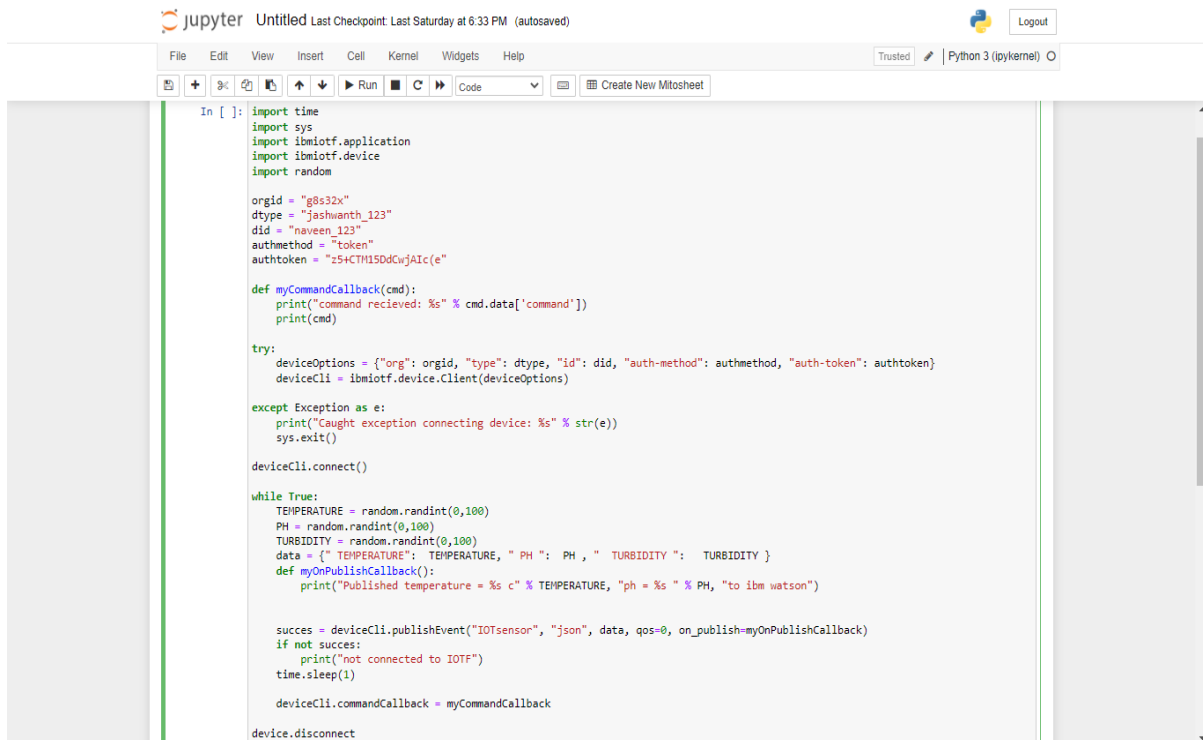


Project Development Phase Sprint-2

Team ID	PNT2022TMID22083
Project Name	Real-Time River Water Quality Monitoring and Control System

Developed the python script:

Developed the python script to connect the iot device with the python script.



```
In [ ]: import time
import sys
import ibmiotf.application
import ibmiotf.device
import random

orgid = "g8s32x"
dtype = "jashwanth_123"
did = "naveen_123"
authmethod = "token"
authtoken = "z5+CTM150dCwJAic(e"

def myCommandCallback(cmd):
    print("command recieved: %s" % cmd.data['command'])
    print(cmd)

try:
    deviceOptions = {"org": orgid, "type": dtype, "id": did, "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()

deviceCli.connect()

while True:
    TEMPERATURE = random.randint(0,100)
    PH = random.randint(0,100)
    TURBIDITY = random.randint(0,100)
    data = {"TEMPERATURE": TEMPERATURE, "PH": PH, "TURBIDITY": TURBIDITY}
    def myOnPublishCallback():
        print("Published temperature = %s c" % TEMPERATURE, "ph = %s" % PH, "to ibm watson")

    succes = deviceCli.publishEvent("IOTsensor", "json", data, qos=0, on_publish=myOnPublishCallback)
    if not succes:
        print("not connected to IOTF")
        time.sleep(1)

    deviceCli.commandCallback = myCommandCallback

device.disconnect
```

Python code:

import time

import sys

import ibmiotf.application

import ibmiotf.device

import random

orgid = "g8s32x"

```

dtype = "jashwanth_123"
did = "naveen_123"
authmethod = "token"
authtoken = "z5+CTM15DdCwjAlc(e"

def myCommandCallback(cmd):
    print("command recieved: %s" % cmd.data['command'])
    print(cmd)

try:
    deviceOptions = {"org": orgid, "type": dtype, "id": did, "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()

deviceCli.connect()

while True:
    TEMPERATURE = random.randint(0,100)
    PH = random.randint(0,100)
    TURBIDITY = random.randint(0,100)

    data = {"TEMPERATURE": TEMPERATURE, "PH ": PH , "TURBIDITY ": TURBIDITY }

    def myOnPublishCallback():
        print("Published temperature = %s c" % TEMPERATURE, "ph = %s " % PH, "to ibm
watson")

        succes = deviceCli.publishEvent("IOTsensor", "json", data, qos=0,
on_publish=myOnPublishCallback)

        if not succes:
            print("not connected to IOTF")

        time.sleep(1)

    deviceCli.commandCallback = myCommandCallback

device.disconnect

