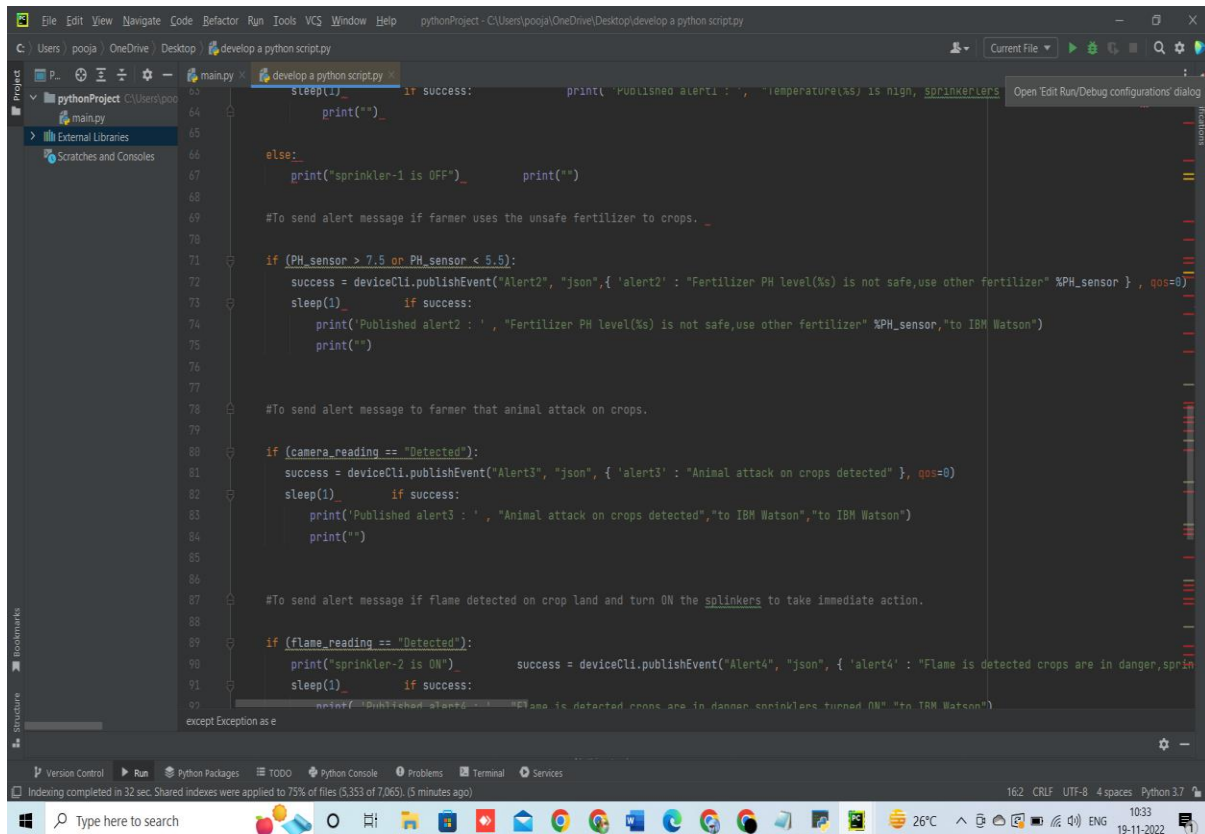


## DELIVERY OF SPRINT-4

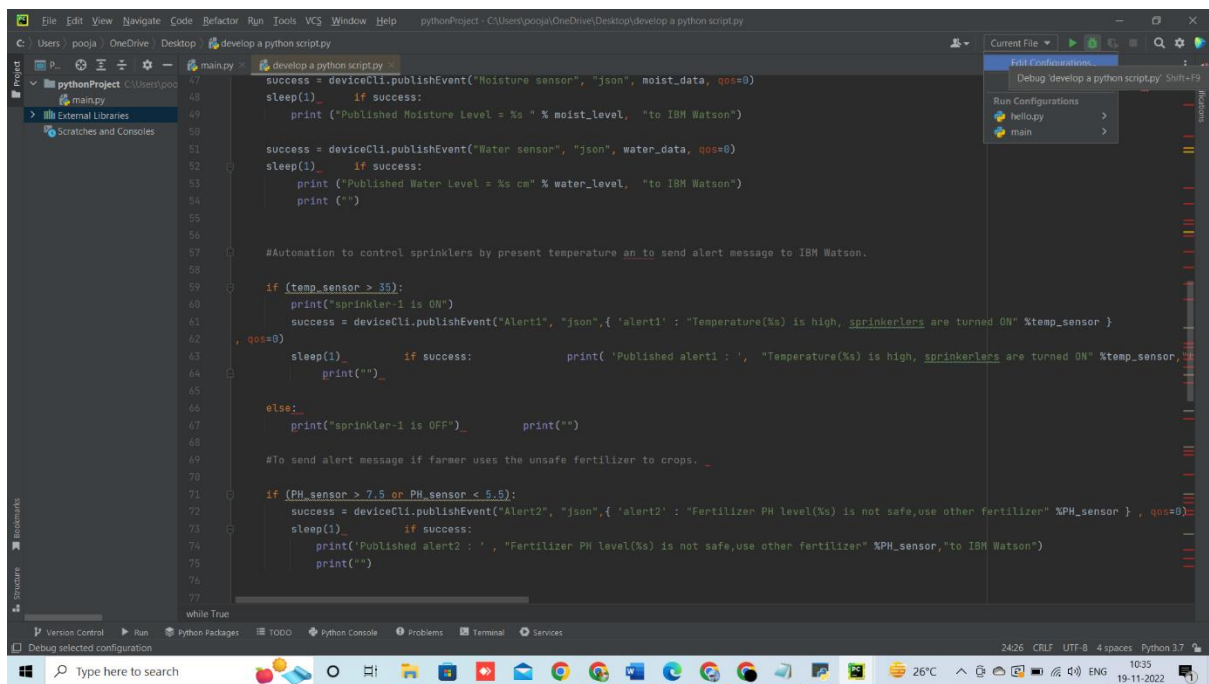
TEAM ID	PNT2022TMID25739
PROJECT NAME	Smart Waste Management For Metropolitan Cities

