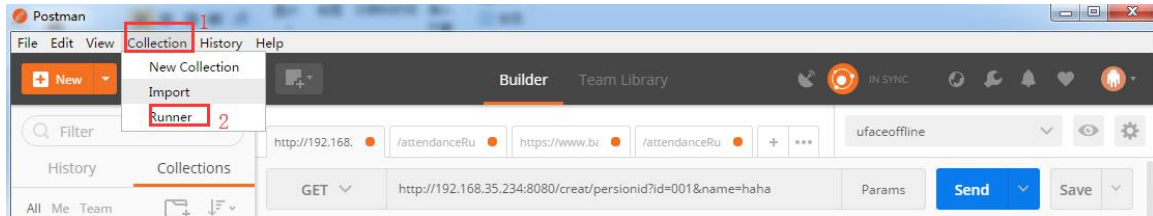


## 11. 批量执行（有三种入口）

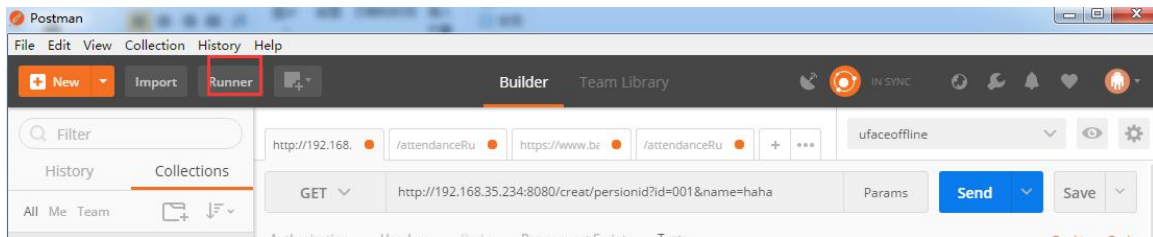
### A. 点击测试集后边的“>”图标



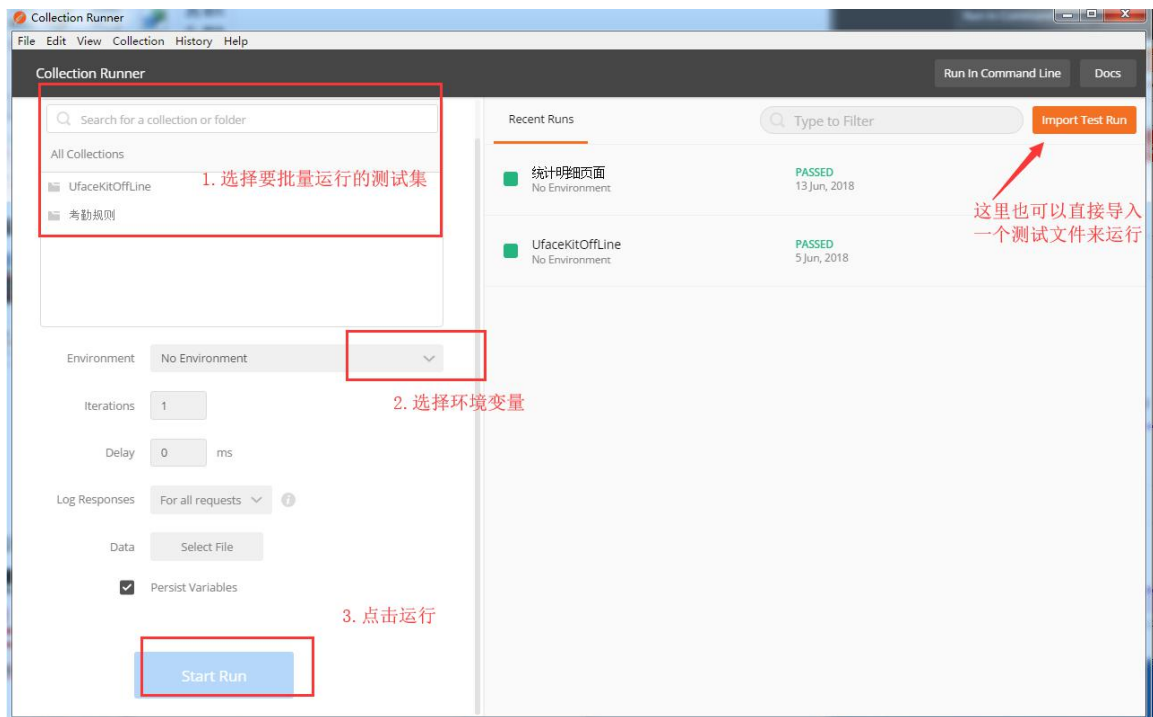
## B. 从窗口导航栏中的 collection 中进入



## C. 直接点击窗口中的 runner 运行



## 批量执行页面



## Environment

此处选择运行集合中的请求时对应使用的环境。

## Iterations

该参数配置我们的集合将被运行的次数。通常通过使用不同的测试数据运行多次，从而来确保 API 的稳定性。

