SPRINT 1 - PYTHON CODING

Date	19 November
Team ID	PNT2022TMID29937
Project name	Project – Smart Farmer-IoT Enabled smart Farming Application

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

#Provide your IBM Watson Device Credentials

orgId = "n1eaxk" deviceType =
"smartfarming"  deviceId =
"TamilNadu" Token =
"q3(u4iv5-4L+OHy@wm"
authMethod = "use-token-auth"

# Initialize GPIO
     def
myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
status=cmd.data['command']      if status=="motoron":
print ("motor is on")     else:
        print ("motor is off")

    #print(cmd)

try:
  deviceOptions = {"org": orgId, "type": deviceType, "id": deviceId,
"auth-method": authMethod, "auth-token": Token}
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(-
20,125)      hum=random.randint(0,100)
soil=random.randint(0,100)
    data = { 'temp' : temp, 'hum': hum , 'soil': soil}
```

```
    #print data def myOnPublishCallback( ):      print (f"Published temp
= {temp} C , hum = {hum} , soil = {soil} deg c to IBM Watson")
    success = deviceCli.publishEvent("IoTSensor", "json", data,
qos=0,on_publish=myOnPublishCallback)      if not success:
        print("Not connected to IoTF")
time.sleep(10)
    deviceCli.commandCallback = myCommandCallback      #
Disconnect the device and application from the cloud
deviceCli.disconnect()
```