### 1.Develop a python script simulation.

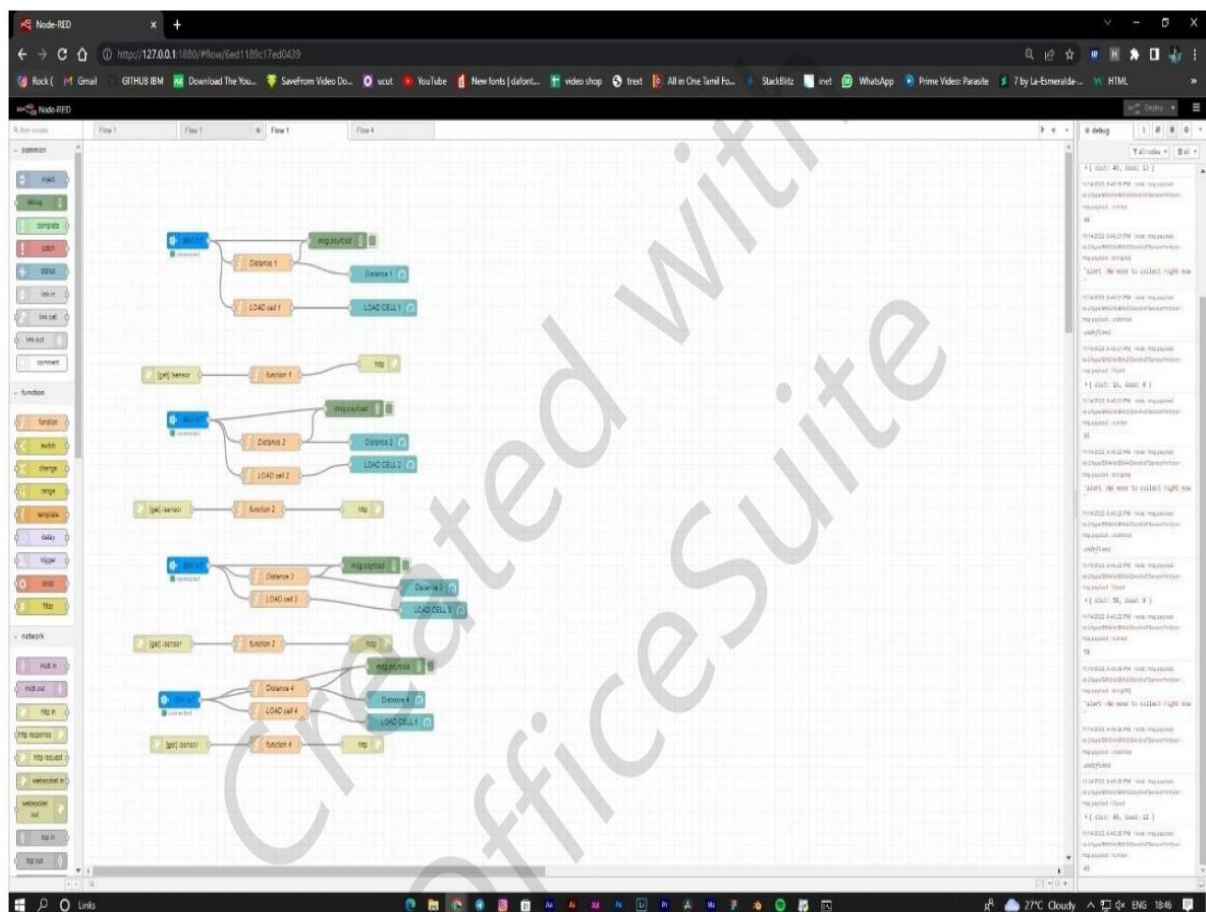


```
File Edit View Navigate Code Refactor Run Tools VCS Window Help pythonProject - C:\Users\pooja\OneDrive\Desktop\develop a python script.py
pythonProject
├── main.py
└── External Libraries
  └── Scratches and Consoles

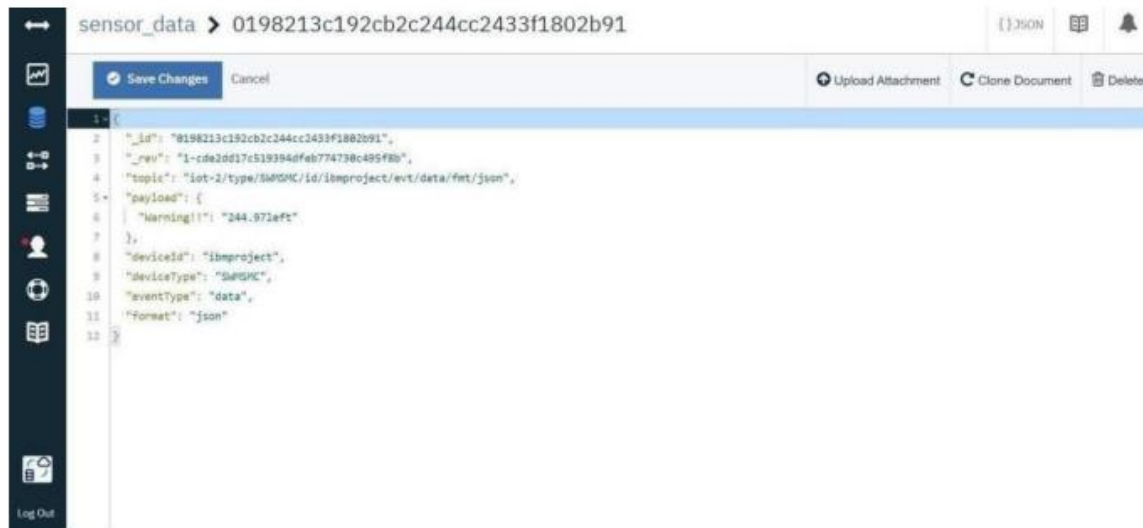
63 step(1) if success: print('Published alert1 : ', 'temperature(°s) is high, sprinklers
64 print("")
65
66 else:
67     print("sprinkler-1 is OFF") print("")
68
69 #To send alert message if farmer uses the unsafe fertilizer to crops.
70
71 if (PH_sensor > 7.5 or PH_sensor < 5.5):
72     success = deviceCli.publishEvent("Alert2", "json", { 'alert2' : "Fertilizer PH level(°s) is not safe,use other fertilizer" %PH_sensor }, qos=0)
73     sleep(1) if success:
74         print('Published alert2 : ', "Fertilizer PH level(°s) is not safe,use other fertilizer" %PH_sensor,"to IBM Watson")
75         print("")
76
77
78 #To send alert message to farmer that animal attack on crops.
79
80 if (camera_reading == "Detected"):
81     success = deviceCli.publishEvent("Alert3", "json", { 'alert3' : "Animal attack on crops detected" }, qos=0)
82     sleep(1) if success:
83         print('Published alert3 : ', "Animal attack on crops detected","to IBM Watson","to IBM Watson")
84         print("")
85
86
87 #To send alert message if flame detected on crop land and turn ON the sprinklers to take immediate action.
88
89 if (flame_reading == "Detected"):
90     print("sprinkler-2 is ON") success = deviceCli.publishEvent("Alert4", "json", { 'alert4' : "Flame is detected crops are in danger,sprin
91     sleep(1) if success:
92         print('Published alert4 : ', "Flame is detected crops are in danger,sprinklers turned ON","to IBM Watson")
93
94 except Exception as e
```



