

# DEVELOPING PYTHON SCRIPT

TEAM ID	PNT2022TMID52099
PROJECT	IOT BASED SMART CROP PROTECTION

## LOCATION DATA:

```
import wiotp.sdk.device
import time
import random
myConfig={
"identity": (
"orgId": "gagtey",
"typeId": "GPS",
"deviceId":"12345"},
"auth": {
"token": "12345678"
}}
def myCommandCallback (cmd):
print ("Message received from IBM IoT Platform: %s" %
cmd.data['command']) m-cmd.data['command']
client= wiotp.sdk.device.DeviceClient (config=myConfig,
logHandlers=None)
client.connect()
def pub (data):
```

```
client.publishEvent (eventId="status", msgFormat="json",  
data=myData, qos=0, print("Published data Successfully: %s",  
myData)
```

```
while True:
```

```
myData={'name': 'Train1', 'lat': 17.6387448, 'lon':  
78.4754336)
```

```
pub (myData)
```

```
time.sleep (3)
```

```
#myData({'name': 'Train2', 'lat': 17.6387448, 'lon':  
78.4754336)
```

```
#pub (myData)
```

```
#time.sleep (3)
```

```
myData={'name': 'Train1', 'lat': 17.6341908, 'lon':  
78.4744722)
```

```
pub (myData)
```

```
time.sleep(3)
```

```
myData={'name': 'Train1', 'lat': 17.6340889, 'lon': 78.4745052)
```

```
pub (myData)
```

```
time.sleep(3)
```

```
myData={'name': 'Train1', 'lat': 17.6248626, 'lon': 78.4720259)
```

```
pub (myData)
```

```
time.sleep (3)
```

```
myData={'name': 'Train1', 'lat': 17.6188577, 'lon': 78.4698726)
```

```
pub (myData)
time.sleep (3)
myData={'name': 'Train1', 'lat': 17.6132382, 'lon':
78.4707318)
pub (myData)
time.sleep (3)
client.commandCallback = myCommandCallback
client.disconnect()
```

### **QR SCANNER CODE:**

```
Import cv2
import numpy as np
import time
Import pyzbar.pyzbar as pyzbar
from ibmcloudant.cloudant_v1 import CloudantV1
from ibmcloudant import CouchDbSessionAuthenticator
from ibm_cloud_sdk_core.authenticators import
BasicAuthenticator
authenticator= BasicAuthenticator ('apikey-v2-
16u3crmdpkghhxefdikvpssoh5fwezrmuup5fv5g3ubz',
'b0ab119f45d3e6255eabb978
service Cloudant V1 (authenticator-authenticator)
service.set_service_url('https://apikey-v2-
```

16u3crmdpkghhxefdikvpssoh5fwezrmuup5fv5g3ubz:b0ab119  
f45d3e6255eabb978e7e2f0

```
cap= cv2.VideoCapture (0)
```

```
font cv2.FONT_HERSHEY_PLAIN
```

```
while True:
```

```
    frame cap.read()
```

```
    decodedobjects pyzbar.decode (frame)
```

```
    for obj in decodedObjects:
```

```
        #print ("Data", obj.data)
```

```
        a=obj.data.decode('UTF-8')
```

```
        cv2.putText (frame, "Ticket", (50, 50), font, 2,  
                    (255, 0, 0), 3)
```

```
        #print (a)
```

```
        try: response = service.get_document (
```

```
            db='booking, doc_id = a
```

```
        ).get_result()
```

```
        print (response) time.sleep(5)
```

```
    except Exception as e:
```

```
        print ("Not a Valid Ticket")
```

```
        time.sleep (5)
```

```
    cv2.imshow("Frame", frame)
```

```
if cv2.waitKey(1) & 0xFF==ord('q'):  
    break  
cap.release()  
cv2.destroyAllWindows ()  
client.disconnect()
```