

The Python Script

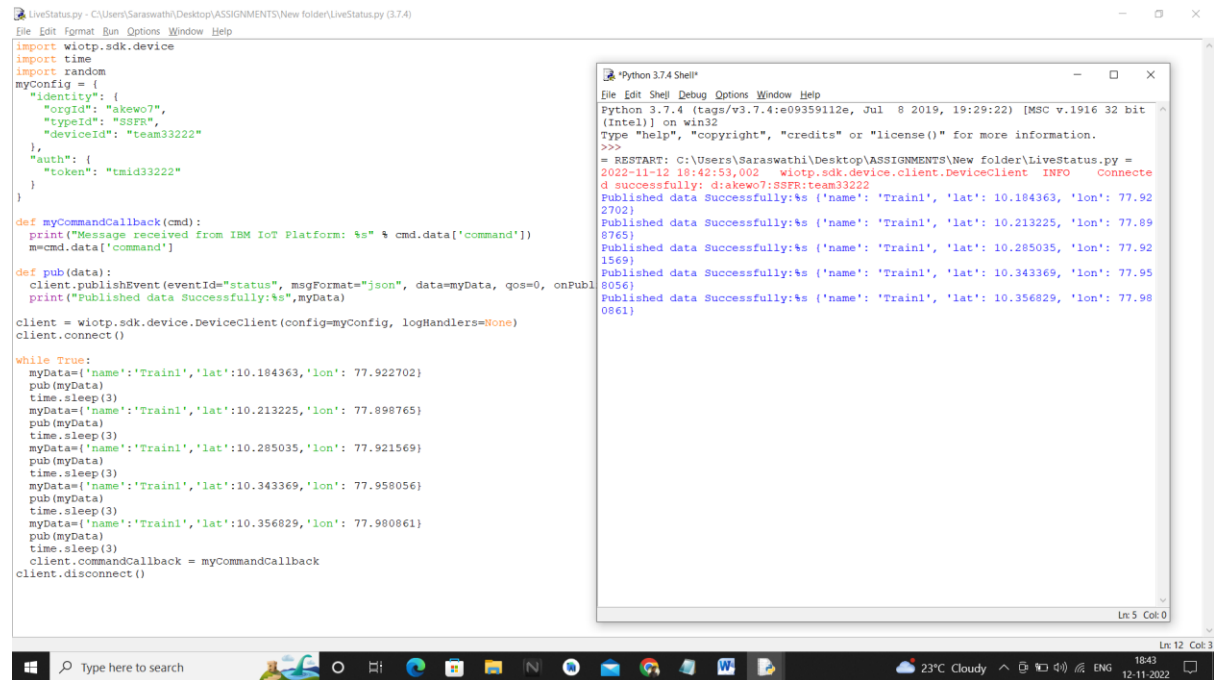
Date	12 November 2022
Team ID	PNT2022TMID33222
Project	Smart Solutions For Railways

Python code for publishing the location (latitude and longitude) data to the IBM IoT Platform.

```
import wiotp.sdk.device
import time
import random
myConfig = {
    "identity": {
        "orgId": "akewo7",
        "typeId": "SSFR",
        "deviceId": "team33222"
    },
    "auth": {
        "token": "tmid33222"
    }
}
def myCommandCallback(cmd):
    print("Message received from IBM IoT Platform: %s" % cmd.data['command'])
    m=cmd.data['command']
def pub(data):
    client.publishEvent(eventId="status", msgFormat="json", data=myData, qos=0, onPublish=None)
    print("Published data Successfully:%s",myData)
client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
client.connect()
while True:
    myData={'name':'Train1','lat':10.184363,'lon': 77.922702}
    pub(myData)
    time.sleep(3)
    myData={'name':'Train1','lat':10.213225,'lon': 77.898765}
    pub(myData)
    time.sleep(3)
    myData={'name':'Train1','lat':10.285035,'lon': 77.921569}
    pub(myData)
    time.sleep(3)
    myData={'name':'Train1','lat':10.343369,'lon': 77.958056}
    pub(myData)
    time.sleep(3)
    myData={'name':'Train1','lat':10.356829,'lon': 77.980861}
    pub(myData)
    time.sleep(3)
    client.commandCallback = myCommandCallback
client.disconnect()
```

Output:

