



Quick Start

Open API > Quick Start

Version: 20200108

Contents

1 Overview	1
2 Environmental Preparation	2
3 Using OpenAPI to Control a Device	4
3.1 Preparation	4
3.2 Getting access_token in Simple Mode	4
3.3 Checking the Command Sets Supported by the Device	5
3.4 Controlling the Device	5
3.5 Checking the Current Device Status	6
4 Using OpenAPI to Pull Data	7
4.1 Creating a User	7
4.2 Getting User Device Data	8
4.3 Getting User Data	8
5 Summary	10

1 Overview

This documentation introduces the Tuya Cloud API (openAPI) and the tool (Postman) used on the application-end (PC) to send HTTPS requests and create users, acquire user devices and their information, remotely control devices, and other functions.

Before starting to code, developers can use the HTTPS request tool Postman to call the openAPI interface and quickly get a feel for Tuya Cloud API's external output ability as well as an understanding of whether it suits your Cloud interfacing needs.

2 Environmental Preparation

For this step, assume that you have already registered a developer account on the Tuya Developer Platform and that you have already applied for Cloud API authorization;

Before starting, you should first prepare the environments required by Postman and Tuya Cloud by doing the following:

- Download Postman: Go to the Postman official website and download the Postman client. Postman official website: <https://www.getpostman.com>
- Import the Tuya Cloud API interface package:

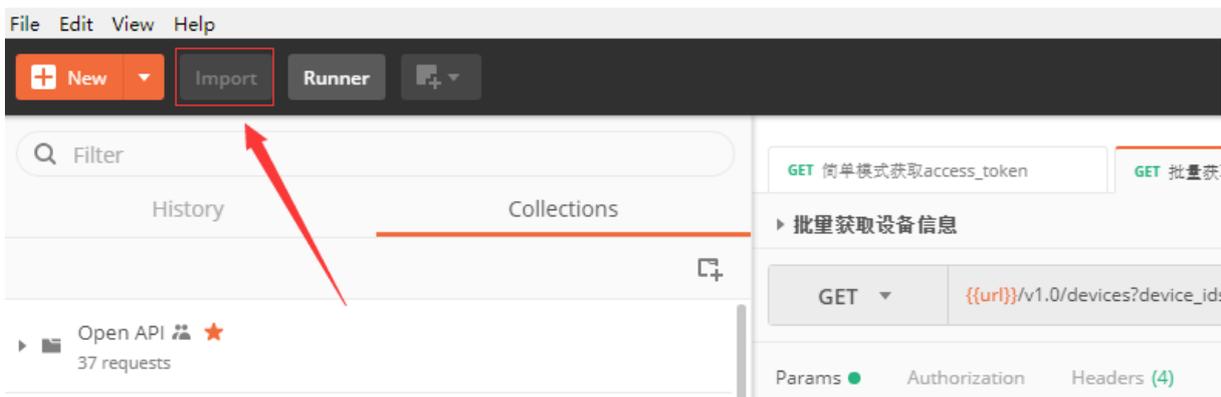


Figure 1: Postman API Import

- Import the Tuya Cloud API environment package:

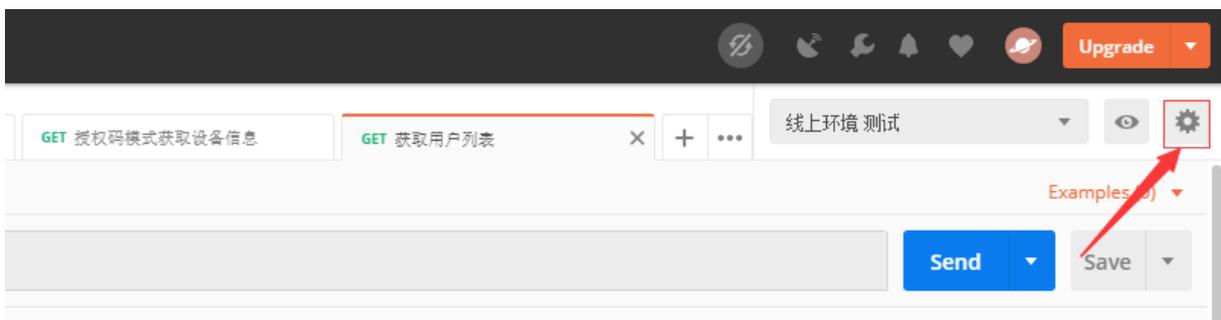


Figure 2: Postman Environment Import

- Add essential data: Just open the environment, select the imported Tuya environment, and enter clientId (accessId), secret (accessKey), deviceId, schema:

