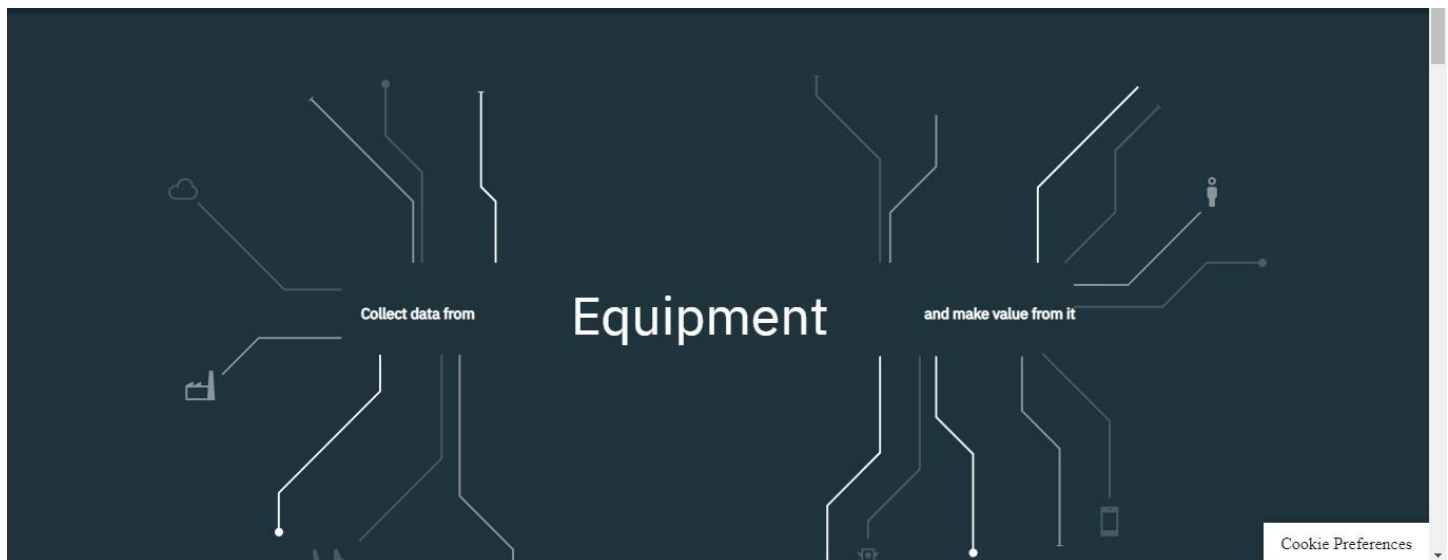


TITLE	IOT based child safety gadget for child safety monitoring and notification
TEAM ID	PNT2022TMID17111
SUBMISSION DATE	11 November 2022

Creating IBM Cloud Service and creating the device:



Creating Python Code:

```
import time

import sys

import ibmiotf.application

import ibmiotf.device

import random

#Provide your IBM Watson Device

Credentials organization = "zwx6lb"

deviceType = "ABCD" deviceId = "13"

authMethod = "token" authToken =

"12345678"

#api key {a-illza1-mbdxqo6z0s}

#api token {zSYzISuAWF&F_x7GkT}


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

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
    a="Child inside the geofence"      b=" Child outside the geofence"      c="High
temperature"      d="Low temperature"      x={'your_child_Zone':a}
y={'your_child_Zone':b}      z={'temp_condition':c}      w={'temp_condition':d}

    data = { 'temp' : temperature, 'lat': latitude, 'lon':longitude, 'name':name }

    #print data      def
myOnPublishCallback():
print ("Published Temperature
=  %s  C"  %  temperature,
"latitude = %s  %" % latitude,

```

```
"longitude = %s %%" % longitude, "to IBM Watson")
```

```
print("\n")
```

```
    success = deviceCli.publishEvent("IoTSensorgpsdata", "json", data, qos=0,  
on_publish=myOnPublishCallback)
```

```
    if latitude>=10.78200 and latitude<=10.786000 and longitude >=79.130000 and longitude  
<=79.133000:
```

```
deviceCli.publishEvent("IoTSensorgpsdata","json",data=x,qos=0,on_publish=myOnPublishCallb  
ack)
```

```
    print(x)
```

```
print("\n")    else:
```

```
deviceCli.publishEvent("IoTSensorgpsdata","json",data=y,qos=0,on_publish=myOnPublishCallb  
ack)
```

```
    print(y)
```

```
print("\n")
```

```
    if (temperature>35):
```

```
deviceCli.publishEvent("IoTSensorgpsdata","json",data=z,qos=0,on_publish=myOnPublishCallb  
ack)
```

```
    print(c)
```

```
print("\n")    else:
```

```
deviceCli.publishEvent("IoTSensorgpsdata","json",data=w,qos=0,on_publish=myOnPublishCall  
back)
```

```
    print(d)
```

```
    print("\n")
```

```
        if not success:            print("Not
connected to IoT")
print("\n")            time.sleep(3)
# Disconnect the device and application from the cloud
deviceCli.disconnect()
```

Connecting IBM Watson and python Code:

The screenshot displays two separate code editing environments. The left environment is a Python 3.7.4 Shell window with a menu bar (File, Edit, Shell, Debug, Options, Window, Help). It contains several lines of Python code, some commented out with '#Pr' or '#ap'. The active code includes logic to check if a child is inside or outside a geofence and to report temperature status (e.g., 'High temperature'). It also mentions coordinates like latitude = 12.130 and longitude = 78.198.

The right environment is another code editor showing a different script. Visible code includes a dictionary key-value pair: '-method': authMethod, "auth-token": authToken}. Below it, there's a comment or instruction: type "greeting" 10 times. At the bottom, a partially visible line suggests a loop or repeated action: ...repeats "Hello" 10 times...



13	Connected	ABCD	Device	Nov 2, 2022 10:55 PM	→ ...
----	-----------	------	--------	----------------------	-------

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
IoTSensorgp...	{"temp_status":"High temperature"}	json	a few seconds ago
IoTSensorgp...	{"your_child_zone":"Outside the geofence"}	json	a few seconds ago
IoTSensorgp...	{"temp":50,"lat":12.132819998043411,"lon":78...	json	a few seconds ago
IoTSensorgp...	{"temp_status":"Low temperature"}	1 Simulation running	
IoTSensorgp...	{"your child zone":"Outside the geofence"}		