```

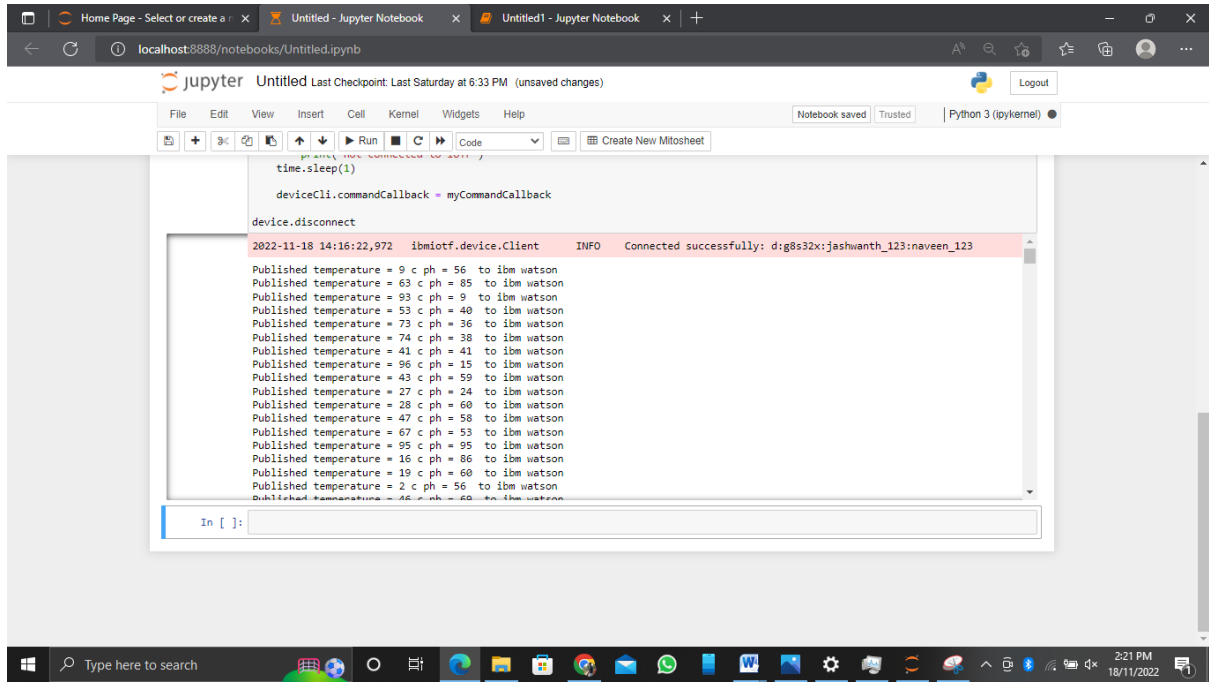
Connected the IBM IOT WATSON platform with python code:

The screenshot displays the IBM Watson IoT Platform dashboard. The browser's address bar shows the URL `g8s32x.internetofthings.ibmcloud.com/dashboard/devices/browse`. The dashboard header includes the IBM logo and a user profile for `vh10385_cse19@velhighitech.com` with ID `g8s32x`. The main navigation bar has tabs for `Browse`, `Action`, `Device Types`, and `Interfaces`, along with an `Add Device` button. The `Browse` tab is active, showing a `Browse Devices` section with `All Devices` and `Diagnose` buttons. A descriptive text states: "This table shows a summary of all devices that have been added. It can be filtered, organized, and searched on using different criteria. To get started, you can add devices by using the Add Device button, or by using API." Below this is a search bar labeled "Search by Device ID" and a `Device Simulator` toggle switch. A table lists the devices:

<input type="checkbox"/>	Device ID	Status	Device Type	Class ID	Date Added
<input type="checkbox"/>	naveen_123	Connected	jashwanth_123	Device	Nov 1, 2022 7:38 PM

At the bottom of the table, it says "Items per page 50 | 1-1 of 1 item". To the right of the table, it indicates "0 Simulations running". The Windows taskbar at the bottom shows the time as 2:16 PM on 18/11/2022.

Transmitted the data from python code to ibm iot Watson platform:



The screenshot shows a Jupyter Notebook running on a local host. The notebook contains Python code that connects to the IBM IoT Watson platform and publishes temperature and pH data. The output shows a successful connection and a series of published data points.

