

Sprint – 3

Team ID: PNT2022TMID10955

Python Code:

```
# Sprint - 3
```

```
# Team ID: PNT2022TMID10955
```

```
import time import
```

```
sys
```

```
import ibmiotf.application
```

```
import      ibmiotf.device
```

```
import random
```

```
#Provide your IBM Watson Device Credentials
```

```
organization = "lcft5g" deviceType = "Final"
```

```
deviceId = "Hello" authMethod = "token"
```

```
authToken = "8300113450"
```

```
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(0,100)

```

```

    Humid=random.randint(0,100)

```

```

    Gas=random.randint(0,100)

```

```

    data = { 'temp' : temp, 'Humid': Humid,'Gas':gas }

```

```

    #print      data      def

```

```

    myOnPublishCallback():

```

```

        print ("Published Temperature = %s C" % temp, "Humidity = %s %" %
Humid, "Gas Concentration = %s"%Gas"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)

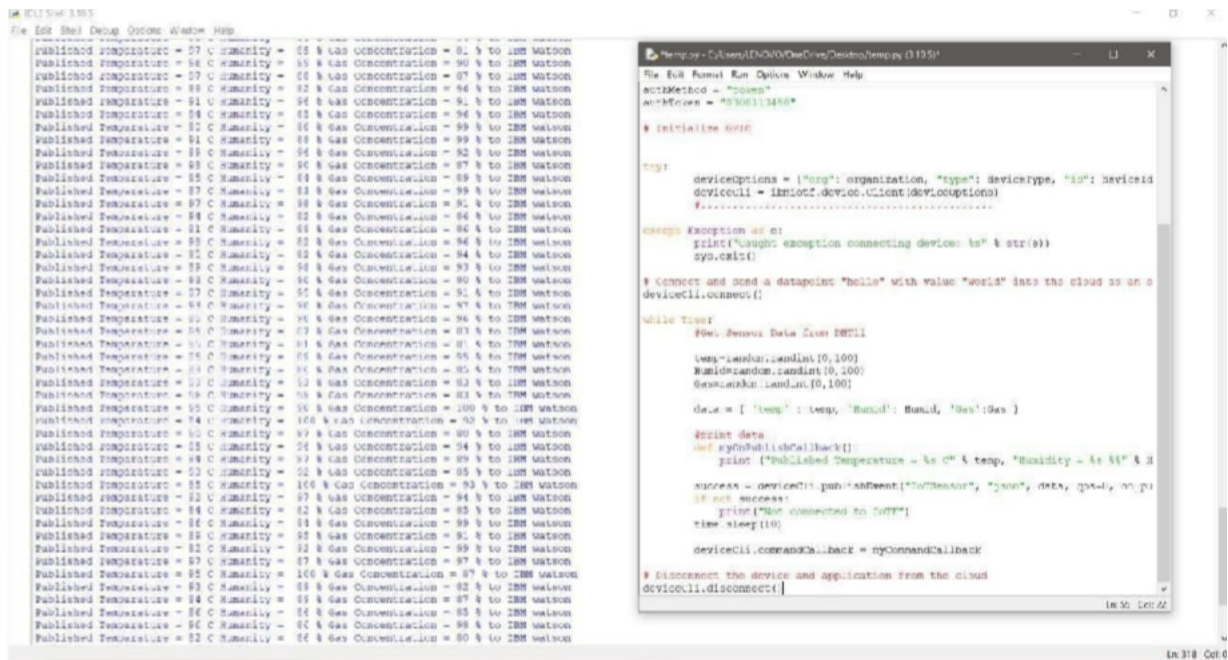
```

```
deviceCli.commandCallback = myCommandCallback
```

```
# Disconnect the device and application from the cloud
```

```
deviceCli.disconnect()
```

Output:



