

PYTHON SCRIPT

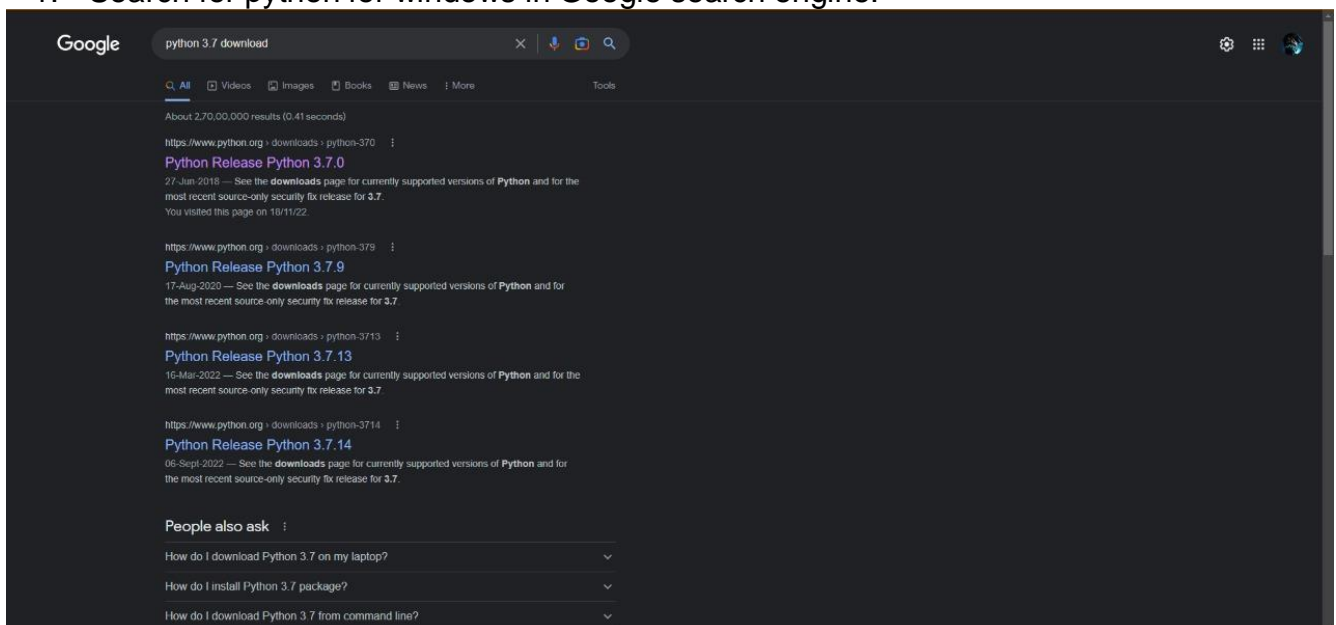
Assignment Date	06 NOVEMBER2022
Team ID	PNT2022TMID32740
Project Name	Gas Leakage Monitoring and Alerting System

AIM:

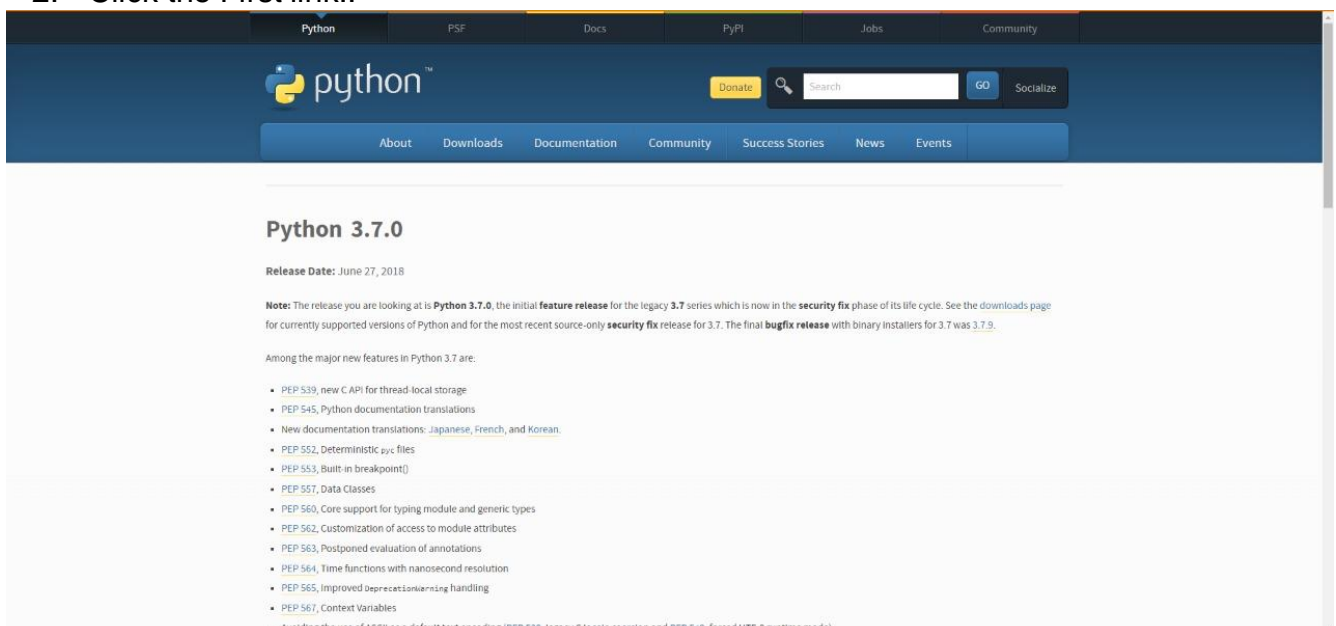
To install python version 3.9.6 and IBM Watson IoT platform packages in python.

STEPS:

1. Search for python for windows in Google search engine.



2. Click the First link..



3. Click the python version 3.9.6 and download.

[Embedded Distribution](#) for more information.

macOS users

- For 3.7.0, we provide two binary installer options for download. The default variant is 64-bit only and works on macOS 10.9 (Mavericks) and later systems. We also continue to provide a 64-bit/32-bit variant that works on all versions of macOS from 10.6 (Snow Leopard) on. Both variants now come with batteries-included versions of Tcl/Tk 8.6 for users of IDLE and other tkinter-based GUI applications, third-party and system versions of Tcl/Tk are no longer used. Consider using the new 10.9 64-bit-only installer variant, unless you are building Python applications that also need to work on older macOS systems.
- Both python.org installer variants include private copies of OpenSSL 1.1.0. Please carefully read the [Important Information](#) displayed during installation for information about SSL/TLS certificate validation and the `Install Certificates.command`.

[Full Changelog](#)

Files

Version	Operating System	Description	MD5 Sum	File Size	GPG
Gzipped source tarball	Source release		41b6995deb4147a1ed517a7d9a580271	22745726	SIG
XZ compressed source tarball	Source release		eb8c2a6b1447d50813c02714af4681f3	16922100	SIG
macOS 64-bit/32-bit installer	macOS	for Mac OS X 10.6 and later	ca3eb64092d0ff6d02e42f63a734338e	34274481	SIG
macOS 64-bit installer	macOS	for OS X 10.9 and later	ae0717a02efea3b0eb34aadcf680dc498	27651276	SIG
Windows help file	Windows		46562a186c2049dd0cc7680348180dca	8547689	SIG
Windows x86-64 embeddable zip file	Windows	for AMD64/EM64T/x64	cb8b4f0d979a36258f73ed541def10a5	6946082	SIG
Windows x86-64 executable installer	Windows	for AMD64/EM64T/x64	531c3f6821ce0a4107b6d2c6a129be3e	26262280	SIG
Windows x86-64 web-based installer	Windows	for AMD64/EM64T/x64	3cfda4c8d3b0475aaec12ba402d04d2	1327160	SIG
Windows x86 embeddable zip file	Windows		ed9a1c028c1e99f5323b9c20723d7d0f	6395982	SIG
Windows x86 executable installer	Windows		ebb6444c284c1447e902e87381afeff0	25506832	SIG
Windows x86 web-based installer	Windows		779c4085464eb3ee5b1a4fffd0eubca4	1298280	SIG

4. Python 3.9.6 version is installed successfully.

[Embedded Distribution](#) for more information.

