

# Postman接口测试实战

欢迎进入 Testing学院

主讲老师：Roy QQ：578225840

技术交流QQ群：691499681

客服老师：卡卡 QQ：632232258





# 目录

CONTENT

Part.1 Postman安装与基本使用

Part.2 Postman接口请求实战

Part.3 结果断言与变量使用

Part.4 测试管理-测试集与数据驱动

Part.5 进阶实战-Cookie篇

Part.6 进阶实战-token与参数传递篇

Part.7 Postman持续集成

## PART 1

# Postman安装与基本使用



# Postman下载安装



## Postman下载

Postman分为Postman native app 和Postman Chrome app两个版本。

目前Chrome app已经停止维护，官方也不推荐使用该版本。

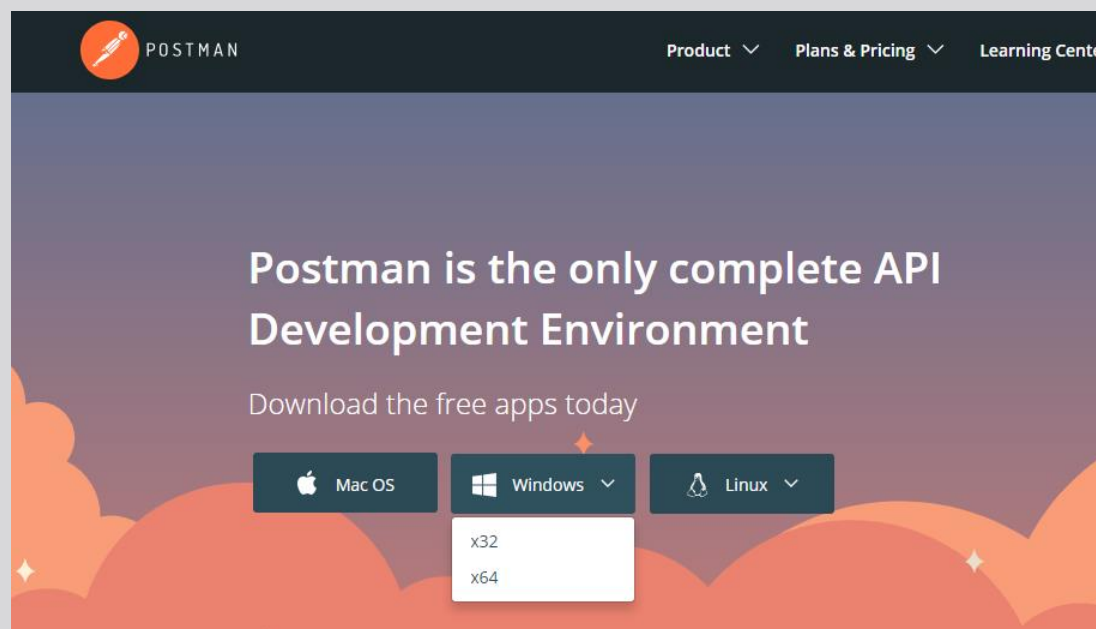
因此建议大家直接选择native app进行下载。

下载地址：<https://www.getpostman.com/postman>

根据自己的操作系统选择对应版本。

## Postman安装

下载完成后的exe文件直接双击运行，按提示逐步完成安装即可。



# Postman基本使用

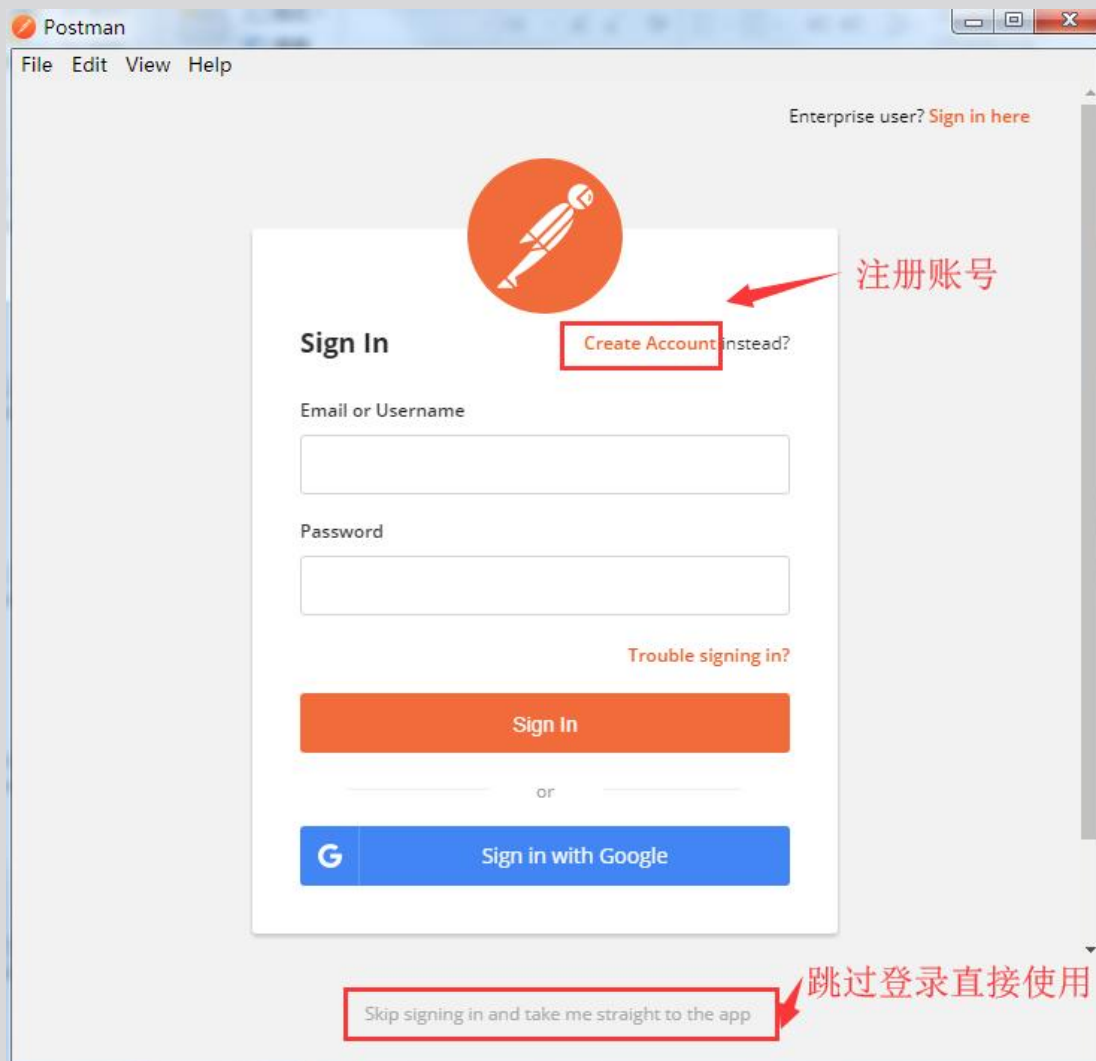
## 初次使用Postman

第一次打开Postman时，会弹出一个账号登录界面，如右图。

Postman基于账号提供了云端同步postman测试数据和团队协作功能，推荐注册一个账号使用，如果不想注册，点击下方skip或者直接关闭窗口即可。

## Postman使用

接下来将会弹出新建窗口引导：Create New 菜单用于创建单个测试、测试集、环境变量等 Template 和 API Network 提供了一系列他人发布的模板，可供测试学习使用。



# 发起第一个接口请求



## 第一个接口请求实战

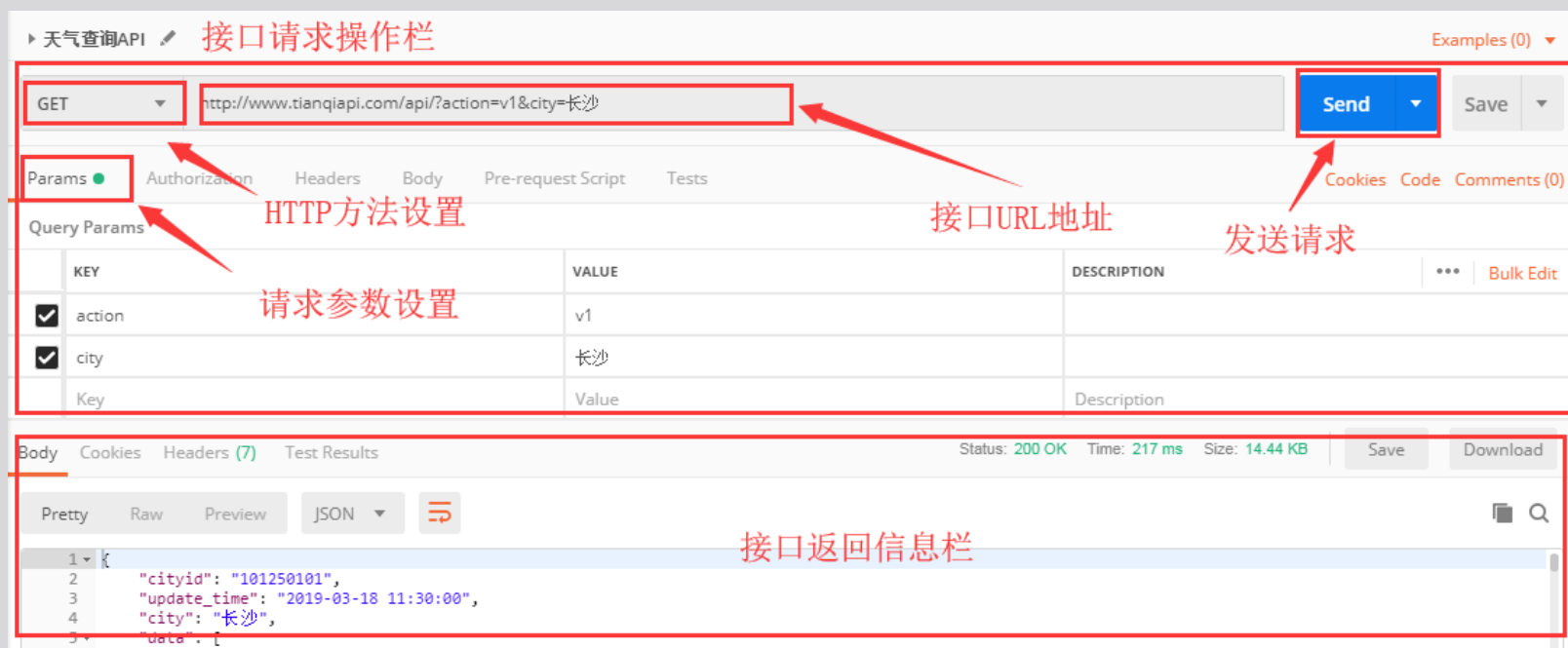
开发的天气API接口地址：<https://www.tianqiapi.com/api>