	VARIABLE	INITIAL VALUE ⓘ	CURRENT VALUE ⓘ	⋮	Persist All	Reset All
<input checked="" type="checkbox"/>	url	https://openapi.tuyac...	https://openapi.tuyacn.com			
<input checked="" type="checkbox"/>	clientId					
<input checked="" type="checkbox"/>	secret					
<input checked="" type="checkbox"/>	deviceId					
<input checked="" type="checkbox"/>	schema					
	Add a new variable					

Figure 3: Postman Add essential data

At this point, the Postman environment is now ready and you can move on to the next stages. If you still have questions about Postman, you can find more information online. Postman is an outstanding tool for allowing programmers to quickly understand an API.

3 Using OpenAPI to Control a Device

Before controlling a device, you need to create a product on the Tuya Developer Platform and create a virtual device under the product. Then you can use the API to control the device.

3.1 Preparation

- Create a product
- Select an App interface
- Scan the QR code below “Please scan with the Tuya Smart App to experience mobile control”
- In the Tuya Smart app, go to the scanned device details page and acquire the device’s virtual ID

3.2 Getting access_token in Simple Mode

Select the “Get access_token in Simple Mode” interface from the cloud-to-cloud interface API folder. Click “Send” as pictured below to send a request:

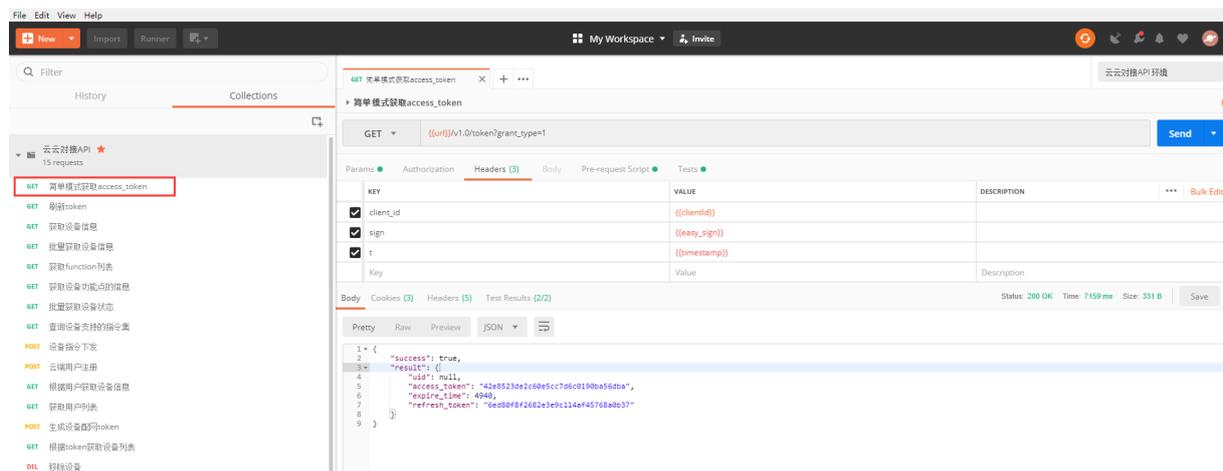


Figure 4: Postman Getting accessToken

If other interfaces prompt that the token has expired, click the interface and get the token again.

3.3 Checking the Command Sets Supported by the Device

Select the “Check command sets supported by the device” interface:

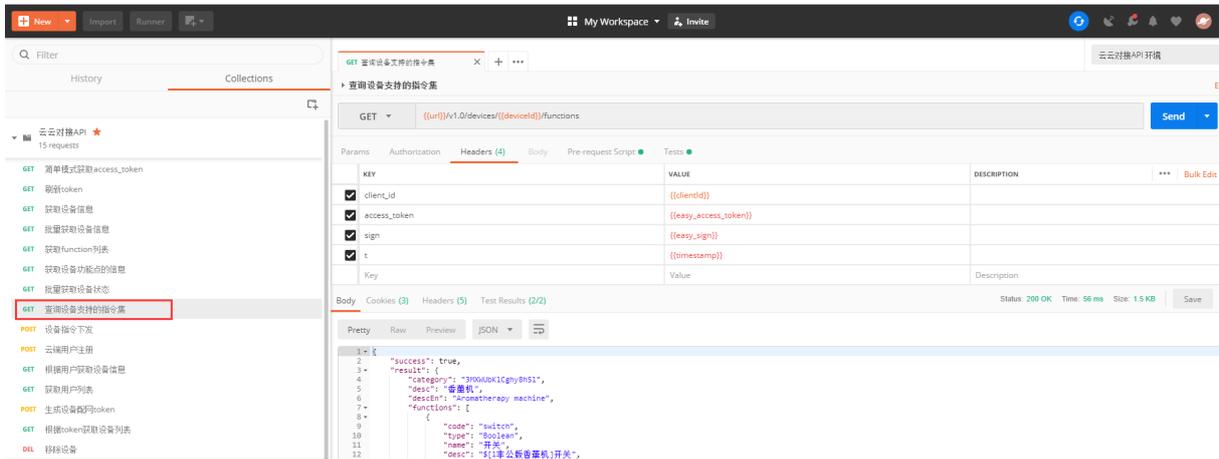


Figure 5: Postman command sets

3.4 Controlling the Device

Select the “Device command issuing” interface, click “Body”, select “raw”-“JSON(application/json)” type, and enter the content in json format:

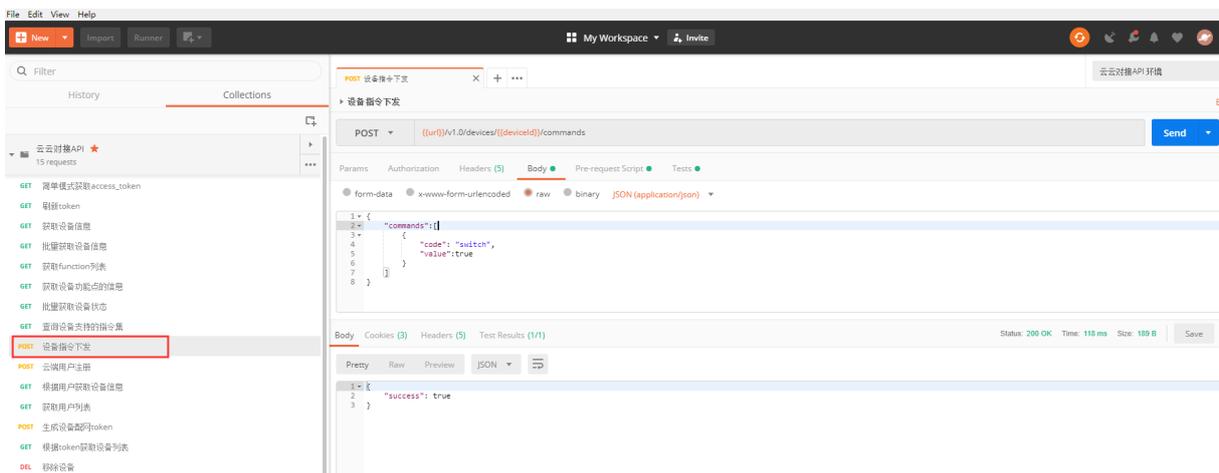


Figure 6: Postman Controlling the Device

3.5 Checking the Current Device Status

Select the “Get device function point info” interface:

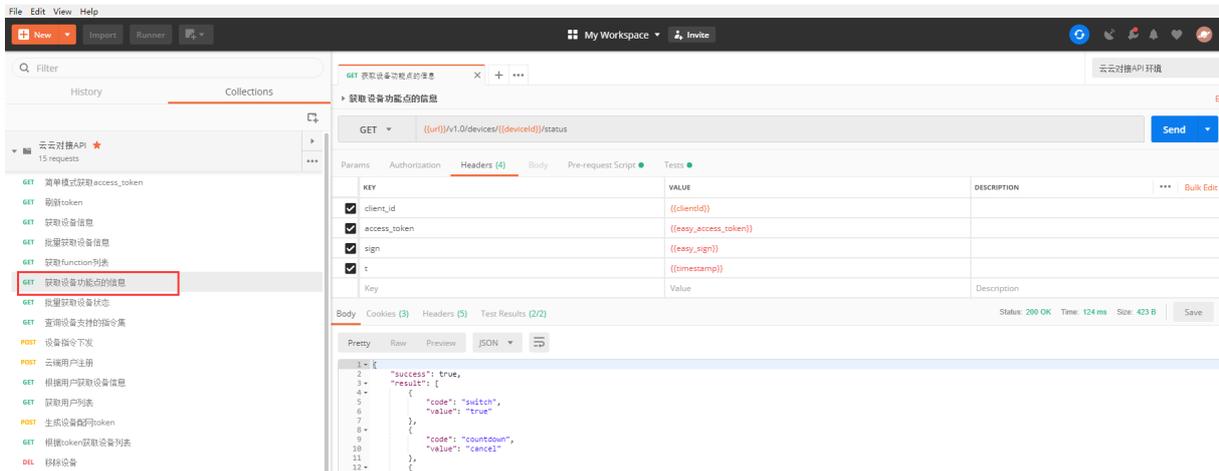


Figure 7: Postman Current Device Status

4 Using OpenAPI to Pull Data

Tuya Cloud provides numerous APIs to allow developers to get data from their users and devices. Users of the Tuya official App should note that the user data from Tuya Smart and Smart Life apps is owned by Tuya Smart, and developers are not entitled to acquire user data from the Tuya official app.

In terms of device data, Tuya provides two dimensions through which developers can pull data: the App dimension and the product dimension.

- App dimension: Users linked with devices are the developer's users on the Tuya Cloud app; developers have indirect permissions to operate the devices of their App users.
- Product dimension: the devices used by the products belong to the developer on Tuya Cloud product devices; developers have operation permissions on these devices.

Note: When you call a device's related interface and are prompted "Permission denied", check the two dimensions of permissions above and ensure you are conforming to them.

4.1 Creating a User

Select the "Cloud user registration" interface, click "Body", select "raw"-
"JSON(application/json)" type, and enter the content in json format:

The schema is the channel identifier of the App application. If you create an OEM App, please get it according to the package name. For example, if your App package name is com.aa.bb, then your schema is: aabb. If your App is based on a Tuya-based SDK, you can query the channel identifier under the SDK App details page on the platform.

- Note: For passwords you are recommended to use MD5 encryption

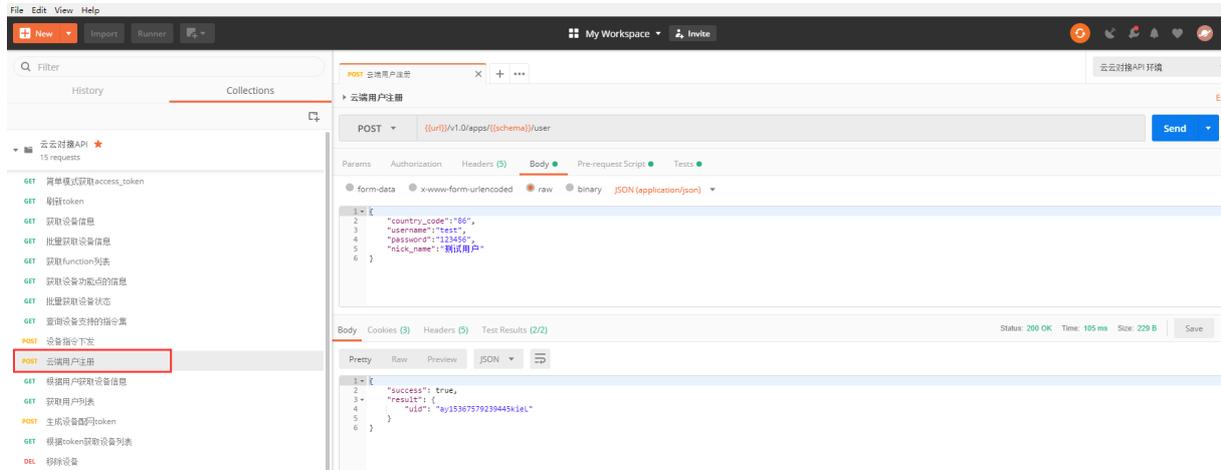


Figure 8: Postman Creating a User

4.2 Getting User Device Data

Select the “Get device info by user” interface:

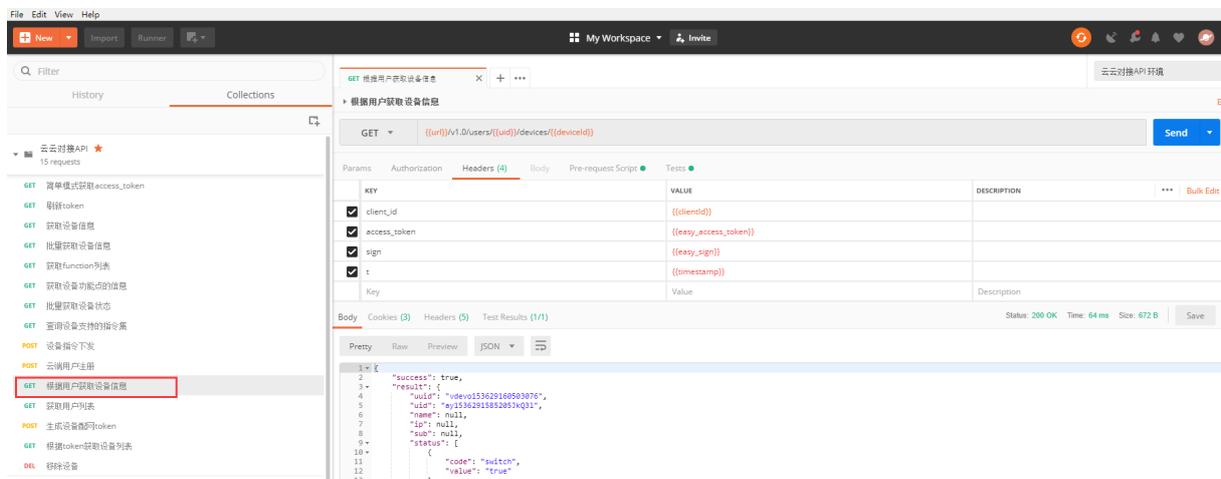


Figure 9: Postman Getting User Device Data

4.3 Getting User Data

Select the “Get user list” interface:

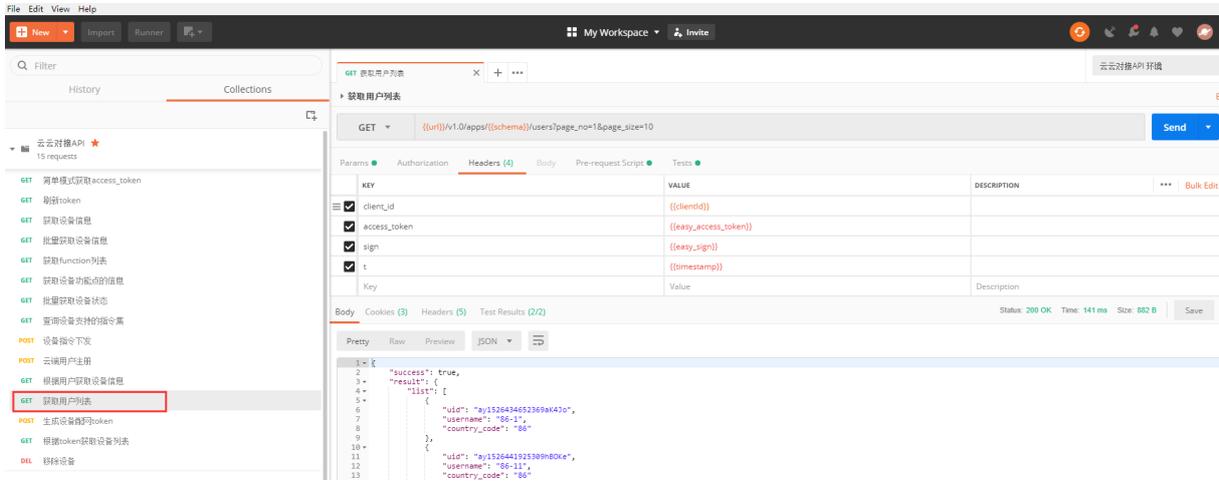


Figure 10: Postman Getting User Data

5 Summary

After following the simple tutorials above, you should have a broad understanding of the abilities provided by Tuya Cloud OpenAPI, and can now get coding, use the API, and do some related development.

For more interfaces, please refer to API lists. If you have any special requirements, please contact Tuya's commercial affairs team.

If you still have questions about calling interfaces, you can submit them to us on the Tuya Developer Platform. Please try to provide interface request parameters when doing so.