## Delay

该参数配置依次运行每个请求，每个请求之间间隔侧时间，单位为毫秒。

## Log Responses

这用于限制集合运行时的响应日志记录。默认情况下，一般调试目的，所有响应都会记录，但对于大型集合，可以更改此设置以提高性能。

## Data

提供集合运行时需要的数据文件，比如一些测试数据等。

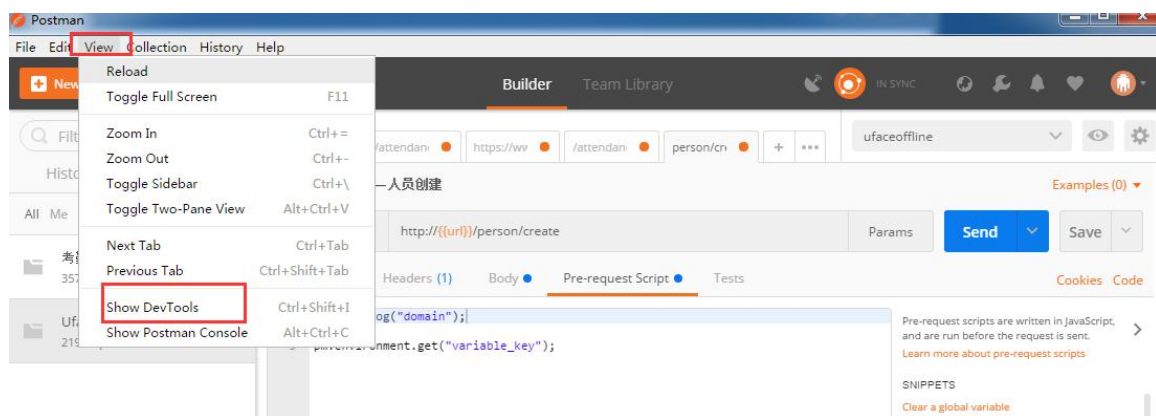
## Persist variables

默认情况下，集合运行器中的任何环境更改都不会反映回请求构建器中。

# 二. postman 开发者模式

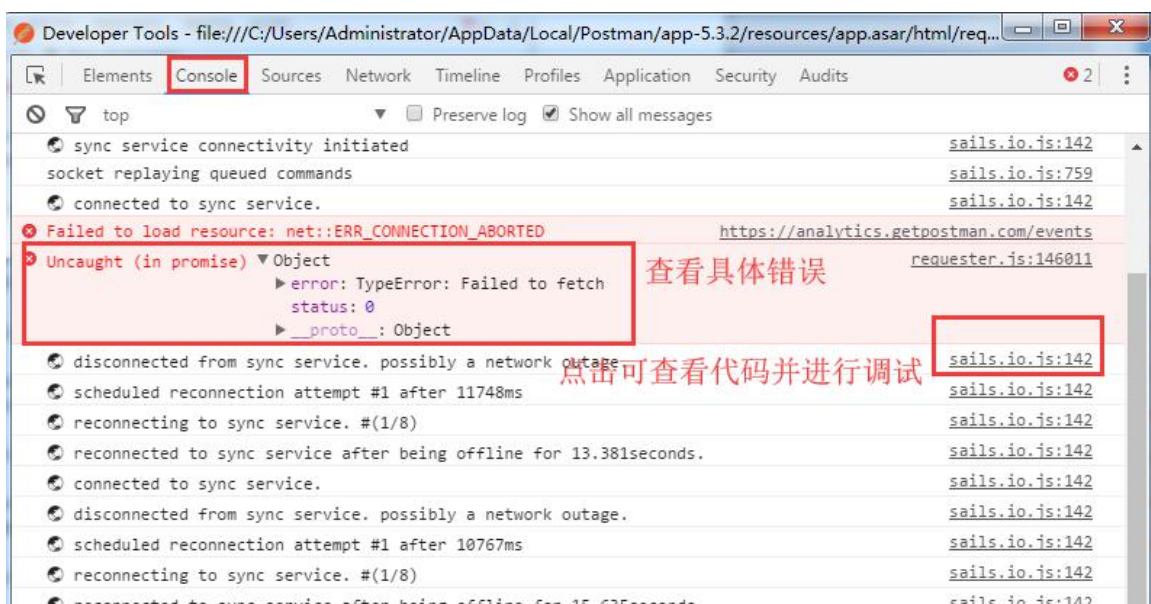
## 1. postman 中的开发者工具 (ShowDevtools):控制台日志

