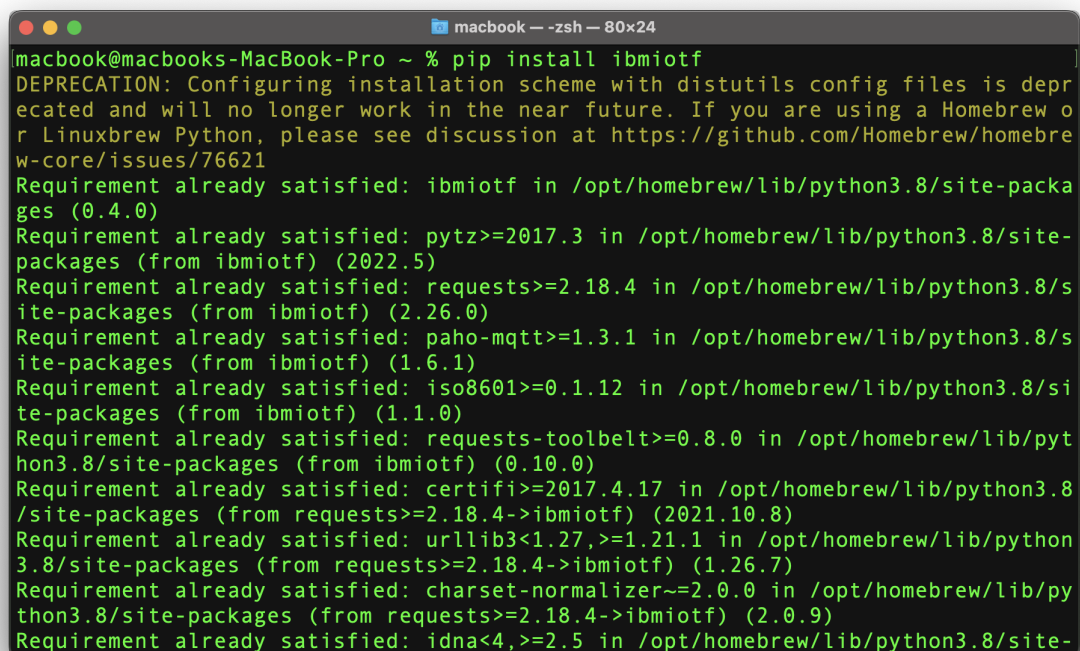


Step1 : Installing ibmiotf python module

A terminal window titled 'macbook - zsh - 80x24' showing the command 'pip install ibmiotf' being executed. The output displays a deprecation warning and a list of requirements that are already satisfied, including ibmiotf, pytz, requests, paho-mqtt, iso8601, requests-toolbelt, certifi, urllib3, charset-normalizer, and idna.

```
macbook@macbooks-MacBook-Pro ~ % pip install ibmiotf
DEPRECATION: Configuring installation scheme with distutils config files is deprecated and will no longer work in the near future. If you are using a Homebrew or Linuxbrew Python, please see discussion at https://github.com/Homebrew/homebrew-core/issues/76621
Requirement already satisfied: ibmiotf in /opt/homebrew/lib/python3.8/site-packages (0.4.0)
Requirement already satisfied: pytz>=2017.3 in /opt/homebrew/lib/python3.8/site-packages (from ibmiotf) (2022.5)
Requirement already satisfied: requests>=2.18.4 in /opt/homebrew/lib/python3.8/site-packages (from ibmiotf) (2.26.0)
Requirement already satisfied: paho-mqtt>=1.3.1 in /opt/homebrew/lib/python3.8/site-packages (from ibmiotf) (1.6.1)
Requirement already satisfied: iso8601>=0.1.12 in /opt/homebrew/lib/python3.8/site-packages (from ibmiotf) (1.1.0)
Requirement already satisfied: requests-toolbelt>=0.8.0 in /opt/homebrew/lib/python3.8/site-packages (from ibmiotf) (0.10.0)
Requirement already satisfied: certifi>=2017.4.17 in /opt/homebrew/lib/python3.8/site-packages (from requests>=2.18.4->ibmiotf) (2021.10.8)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in /opt/homebrew/lib/python3.8/site-packages (from requests>=2.18.4->ibmiotf) (1.26.7)
Requirement already satisfied: charset-normalizer~=2.0.0 in /opt/homebrew/lib/python3.8/site-packages (from requests>=2.18.4->ibmiotf) (2.0.9)
Requirement already satisfied: idna<4,>=2.5 in /opt/homebrew/lib/python3.8/site-
```

Step 2 : Develop the python script

Python program :