接口文档地址：<http://www.tianqiapi.com/?action=v1>

网上有非常多的开放接口可用，比如用友开放接口平台：<https://api.yonyoucloud.com>

但是需要注册得到appcode之后才能进行测试，有兴趣的同学可以申请进行测试。

## Postman基本功能使用



## PART 2

# 接口测试实战





# 接口测试



## 什么是接口测试？

接口测试就是针对软件对外提供服务的接口的输入输出进行测试，以及接口间相互逻辑的测试，验证接口功能与接口描述文档的一致性。

## 为什么做接口测试

接口测试通常能够对系统测试得更为彻底，更好地保障产品质量，同时越早越底层的发现问题，修改和维护的代价也越小。

## 如何做接口测试

根据接口文档设计用例，调用接口，验证结果。

## 第二个接口测试实战

百度ip查询接口从抓包到测试。





# 接口测试流程

## 获取接口信息

通过接口文档和抓包来获取接口的基本调用方式和返回

## 接口测试用例设计

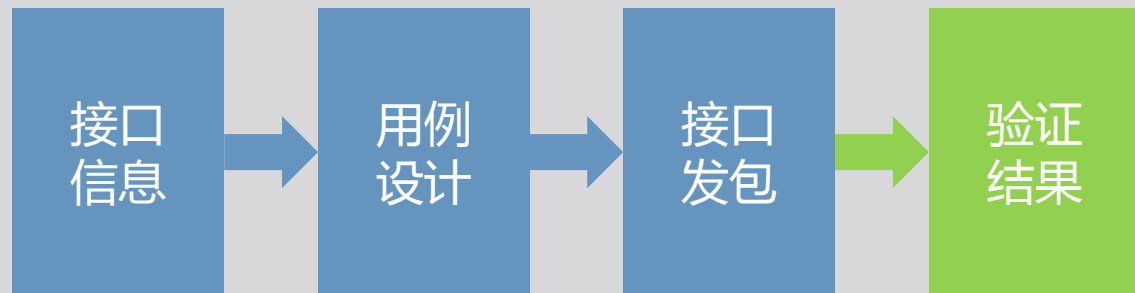
根据获取到的接口信息，按照接口测试用例设计方法，设计参数和预期返回结果。

## 接口发包

使用工具或者编程向接口传递参数。

## 返回信息验证

获取接口返回的结果，进行解析和验证。



# Postman发起接口请求

通过chrome浏览器中按下F12打开开发者工具，切换到network菜单，对浏览器中的动作进行抓包。

## 请求部分

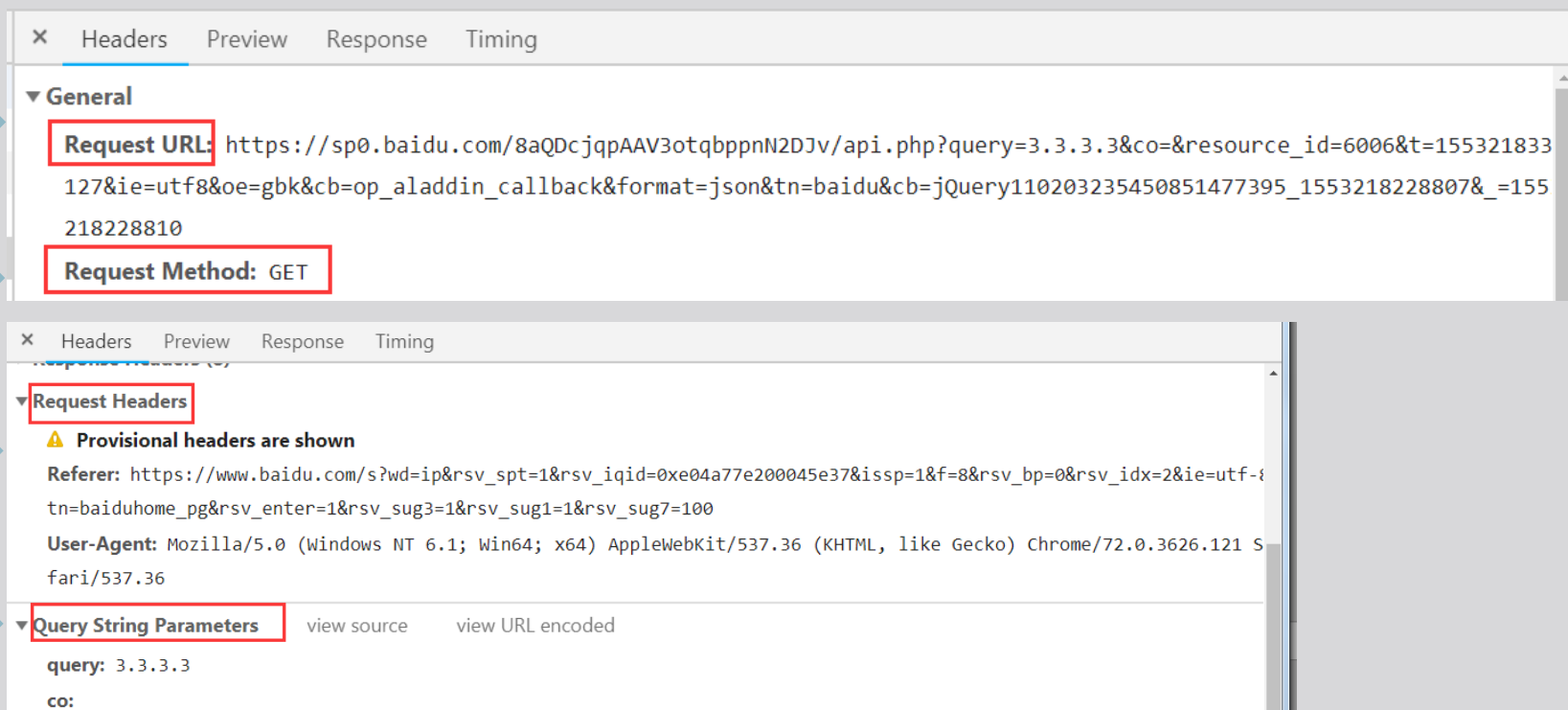
在接口测试中，接口请求信息中，重点需要关注4大信息：接口URL地址、请求方法、请求头以及请求参数。