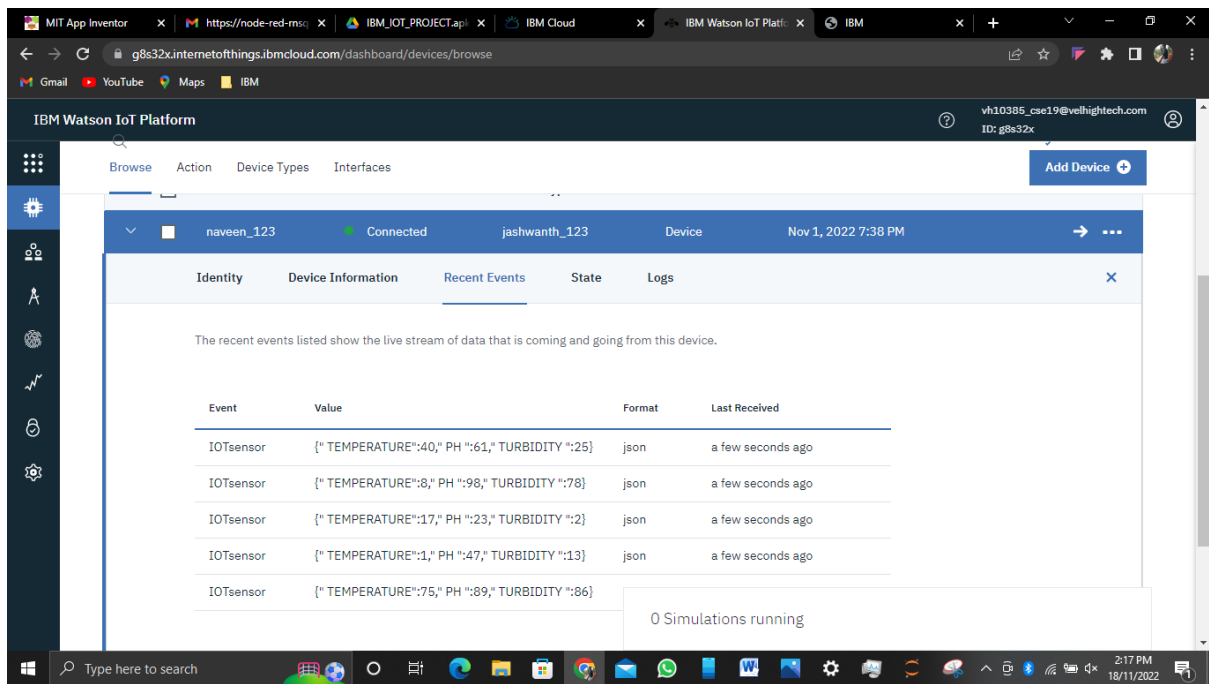
```
time.sleep(1)

deviceCli.commandCallback = myCommandCallback

device.disconnect
```

2022-11-18 14:16:22,972 ibmiotf.device.Client INFO Connected successfully: d:gs32x:jashwanth_123:naveen_123

Published temperature = 9 c ph = 56 to ibm watson
Published temperature = 63 c ph = 85 to ibm watson
Published temperature = 93 c ph = 9 to ibm watson
Published temperature = 53 c ph = 40 to ibm watson
Published temperature = 73 c ph = 36 to ibm watson
Published temperature = 74 c ph = 38 to ibm watson
Published temperature = 41 c ph = 41 to ibm watson
Published temperature = 96 c ph = 15 to ibm watson
Published temperature = 43 c ph = 59 to ibm watson
Published temperature = 27 c ph = 24 to ibm watson
Published temperature = 28 c ph = 60 to ibm watson
Published temperature = 47 c ph = 58 to ibm watson
Published temperature = 67 c ph = 53 to ibm watson
Published temperature = 95 c ph = 95 to ibm watson
Published temperature = 16 c ph = 86 to ibm watson
Published temperature = 19 c ph = 60 to ibm watson
Published temperature = 2 c ph = 56 to ibm watson
Published temperature = 46 c ph = 60 to ibm watson



The screenshot shows the IBM Watson IoT Platform dashboard. The device 'naveen_123' is listed as 'Connected'. The 'Recent Events' tab is selected, showing a live stream of data from the device. The events are listed in a table with columns: Event, Value, Format, and Last Received.

Event	Value	Format	Last Received
IOTsensor	{"TEMPERATURE":40,"PH":61,"TURBIDITY":25}	json	a few seconds ago
IOTsensor	{"TEMPERATURE":8,"PH":98,"TURBIDITY":78}	json	a few seconds ago
IOTsensor	{"TEMPERATURE":17,"PH":23,"TURBIDITY":2}	json	a few seconds ago
IOTsensor	{"TEMPERATURE":1,"PH":47,"TURBIDITY":13}	json	a few seconds ago
IOTsensor	{"TEMPERATURE":75,"PH":89,"TURBIDITY":86}	json	a few seconds ago

0 Simulations running