



**GOVERNMENT COLLEGE OF ENGINEERING
CHETTIKARAI, DHARMAPURI**



Safety Gadget for Child Safety Monitoring and Notification

IBM NALAIYATHIRAN

Project Development –Delivery of Sprint 2

Creating Node –Red service and connect with IBM cloud and Web UI

TITLE	IOT based child safety gadget for child safety monitoring and notification
DOMAIN NAME	INTERNET OF THINGS
TEAM ID	PNT2022TMID41271
TEAM LEADERNAME	Dharshanraj V
TEAM MEMBER NAME	Chinnan S Keerthiprasath S Koushik R P
MENTOR NAME	Dr. DINESH G

Creating Node-Red service:

The screenshot displays a Node-RED web interface in a browser. The address bar shows the URL: `node-red-knnqv-2022-11-07.au-syd.mybluemix.net/red/#flow/78f02f688dd8e1d2`. The interface is divided into several sections:

- Top Bar:** Contains various icons for chat, IBM, and other applications, along with a search bar.
- Left Panel:** Shows a sidebar with a search bar and a list of nodes. The 'function' node is highlighted.
- Flow 1:** The main workspace where the flow is built. It starts with an 'IBM IoT' node (connected). This node branches into several parallel paths:
 - A path through a 'function' node to a 'childsafty' node.
 - A path through a 'temp' node to a 'gauge' node, which then connects to a 'Temperature' node.
 - A path through a 'temp_status' node to a 'temp_status' node.
 - A path through a 'latitude' node to a 'Latitude' node.
 - A path through a 'your_child_zone' node to a 'Child Zone' node.
 - A path through a 'longitude' node to a 'Longitude' node.
 - A path through a 'worldmap' node (connected 0) to a 'worldmap' node.
- Right Panel:** Contains a 'debug' console showing logs for the flow. The logs include timestamps and data payloads, such as:


```
11/10/2022, 10:00:45 PM node: 65909d20f5fd4648
iot-2/type/ABCD/id/13/levfIoTSensorgpsdata/fmt/json :
msg.payload : Object
  > { temp: 50, lat:
    12.131629972663186, lon:
    78.19606388397351, name: "Child" }
```

Connecting with IBM Cloud:

Using IBM IOT node through API key

The screenshot shows the IBM Watson IoT Platform dashboard. The URL is `zwx6lb.internetofthings.ibmcloud.com/dashboard/apps/browse/add`. The page title is "The API key has been added." Below the title, a message states: "Authentication tokens are non-recoverable. If you misplace this token, you will need to re-register the API key to generate a new authentication token." The "Generated Details" section shows the API Key as `a-zwx6lb-z7sryerler` and the Authentication Token as `dO&H(qcUv)icaFOYcb`. The "API Key Information" section shows the Role as "Standard Application" and Expires as "Never". A warning icon and text state: "Make a note of the generated authentication token. Lost authentication tokens cannot be recovered. If you lose the token, you must reregister the API to generate a new token." A "1 Simulation running" notification is visible at the bottom right.

The screenshot shows the IBM Watson IoT Platform dashboard. The URL is `zwx6lb.internetofthings.ibmcloud.com/dashboard/apps/browse/add`. The page title is "API Key Information". The "Key" is `a-zwx6lb-97epyzrfc`. The "Description" is "-". The "Role" is "Standard Application". The "Expires" is "Never". The "Date Added" is "Nov 7, 2022 5:54 PM". The "Last Update" is "Nov 7, 2022 5:54 PM". The "Last Edited By" is `613519106013@smartinternz.com`. A "1 Simulation running" notification is visible at the bottom right.

Transferring values from Python Code:

```
Child Safety device.py - C:/Users/kutta/Desktop/IBM-Dr/Child Safety device.py (3.7.4)
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 = "illzai"
deviceType = "latlonitem"
deviceId = "613510"
authMethod = "token"
authToken = "1092837465"
#api key (a-illzai-mbdxqo6z0s)
#api token (zSYzISuANF&F_x7GkT)

try:
    deviceOptions = {"org": organization, "type": deviceType, "id": deviceId, "auth-metho
    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
print("power on ")
print("checking connection to waston iot...")
time.sleep(2)
deviceCli.connect()
print("dear user ... welcome to IBM-IOT ")
print("i can provide your children live location and temperature ")
print()
name=str(input("enter your child name:"))

while True:

    temperature=random.randint(20,50)#random temperature for your child
    latitude=random.uniform(10.781377,10.78643)#random latitude for your child
    longitude=random.uniform(79.129113,79.134014)#random longitude for your child

    data = { 'temp' : temperature, 'lat': latitude, 'lon':longitude, 'name':name }
    #print data
    def myOnPublishCallback():
        #Print the temperature, latitude, longitude, name, and longitude

Python 3.7.4 Shell
File Edit Shell Debug Options Window Help
Python 3.7.4 (tags/v3.7.4:e09359112e, Jul 8 2019, 20:34:20) [MSC v.1916 64 i
(AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/kutta/Desktop/IBM-Dr/Child Safety device.py =====
power on
checking connection to waston iot...
2022-11-10 22:14:21,799 ibmiotf.device.Client INFO Connected succe
lly: d:illzai:latlonitem:613510
dear user ... welcome to IBM-IOT
i can provide your children live location and temperature

enter your child name:child
Published Temperature = 39 C latitude = 10.782749628132827 & longitude = 79.
867253162 & to IBM Watson
Published Temperature = 39 C latitude = 10.782669248109656 & longitude = 79.
1255540076 & to IBM Watson
Published Temperature = 43 C latitude = 10.781765104656792 & longitude = 79.
077864707 & to IBM Watson
Published Temperature = 30 C latitude = 10.786083936690018 & longitude = 79.
2366715787 & to IBM Watson
Published Temperature = 31 C latitude = 10.784810558975826 & longitude = 79.
0117359415 & to IBM Watson
Published Temperature = 45 C latitude = 10.785949922923024 & longitude = 79.
5563867668 & to IBM Watson
Published Temperature = 24 C latitude = 10.784168891438233 & longitude = 79.
9528906442 & to IBM Watson
Published Temperature = 23 C latitude = 10.786248060883958 & longitude = 79.
4368596464 & to IBM Watson
Published Temperature = 27 C latitude = 10.783808327214418 & longitude = 79.
951933729 & to IBM Watson
Published Temperature = 43 C latitude = 10.786340416981865 & longitude = 79.
7748803969 & to IBM Watson
Published Temperature = 49 C latitude = 10.786208956579015 & longitude = 79.
2192551409 & to IBM Watson
Published Temperature = 45 C latitude = 10.783690544907325 & longitude = 79.
504415061 & to IBM Watson
```

Node-Red:

The screenshot shows the Node-RED web interface in a browser. The main workspace displays a flow named 'Flow 1'. The flow starts with an 'IBM IoT' node (connected). It branches into several parallel paths, each representing a different data field from the device: 'Child name', 'temp', 'temp_status', 'latitude', 'your_child_zone', and 'longitude'. Each path consists of a function node (labeled with 'f') followed by a 'gauge' node (labeled with 'abc'). The 'worldmap' node is also present at the bottom of the flow. The right sidebar shows the 'debug' console with several log entries, including timestamps and JSON payloads for the data being received from the device. The bottom status bar shows the system clock as 10:01 PM on 11/10/2022.

Node-Red Dashboard:



