
Govee Developer API

Reference

Version 2.0

You can always get the latest API guide through the link below:

[https://govee-public.s3.amazonaws.com/developer-docs/
GoveeDeveloperAPIReference.pdf](https://govee-public.s3.amazonaws.com/developer-docs/GoveeDeveloperAPIReference.pdf)

Govee Team

Table of Contents

F&Q	1
Getting Started with the Govee Developer API	2
How to Use Govee Developer API Key.....	3
Rate Limiting (Lights, Plugs, Switches).....	4
Rate Limiting (Appliances)	5
Govee Developer API (Lights, Plugs, Switches)	6
DeviceList	6
DeviceControl	10
DeviceState	15
Govee Developer API (Appliances)	19
DeviceList	19
DeviceControl	20
Version History	21

F&Q

1. I have a Bluetooth device. It does not support Wi-Fi. Is there an available API?

No, only Wi-Fi devices support API.

2. When I use Govee Home App to control the brightness of the device, the API will not display the changed status.

The changed status can only be synchronized when the light is controlled via Wi-Fi, which means your phone Bluetooth needs to be turned off. Please recheck if you have turned off your phone Bluetooth.

3. I have a problem using a third-party software, I am not a developer, can you solve the problem with a third-party software.

We do not have the supported 3rd party software, who use our API to integrate. Since we are not involved in it, we could not diagnose the issue and solve the problem. It is advised to contact the third-party support directly to solve the problem.

4. How to refresh the API key?

Govee Developer API Key can be applied through Govee Home App. The key is refreshed every time you apply, and will be sent to the email address.

5. What can I do if the response status is not 200?

Please read the “message” in the Response Body first. If your question persists, please send us your Request Params and the value of “Date” and “X-traceId” showed in the Response Header, which would be much helpful for the debug.

Getting Started with the Govee Developer API

Use the Govee Developer API to control your devices and query your device state.

The device models supported by the current version are:

Lights, Plugs, Switches:

H6160, H6163, H6104, H6109, H6110, H6117, H6159, H7022, H6086, H6089, H6182, H6085, H7014, H5081, H6188, H6135, H6137, H6141, H6142, H6195, H7005, H6083, H6002, H6003, H6148, H6052, H6143, H6144, H6050, H6199, H6054, H5001, H6154, H6072, H6121, H611A, H5080, H6062, H614C, H615A, H615B, H7020, H7021, H614D, H611Z, H611B, H611C, H615C, H615D, H7006, H7007, H7008, H7012, H7013, H7050, H6051, H6056, H6061, H6058, H6073, H6076, H619A, H619C, H618A, H618C, H6008, H6071, H6075, H614A, H614B, H614E, H618E, H619E, H605B, H6087, H6172, H619B, H619D, H619Z, H61A0, H7060, H610A, H6059, H7028, H6198, H6049, H7031, H7032, H61A1, H61A2, H61B2, H7061, H6067, H6066, H6009, H7041, H7042, H604A, H6173, H615E, H604B, H6091, H7051, H7062, H618F, H605D, H6046, H601A, H61A3, H610B, H6047, H7065, H61E1, H6057, H604C, H6065, H605C, H705A, H705B, H7055, H61A5, H6078, H604D, H6168, H6601, H70B1, H61A8

Appliances:

H7121, H7122, H7123, H7120, H7141, H7142, H7130, H7131, H7132, H7150, H7160, H7101, H7111

Here are a few things to know before you start.

1. How to use Govee Developer API key
2. Rate Limiting

How to Use Govee Developer API Key

The Govee Developer API Key is used for authentication. Requests that require authentication will return *401Unauthorized*.

Use http header 'Govee-API-Key' to send Govee Developer API Key. You can learn how to use it in the API examples.

Rate Limiting (Lights, Plugs, Switches)

The returned HTTP headers show your current rate limit status:

RateLimit Status	Description
RateLimit-Limit	The maximum number of requests you're permitted to make.
RateLimit-Remaining	The number of requests remaining in the current rate limit window.
RateLimit-Reset	The time at which the current rate limit window resets in UTC epoch seconds .

Note: There are two levels of limits working together. Please make sure your design would not reach any level of the limits. Otherwise, the HTTP response status will be 429:

- (1) X-RateLimit-Limit, X-RateLimit-Remaining, X-RateLimit-Reset (this level of limit works on all APIs, which means total requests from one user can not be more than 10000 per day).

Works on All APIs	10000 per day
-------------------	---------------

- (2) API-RateLimit-Limit, API-RateLimit-Remaining, API-RateLimit-Reset.

DeviceList	10 per minute
DeviceControl	10 per minute per device
DeviceState	10 per minute per device

Rate Limiting (Appliances)

The rules are the same as that of “Rate Limiting” on page 4.

(2) X-RateLimit-Limit, X-RateLimit-Remaining, X-RateLimit-Reset

Works on All APIs	100 per day
-------------------	-------------

(2) API-RateLimit-Limit, API-RateLimit-Remaining, API-RateLimit-Reset.

DeviceList	10 per minute
DeviceControl	10 per minute per device

An example of HTTP response by Postman (a HTTP test tool)

Cookies	Headers (15)	Test Results	Status: 200 OK	Time: 2.50 s
KEY	VALUE			
API-RateLimit-Remaining	9			
API-RateLimit-Reset	1648537626			
API-RateLimit-Limit	10			
Vary	Origin			
Access-Control-Allow-Origin	*			
X-RateLimit-Limit	10000			
X-RateLimit-Remaining	9990			
X-RateLimit-Reset	1648623756			
Content-Type	application/json; charset=utf-8			
X-Response-Time	2488ms			
X-traceId	b41600a0-af2e-11ec-95cf-4fbe2c61925f			

Govee Developer API (Lights, Plugs, Switches)

DeviceList

GET /v1/devices

Get information about all supported devices in the account.

Request Headers

Govee-API-Key : {your_Govee_API_KEY}

Response Body Parameters

Content Type : application/json

code

The code value will be 200 when the request process successfully.

Type: number

Required: Yes

message

The message value will be "success" when the request process successfully. Or it will be the error message.

Type: string

Required: Yes

data

The response data.

Type: object

Required: Yes

data.devices

Return your devices supported by the Govee Developer API.

Type: array

Required: Yes

data.devices.model

Product Model of your device.

Type: string

Required: Yes

data.devices.device

Mac address of your device. Use device and model to identify the target one device.

Type: string

Required: Yes

data.devices.controllable

Controllable will be true when the device support commands to control.

Valid Values: true, false

Type: bool

Required: Yes

data.devices.retrievable

Retrievable will be true when the device support querying the current device state.

Valid Values: true, false

Type: bool

Required: Yes

data.devices.supportCmds

Commands supported by the device.

Valid Values in the array: "turn", "brightness", "color", "colorTem"

Type: array

Required: Yes

Errors

AuthorizationError

Indicates that the Govee Developer API key is invalid.

HTTP Status Code: 403

InternalError

Indicates an internal service error.

HTTP Status Code: 500

Example

Sample Request

Request URL: <https://developer-api.govee.com/v1/devices>

Request Method: GET

Request Header:

Govee-API-Key: 00000000-729c-4b82-b536-0000000000

Sample Response

```
{
  "data": {
    "devices": [
      {
        "device": "99:E5:A4:C1:38:29:DA:7B",
        "model": "H6159",
        "deviceName": "test light",
        "controllable": true,
        "retrievable": true,
        "supportCmds": [
          "turn",
          "brightness",
          "color",
          "colorTem"
        ],
        "properties": {
```

```
        "colorTem": {
          "range": {
            "min": 2000,
            "max": 9000
          }
        }
      },
      {
        "device": "34:20:03:2e:30:2b",
        "model": "H5081",
        "deviceName": "Smart Plug",
        "controllable": true,
        "retrievable": true,
        "supportCmds": [
          "turn"
        ]
      }
    ],
    "message": "Success",
    "code": 200
  }
}
```

DeviceControl

Use supported commands to control the device.

Request Headers

Content-Type : application/json

Govee-API-Key : {*your_Govee_API_KEY*}

Request Body Parameters

device

The Mac address of the device you want to control.

Type: String

Required: Yes

model

The Model of the device you want to control

Type: String

Required: Yes

cmd

Device commands.

Type: object

Required: Yes

cmd.name

Device commands. Use with `cmd.value`.

Valid values: "turn", "brightness", "color", "colorTem"

Type: object

Required: Yes

cmd.value

Device commands. Use with `cmd.name`.

Type: number

Required: Yes

Demos:

```
{"name": "turn", value: "off"}
```

```
{"name": "brightness", value: 10}
```

```
{"name": "color", value: {"r": 255, "g": 255, "b": 255}}
```

```
{"name": "colorTem", value: 7000}
```

When cmd.name is "turn":
the valid values are "on" and "off".

Type: string

When cmd.name is "brightness":
the valid values are between 0 and 100, and 0 will turn off the device

Type: number

When cmd.name is "color":

Type: object

cmd.value.r

Red value. Valid range is 0~255.

Type: number

cmd.value.g

Green value. Valid range is 0~255.

Type: number

cmd.value.b

Blue value. Valid range is 0~255.

Type: number

When cmd.name is "colorTem":

the valid values are returned by DeviceList, there will be one more field
named "properties".

```
"properties": {  
    "colorTem": {  
        "range": {  
            "min": 2000,  
            "max": 9000  
        }  
    }  
}
```

Response Body Parameters

Content Type : application/json

code

The code value will be 200 when the request process successfully.

Type: number

Required: Yes

message

The message value will be "success" when the request process successfully. Or it will be the error message.

Type: string

Required: Yes

data

The response data.

Type: object

Required: Yes

Errors

AuthorizationError

Indicates that the Govee Developer API key is invalid.

HTTP Status Code: 403

InternalError

Indicates an internal service error.

HTTP Status Code: 500

InvalidParameter

Indicates that a request parameter does not comply with the associated constraints.

HTTP Status Code: 400

Example

Sample Request 1

Request URL: <https://developer-api.govee.com/v1/devices/control>

Request Method: PUT

Request Header:

Content-Type : application/json

Govee-API-Key : 00000000-729c-4b82-b536-0000000000

Request body

```
{
  "device": "34:20:03:15:82:ae",
  "model": "H6089",
  "cmd": {
    "name": "turn",
    "value": "on"
  }
}
```

Sample Request 2

Request URL: <https://developer-api.govee.com/v1/devices/control>

Request Method: PUT

Request Header:

Content-Type : application/json

Govee-API-Key : 00000000-729c-4b82-b536-0000000000

Request body

```
{
  "device": "34:20:03:15:82:ae",
  "model": "H6089",
  "cmd": {
    "name": "brightness",
    "value": 50
  }
}
```

Sample Request 3

Request URL: <https://developer-api.govee.com/v1/devices/control>

Request Method: PUT

Request Header:

Content-Type : application/json

Govee-API-Key : 00000000-729c-4b82-b536-0000000000

Request body:

```
{
  "device": "34:20:03:15:82:ae",
  "model": "H6089",
  "cmd": {
    "name": "color",

```

```
{
  "value": {
    "r": 255,
    "g": 255,
    "b": 255
  }
}
```

Sample Request 4

Request URL: <https://developer-api.govee.com/v1/devices/control>

Request Method: PUT

Request Header:

Content-Type : application/json

Govee-API-Key : 00000000-729c-4b82-b536-0000000000

Request body

```
{
  "device": "34:20:03:15:82:ae",
  "model": "H6089",
  "cmd": {
    "name": "colorTem",
    "value": 7000
  }
}
```

Sample Response

```
{
  "code": 200,
  "message": "Success",
  "data": {}
}
```

DeviceState

Query the device state.

Request Headers

Content-Type : application/json

Govee-API-Key : {your_Govee_API_KEY}

Request Query Parameters

device

The Mac address of the device you want to control.

Type: String

Required: Yes

model

The Model of the device you want to control

Type: String

Required: Yes

Response Body Parameters

Content Type : application/json

code

The code value will be 200 when the request process successfully.

Type: number

Required: Yes

message

The message value will be "success" when the request process successfully. Or it will be the error message.

Type: string

Required: Yes

data

The response data.

Type: object

Required: Yes

data.model

Product Model of your device.

Type: array

Required: Yes

data.device

Mac address of your device.

Type: string

Required: Yes

data.name

Your device name.

Type: string

Required: Yes

data.properties

Device states.

Demos:

```
{"online": "false"}  
{"powerState" : "off"}  
{"brightness" : 60}
```

```
{"color": {"r": 255, "g": 255, "b": 255}}  
{"colorTem": 5000}
```

Type: arr

Required: Yes

Note: "online" is implemented through the cache. Sometimes it may return wrong state. We suggest the third-party developers to ensure that even if online returns "false", the users are still allowed to send control commands. Even if the cache is wrong, the users can still control the device then.

