**AMLAN CODE and OUTPUT:**

import random

import time

from iotc.models import Command, Property

from iotc import IoTCClient, IOTCConnectType, IOTCEvents

scope\_id = '0ne00AE04F2'

device\_id = '267a860qdx1'

device\_key = 'kiT+AhhUNj7Nl7+ef3F5oir3OSleJzfi6RQvqPE+V+A='

LastTurnedOn=time.time()

temp = str(random.randint(0, 40))

pressure = str(random.randint(1, 10))

humidity =  str(random.randint(0,100))

telemetry\_str = "Temperature: {}, Pressure: {}, Humidity: {}".format(temp, pressure, humidity)

def on\_commands(command: Command):

    print(f"{command.name} command was sent")

    command.reply()

    iotc.send\_property({

    "LastPowerOn": LastTurnedOn

    })

    iotc.send\_property({

    "LastCommandReceived": time.time()

    })

    iotc.send\_property({

    "SendData" : telemetry\_str

    })

iotc = IoTCClient(

    device\_id,

    scope\_id,

        IOTCConnectType.IOTC\_CONNECT\_DEVICE\_KEY,

        device\_key)

iotc.connect()

iotc.on(IOTCEvents.IOTC\_COMMAND, on\_commands)

while iotc.is\_connected():

    iotc.send\_telemetry({

        'Temperature': str(random.randint(0, 40)),

        'Pressure': str(random.randint(0, 40)),

        'Humidity': str(random.randint(0, 40))

    })

    time.sleep(60)

A screen shot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**SHASWATI CODE and OUTPUT:**

import random

import time

from iotc.models import Command, Property

from iotc import IoTCClient, IOTCConnectType, IOTCEvents

scopeId = '0ne00AE04F2'

device\_id = 'ec3aqb46ip'

device\_key = 'WDhq8Ep7MheHsWci1qGYqq26XrrimZfJ8nquRgT7A4Y='

LastTurnedOn = time.time()

temp=str(random.randint(0,40))

pressure=str(random.randint(1,10))

humidity=str(random.randint(0,100))

telemetry\_str="Temperature: {}, Pressure: {}, Humidity: {}".format(temp, pressure, humidity)

def on\_commands(command: Command):

    print(f"{command.name} command was sent")

    iotc.send\_property({

    "LastPowerOn": LastTurnedOn

    })

    iotc.send\_property({

    "LastCommandReceived": time.time()

    })

    iotc.send\_property({

    "SendData": telemetry\_str

    })

    command.reply()

iotc = IoTCClient(

    device\_id,

    scopeId,

            IOTCConnectType.IOTC\_CONNECT\_DEVICE\_KEY,

            device\_key)

iotc.connect()

iotc.on(IOTCEvents.IOTC\_COMMAND, on\_commands)

while iotc.is\_connected():

    iotc.send\_telemetry({

    'Temperature': str(random.randint(0, 40)),

    'Pressure':str(random.randint(0, 40)),

    'Humidity':str(random.randint(0, 40))

    })

    time.sleep(60)

A screenshot of a computer screen

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**DATA EXPLORER:**

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated