ASSIGNMENT-4

Code:

```
import ibmiotf.application
import ibmiotf.device
import random
import json
import time

#Provide your IBM Watson Device Credentials
organization = &amp;quot;hvqy0x&amp;quot;
deviceType = &amp;quot;iotdevice&amp;quot;
deviceId = &amp;quot;2001&amp;quot;
authMethod = &amp;quot;token&amp;quot;
authToken = &amp;quot;12345678910&amp;quot;

# Initialize the device client.

T=0
H=0

def myCommandCallback(cmd):
print(&amp;quot;Command received: %s&amp;quot; % cmd.data[&amp;#39;command&amp;#39;])

if cmd.data[&amp;#39;command&amp;#39;]==&amp;#39;lighton&amp;#39;:
print(&amp;quot;LIGHT ON IS RECEIVED&amp;quot;)

elif cmd.data[&amp;#39;command&amp;#39;]==&amp;#39;lightoff&amp;#39;:
print(&amp;quot;LIGHT OFF IS RECEIVED&amp;quot;)
```

if cmd.command == "setInterval":

if 'interval' not in cmd.data:

print("Error - command is missing required information: 'interval'")

else:

interval = cmd.data['interval']

elif cmd.command == "print":

if 'message' not in cmd.data:

print("Error - command is missing required information: 'message'")

else:

print(cmd.data['message'])

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

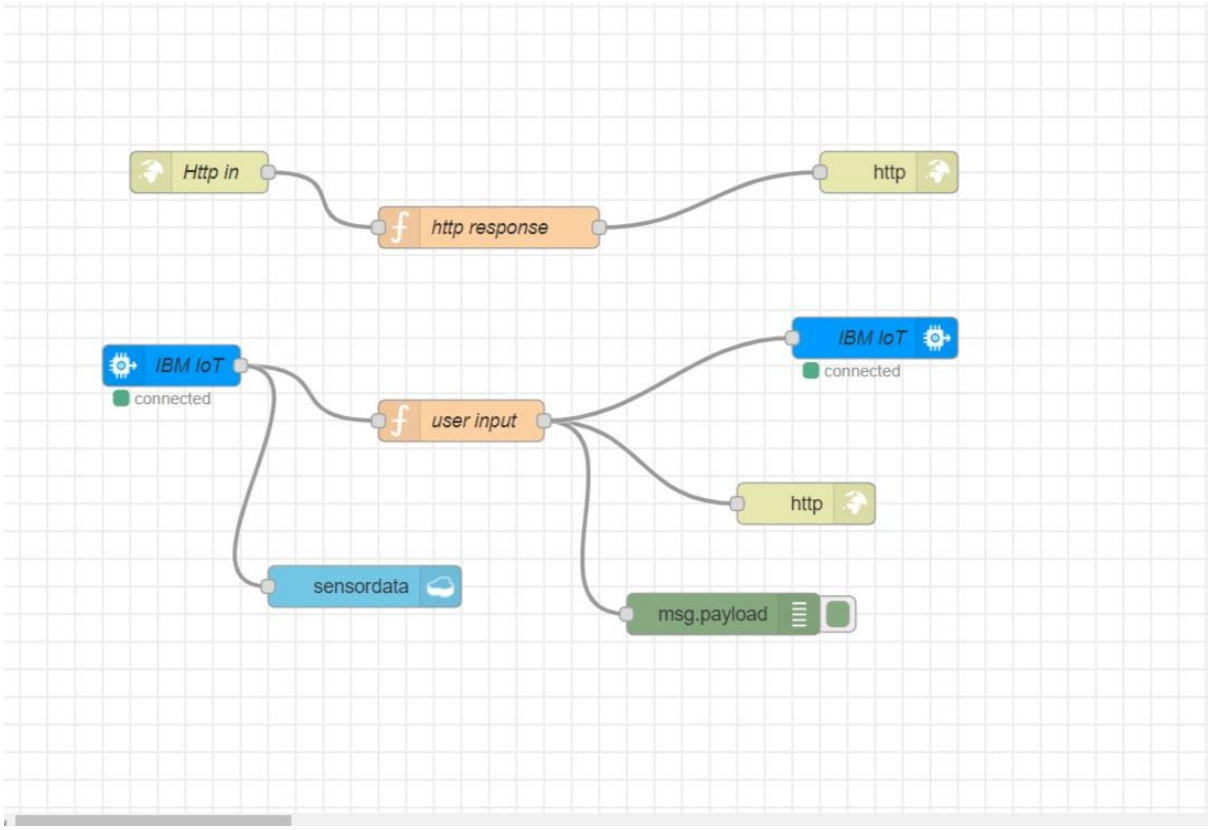
deviceCli.connect()

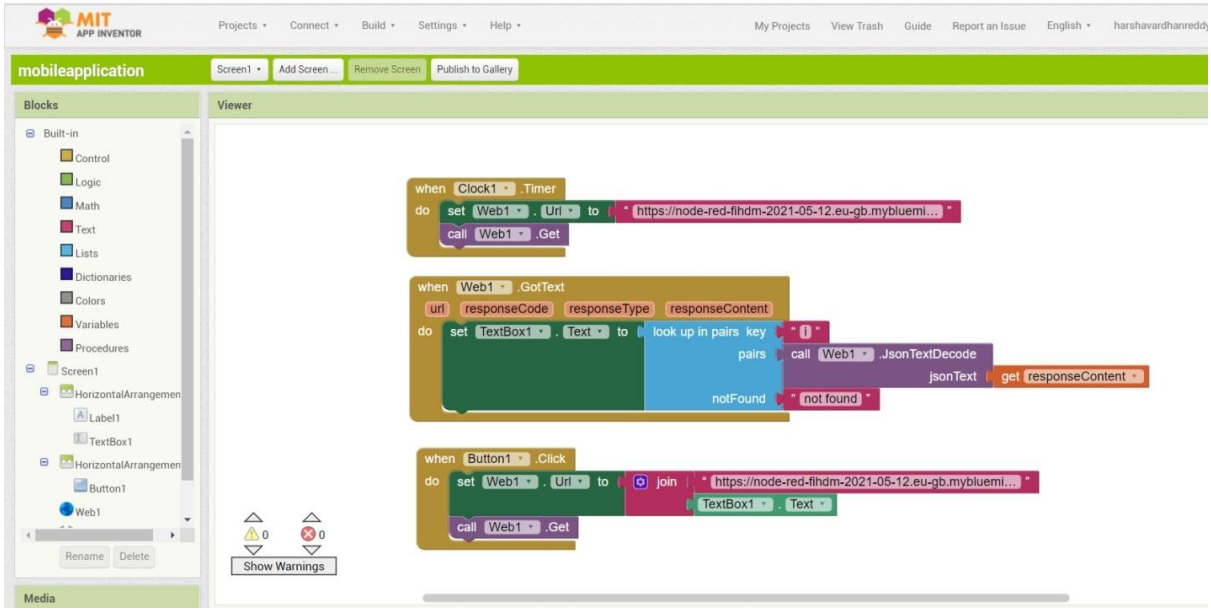
while True:

T=23

H=45

```
#Send Temperature &amp; Humidity to IBM Watson

data = {&quot;d&quot;:{ &#39;temperature&#39; : T, &#39;humidity&#39;: H }}

#print (data)

def myOnPublishCallback():

print (&quot;Published Temperature = %s C&quot; % T, &quot;Humidity = %s %%&quot; % H,
&quot;to IBM Watson&quot;)


success = deviceCli.publishEvent(&quot;Data&quot;, &quot;json&quot;, data, qos=0,

on_publish=myOnPublishCallback)

if not success:

print(&quot;Not connected to IoTF&quot;)

time.sleep(1)


deviceCli.commandCallback = myCommandCallback


# Disconnect the device and application from the cloud

deviceCli.disconnect()
```


Http in
http response
http
IBM IoT
connected
IBM IoT
connected
user input
http
sensordata
msg.payload


MIT APP INVENTOR
Projects
Connect
Build
Settings
Help
My Projects
View Trash
Guide
Report an Issue
English
harshavardhanreddy
mobileapplication
Screen1
Add Screen ...
Remove Screen
Publish to Gallery
Blocks
Viewer
Built-in
Control
Logic
Math
Text
Lists
Dictionaries
Colors
Variables
Procedures
Screen1
HorizontalArrangemen
Label1
TextBox1
HorizontalArrangemen
Button1
Web1
Rename
Delete
Media
Show Warnings
when Clock1 .Timer
do set Web1 . Url to " https://node-red-fihdm-2021-05-12.eu-gb.mybluemi... "
call Web1 .Get
when Web1 .GotText
url responseCode responseType responseContent
do set TextBox1 . Text to look up in pairs key " {} "
pairs call Web1 .JsonTextDecode
jsonText get responseContent
notFound " not found "
when Button1 .Click
do set Web1 . Url to join " https://node-red-fihdm-2021-05-12.eu-gb.mybluemi... "
TextBox1 . Text
call Web1 .Get


IDLE Shell 3.9.2
File Edit Shell Debug Options Window Help
Python 3.9.2 (tags/v3.9.2:1a79785, Feb 19 2021, 13:44:55) [MSC v.1928 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/admin/AppData/Local/Programs/Python/Python39/mobile app.py =
2021-05-25 17:11:25,368 ibmiotf.device.Client INFO Connected successfully: d:frtx4v:iotdevice:1001
Published Temperature = 23 C Humidity = 45 % to IBM Watson
Published Temperature = 23 C Humidity = 45 % to IBM Watson
Published Temperature = 23 C Humidity = 45 % to IBM Watson
Published Temperature = 23 C Humidity = 45 % to IBM Watson
Published Temperature = 23 C Humidity = 45 % to IBM Watson
Command received: hiii
Published Temperature = 23 C Humidity = 45 % to IBM Watson