

Sprint - 1

Team Id	PNT2022TMID23518
Title	Hazardous Area Monitoring for Industrial Plant using IoT

Configuring the IBM cloud services

IBM Watson Platform

The screenshot shows the IBM Cloud IoT Platform console. The top navigation bar includes the IBM Cloud logo, a search bar, and links to Catalog, Manage, and Jaganath A's Account. The main content area is titled 'IoT Platform-i7' and shows a 'Manage' tab selected. A sidebar on the left contains links for Manage, Plan, and Connections. The main content area features a large graphic of a network of devices and a section titled 'Let's get started with IBM Watson IoT Platform' with a 'Launch' button. Below this, there is a section titled 'Ready for the next level?' and 'IBM Watson IoT Platform Journey' with three stages: Lite, Non-Production, and Production, each with a brief description and a list of features.

Let's get started with IBM Watson IoT Platform

Securely protect, control, and manage devices. Quickly build IoT applications that analyze data from the physical world.

[Launch](#) [Docs](#)

Ready for the next level?

IBM Watson IoT Platform Journey

- Lite**
The Lite service plan provides a lightweight development environment to get you started with the connectivity capabilities of Watson IoT Platform.
 - Free
 - 200 MB data-transfer limit
 - 500 application bindings limit
- Non-Production**
The Non-Production service plan is a full-featured, fully-integrated offering that enables you to explore Watson IoT Platform to see how the service can fit into your IoT environment.
 - Starts at \$300 per month
 - Capacity limit based on device type
 - Optional Analytics Service and Blockchain
- Production**
The Production service is a fully managed SaaS offering that enables you to manage and analyze enterprise IoT data.
 - Includes IBM Service & Support
 - Pricing based on number of devices per device type

The screenshot shows the 'Browse Devices' page in the IBM Watson IoT Platform console. The top navigation bar includes the IBM Watson IoT Platform logo, a search bar, and links to Browse, Action, Device Types, and Interfaces. The main content area is titled 'Browse Devices' and has a 'Diagnose' button. Below this, there is a table of devices with columns for Device ID, Status, Device Type, Class ID, Date Added, and Descriptive Location. The table shows three devices: ESP32_sensor, IoT001, and TH-01, all with a status of 'Disconnected'. The bottom of the page shows pagination information: 'Items per page: 50' and '1 of 1 page'.

Browse Devices

[All Devices](#) [Diagnose](#)

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.

Search by Device ID

Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location
ESP32_sensor	Disconnected	ESP_Controller	Device	18 Nov 2022 6:33 PM	
IoT001	Disconnected	NodeMCU	Device	18 Nov 2022 5:24 PM	
TH-01	Disconnected	TempAndHumid	Device	18 Nov 2022 5:38 PM	

Items per page: 50 | 1-3 of 3 items

1 of 1 page

Node Red Configuration

The screenshot displays the IBM Cloud console for a resource named "Node RED YENDV 2022-11-10". The interface is divided into several sections:

- Details:** Shows the App URL (<http://159.122.177.234:30241>), Source (<https://us-south-gf.cloud.ibm.com/jagangrtnls/NodeREDYE...>), Resource group (Default), Deployment target (Kube/Helm), and Created date (10/11/2022).
- Services:** Features a "Cloudant" service card with links to "Open dashboard", "Documentation", and "API reference", along with a "Credentials" dropdown. Buttons for "Connect existing services" and "Create service" are at the bottom.
- Deployment Automation:** Lists two pipelines: "pr-pipeline" (No stages selected) and "ci-pipeline" (Status: Success).
- Getting started quickly:** A sidebar guide with five steps: 1. Use the Services card to connect a service to your app. 2. If you want to view the code before your app is deployed, click Download code to obtain the .zip file. 3. Click Deploy your app in the Deployment Automation card to select the deployment target and configure the Continuous Delivery service. 4. After the deployment begins, you can view the status of the deployment, modify your app, view your repo, or view the app's URL. 5. If you make any changes to your app, be sure to deploy it again.







Cloudant dB

The screenshot shows the IBM Cloud console for a resource named "node-red-yendv-2022--cloudant-1668085203840", which is marked as "Active". The interface includes a left sidebar with "Manage", "Service credentials", "Plan", and "Connections". The main content area has tabs for "Overview", "Capacity", and "Docs", with "Overview" selected. A "Launch Dashboard" button is in the top right.

Deployment details:

- CRN:** orxv1:bluemix:public:cloudantnosqldb:eu-gb-a/12649ecbca094c49b02e42f8b7bce601b924af7e-28e6-4cb4-bff9-1ad00b7c1600::
- Location:** London
- External endpoint:** <https://68891c99-a3d3-4b5f-bfff-7e82c4038355-bluemix.cloudant.com>
- External endpoint (preferred):** <https://68891c99-a3d3-4b5f-bfff-7e82c4038355-bluemix.cloudantnosqldb.appdomain.cloud>
- Authentication methods:** IBM Cloud IAM and Cloudant credentials. A "Migrate to IAM Only" button is present.
- Activity Tracker event types:** Management (selected) and Error.
- Disk encryption:** Yes. Automatically generated disk encryption key.