URL

方法

头域

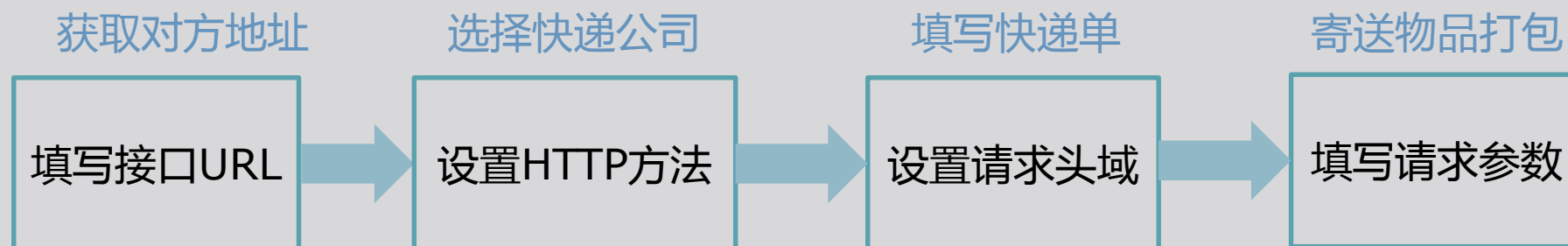
参数



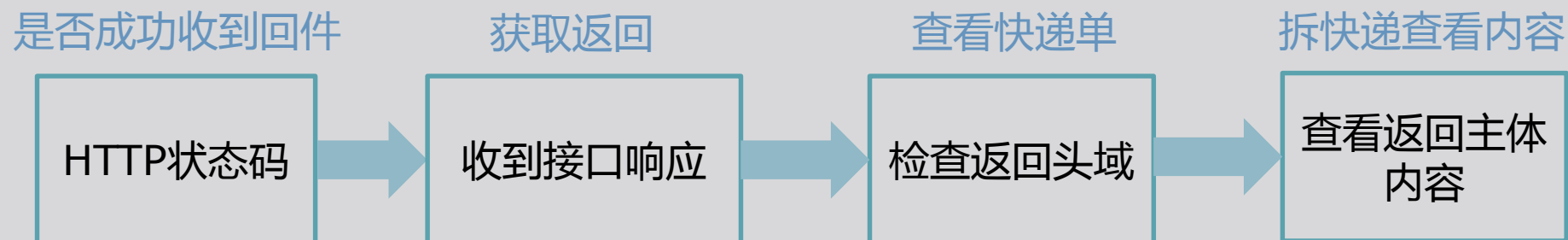
# 接口收发包

简单理解，接口收发包的过程就像通过快递公司寄快递，需要知道对方的地址（URL）、选择快递公司（HTTP方法）、填写快递单（头域信息）、包装快递物品（发送请求体参数）。

## Postman接口请求过程



## Postman接口响应验证



# 发起接口请求



根据抓包信息，填写  
HTTP方法、URL、请  
求参数、以及请求地址  
之后，点击send发包，  
即可得到请求的响应返  
回信息。

百度IP接口 Examples (0)

GET https://sp0.baidu.com/8aQDcjqpAAV3otqbppnN2Djv/api.php?query=12.... Send Save

Params Authorization Headers (1) Body Pre-request Script Tests Cookies Code Comments (0)

Query Params

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/>	query	12.1.1.1			
<input type="checkbox"/>	co				
<input checked="" type="checkbox"/>	resource_id	6006			
<input type="checkbox"/>	t	1552892901207			
<input type="checkbox"/>	ie	utf8			
<input type="checkbox"/>	oe	gbk			
<input type="checkbox"/>	format	json			
<input type="checkbox"/>	tn	baidu			
<input type="checkbox"/>	-	1552878312084			
	Key	Value	Description		

Body Cookies (1) Headers (8) Test Results (1/1) Status: 200 OK Time: 100 ms Size: 573 B Save Download

Pretty Raw Preview JSON ≡ 🔍

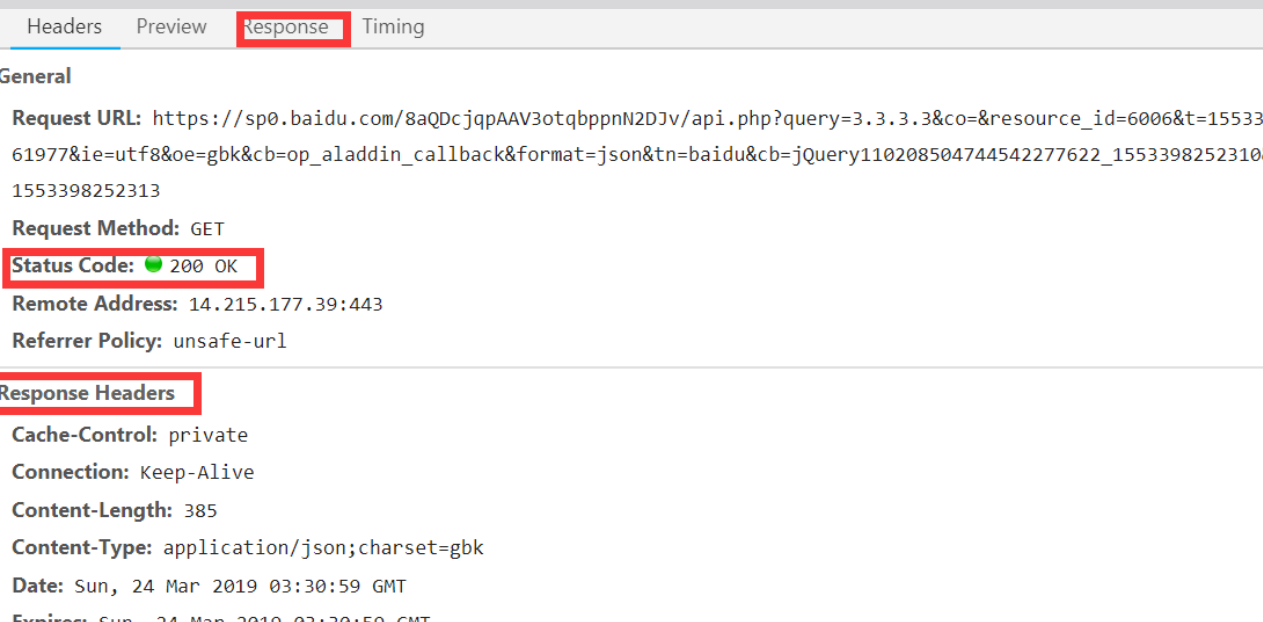
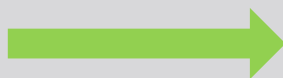
```
1 {  
2   "status": "0",
```



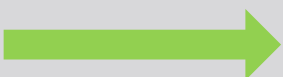
# 获取响应返回信息

接口的响应包中，测试关注的重点在于响应的正文主体，同时可能对响应头和状态码进行校验。

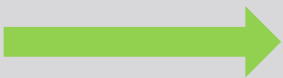
状态码



响应头



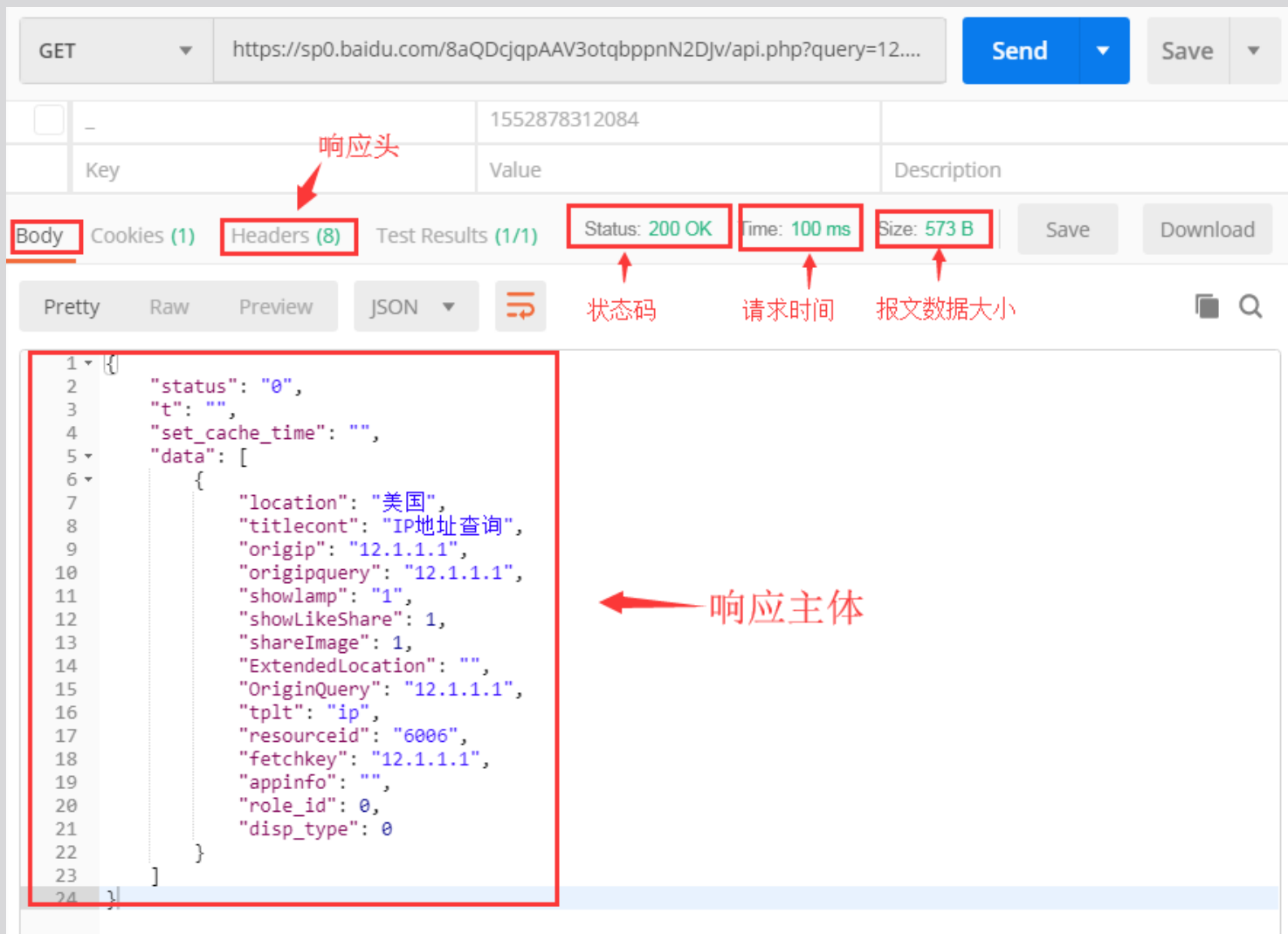
响应正文



# Postman获取响应

Postman请求发起之后将得到响应结果，包括响应主体、响应头、cookies、状态码、请求耗时、报文数据大小等信息。

而接口测试，主要就是  
对响应得到的信息进行验证，  
是否与用例设计预期相符。



GET https://sp0.baidu.com/8aQDcjqpAAV3otqbppnN2DJv/api.php?query=12.... Send Save