### 打开 showDevtools

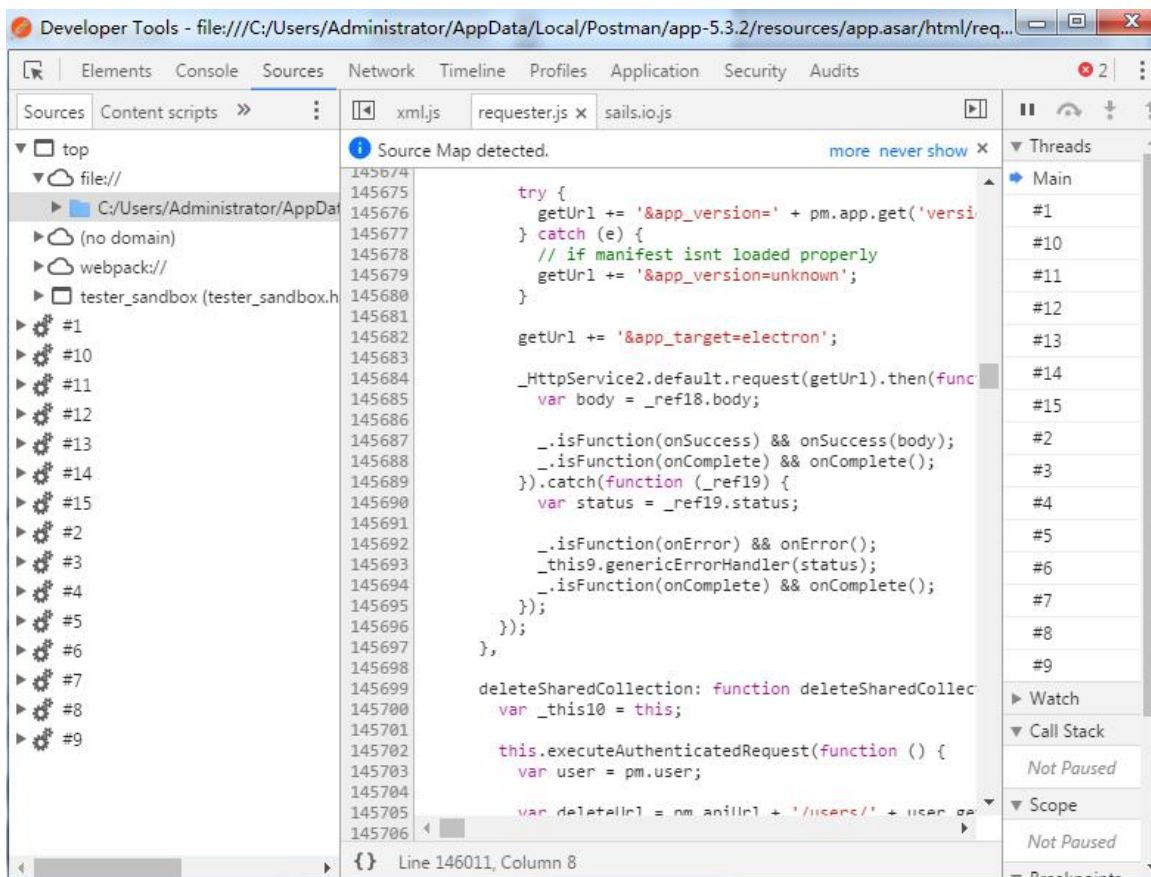




在 DevTools 窗口中，单击 top 控制台选项卡显示应用程序的调试日志。

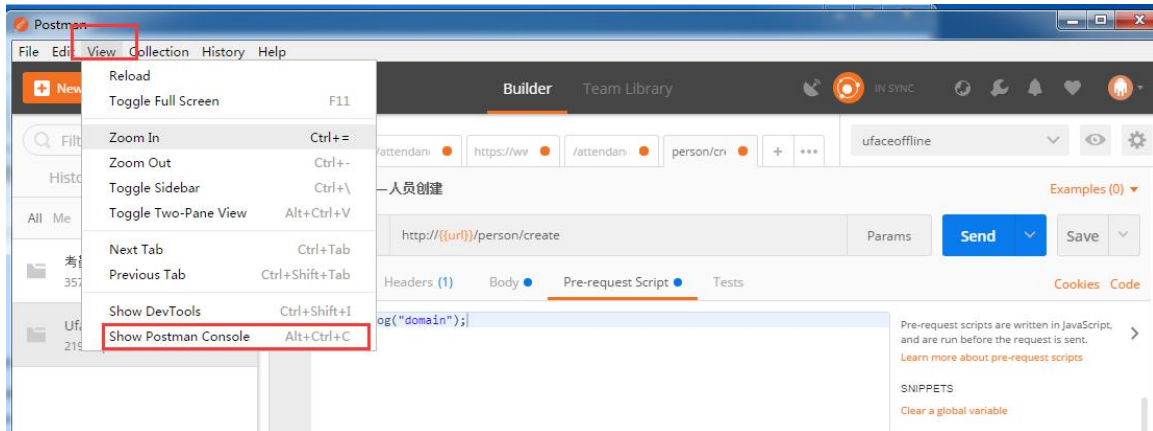


## 具体调试

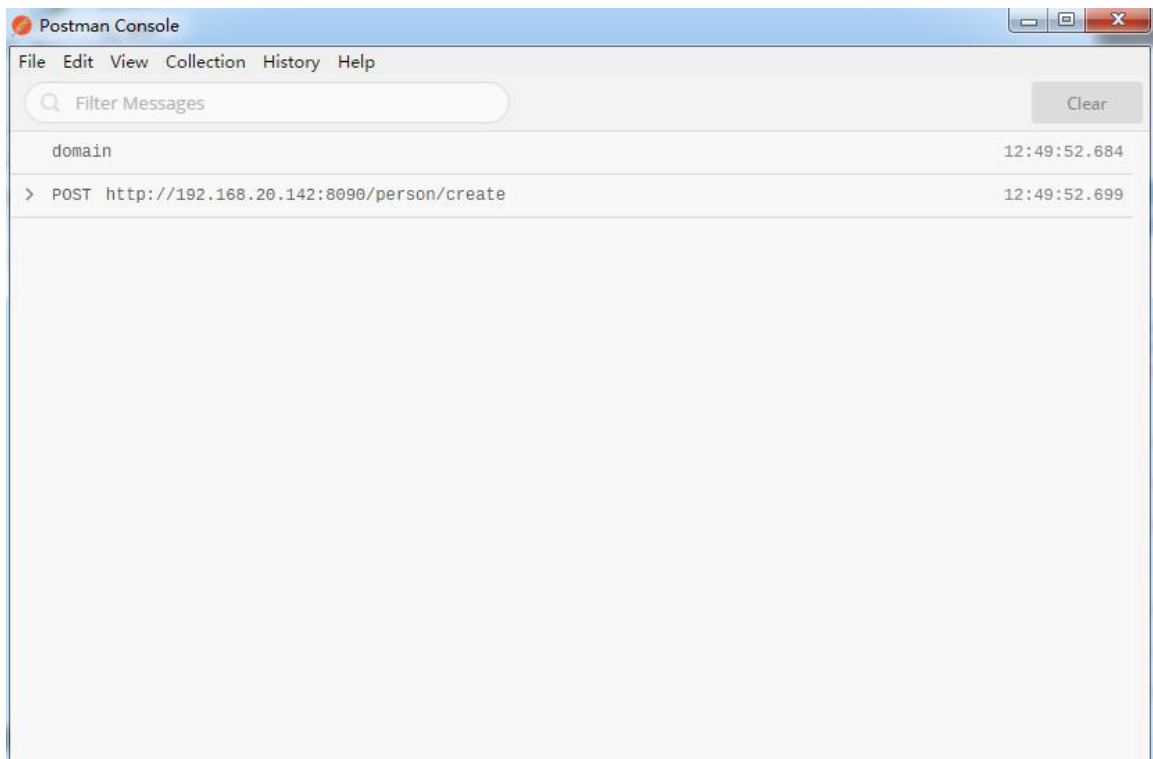


