

SmartHome x Service Del x IBM Watson x IBM Cloud x Untitled - J x en.wikiped x pytz - PyPI x Authorize x How do I x

wuhx3q.internetofthings.ibmcloud.com/dashboard/devices/browse

CHANUKVADEM b... Watch Boruto: Naru... Administration Fertilizers Recomm... IBM Watson Service... Projects - DataCam... YouTube Gmail Postimages — free... Copy of Meet Our T...

IBM Watson IoT Platform

secretlovewithkd9542@gmail.com ID: wuhx3q

Browse Action Device Types Interfaces

Add Device

12345 Connected NodeMCU Device Nov 12, 2022 9:44 AM

Identity Device Information Recent Events State Logs

Device ID 12345

Device Type NodeMCU

Date Added Nov 12, 2022 9:44 AM

Added By secretlovewithkd9542@gmail.com

Connection Status **Connected**  
Connection Time: Nov 12, 2022 9:50 AM  
Client Address: 103.130.91.131 SecureToken

Items per page 50 | 1-1 of 1 item

1 of 1 page

22°C Cloudy

Q Search

ENG IN 09:57 AM 12-11-2022

IBM Watson IoT Platform dashboard showing device details for device ID 123456. The device is connected and a simulation is running. The recent events table shows a stream of data with random values for pH, turbidity, and event ID.

Event	Value	Format	Last Received
event_1	{"randomNumber":89,"phvalue":23,"turbidity":72}	json	a few seconds ago
event_1	{"randomNumber":59,"phvalue":48,"turbidity":8}	json	a few seconds ago
event_1	{"randomNumber":50,"phvalue":18,"turbidity":92}	json	a few seconds ago
event_1	{"randomNumber":49,"phvalue":37,"turbidity":79}	json	a few seconds ago
event_1	{"randomNumber":21,"phvalue":61,"turbidity":55}	json	a minute ago

1 Simulation running

Jupyter Notebook interface showing Python code for connecting to the IBM Watson IoT Platform and publishing data. The code includes a configuration dictionary, a command callback function, and a loop for publishing data.

```
In [2]: import random
import time
import wiotp.sdk.device

myConfig = {
    "identity": {
        "orgid": "wuhx3q",
        "typeId": "NodeMCU",
        "deviceId": "123456"
    },
    "auth": {
        "token": "v_I5Wzhs-oq8igc0B9"
    }
}

def myCommandCallback(cmd):
    print("Message received from IBM IoT Platform: %s" % cmd.data['command'])
    m=cmd.data['command']

client = wiotp.sdk.device.DeviceClient(config=myConfig, loghandlers=None)
client.connect()

while True:
    ph=random.randint(1,5)
    turb=random.randint(0,5)
    myData={"phvalue":ph,"turbidity":turb}
    client.publishEvent(eventId="status", msgFormat="json", data=myData, qos=0, onPublish=None)
    print("Published data Successfully: %s" % myData)
    client.commandsCallback = myCommandCallback
    time.sleep(2)
devicecli.commandsCallback = myCommandCallback
devicecli.disconnect()
```

2022-11-18 14:09:36,969 wiotp.sdk.device.client.DeviceClient INFO Connected successfully: d:wuhx3q:NodeMCU:123456

2022-11-18 14:09:48,042 wiotp.sdk.device.client.DeviceClient ERROR Unexpected disconnect from IBM Watson IoT Platform: 7

2022-11-18 14:10:00,908 wiotp.sdk.device.client.DeviceClient WARNING Unable to send event status because client is in disconnected state