Errors

AuthorizationError

Indicates that the Govee Developer API key is invalid.

HTTP Status Code: 403

InternalServerError

Indicates an internal service error.

HTTP Status Code: 500

InvalidParameter

Indicates that a request parameter does not comply with the associated constraints.

HTTP Status Code: 400

Example

Sample Request

Request URL: <https://developer-api.govee.com/v1/devices/state?device=06%3A7A%3AA4%3AC1%3A38%3A5A%3A2A%3A8D&model=H6148>

Request Method: GET

Request Header:

Govee-API-Key : 00000000-729c-4b82-b536-0000000000

Request Query Parameters:

device: 06:7A:A4:C1:38:5A:2A:8D

model: H6148

Sample Response

```
{  
  "data": {  
    "device": "C6:EA:B8:56:C8:C6:89:BE",
```

```
"model": "H6188",
"properties": [
  {
    "online": "false"
  },
  {
    "powerState": "off"
  },
  {
    "brightness": 82
  },
  {
    "color": {
      "r": 0,
      "b": 0,
      "g": 0
    }
  }
],
"message": "Success",
"code": 200
}
```

Govee Developer API

(Appliances)

DeviceList

GET /v1/appliance/devices

The data structure is the same as that of “/v1/devices” on page 6.

data.devices.supportCmds

Commands supported by the device.

Valid Values in the array: "turn", "mode"

data.devices.properties.mode.options.name

the Displayed Name of the mode option..

data.devices.properties.mode.options.value

the ID of the mode option.

Sample Response

```
{
  "data": {
    "devices": [
      {
        "device": "01:B2:7C:A6:B0:6xxxxx",
        "model": "H7121",
        "deviceName": "Lucy",
        "controllable": true,
        "retrievable": false,
        "properties": {
          "mode": {
            "options": [
              {
                "name": "Low",
                "value": 1
              }
            ]
          }
        }
      }
    ]
  }
}
```

```
{
  {
    "name": "Medium",
    "value": 2
  },
  {
    "name": "High",
    "value": 3
  },
  {
    "name": "Sleep",
    "value": 16
  }
]
},
"supportCmds": [
  "turn",
  "mode"
]
}
],
},
"message": "Success",
"code": 200
}
```

DeviceControl

PUT /v1/appliance/devices/control

The data structure is the same as that of “/v1/devices/control” on page 10.

Sample Request

```
{
  "device": "01:B2:7C:A6:B0:6xxxxx",
  "model": "H7121",
  "cmd": {
    "name": "mode",
    "value": 16
  }
}
```

}

Version History

Version 2.0

1. Supports Lights: H61A5, H6078, H604D, H6168, H6601, H70B1, H61A8
2. Supports Appliances: H7121, H7122, H7123, H7120, H7141, H7142, H7130, H7131, H7132, H7150, H7160, H7101, H7111

Please see “Govee Developer API (Appliances)” on page 19 and “Rate Limiting (Appliances)” on page 5

Version 1.8

1. Supports H6046, H601A, H61A3, H610B, H6047, H7065, H61E1, H6057, H604C, H6065, H605C, H705A, H705B, H7055

Version 1.7

1. Rate limit rules update (see Rate Limiting)

Version 1.5

1. Supports H7031, H7032, H61A1, H61A2, H61B2, H7061, H6067, H6066, H6009, H7041, H7042, H604A, H6173, H615E, H604B, H6091, H7051, H7062, H618F, H605D

Version 1.4

1. Supports H605B, H6087, H6172, H619B, H619D, H619Z, H61A0, H7060, H610A, H6059, H7028, H6198, H6049

Version 1.3

1. Supports H7050, H6051, H6056, H6061, H6058, H6073, H6076, H619A, H619C, H618A, H618C, H6008, H6071, H6075, H614A, H614B, H614E, H618E, H619E

Version 1.2

1. Supports H6050, H6154, H6143, H6144, H6072, H6121, H611A, H5080, H6062, H614C, H615A, H615B, H7020, H7021, H614D, H611Z, H611B, H611C, H615C, H615D, H7006, H7007, H7008, H7012, H7013

2. The change of DeviceState:

When cmd.name is "colorTem":

the valid values are returned by DeviceList, there will be one more field named "properties".

```
"properties": {  
    "colorTem": {  
        "range": {  
            "min": 2000,  
            "max": 9000  
        }  
    }  
}
```