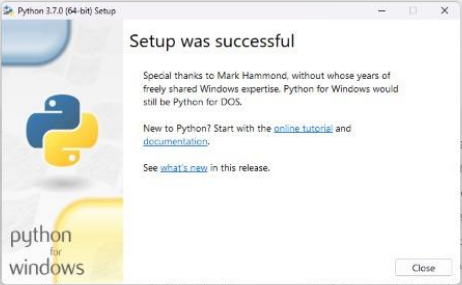
macOS users

- For 3.7.0, we provide two binary installer options for download. The default variant is 64-bit-only and works on macOS 10.9 (Mavericks) and later systems. We also continue to provide a 64-bit/32-bit variant that works on all versions of macOS from 10.6 (Snow Leopard) on. Both variants now come with batteries-included versions of Tcl/Tk 8.6 for users of IDLE and other tkinter-based GUI applications; third-party and system versions of Tcl/Tk are no longer used. Consider using the new 10.9 64-bit-only installer variant, unless you are building Python applications that also need to work on older macOS systems.
- Both python.org installer variants include private copies of OpenSSL 1.1.0. Please carefully read the [Important Information](#) displayed during installation for information about SSL/TLS certificate validation and the `Install Certificates.command`.

[Full Changelog](#)

Files

Version	File Size	GPG
Gzipped source tarball	22745726	SIG
XZ compressed source tarball	16922100	SIG
macOS 64-bit/32-bit installer	34274481	SIG
macOS 64-bit installer	27651276	SIG
Windows help file	8547689	SIG
Windows x86-64 embeddable zip file	6946082	SIG
Windows x86-64 executable installer	26262280	SIG
Windows x86-64 web-based installer	1327160	SIG
Windows x86 embeddable zip file	6395982	SIG
Windows x86 executable installer	25506832	SIG
Windows x86 web-based installer	1298280	SIG



Python 3.7.0 (64-bit) Setup

Setup was successful

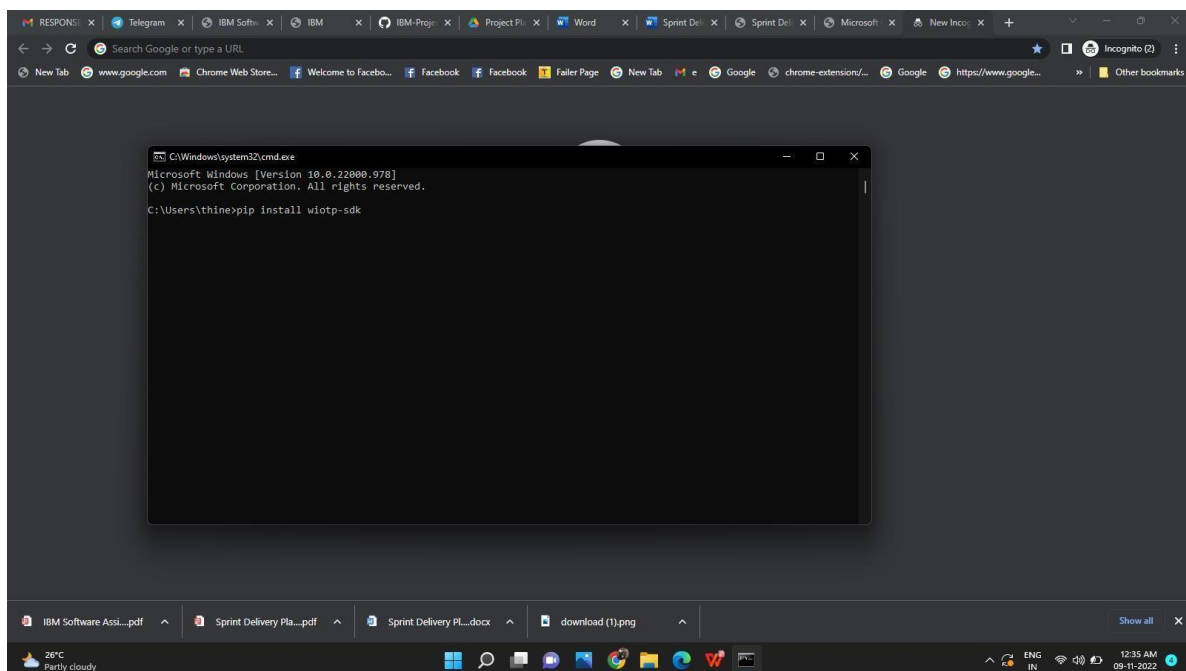
Special thanks to Mark Hammond, without whose years of freely shared Windows expertise, Python for Windows would still be Python for DOS.