2. postman 中的 show Postman Console: 打印变量的值（也可以在 request 的 Pre-request Script 中填写 `console.log("domain")`，当发送请求时，此脚本将会执行，参数对应的值也会打印在 Postman Console 中）。

打开 show Postman Console 会弹出图二界面——》点击 postman 中的 send 按钮，变量值就会在图二界面打印（如图三）

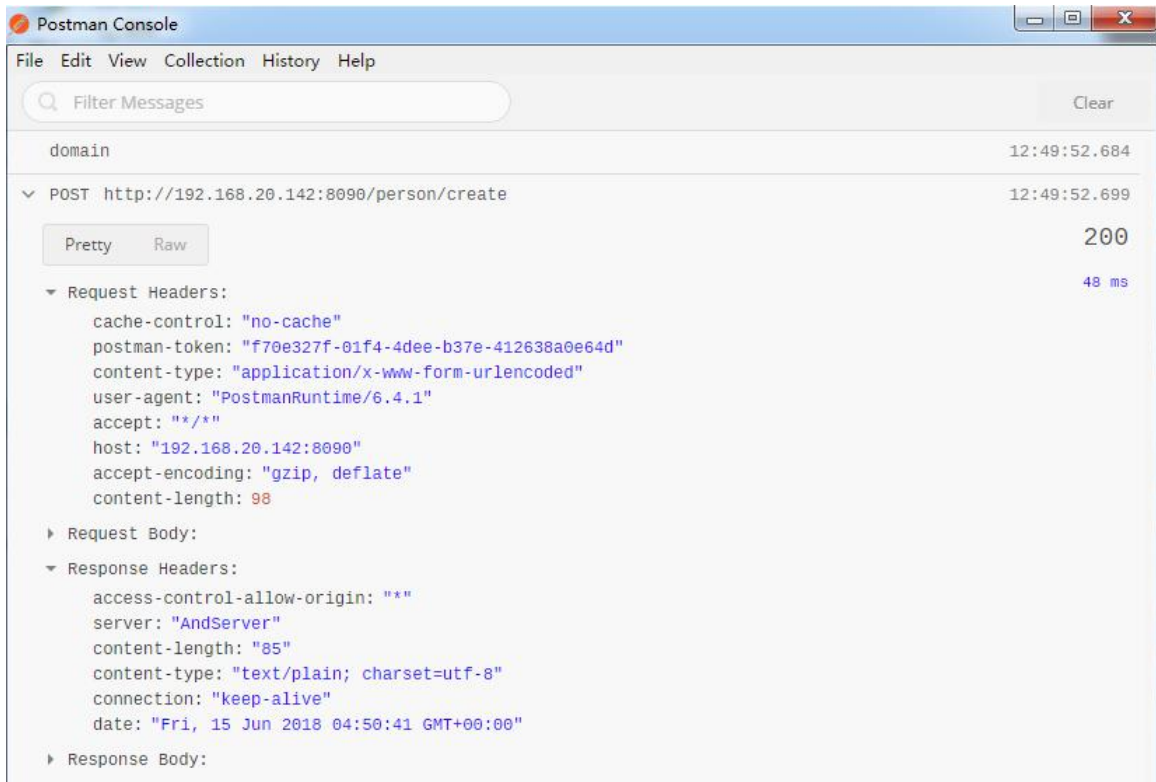


图二：



图三：





### 3. NewMan

(1) 首先需要下载安装 node.js 插件

下载地址: windows node-v6.7.0-x64:

<http://pan.baidu.com/s/1nvm69E5>

Mac node-v7.8.0.pkg 下载: <http://pan.baidu.com/s/1c25VoXy>

(2) 打开 cmd 控制台输入 `npm install -g newman` 命令显示界面如下:  
安装成功

```
C:\Users\Administrator>npm install -g newman
C:\Users\Administrator\AppData\Roaming\npm\newman -> C:\Users\Administrator\AppData\Roaming\npm\node_modules\newman\bin\newman.js
C:\Users\Administrator\AppData\Roaming\npm
+-- newman@3.9.4
+-- async@2.6.1
+-- lodash@4.17.10
+-- cli-progress@1.8.0
+-- cli-table2@0.2.0
+-- lodash@3.10.1
+-- string-width@1.0.2
+-- code-point-at@1.1.0
+-- is-fullwidth-code-point@1.0.0
+-- number-is-nan@1.0.1
+-- strip-ansi@3.0.1
+-- ansi-regex@2.1.1
+-- colors@1.3.0
+-- commander@2.15.1
+-- csv-parse@1.2.4
+-- EventEmitter3@3.1.0
+-- filesize@3.6.1
+-- handlebars@4.0.11
+-- async@1.5.2
+-- optimist@0.6.1
+-- minimist@0.0.10
+-- wordwrap@0.0.3
+-- source-map@0.4.4
```

(3) 执行测试集(newman 的常用命令):

### 1. 本地运行自动化测试命令

输入 `newman run {json filepath}` //绝对路径

(备注: Windows 系统后面不需要分号, linux 系统需要, 例如: `newman run C:\Users\xingzheng\Desktop\dafmng-api.json`).

```
管理员: C:\Windows\system32\cmd.exe
inside "人员管理 / person/create-人员创建-id未填"
04. AssertionError... 返回msg正确
    expected false to be truthy
    at assertion:1 in test-script
    inside "人员管理 / person/create-人员创建-id重复"
05. AssertionError... 返回msg正确
    expected false to be truthy
    at assertion:1 in test-script
    inside "人员管理 / person/create-人员创建-id数字字母"
06. AssertionError... 返回success正确
    expected false to be truthy
    at assertion:3 in test-script
    inside "人员管理 / person/create-人员创建-id数字字母"
07. TypeError        Cannot read property 'id' of undefined
    at test-script
    inside "人员管理 / person/create-人员创建-id数字字母"
08. AssertionError... 返回msg正确
    expected false to be truthy
    at assertion:1 in test-script
    inside "人员管理 / person/create-人员创建-id特殊字符"
09. AssertionError... 返回msg正确
    expected false to be truthy
    at assertion:1 in test-script
    inside "人员管理 /
```

2、生成测试报告，支持 html, xml, json, cli4 种。

newman run

```
C:\Users\hexueqin\Desktop\serverApiTest.postman_collection.js
on --reporters html,junit --reporter-html-export
d:\html01.html --reporter-junit-export d:\xml01.xml
```

3、含有环境变量，全局变量的自动化测试命令：指定执行对应的 folder  
常见命令：

run --运行 newman 的命令

-e 加载环境变量的数据

-g 加载全局变量的数据

-n case 运行次数，默认运行次数为 1 次

--folder 运行 collection 中的指定 folder

--reporters html, junit 指定生成报告的类型

其它命令可在 github 上查看：

<https://github.com/postmanlabs/newman#configuring-reporters>

举例：

newman run

```
C:\Users\hexueqin\Desktop\apiTest.postman_collection.json -e  
C:\Users\hexueqin\Desktop\dev-environment.postman_environment.  
json -g C:\Users\hexueqin\Desktop\globals.postman_globals.json  
--folder visitApi --reporters html --reporter-html-export  
d:\result.htm
```

### 三. 其他小功能

#### 1. 窗口导航栏中的快捷方法使用