Key	Value	Description
-	1552878312084	

Body Cookies (1) Headers (8) Test Results (1/1) Status: 200 OK Time: 100 ms Size: 573 B Save Download

Pretty Raw Preview JSON

```
1 {
2   "status": "0",
3   "t": "",
4   "set_cache_time": "",
5   "data": [
6     {
7       "location": "美国",
8       "titlecont": "IP地址查询",
9       "origip": "12.1.1.1",
10      "origipquery": "12.1.1.1",
11      "showlamp": "1",
12      "showLikeShare": 1,
13      "shareImage": 1,
14      "ExtendedLocation": "",
15      "OriginQuery": "12.1.1.1",
16      "tplt": "ip",
17      "resourceid": "6006",
18      "fetchkey": "12.1.1.1",
19      "appinfo": "",
20      "role_id": 0,
21      "disp_type": 0
22    }
23  ]
24 }
```

响应头

状态码

请求时间

报文数据大小

响应主体





# Postman接口请求实战

## 百度IP接口实战

接口地址：<https://sp0.baidu.com/8aQDcjqpAAV3otqbppnN2DJv/api.php>

## 需要设置头域的请求实战

腾讯课堂评论接口，获取信息时需要添加头域referer，才能获取到正确的评论信息。

接口地址：[https://ke.qq.com/cgi-bin/comment\\_new/course\\_comment\\_list](https://ke.qq.com/cgi-bin/comment_new/course_comment_list)

访问地址：<https://ke.qq.com/course/315793?tuin=227706b0>

## Post接口实战

使用用友云提供的新发债券查询接口进行测试。

接口地址：

[https://api.yonyoucloud.com/apis/pte/New\\_Issue\\_Bond\\_Information/New\\_Issue\\_Bond\\_Information](https://api.yonyoucloud.com/apis/pte/New_Issue_Bond_Information/New_Issue_Bond_Information)

接口申请描述地址：

[https://api.yonyoucloud.com/apilink/tempServicePages/b1277b7e-b1e3-445c-818a-cdcd457f298c\\_true.html](https://api.yonyoucloud.com/apilink/tempServicePages/b1277b7e-b1e3-445c-818a-cdcd457f298c_true.html)

## Json格式请求发包实战

接口地址：[http://api.test.zhulogic.com/designer\\_api/account/login\\_quick](http://api.test.zhulogic.com/designer_api/account/login_quick)

网站访问地址：<http://test.zhulogic.com/login/>





# Post接口请求实战

## 请求体与Content-Type头域

之前案例中使用的GET方法接口都是通过在URL中添加请求参数的方式完成参数传递。在对使用POST方法的接口进行测试时，需要特别注意请求参数在body中的格式以及请求头中的Content-Type头域，该头域规定了接口接受的请求参数传值格式。

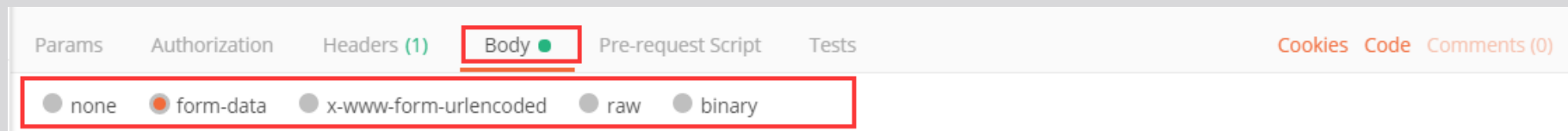
在Postman请求体中传递参数时，在body菜单中设置请求体的格式。

x-www-form-urlencoded对应于Content-Type头域为x-www-form-urlencoded的类型，是以键值对形式发送的表单参数，同时参数会携带在url中。

form-data对应于Content-Type的multipart/form-data类型，既可以发送键值对也可以进行文件参数传递。

raw选项中可以使用的请求体原始格式编辑各Content-Type类型对应的参数格式，直接按请求体的格式来进行内容发送。

binary选项用于发送文件内容请求。



# Post接口请求实战



## Post接口实战

使用用友云提供的新发债券查询接口进行测试。

接口地址：

[https://api.yonyoucloud.com/apis/pte/New\\_Issue\\_Bond\\_Information/New\\_Issue\\_Bond\\_Information](https://api.yonyoucloud.com/apis/pte/New_Issue_Bond_Information/New_Issue_Bond_Information)

接口申请描述地址：

[https://api.yonyoucloud.com/apilink/tempServicePages/b1277b7e-b1e3-445c-818a-cdcd457f298c\\_true.html](https://api.yonyoucloud.com/apilink/tempServicePages/b1277b7e-b1e3-445c-818a-cdcd457f298c_true.html)

## 文件上传接口发包实战

接口地址：

[http://www.testingedu.com.cn:8000/index.php/home/Uploadify/imageUp/savepath/head\\_pic/pictitle/banner/dir/images.html](http://www.testingedu.com.cn:8000/index.php/home/Uploadify/imageUp/savepath/head_pic/pictitle/banner/dir/images.html)

## Json格式请求发包实战

接口地址：[http://api.test.zhulogic.com/designer\\_api/account/login\\_quick](http://api.test.zhulogic.com/designer_api/account/login_quick)

网站访问地址：<http://test.zhulogic.com/login/>



## PART 3

# 断言与变量

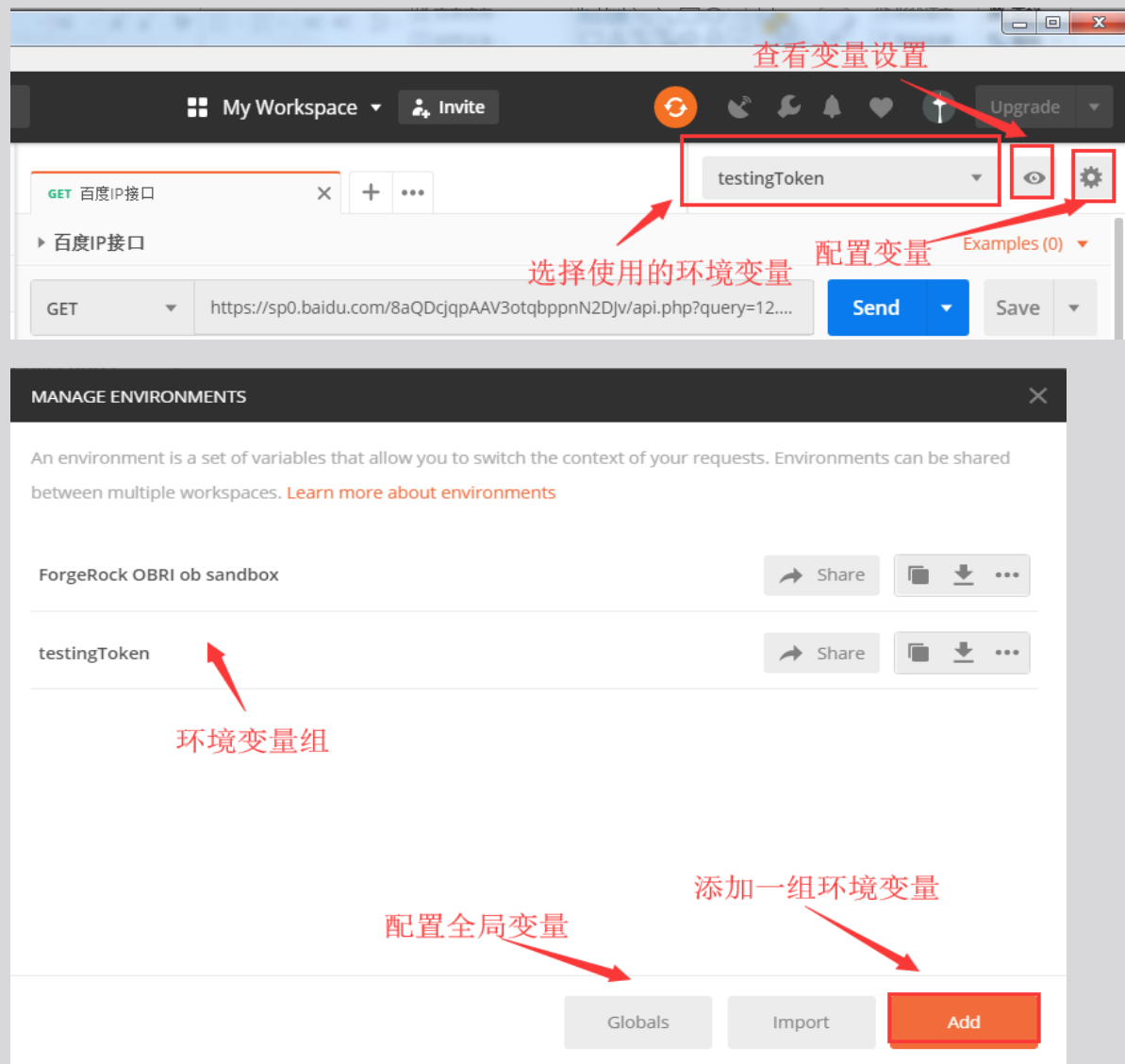


# 环境变量与全局变量



接口测试过程中，经常需要使用已经设定好的参数或者从其他接口的返回中获取变化的参数。Postman提供了变量的机制供这种场景使用。

Postman变量分为环境变量和全局变量两种，在接口请求中可以选择一组环境变量使用。而全局变量可供所有请求使用。  
使用变量的格式为 `{{变量名}}` 调用变量的值。