New to Python? Start with the [online tutorial](#) and [documentation](#).

See [what's new](#) in this release.

Close

5. Now install the IBM Watson Platform package through command prompt.



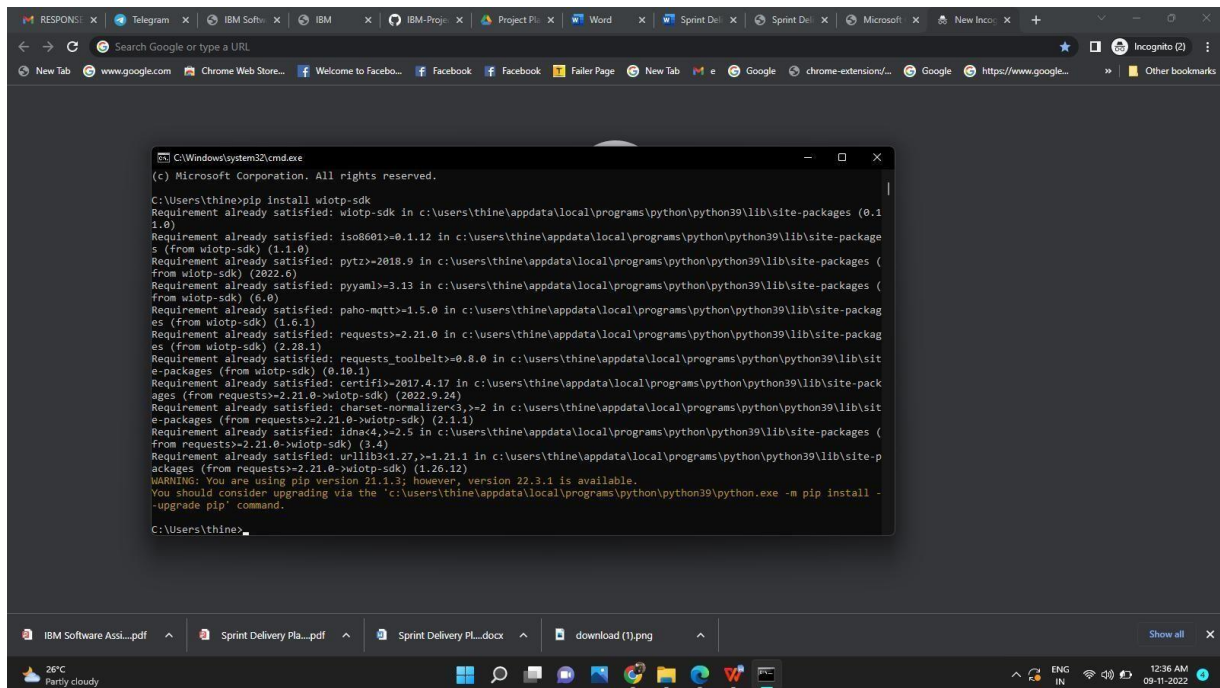
The screenshot shows a Windows desktop environment. In the background, a web browser (Incognito mode) is open with several tabs, including 'RESPONSE...', 'Telegram', 'IBM Software...', 'IBM', 'IBM-Proj...', 'Project P...', 'Word', 'Sprint Del...', 'Sprint Del...', 'Microsoft', 'New Inco...', and 'New Inco...'. The browser's address bar shows 'https://www.google...'. In the foreground, a command prompt window is open, displaying the following text:

```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 10.0.22000.978]
(c) Microsoft Corporation. All rights reserved.

C:\Users\thine>pip install wattp-sdk
```

The taskbar at the bottom shows the system clock as 12:35 AM on 09-11-2022, with a temperature of 28°C and 'Partly cloudy' weather.