The screenshot shows a Python IDE with two windows. The left window displays a list of 40 published data points, each containing Temperature, Humidity, and Gas Concentration values. The right window shows the Python script that generates this data. The script includes a loop that publishes data points to the cloud and a function to disconnect the device and application from the cloud.

```
File Edit Shell Debug Options Window Help
Python 3.7.4 Shell
Published Temperature = 27 C Humidity = 55 % Gas Concentration = 81 % to IBM Watson
Published Temperature = 26 C Humidity = 55 % Gas Concentration = 90 % to IBM Watson
Published Temperature = 27 C Humidity = 55 % Gas Concentration = 87 % to IBM Watson
Published Temperature = 28 C Humidity = 52 % Gas Concentration = 84 % to IBM Watson
Published Temperature = 24 C Humidity = 54 % Gas Concentration = 91 % to IBM Watson
Published Temperature = 24 C Humidity = 53 % Gas Concentration = 96 % to IBM Watson
Published Temperature = 32 C Humidity = 50 % Gas Concentration = 99 % to IBM Watson
Published Temperature = 21 C Humidity = 58 % Gas Concentration = 99 % to IBM Watson
Published Temperature = 28 C Humidity = 54 % Gas Concentration = 92 % to IBM Watson
Published Temperature = 28 C Humidity = 50 % Gas Concentration = 87 % to IBM Watson
Published Temperature = 25 C Humidity = 51 % Gas Concentration = 89 % to IBM Watson
Published Temperature = 27 C Humidity = 58 % Gas Concentration = 91 % to IBM Watson
Published Temperature = 24 C Humidity = 52 % Gas Concentration = 94 % to IBM Watson
Published Temperature = 21 C Humidity = 55 % Gas Concentration = 86 % to IBM Watson
Published Temperature = 22 C Humidity = 52 % Gas Concentration = 96 % to IBM Watson
Published Temperature = 25 C Humidity = 52 % Gas Concentration = 94 % to IBM Watson
Published Temperature = 25 C Humidity = 58 % Gas Concentration = 93 % to IBM Watson
Published Temperature = 28 C Humidity = 56 % Gas Concentration = 90 % to IBM Watson
Published Temperature = 27 C Humidity = 55 % Gas Concentration = 91 % to IBM Watson
Published Temperature = 24 C Humidity = 50 % Gas Concentration = 87 % to IBM Watson
Published Temperature = 25 C Humidity = 51 % Gas Concentration = 96 % to IBM Watson
Published Temperature = 24 C Humidity = 51 % Gas Concentration = 85 % to IBM Watson
Published Temperature = 22 C Humidity = 52 % Gas Concentration = 83 % to IBM Watson
Published Temperature = 25 C Humidity = 58 % Gas Concentration = 83 % to IBM Watson
Published Temperature = 25 C Humidity = 50 % Gas Concentration = 100 % to IBM Watson
Published Temperature = 24 C Humidity = 100 % Gas Concentration = 92 % to IBM Watson
Published Temperature = 24 C Humidity = 97 % Gas Concentration = 80 % to IBM Watson
Published Temperature = 25 C Humidity = 76 % Gas Concentration = 94 % to IBM Watson
Published Temperature = 24 C Humidity = 97 % Gas Concentration = 89 % to IBM Watson
Published Temperature = 22 C Humidity = 52 % Gas Concentration = 85 % to IBM Watson
Published Temperature = 25 C Humidity = 58 % Gas Concentration = 99 % to IBM Watson
Published Temperature = 32 C Humidity = 52 % Gas Concentration = 99 % to IBM Watson
Published Temperature = 27 C Humidity = 57 % Gas Concentration = 97 % to IBM Watson
Published Temperature = 25 C Humidity = 100 % Gas Concentration = 87 % to IBM Watson
Published Temperature = 23 C Humidity = 59 % Gas Concentration = 82 % to IBM Watson
Published Temperature = 24 C Humidity = 59 % Gas Concentration = 87 % to IBM Watson
Published Temperature = 26 C Humidity = 56 % Gas Concentration = 85 % to IBM Watson
Published Temperature = 26 C Humidity = 50 % Gas Concentration = 98 % to IBM Watson
Published Temperature = 22 C Humidity = 56 % Gas Concentration = 80 % to IBM Watson

Python 3.7.4 Shell
File Edit Shell Debug Options Window Help
Python 3.7.4 Shell
def publish_data():
    # Initialize data
    data = {}
    data['temp'] = random.randint(20, 30)
    data['humidity'] = random.randint(50, 100)
    data['gas'] = random.randint(80, 100)

    # Publish data
    deviceCli.publish_data(data)
    print("Published Temperature = %s C & temp, Humidity = %s % & gas" % (data['temp'], data['humidity'], data['gas']))
    if not success:
        print("Not connected to IoT")
    time.sleep(10)

deviceCli.commandCallback = myCommandCallback

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