# 测试沙箱实战



## 测试沙箱常用功能

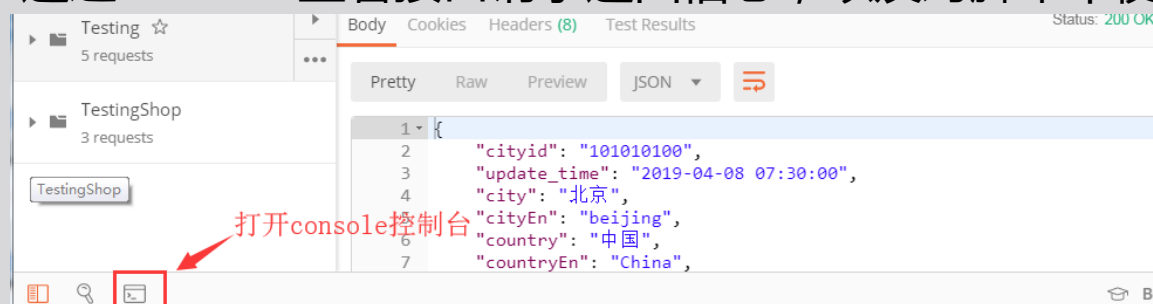
Postman测试沙箱其实是结合JS脚本完成测试中的功能，在请求发起前后实现部分测试操作。

常用功能：

- 请求前脚本（pre-request scripts）设置请求前置操作如设置变量等。
- 请求后对状态码、响应头、响应正文等信息进行断言操作。

## 使用console控制台进行调试

通过console查看接口请求返回信息，以及对脚本中使用的变量进行输出调试等操作。



## 案例实战



# 返回信息断言验证



为了更直观地进行接口测试，可以设置断言，来完成对响应结果的校验。

- 设置tests脚本，在右侧snippets中，提供了许多场景的脚本模板可以直接套用。
- 发包之后在响应面板中将得到测试结果。
- 官方文档地址：  
<https://go.pstmn.io/docs-test-scripts>

百度IP接口

GET <https://sp0.baidu.com/8aQDcjqpAAV3otqbppnN2DJv/api.php?query=12....> Send Save

Params Authorization Headers (1) Body Pre-request Script Tests Cookies Code Comments (0)

```
1 pm.test("IP地址是美国", function () {
2   var jsonData = pm.response.json();
3   console.log(jsonData.data);
4   pm.expect(jsonData.data[0].location).to.eql("美国");
5 });
6
7 pm.test("状态码是 200", function () {
8   pm.response.to.have.status(200);
9 });
```

Test scripts are written in JavaScript, and are run after the response is received.  
[Learn more about tests scripts](#)

SNIPPETS

- Clear a global variable
- Send a request
- Status code: Code is 200
- Response body: Contains string
- Response body: JSON value check
- Response body: Is equal to a string
- Response headers: Content-Type header check
- Response time is less than 200ms

Body Cookies (1) Headers (8) Test Results (2/2) Status: 200 OK Time: 51 ms Size: 573 B Save Download