6. The package will be installed.

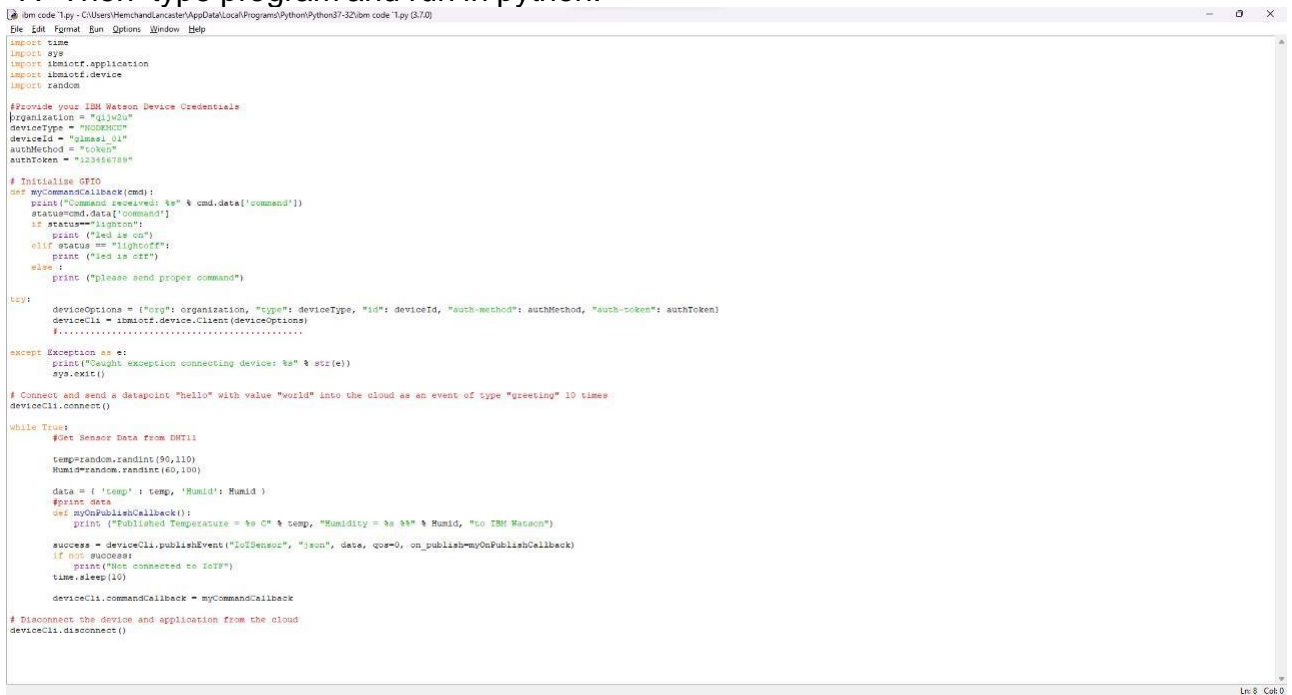


```
C:\Windows\system32\cmd.exe
(c) Microsoft Corporation. All rights reserved.

C:\Users\thine>pip install wiotp-sdk
Requirement already satisfied: wiotp-sdk in c:\users\thine\appdata\local\programs\python\python39\lib\site-packages (0.1.0)
Requirement already satisfied: iso8601>=0.1.12 in c:\users\thine\appdata\local\programs\python\python39\lib\site-packages (from wiotp-sdk) (0.1.10)
Requirement already satisfied: pytz>=2018.9 in c:\users\thine\appdata\local\programs\python\python39\lib\site-packages (from wiotp-sdk) (2022.6)
Requirement already satisfied: pyyaml>=3.13 in c:\users\thine\appdata\local\programs\python\python39\lib\site-packages (from wiotp-sdk) (6.0)
Requirement already satisfied: paho-mqtt>=1.5.0 in c:\users\thine\appdata\local\programs\python\python39\lib\site-packages (from wiotp-sdk) (1.6.1)
Requirement already satisfied: requests>=2.21.0 in c:\users\thine\appdata\local\programs\python\python39\lib\site-packages (from wiotp-sdk) (2.28.1)
Requirement already satisfied: requests-toolbelt>=0.8.0 in c:\users\thine\appdata\local\programs\python\python39\lib\site-packages (from wiotp-sdk) (0.10.1)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\thine\appdata\local\programs\python\python39\lib\site-packages (from requests>=2.21.0->wiotp-sdk) (2022.9.24)
Requirement already satisfied: charset-normalizer<3,>=2 in c:\users\thine\appdata\local\programs\python\python39\lib\site-packages (from requests>=2.21.0->wiotp-sdk) (1.26.12)
Requirement already satisfied: idna<4,>=2.5 in c:\users\thine\appdata\local\programs\python\python39\lib\site-packages (from requests>=2.21.0->wiotp-sdk) (3.4)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in c:\users\thine\appdata\local\programs\python\python39\lib\site-packages (from requests>=2.21.0->wiotp-sdk) (1.26.12)
WARNING: You are using pip version 21.1.3; however, version 22.3.1 is available.
You should consider upgrading via the 'c:\users\thine\appdata\local\programs\python\python39\python.exe -m pip install --upgrade pip' command.

C:\Users\thine>
```