Python IDLE output:



```
LiveStatus.py - C:\Users\Saraswathi\Desktop\ASSIGNMENTS\New folder\LiveStatus.py (3.7.4)
File Edit Format Run Options Window Help

import wiotp.sdk.device
import time
import random
myConfig = {
    "identity": {
        "orgId": "akewo7",
        "typeId": "SSFR",
        "deviceId": "team33222"
    },
    "auth": {
        "token": "tmid33222"
    }
}

def myCommandCallback(cmd):
    print("Message received from IBM IoT Platform: %s" % cmd.data['command'])
    m=cmd.data['command']

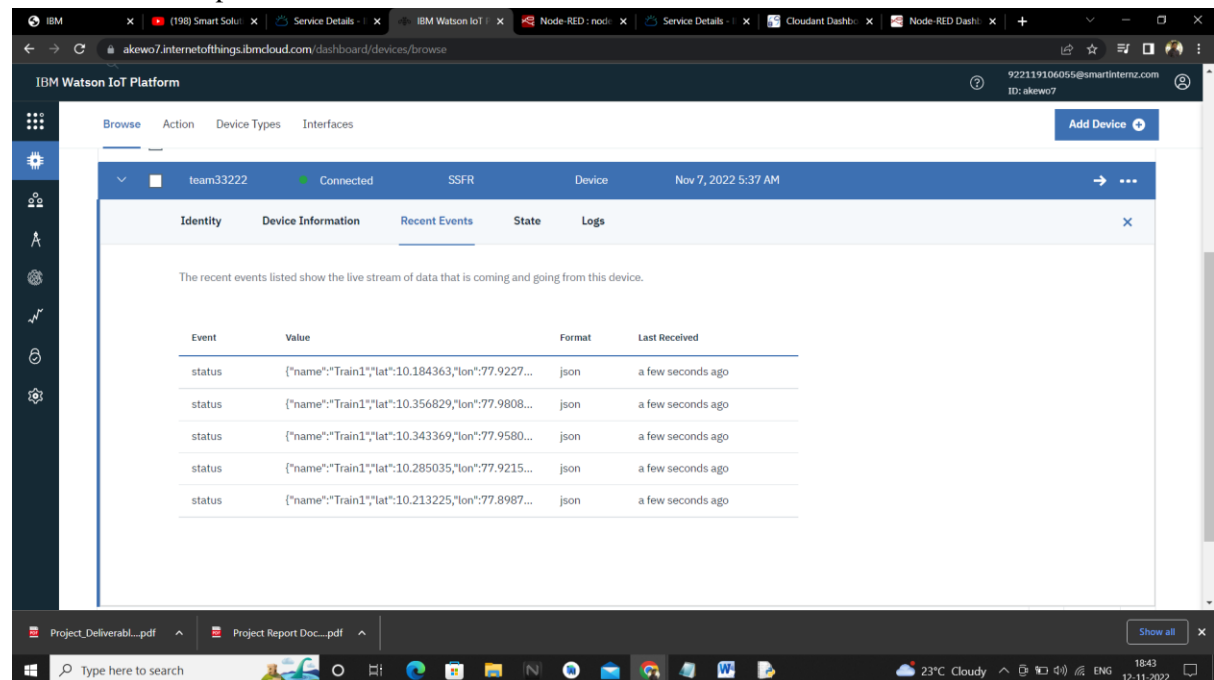
def pub(data):
    client.publishEvent(eventId="status", msgFormat="json", data=myData, qos=0, onPubl
    print("Published data Successfully:%s",myData)

client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
client.connect()

while True:
    myData={"name":'Train1','lat':10.184363,'lon': 77.922702}
    pub(myData)
    time.sleep(3)
    myData={"name":'Train1','lat':10.213225,'lon': 77.898765}
    pub(myData)
    time.sleep(3)
    myData={"name":'Train1','lat':10.285035,'lon': 77.921569}
    pub(myData)
    time.sleep(3)
    myData={"name":'Train1','lat':10.343369,'lon': 77.958056}
    pub(myData)
    time.sleep(3)
    myData={"name":'Train1','lat':10.356829,'lon': 77.980861}
    pub(myData)
    time.sleep(3)
    client.commandCallback = myCommandCallback
    client.disconnect()

Python 3.7.4 Shell
File Edit Shell Debug Options Window Help
Python 3.7.4 (tags/v3.7.4:e09359112e, Jul 8 2019, 19:29:22) [MSC v.1916 32 bit
(Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\Saraswathi\Desktop\ASSIGNMENTS\New folder\LiveStatus.py =
2022-11-12 18:42:53,002 wiotp.sdk.device.client.DeviceClient INFO Connecte
d successfully: d:akewo7:SSFR:team33222
Published data Successfully:%s ('name': 'Train1', 'lat': 10.184363, 'lon': 77.92
2702)
Published data Successfully:%s ('name': 'Train1', 'lat': 10.213225, 'lon': 77.89
8765)
Published data Successfully:%s ('name': 'Train1', 'lat': 10.285035, 'lon': 77.92
1569)
Published data Successfully:%s ('name': 'Train1', 'lat': 10.343369, 'lon': 77.95
8056)
Published data Successfully:%s ('name': 'Train1', 'lat': 10.356829, 'lon': 77.98
0861)
Ln 5 Col 0
```

IBM Watson Output:



IBM Watson IoT Platform

922119106055@smartinternz.com
ID: akewo7

team33222 Connected SSFR Device Nov 7, 2022 5:37 AM

Identity Device Information Recent Events State Logs

The recent events listed show the live stream of data that is coming and going from this device.

Event	Value	Format	Last Received
status	{"name":"Train1","lat":10.184363,"lon":77.9227...	json	a few seconds ago
status	{"name":"Train1","lat":10.356829,"lon":77.9808...	json	a few seconds ago
status	{"name":"Train1","lat":10.343369,"lon":77.9580...	json	a few seconds ago
status	{"name":"Train1","lat":10.285035,"lon":77.9215...	json	a few seconds ago
status	{"name":"Train1","lat":10.213225,"lon":77.8987...	json	a few seconds ago

Python code to read the QR Code and fetch the data from Cloudant DB

```
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-
iufbfeavfqzri2a1toob4eqpp1o0os7qa33jorulvz','4b171c068dcccc74c1a0155dae6533f5')
service = CloudantV1(authenticator=authenticator)
service.set_service_url('https://245c54f0-3778-46cb-bfd9-6e1bfca37902-
bluemix.cloudantnosqldb.appdomain.cloud')

cap = cv2.VideoCapture(0)
font = cv2.FONT_HERSHEY_PLAIN

while True:
    _, frame = cap.read()
    decodedObjects = pyzbar.decode(frame)
    for obj in decodedObjects:
        a=obj.data.decode('UTF-8')
        cv2.putText(frame, "Ticket", (50,50), font, 2, (255, 0, 0), 3)
        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()
```

Output:

