

## Sprint – 2

Date	10- Nov-2022
Team ID	PNT2022TMID45881
Project Name	<b>Project - IoT Based Safety Gadget for Child Safety Monitoring &amp; Notification</b>
Maximum Marks	<b>8 Marks</b>

**USN- 4 :** Integrating the IBM Watson IoT Platform and Cloudant DB with the node red.

- Launching IBM IoT Watson

The screenshot displays the IBM Watson IoT Platform interface. The main heading is 'Browse Devices'. Below it, there are two buttons: 'All Devices' (selected) and 'Diagnose'. 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 text is a search bar labeled 'Search by Device ID' and a 'Device Simulator' toggle switch. The table below lists the following data:

Device ID	Status	Device Type	Class ID	Date Added
28	Disconnected	Tracker	Device	Nov 6, 2022 11:54 AM

At the bottom of the table, it shows 'Items per page: 50' and '1-1 of 1 item'. The pagination controls show '1 of 1 page'.

- Implementing the node-red in IBM cloud.

- 

Launch the cloudbant DB and create a database to store the location data.

**Databases**

Database name

Create Database {} JSON

Your Databases

Name	Size	# of Docs	Partitioned	Actions
child_location	0 bytes	0	Yes	
noderedrvwbe20221105	30.4 KB	4	No	
sample	0 bytes	0	Yes	

Showing 1-3 of 3 databases. Databases per page 20

- For our project we are creating a database called child\_location.

**USN – 5** : Developing the Python code for connecting with IBM Watson IoT platform.

**Databases**

Database name

Create Database {} JSON

Your Databases

Name	Size	# of Docs	Partitioned	Actions
child_location	0 bytes	0	Yes	

```
1 import time
2 import wiotp.sdk.application
3 print("Hello")
4 myConfig = {
5     "identity": {
6         "orgId": "fjde2i",
7         "typeId": "Tracker",
8         "deviceId": "28",
9     },
10    "auth": {
11        "token": "123456789"
12    }
13 }
14 client = wiotp.sdk.device.DeviceClient(config = myConfig, logHandlers = None)
15 client.connect()
16
17 while True:
18     name = "Child"
19     #in area location
20
21     latitude = 17.4219272
22     longitude = 78.5488783
23
24
25     #out area location
26
27     #latitude = 17.4219272
28     #longitude = 78.5488783
29     myData = {'name': name, 'lat': latitude, 'lon': longitude}
30     client.publishEvent(eventId = "status", msgFormat = "json", data = myData, qos = 0, onPublish = None)
31     print("Data published to IBM IoT Platform: ", myData)
32     time.sleep(5)
33
34 client.disconnect()
```

- Connected successfully with IBM IoT Watson.



```
Run: child x
C:\Users\dell\AppData\Local\Programs\Python\Python311\python.exe C:/Users/dell/AppData/Local/Programs/Python/child.py
Data published to IBM IoT Platform: {'name': 'Child', 'lat': 17.4219272, 'lon': 78.5488783}
2022-11-08 20:56:53,786 wiotp.sdk.device.client.DeviceClient INFO Connected successfully: d:fjde2i:Tracker:28
Data published to IBM IoT Platform: {'name': 'Child', 'lat': 17.4219272, 'lon': 78.5488783}
Data published to IBM IoT Platform: {'name': 'Child', 'lat': 17.4219272, 'lon': 78.5488783}
Data published to IBM IoT Platform: {'name': 'Child', 'lat': 17.4219272, 'lon': 78.5488783}
Data published to IBM IoT Platform: {'name': 'Child', 'lat': 17.4219272, 'lon': 78.5488783}
```

- IBM IoT Watson platform receiving the details of the child's location.

Event	Value	Format	Last Received
status	{"name":"Child","lat":17.4219272,"lon":78.5488...	json	a few seconds ago
status	{"name":"Child","lat":17.4219272,"lon":78.5488...	json	a few seconds ago
status	{"name":"Child","lat":17.4219272,"lon":78.5488...	json	a few seconds ago
status	{"name":"Child","lat":17.4219272,"lon":78.5488...	json	a few seconds ago
status	{"name":"Child","lat":17.4219272,"lon":78.5488...	json	a few seconds ago