All Passed Skipped Failed

- PASS IP地址是美国
- PASS 状态码是 200

脚本编辑窗

脚本模板

断言脚本执行结果





## PART 4

# 测试管理-测试集与数据驱动



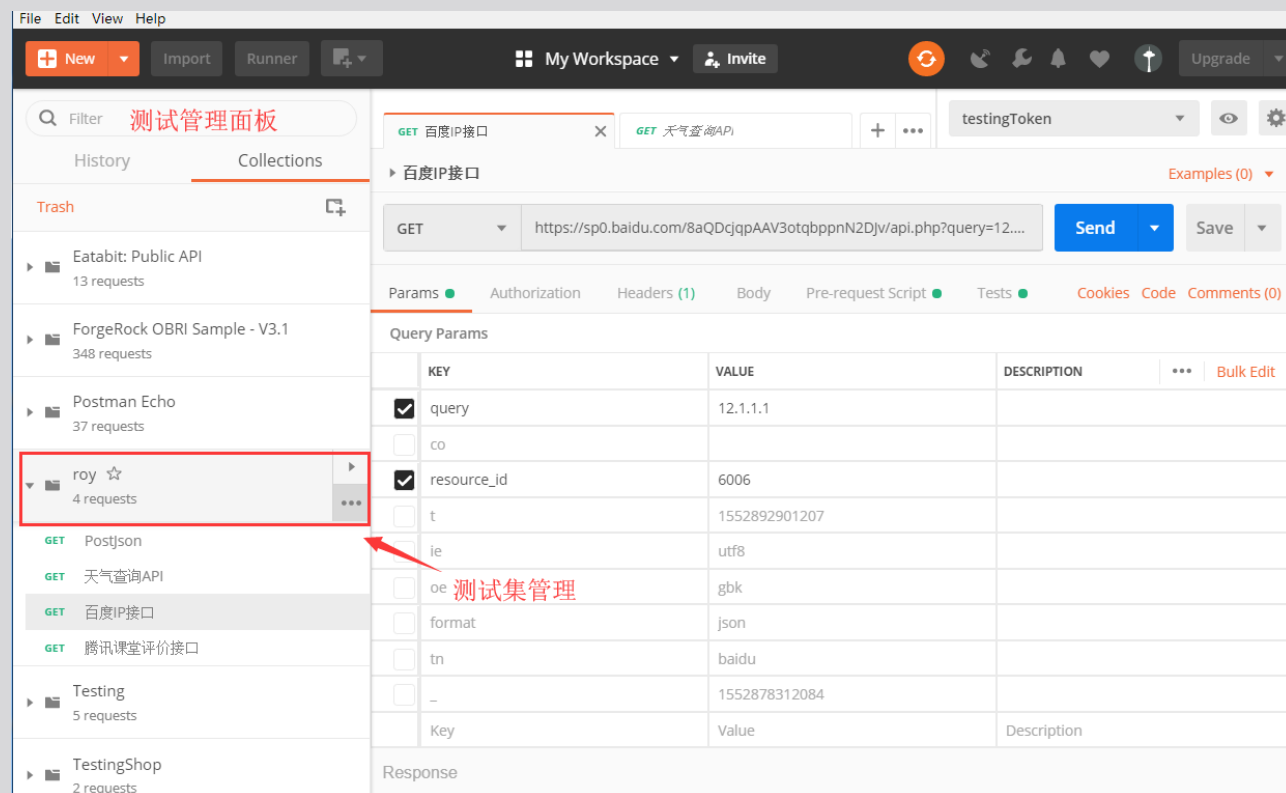


# Postman测试集

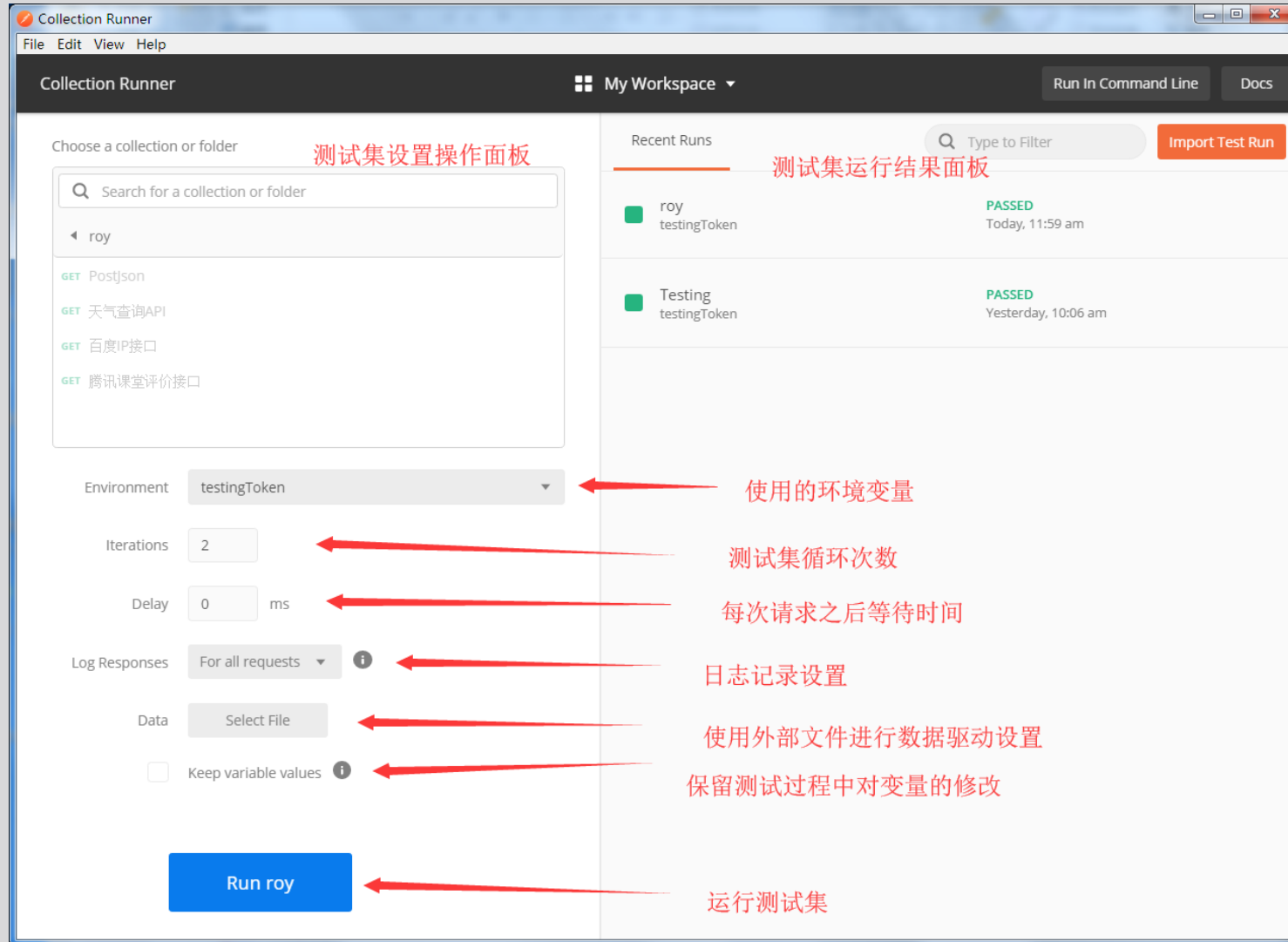


## 测试集使用

为了方便地管理接口请求的执行，可以通过postman测试集（collection）来完成测试的操作，每一个测试请求可以看成一个测试用例（testcase），collections能够一次管理多个测试用例来执行。通过save，能够将请求保存到collections中使用。



# Postman测试集使用



# Postman数据驱动



当需要进行数据驱动方式运行多次Postman脚本时，可以通过csv文件或者json格式的文件对接口测试数据进行管理，以方便用例管理执行。

## 接口请求使用数据驱动数据

在接口请求中，参数使用变量名获取，之后在创建数据文件时，使用这些变量名编辑测试数据。

Params Authorization Headers Body Pre-request Script Tests Cookies			
Query Params			
	KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/>	version	v1	
<input checked="" type="checkbox"/>	city	{{cityname}}	

在测试沙箱中，参数的获取用data.参数名的形式来进行读取。

```
1 pm.test("验证返回中city的值是", data.cityname, function () {
2     var jsonData = pm.response.json();
3     pm.expect(jsonData.city).to.eql(data.cityname);
4 });
```



# Postman数据驱动



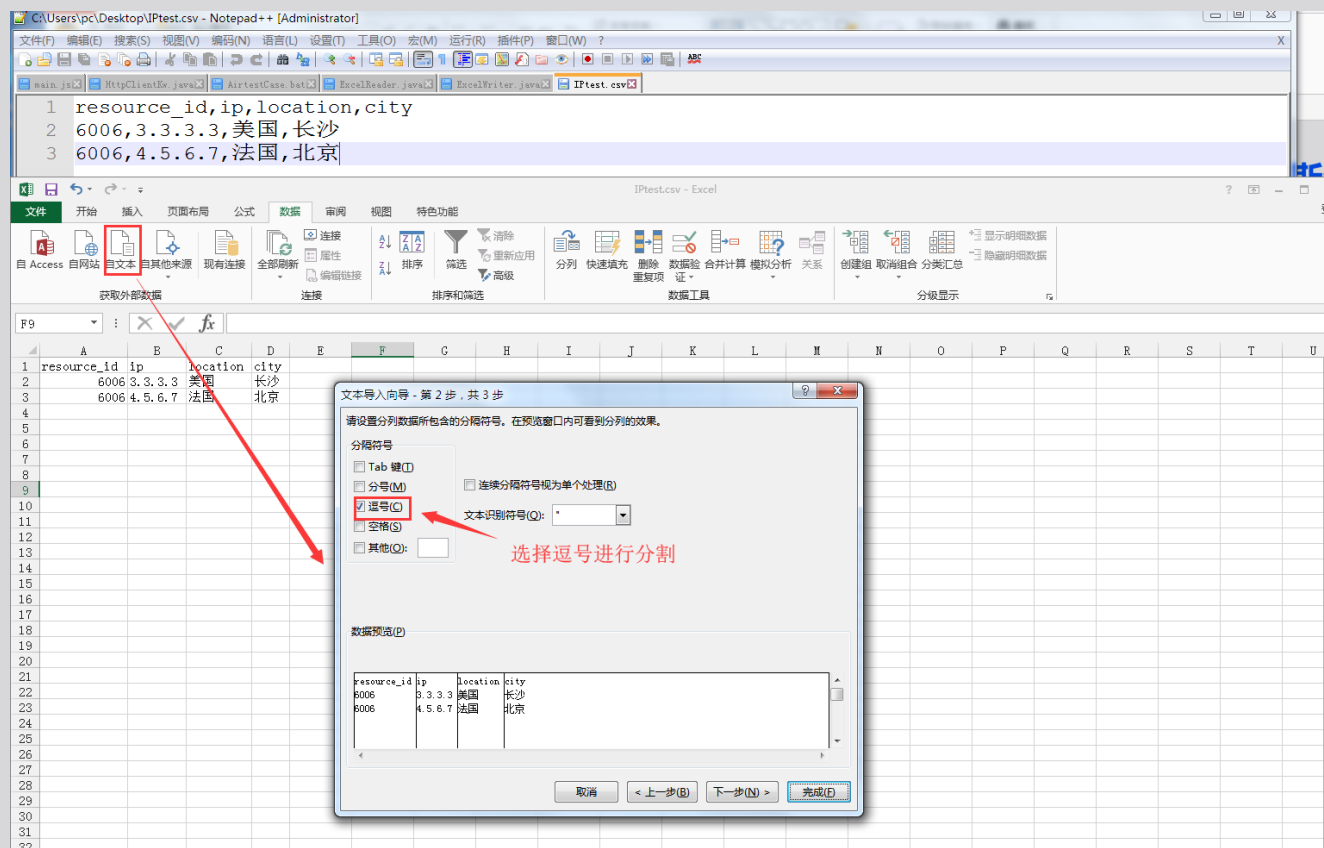
当需要进行数据驱动方式运行多次Postman脚本时，可以通过csv文件或者json格式的文件对接口测试数据进行管理，以方便用例管理执行。

## 创建测试数据文件

创建一个csv格式文件之后，通过notepad++或者excel进行编辑。

注意:文件需要使用**UTF-8编码**，创建了csv文件之后，在notepad++中使用编码菜单中的转为UTF-8编码功能将文件转为UTF-8编码使用。

文件中第一行必须是接口请求使用的变量名，后续每行填写本次执行使用的变量值。



# Postman数据驱动



## 使用测试数据文件

在collection runner中data选项选择编辑好的测试数据文件，preview能够查看文件预览，确保文件读取内容无误。

Collection Runner

My Workspace

Run In Command Line Docs

DDTTest

GET 百度ip

GET 天气API

Recent Runs

Type to Filter

Import Test Run

You don't have any runs yet. Select a collection or folder to start a run.

PREVIEW DATA

Iteration	resource_id	ip	location	city
1	6006	"3.3.3.3"	"美国"	"长沙"
2	6006	"4.5.6.7"	"法国"	"北京"

Environment: No Environment

Iterations: 2

Delay: 0 ms

Log Responses: For all requests

Data: Select File IPtest.csv X

Data File Type: text/csv Preview

☐ Keep variable values

预览面板

选择测试数据管理文件

选择文件格式，preview进行预览



## PART 5

### 进阶实战-cookie篇





# 进阶实战-cookie篇



在接口测试中某些接口的调用需要使用已有cookie，Postman操作cookie可以有两种方式。

## Postman接口请求使用Cookie

- 1、直接在头域中添加cookie头域，适用于已知请求时的cookie头域的情况
- 2、使用Postman的cookie管理机制，既可以手动添加，同时Postman也会将操作流程中获取的参数自动保存，因此可以通过调用前置接口来完成cookie获取。

## 电商接口测试实例

使用电商项目进行测试，在登录接口完成登录之后，通过购物车接口获取购物车中的信息。由于购物车接口需要使用登录后的cookie保持登录状态。

登录接口：

`http://www.testingedu.com.cn:8000/index.php?m=Home&c=User&a=do_login&t=0.5404187243051779`

购物车接口：

`http://www.testingedu.com.cn:8000/index.php?m=Home&c=Cart&a=header_cart_list`





## PART 6

### 进阶实战-token与参数传递篇



# 进阶实战-token篇



## 接口测试项目实例

使用接口测试项目实例，注册、登录、获取用户信息、登出接口均需要获取auth接口获取到的token进行使用，因此通过auth接口返回信息获取token之后，存为变量供后续接口调用。

鉴权接口：<http://testingedu.com.cn/inter/HTTP/auth>

该接口返回一个token值，之后的所有接口请求均需添加token头域使用该token

注册接口：<http://www.testingedu.com.cn/inter/HTTP/register>

登录接口：<http://testingedu.com.cn/inter/HTTP/login>

该接口返回一个userid，在请求用户信息时使用，因此保存为变量。

获取用户信息接口：<http://www.testingedu.com.cn/inter/HTTP/getUserInfo>

该接口需要使用登录获取的userid，因此使用保存的userid变量值。

注销接口：<http://testingedu.com.cn/inter/HTTP/logout>



## PART 7

# Postman持续集成



# Newman安装



## Newman安装

Postman使用newman插件完成命令行执行postman脚本，因此如果需要用命令行运行脚本，需要首先安装Newman。

官方文档：<https://www.npmjs.com/package/newman>

### 1、安装nodejs

官网下载地址：<http://nodejs.cn/download/>，下载之后直接运行安装。

安装之后确认环境变量中已经添加了node所在目录，否则手动添加。

在cmd中通过node -v 和 npm -v命令确认node环境已经搭建成功

```
管理员: C:\Windows\system32\cmd.exe
Microsoft Windows [版本 6.1.7601]
版权所有 (c) 2009 Microsoft Corporation。保留所有权利。

C:\Users\pc>node -v
v9.9.0

C:\Users\pc>npm -v
5.6.0
```



# Newman安装



## Newman安装

Postman使用newman插件完成命令行执行postman脚本，因此如果需要用命令行运行脚本，需要首先安装Newman。

官方文档：<https://www.npmjs.com/package/newman>

## 2、安装newman

完成nodejs安装之后，在命令行中输入`npm install newman -g`自动完成newman的安装。

```
C:\Users\pc>npm install newman -g
F:\android\npm\newman -> F:\android\npm\node_modules\newman\bin\newman.js
+ newman@4.4.1
```



# Newman执行测试



## Newman运行postman collections

安装好newman之后，可以通过命令行执行postman测试集中的脚本

官方文档：<https://www.npmjs.com/package/newman>

Postman命令行运行文档：

[https://learning.getpostman.com/docs/postman/collection\\_runs/command\\_line\\_integration\\_with\\_newman/](https://learning.getpostman.com/docs/postman/collection_runs/command_line_integration_with_newman/)

### 1、导出collections

在collection选项菜单中选择export，导出为json文件

### 2、newman执行collection导出的json文件

命令行中输入 `newman run 导出的json文件名` 通过newman命令行运行测试集

```
C:\Users\pc>newman run D:\DDTtest.postman_collection.json  
newman
```



# Newman执行测试

Newman运行时，可以通过运行参数完成某些场景的操作。所有参数可参考官方文档。或者通过newman run -h命令来查看参数列表说明。

官方文档：<https://www.npmjs.com/package/newman>

## Newman常用参数

**-e, --environment** 使用环境变量文件或url

使用时，首先在环境变量中通过导出功能将环境变量导出，然后用如下命令引用。

```
C:\Users\pc>newman run D:\Roy.postman_collection.json -e D:\百度ip环境.postman_environment.json  
newman
```

**-g, --global** 使用全局变量文件或url

类似环境变量使用，在使用前，先导出全局变量文件，再通过-g指定文件使用

**-d, --iteration-data** 指定使用的数据驱动文件

```
C:\Users\pc>newman run D:\DDTtest.postman_collection.json -d D:\DDTData.csv  
newman
```

**-n** 指定测试集循环次数iterations

```
C:\Users\pc>newman run D:\DDTtest.postman_collection.json -d D:\DDTData.csv -n 2  
newman
```





# Newman测试报告



Newman运行时，可以通过运行参数完成某些场景的操作。所有参数可参考官方文档。  
或者通过newman run -h命令来查看参数列表说明。

官方文档：<https://www.npmjs.com/package/newman>

## Newman测试报告

Newman提供的测试报告中，常用的包括cli、json和html三种格式，  
可以通过-r或者--reporter命令指定所用格式

cli格式为在cmd客户端呈现的报告模板，newman默认使用该格式。

json格式可以导出json格式的报告，通过-r json进行设置，  
并加上--reporter-json-export参数指定报告生成的路径。

```
C:\Users\pc>newman run D:\DDTtest.postman_collection.json -d D:\DDTData.csv -r json --reporter-json-export D:\result.json
```

html格式可以导出html格式的报告，通过-r html进行设置，  
并加上--reporter-html-export参数指定报告生成的路径。  
在使用之前需要通过npm install -g newman-reporter-html先完成安装。

```
C:\Users\pc>npm install -g newman-reporter-html
```

```
C:\Users\pc>newman run D:\DDTtest.postman_collection.json -d D:\DDTData.csv -r html --reporter-html-export D:\result.html
```



# 定时任务执行脚本

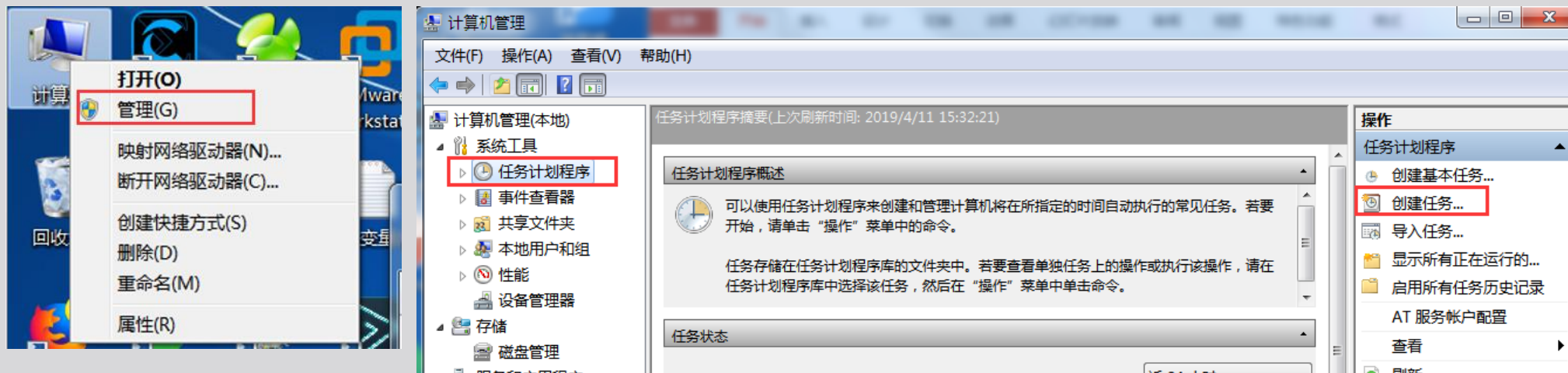
在实现了newman对测试集的命令行运行之后，可以通过编写bat脚本，将其设置为windows定时任务，以完成测试脚本的定时运行。

## 1、编写bat脚本

将newman运行测试集的命令编写为bat脚本保存

## 2、设置定时任务

在计算机上右键->管理，在弹出菜单里的任务计划程序里创建定时任务配置执行。



# Jenkins持续集成



在已经实现newman通过命令行方式执行postman测试集的基础上，可以方便的在Jenkins上完成持续集成构建。

Jenkins官网下载地址：<https://jenkins.io/download/>

持续集成及Jenkins安装过程需要学习的同学可以添加技术交流QQ群：691499681  
或联系卡卡老师QQ：632232258领取免费视频课程。

## 构建任务完成Jenkins定时执行Newman运行测试集

创建自由风格项目后，在构建触发器里，配置定时构建选项，指定任务定时执行。  
五个参数分别代表 分 时 天 月 星期 的周期维度。

☒ 定时构建

Schedule

在构建时，添加构建步骤选择执行windows批处理命令，将newman命令写入即可

Build

执行 Windows 批处理命令

Command

编辑系统变量

变量名 (N):

变量值 (V):

确定 取消

注意：如果Jenkins输出中文乱码，在环境变量中，添加一个变量JAVA\_TOOL\_OPTIONS  
值为-Dfile.encoding=UTF-8，重启Jenkins。



## 附录

### Postman不常用功能



# Postman抓包

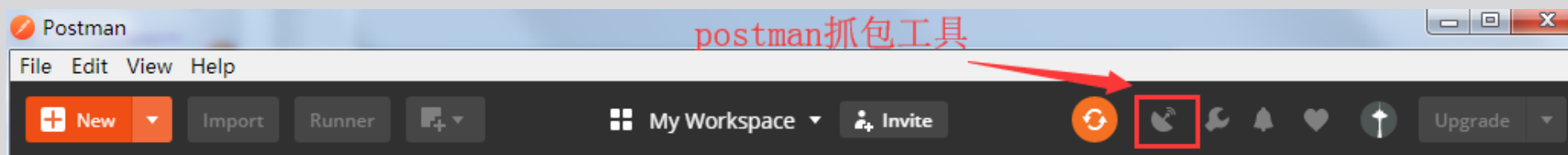


## 抓包获取接口信息

对接口进行测试的第一步是获取接口信息，最好通过接口文档获取，并且通过抓包确认。  
抓包工具：fiddler、charles、chrome浏览器开发者工具，也可使用postman进行抓包。