Capacity details:

Databases					Database name	Create Database	JSON		
Your Databases									
Name	Size	# of Docs	Partitioned	Actions					
industryiot	138.7 KB	795	No	  					
ordered	29.1 KB	4	No	  					

Showing 1-2 of 2 databases. Databases per page 20

Generation of Python code for publishing the random sensor data to the IBM IoT Platform.

```

Python Script IoT.py - C:\Users\91934\Downloads\Python Script IoT.py (3.7.8)
File Edit Format Run Options Window Help

import time
import sys
import ibmiotf.application
import ibmiotf.device
import random

#Provide your IBM Watson Device Credentials
organization = "4u10ex"
deviceType = "ModemCTU"
deviceId = "1c1f0b1"
authMethod = "token"
authToken = "1134347446"

# Initialize MQTT
def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status=="motoron":
        print("Motor on on")
    else:
        print("Motor is off")

try:
    deviceOptions = {"org": organization, "type": deviceType, "id": deviceId, "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()

# Connect and send a datapoint "hello" with value "world" into the cloud as an event of type "greeting" 10 times
deviceCli.connect()

while True:
    #Get Sensor Data from IOTUI
    temp=random.randint(10,100)
    humid=random.randint(0,100)
    data = {'temp': temp, 'humid': humid}
    #print data
    def myOnPublishCallback():
        print ("Published Temperature = %s C° & temp, "Humidity:%s" %humid)

    success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0, on_publish=myOnPublishCallback)
    if not success:
        print("Not connected to IoTP")
        time.sleep(1)
        deviceCli.commandCallback = myCommandCallback

# Disconnect the device and application from the cloud
deviceCli.disconnect()

```

Random Temperature and humidity values generated and published to IBM IoT platform

```
----- RESTART: C:\Users\91934\Downloads\Python Script IOT.py -----
2022-11-18 20:59:00,364 idmiotf.device.Client INFO Connected successfully: d:4wj0mx:NodeMCU:IoT001
Published Temperature = 7 C Humidity:71
Published Temperature = 96 C Humidity:20
Published Temperature = 69 C Humidity:8
Published Temperature = 32 C Humidity:11
Published Temperature = 64 C Humidity:26
Published Temperature = 94 C Humidity:71
Published Temperature = 51 C Humidity:10
Published Temperature = 58 C Humidity:46
Published Temperature = 94 C Humidity:27
Published Temperature = 93 C Humidity:52
Published Temperature = 85 C Humidity:63
Published Temperature = 55 C Humidity:44
Published Temperature = 36 C Humidity:76
Published Temperature = 34 C Humidity:54
Published Temperature = 35 C Humidity:34
Published Temperature = 60 C Humidity:14
Published Temperature = 77 C Humidity:18
Published Temperature = 69 C Humidity:73
Published Temperature = 32 C Humidity:80
Published Temperature = 41 C Humidity:89
Published Temperature = 93 C Humidity:57
```

Identity	Device Information	Recent Events	State	Logs
Device ID	IoT001			
Device Type	NodeMCU			
Date Added	8 Nov 2022 5:24 PM			
Added By	jagangang@nra@gmail.com			
Connection Status	Connected			

Data stored in Database

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	
IoTSensor	{"temp":67,"humid":61}	json	a few seconds ago	
IoTSensor	{"temp":20,"humid":88}	json	a few seconds ago	
IoTSensor	{"temp":51,"humid":91}	json	a few seconds ago	
IoTSensor	{"temp":59,"humid":7}	json	a few seconds ago	
IoTSensor	{"temp":65,"humid":8}	json	a few seconds ago	

Monitoring

Databases

Replication

Active Tasks

Account

Support

Documentation

industryiot > 000245d59da9ac61433e5f634f503b51

Save Changes

Cancel

Upload Attachment

Clone Document

Delete

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

```
1  {
2    "_id": "000245d59da9ac61433e5f634f503b51",
3    "_rev": "1-9513887f1ae94c2d8163c2576f00a803",
4    "topic": "iot-3/types/NodeMCU/1d/1d1001/evt/IoTSensor/fet/json",
5    "payload": {
6      | "temp": 38,
7      | "humid": 34
8    },
9    "deviceId": "1d1001",
10   "deviceType": "NodeMCU",
11   "eventType": "IoTSensor",
12   "format": "json"
13 }
14
15
```

IBM Cloudant

Log Out IBMid-6610044C12