## 2.Node-red deployment.



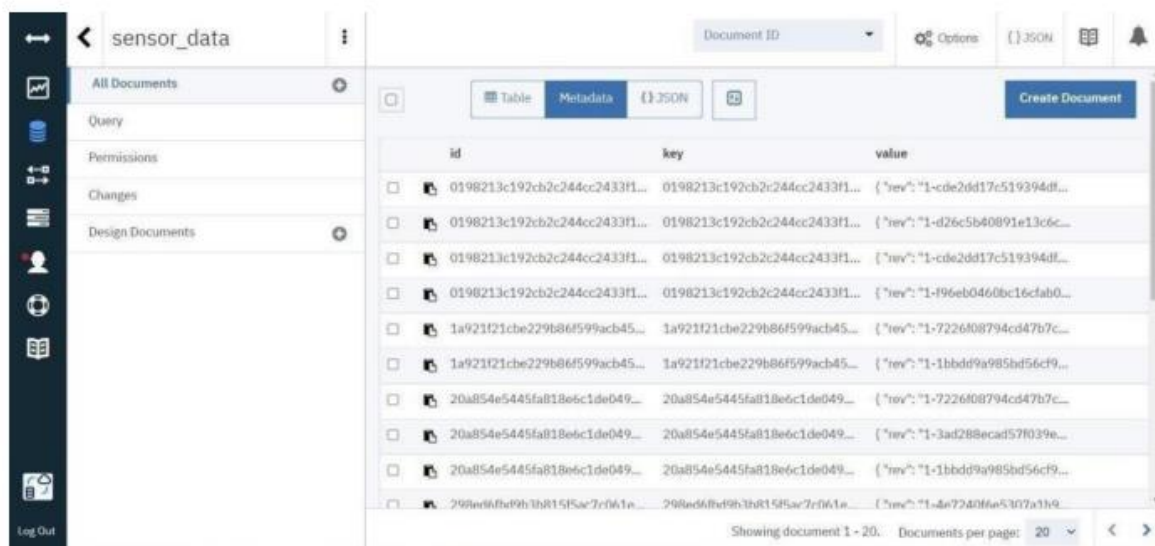
### 3. Data stored in json format



The screenshot shows the IBM Cloudant document editor interface. The document title is "sensor\_data" with a document ID "0198213c192cb2c244cc2433f1802b91". The document is in JSON format. The JSON payload is as follows:

```
{
  "_id": "0198213c192cb2c244cc2433f1802b91",
  "_rev": "1-cde2dd17c519394dfab774738c495f8b",
  "topic": "iot-2/Type/Sensor/Id/ibmproject/evt/data/fmt/json",
  "payload": {
    "warning!!": "244.971e4"
  },
  "deviceId": "ibmproject",
  "deviceType": "Sensor",
  "eventType": "data",
  "format": "json"
}
```

### 4. Storing database in ibm cloudant



The screenshot shows the IBM Cloudant document list view for the "sensor\_data" database. The documents are displayed in a table with columns for id, key, and value. The table shows 20 documents, with the first 10 visible in the screenshot.

id	key	value
0198213c192cb2c244cc2433f1...	0198213c192cb2c244cc2433f1...	{ "rev": "1-cde2dd17c519394dfab774738c495f8b", "topic": "iot-2/Type/Sensor/Id/ibmproject/evt/data/fmt/json", "payload": { "warning!!": "244.971e4" }, "deviceId": "ibmproject", "deviceType": "Sensor", "eventType": "data", "format": "json" }
0198213c192cb2c244cc2433f1...	0198213c192cb2c244cc2433f1...	{ "rev": "1-d26c5b40891e13c6c...", "topic": "iot-2/Type/Sensor/Id/ibmproject/evt/data/fmt/json", "payload": { "warning!!": "244.971e4" }, "deviceId": "ibmproject", "deviceType": "Sensor", "eventType": "data", "format": "json" }
0198213c192cb2c244cc2433f1...	0198213c192cb2c244cc2433f1...	{ "rev": "1-cde2dd17c519394dfab774738c495f8b", "topic": "iot-2/Type/Sensor/Id/ibmproject/evt/data/fmt/json", "payload": { "warning!!": "244.971e4" }, "deviceId": "ibmproject", "deviceType": "Sensor", "eventType": "data", "format": "json" }
0198213c192cb2c244cc2433f1...	0198213c192cb2c244cc2433f1...	{ "rev": "1-f96eb0460bc16cfab0...", "topic": "iot-2/Type/Sensor/Id/ibmproject/evt/data/fmt/json", "payload": { "warning!!": "244.971e4" }, "deviceId": "ibmproject", "deviceType": "Sensor", "eventType": "data", "format": "json" }
1a921f21cbe229b86f599acb45...	1a921f21cbe229b86f599acb45...	{ "rev": "1-7226808794cd47b7c...", "topic": "iot-2/Type/Sensor/Id/ibmproject/evt/data/fmt/json", "payload": { "warning!!": "244.971e4" }, "deviceId": "ibmproject", "deviceType": "Sensor", "eventType": "data", "format": "json" }
1a921f21cbe229b86f599acb45...	1a921f21cbe229b86f599acb45...	{ "rev": "1-1bbdd9a985bd56cf9...", "topic": "iot-2/Type/Sensor/Id/ibmproject/evt/data/fmt/json", "payload": { "warning!!": "244.971e4" }, "deviceId": "ibmproject", "deviceType": "Sensor", "eventType": "data", "format": "json" }
20a854e5445fa818e6c1de049...	20a854e5445fa818e6c1de049...	{ "rev": "1-7226808794cd47b7c...", "topic": "iot-2/Type/Sensor/Id/ibmproject/evt/data/fmt/json", "payload": { "warning!!": "244.971e4" }, "deviceId": "ibmproject", "deviceType": "Sensor", "eventType": "data", "format": "json" }
20a854e5445fa818e6c1de049...	20a854e5445fa818e6c1de049...	{ "rev": "1-3ad288ecad57f039e...", "topic": "iot-2/Type/Sensor/Id/ibmproject/evt/data/fmt/json", "payload": { "warning!!": "244.971e4" }, "deviceId": "ibmproject", "deviceType": "Sensor", "eventType": "data", "format": "json" }
20a854e5445fa818e6c1de049...	20a854e5445fa818e6c1de049...	{ "rev": "1-1bbdd9a985bd56cf9...", "topic": "iot-2/Type/Sensor/Id/ibmproject/evt/data/fmt/json", "payload": { "warning!!": "244.971e4" }, "deviceId": "ibmproject", "deviceType": "Sensor", "eventType": "data", "format": "json" }
298ed6f8d9b3b815f5ac7c061e...	298ed6f8d9b3b815f5ac7c061e...	{ "rev": "1-1de7240f6e5307a1b9...", "topic": "iot-2/Type/Sensor/Id/ibmproject/evt/data/fmt/json", "payload": { "warning!!": "244.971e4" }, "deviceId": "ibmproject", "deviceType": "Sensor", "eventType": "data", "format": "json" }

## 5. Detecting the garbage level