7. Then type program and run in python.



```
Python 3.9.6 Shell
File Edit Format Run Options Window Help

import time
import sys
import ibmiot.application
import ibmiot.device
import random

#Provide your IBM Watson Device Credentials
organization = "q1v2u"
deviceType = "WU282CU"
deviceId = "q1maei_01"
authMethod = "token"
authToken = "123456789"

# Initialise GPIO
def myCommandCallback(cmd):
    print("Command received %s" % cmd.data['command'])
    status=cmd.data['command']
    if status=="lighton":
        print ("led is on")
    elif status == "lightoff":
        print ("led is off")
    else:
        print ("please send proper command")

try:
    deviceOptions = {"org": organization, "type": deviceType, "id": deviceId, "auth-method": authMethod, "auth-token": authToken}
    deviceCli = ibmiot.device.Client(deviceOptions)
    #.....

except Exception as e:
    print("Caught exception connecting device: %s" % str(e))
    sys.exit()

# Connect and send a datapoint "hello" with value "world" into the cloud as an event of type "greeting" 10 times
deviceCli.connect()

while True:
    #Get Sensor Data from DHT11
    temp=random.randint(90,110)
    humid=random.randint(60,100)

    data = { 'temp': temp, 'Humid': humid }
    #print data
    def myOnPublishCallback():
        print ("Published Temperature = %s C & temp, "Humidity = %s %& " & humid, "to IBM Watson")

    success = deviceCli.publishEvent("IoTsensor", "json", data, qos=0, on_publish=myOnPublishCallback)
    if not success:
        print("Not connected to IoT")
        time.sleep(10)

    deviceCli.commandCallback = myCommandCallback

# Disconnect the device and application from the cloud
deviceCli.disconnect()
```

RESULT:

The python version 3.9.6 and IBM Watson IoT platform package are installed successfully.

