SPRINT-1

Python Script

Date	24 October 2022
Team ID	PNT2022TMID31899
Project Name	IOT BASED CROP PROTECTION SYSTEM FOR AGRICULTURE

Description:

The motor status data's are generated and automation has been implemented through the python code instead of using hardware to implement IOT based crop protection system. And the python code need to upload the data's in IBM cloud are written in this python script.

Python Code:

```
import time
import sys
import ibmiotf.application # to install pip install ibmiotf
import ibmiotf.device
# Provide your IBM Watson Device Credentials
organization = "ebf2oy" # replace the ORG ID
deviceType = "Humidity" # replace the Device type
deviceId = "123456" # replace Device ID
authMethod = "token"
authToken = "C4b(zFIpnKm_OT_C+c" # Replace the authtoken
def myCommandCallback(cmd): # function for Callback
    if cmd.data['command'] == 'motoron':
        print("MOTOR ON IS RECEIVED")
    elif cmd.data['command'] == 'motoroff':
        print("MOTOR OFF IS RECEIVED")
    if cmd.command == "setInterval":
        if 'interval' not in cmd.data:
```

```
            print("Error - command is missing required information: 'interval'")
else:
            interval = cmd.data['interval']
    elif cmd.command == "print":

        if 'message' not in cmd.data:
            print("Error - command is missing required information: 'message'")
        else:
            output = cmd.data['message']
            print(output)
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:
      deviceCli.commandCallback = myCommandCallback
# Disconnect the device and application from the cloud
deviceCli.disconnect()
```

Python Sprint Output: