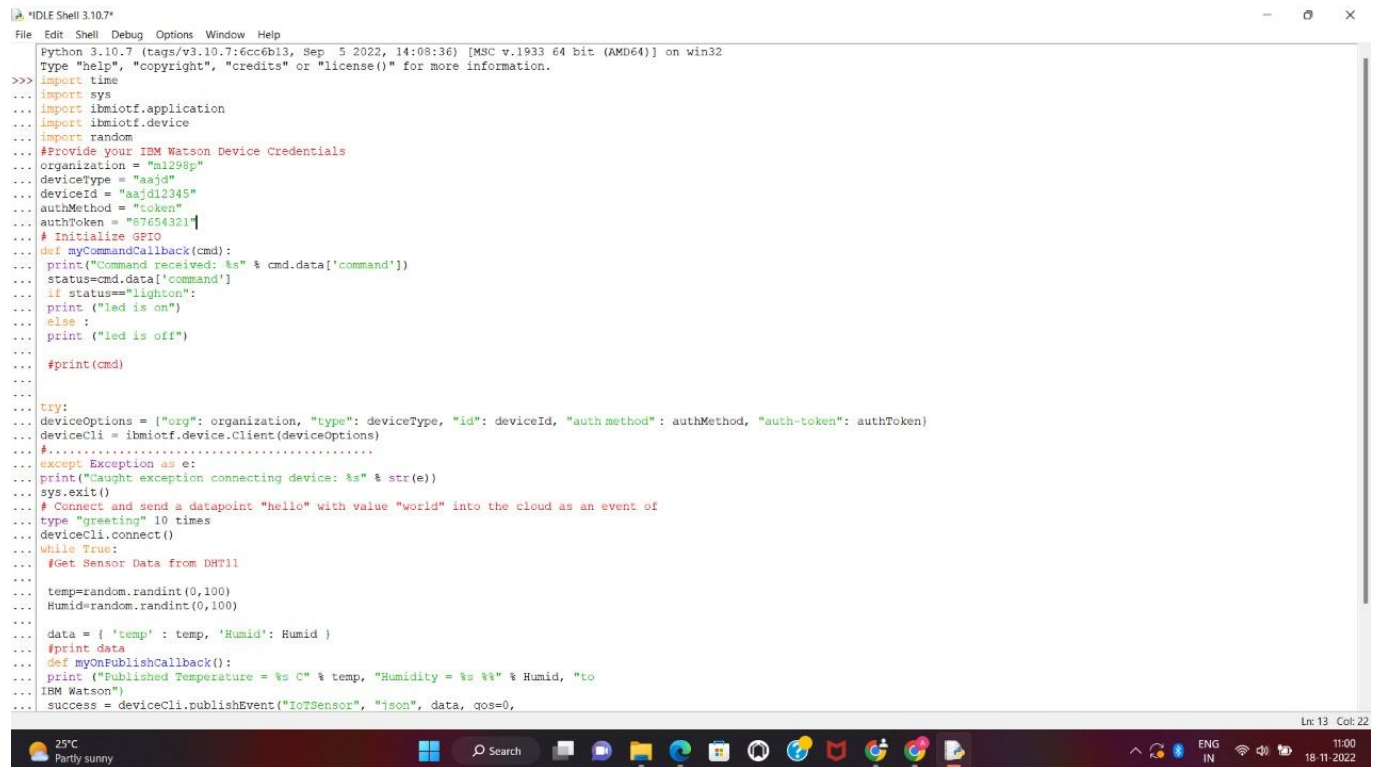


# Project Development Phase II

## Sprint II

Date	18 <sup>th</sup> November 2022
Team Id	PNT2022TMID26470



```
Python 3.10.7 (tags/v3.10.7:6cc6b13, Sep 5 2022, 14:08:36) [MSC v.1933 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> import time
... import sys
... import ibmiotf.application
... import ibmiotf.device
... import random
... #Provide your IBM Watson Device Credentials
... organization = "ml298p"
... deviceType = "aajd"
... deviceId = "aajd12345"
... authMethod = "token"
... authToken = "87654321"
... # Initialize GPIO
... def myCommandCallback(cmd):
...     print("Command received: %s" % cmd.data['command'])
...     status=cmd.data['command']
...     if status=="lighton":
...         print ("led is on")
...     else :
...         print ("led is off")
...
...     #print(cmd)
...
... 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)
...
...     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,
```

```

IDLE Shell 3.10.7
File Edit Shell Debug Options Window Help
...
organization = "m1298p"
...
deviceType = "aa3d"
...
deviceId = "aa3d12345"
...
authMethod = "token"
...
authToken = "07654321"
...
# Initialize GPIO
...
def myCommandCallback(cmd):
...
    print("Command received: %s" % cmd.data['command'])
...
    status=cmd.data['command']
...
    if status=="lighton":
...
        print ("led is on")
...
    else :
...
        print ("led is off")
...
    #print(cmd)
...
...
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)
...
    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()

```

Ln: 13 Col: 22

25°C  
Partly sunny



ENG  
IN 11:00  
18-11-2022

```
{"temperature":8,"Humidity":24}
```

---

```
{"temperature":92,"Humidity":59}
```

---

```
{"temperature":37,"Humidity":22}
```

---

```
{"temperature":53,"Humidity":48}
```

---

```
{"temperature":81,"Humidity":52}
```

---

```
{"temperature":19,"Humidity":86}
```

---

```
{"temperature":3,"Humidity":36}
```

---

```
{"temperature":18,"Humidity":71}
```

---

```
{"temperature":90,"Humidity":31}
```

---

```
{"temperature":100,"Humidity":91}
```

---