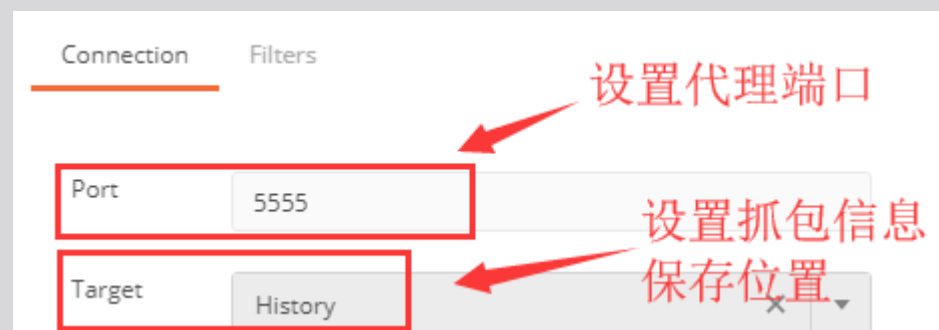
需要抓包工具课程的同学可以添加技术交流QQ群：691499681  
或联系卡卡老师QQ：632232258领取免费视频课程。

## Postman抓包方法



1、打开抓包工具后，设置postman代理使用的端口以及抓包信息保存的文件夹。

2、设置浏览器使用postman配置的127.0.0.1:5555代理进行访问，抓包信息就能在指定的文件夹中显示。



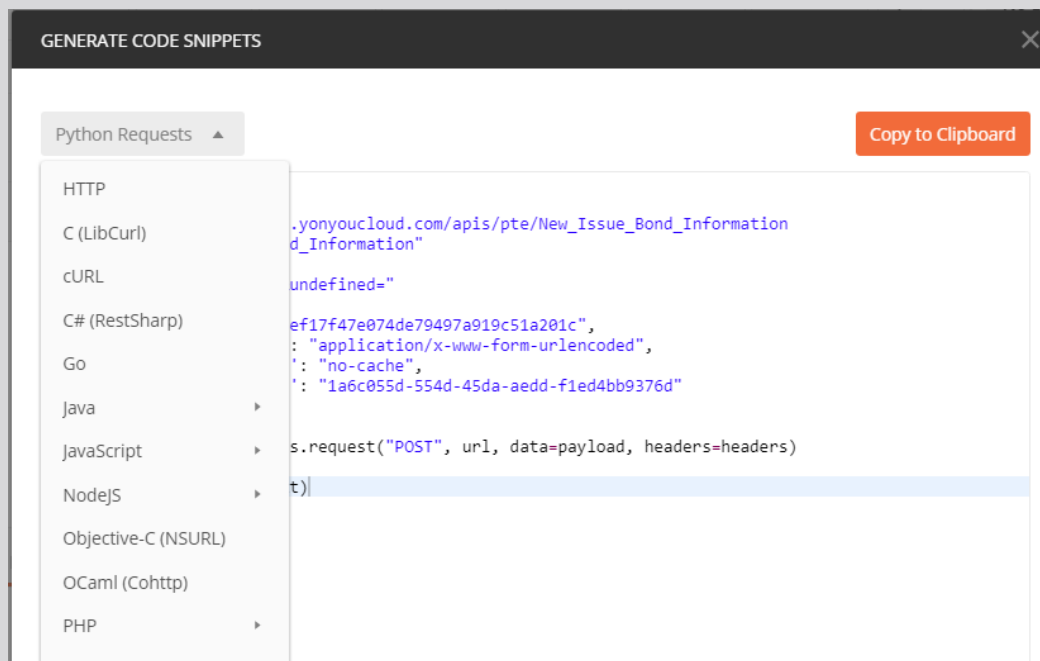
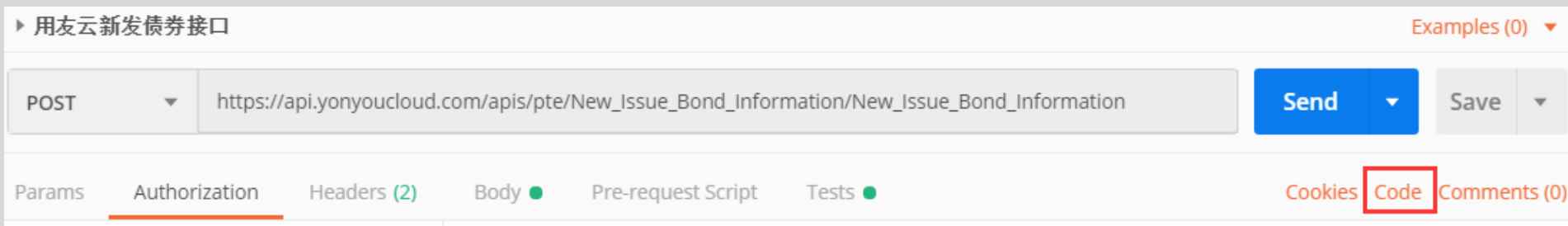


# Postman导出脚本



## Postman脚本导出

Postman的脚本可以通过导出功能导出为多种语言代码脚本



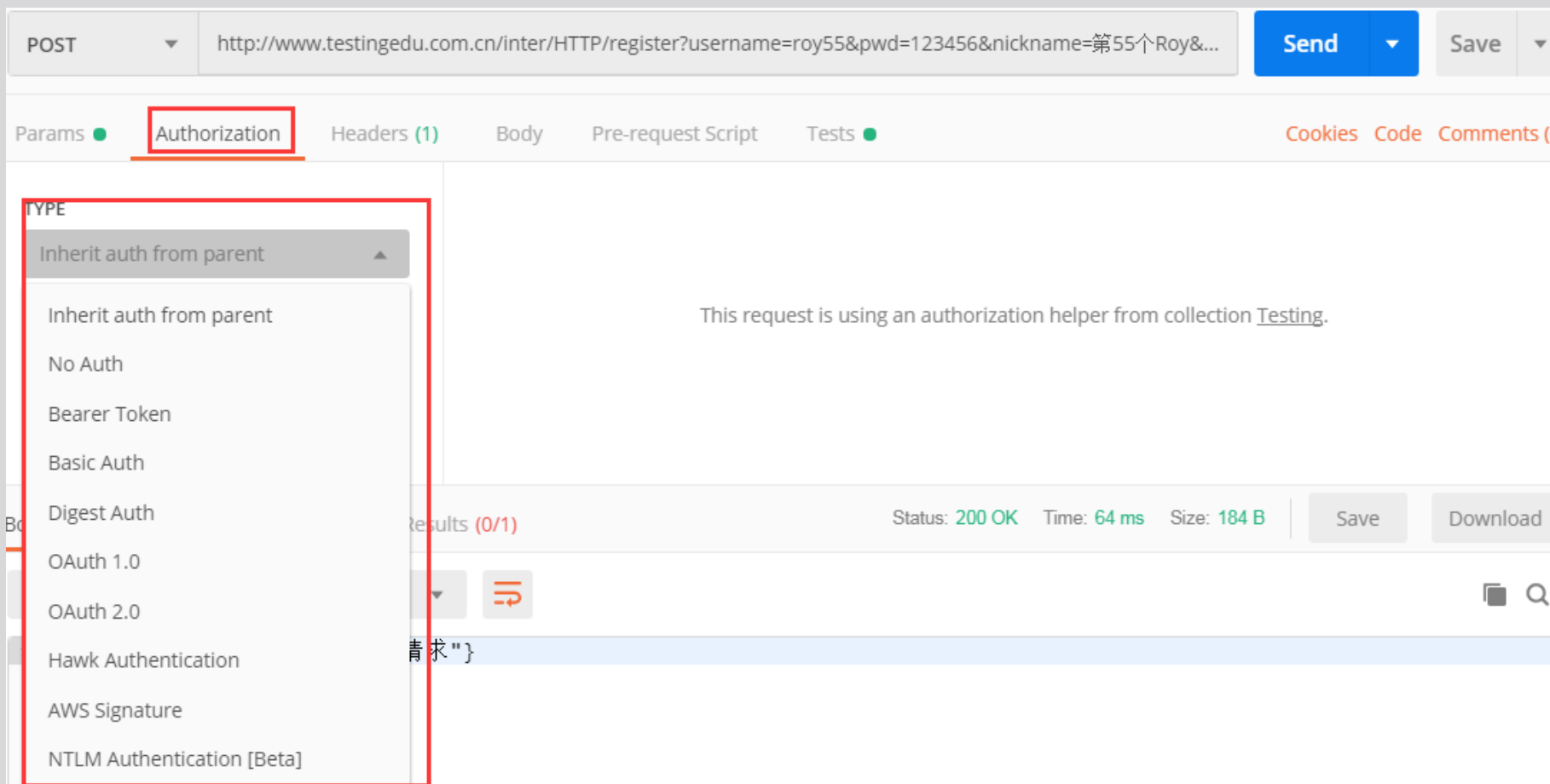


# Postman接口鉴权



## Postman接口鉴权

某些接口可能在调用时需要鉴权之后使用，根据接口规则，在postman中可以通过请求面板中的authorization菜单进行相应设置，通常无须设置。



# THANKS

特斯汀软件测试