

# Smart Farmer - IOT Enabled Smart Farming Application

## SPRINT -1

<b>Team ID</b>	PNT2022TMID51719
<b>Team Leader</b>	SUREKHA S.K
<b>Team Members</b>	DENSIYA .I FENILDA RIJU R.F NISHA M.N

### Python Code:-

```
import time

import sys

import ibmiotf.application # to install pip install ibmiotf
import ibmiotf.device

#Provide your IBM Watson Device Credentials

organization = "0lsrz8" # repalce it with organization ID
deviceType = "Agriculture_1" #replace it with device type
deviceId = "Device_2" #repalce with device id
authMethod = "token"
```

```
authToken = "Surideni@123"#repalce with token
```

```
def myCommandCallback(cmd): # function for Callback print("Command  
received: %s" % cmd.data)
```

```
if cmd.data['command']=='motoron'
```

```
    print("Turn Motor ON")
```

```
elif cmd.data['command']=='motoroff':
```

```
    print("Turn Motor OFF")
```

```
elif cmd.data['command']=='lighton':
```

```
    print("Turn Light ON")
```

```
elif cmd.data['command']=='lightoff':
```

```
    print("Turn Light OFF")
```

```
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()
```

**Output:-**