Python Code:

```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random

#Provide your IBM Watson Device Credentials
organization = "qijw2u"
deviceType = "NODEMCU"
deviceId = "glmas1_01"
authMethod = "token"
authToken = "123456789"

# Initialize GPIO
def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status=="lighton":
        print ("led is on")
    elif status == "lightoff":
        print ("led is off")
    else :
        print ("please send proper command")

try:
    deviceOptions = {"org": organization, "type": deviceType, "id": deviceId, "auth-method":
authMethod, "auth-token": authToken}
    deviceCli = ibmiotf.device.Client(deviceOptions)
    #.....

except Exception as e:
    print("Caught exception connecting device: %s" % str(e))
    sys.exit()

# Connect and send a datapoint "hello" with value "world" into the cloud as an event of type
"greeting" 10 times
deviceCli.connect()

while True:
    #Get Sensor Data from DHT11

    temp=random.randint(90,110)
    Humid=random.randint(60,100)

    data = { 'temp' : temp, 'Humid': Humid }
    #print data
    def myOnPublishCallback():
        print ("Published Temperature = %s C" % temp, "Humidity = %s %" % Humid, "to IBM
Watson")

    success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0,
on_publish=myOnPublishCallback)
    if not success:
        print("Not connected to IoT")
    time.sleep(10)
```

```
deviceCli.commandCallback = myCommandCallback
```

```
# Disconnect the device and application from the cloud  
deviceCli.disconnect()
```

```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Command received: lightoff
led is off
Published Temperature = 105 C Humidity = 85 % to IBM Watson
Published Temperature = 100 C Humidity = 92 % to IBM Watson
Command received: lightoff
led is off
Published Temperature = 101 C Humidity = 92 % to IBM Watson
Command received: lighton
led is on
Published Temperature = 92 C Humidity = 100 % to IBM Watson
Published Temperature = 97 C Humidity = 74 % to IBM Watson
Command received: lighton
led is on
Published Temperature = 93 C Humidity = 77 % to IBM Watson
Published Temperature = 97 C Humidity = 85 % to IBM Watson
Published Temperature = 95 C Humidity = 79 % to IBM Watson
Published Temperature = 91 C Humidity = 66 % to IBM Watson
Published Temperature = 105 C Humidity = 95 % to IBM Watson
Command received: lighton
led is on
Published Temperature = 93 C Humidity = 80 % to IBM Watson
Published Temperature = 92 C Humidity = 72 % to IBM Watson
Command received: lighton
led is on
Published Temperature = 90 C Humidity = 100 % to IBM Watson
Command received: lighton
led is on
Published Temperature = 92 C Humidity = 78 % to IBM Watson
Published Temperature = 95 C Humidity = 85 % to IBM Watson
Published Temperature = 108 C Humidity = 80 % to IBM Watson
Published Temperature = 97 C Humidity = 95 % to IBM Watson
Published Temperature = 93 C Humidity = 75 % to IBM Watson
Published Temperature = 99 C Humidity = 93 % to IBM Watson
Published Temperature = 102 C Humidity = 59 % to IBM Watson
Published Temperature = 106 C Humidity = 49 % to IBM Watson
Published Temperature = 108 C Humidity = 98 % to IBM Watson
Published Temperature = 96 C Humidity = 76 % to IBM Watson
Published Temperature = 103 C Humidity = 85 % to IBM Watson
Published Temperature = 90 C Humidity = 97 % to IBM Watson
Published Temperature = 91 C Humidity = 53 % to IBM Watson
Published Temperature = 103 C Humidity = 75 % to IBM Watson
Published Temperature = 93 C Humidity = 65 % to IBM Watson
Published Temperature = 100 C Humidity = 62 % to IBM Watson
Published Temperature = 96 C Humidity = 60 % to IBM Watson
Published Temperature = 103 C Humidity = 70 % to IBM Watson
Published Temperature = 105 C Humidity = 96 % to IBM Watson
Published Temperature = 96 C Humidity = 80 % to IBM Watson
Published Temperature = 93 C Humidity = 67 % to IBM Watson
Published Temperature = 104 C Humidity = 79 % to IBM Watson
Published Temperature = 100 C Humidity = 71 % to IBM Watson
Published Temperature = 93 C Humidity = 85 % to IBM Watson
Published Temperature = 104 C Humidity = 62 % to IBM Watson
Published Temperature = 104 C Humidity = 84 % to IBM Watson
Published Temperature = 92 C Humidity = 80 % to IBM Watson
Published Temperature = 105 C Humidity = 71 % to IBM Watson
Published Temperature = 108 C Humidity = 68 % to IBM Watson
Published Temperature = 92 C Humidity = 89 % to IBM Watson
Published Temperature = 107 C Humidity = 74 % to IBM Watson
Published Temperature = 97 C Humidity = 78 % to IBM Watson
Published Temperature = 96 C Humidity = 64 % to IBM Watson
Ln: 151 Col: 9
```

IBM Watson IoT Platform

813819106038@smarterinternz.com
ID: qjw2u

Browse Action Device Types Interfaces

Search by Device ID

Device Simulator

Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location	Added By	Device Class
g1mas1_01	Connected	NODEMCU	Device	18 Nov 2022 6:57 PM		813819106038@smarterinternz.com	

Identity Device Information Recent Events State Logs

The recent events listed show the live stream of data that is coming and going from this device.

Event	Value	Format	Last Received
IoTSensor	["temp":107,"Humid":71]	json	a few seconds ago
IoTSensor	["temp":106,"Humid":95]	json	a few seconds ago
IoTSensor	["temp":105,"Humid":94]	json	a few seconds ago
IoTSensor	["temp":97,"Humid":94]	json	a few seconds ago
IoTSensor	["temp":110,"Humid":68]	json	a few seconds ago

Items per page 50 | 1-1 of 1 item

1 of 1 page