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

```
#Provide your IBM Watson Device Credentials
organization = "95kfev"
deviceType = "NodeMCU"
deviceId = "12345"
authMethod = "token"
authToken = "12345678"
```

```
# Initialize GPIO
```

```
def myCommandCallback(cmd):
```

```

print("Command received: %s" % cmd.data['command'])
status=cmd.data['command']
if status=="lighton":
    print ("led is on")
else :
    print ("led is off")

#print(cmd)

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 DHT11

    temp=random.randint(0,100)
    Humid=random.randint(0,100)

    data = { 'temp' : temp, 'Humid': Humid }
    #print data
    def myOnPublishCallback():
        print ("Published Temperature = %s C" % temp, "Humidity = %s %" % Humid, "to
IBM Watson")

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

        deviceCli.commandCallback = myCommandCallback

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

```

Step 3 : Creation of IOT Device in Watson IoT Platform

The screenshot shows the IBM Watson IoT Platform dashboard. The URL is `95kfev.internetofthings.ibmcloud.com/dashboard/devices/browse`. The user is logged in as `sendhanamudhanmani@gmail.com` with ID `95kfev`. The dashboard has tabs for `Browse`, `Action`, `Device Types`, and `Interfaces`. A sidebar on the left contains icons for various functions. The main content area shows a table of devices. One device is listed with ID `12345`, status `Disconnected`, type `NodeMCU`, and class `Device`. A modal window is open for this device, showing details under the `Identity` tab.

Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location
12345	Disconnected	NodeMCU	Device	15 Nov 2022 00:36	

Search by Device ID:

Device Simulator: ☐

Modal Window Details:

- Device ID: 12345
- Device Type: NodeMCU
- Date Added: 15 Nov 2022 00:36
- Added By: sendhanamudhanmani@gmail.com
- Connection Status: Disconnected
- Last Connected: 15 Nov 2022 01:06
- Client Address: 49.37.215.211 SecureToken
- Duration: 4 minutes
- Data Transferred: 17.9 KB

Items per page: 50 | 1-1 of 1 item | 1 of 1 page

Step 4 : Note the Device Specifications in a Notepad

The screenshot shows a Notepad application with a dark theme. The left pane contains a list of notes, and the right pane shows the content of a selected note. The note contains the following information:

Organization ID : 95kfev
12:05 AM Device Type : NodeMCU
Quick Notes

Device Credentials
24/10/22 You registered your device...
Notes

import turtle
14/02/22 import os
Notes

Stacks Using Python
14/02/22 Let's have a class...
Notes

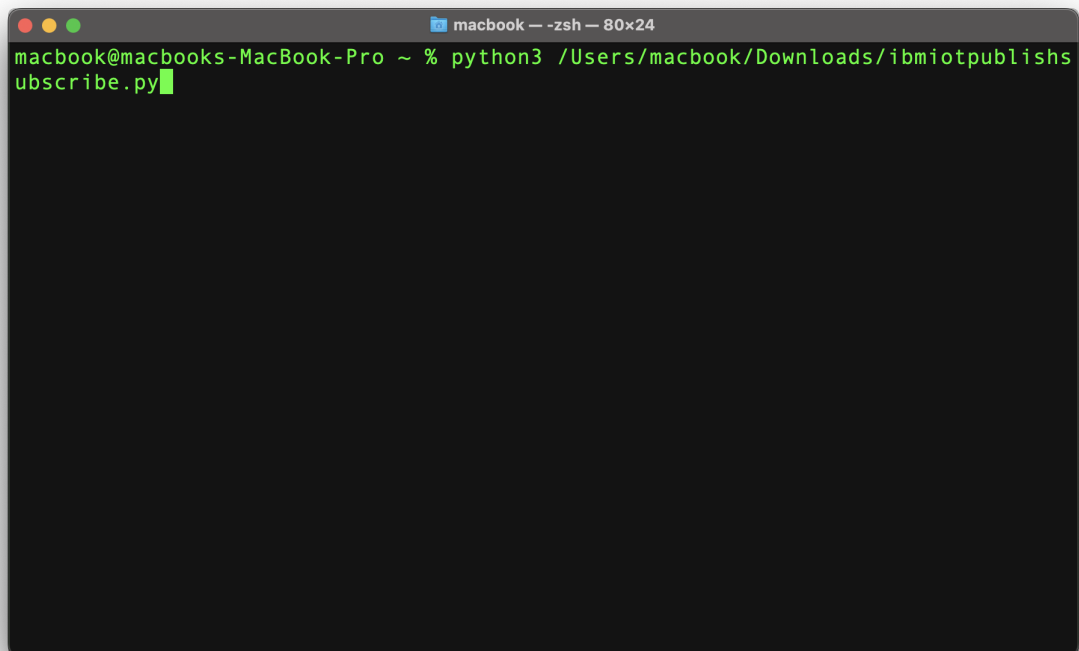
Linked List Using Python:
26/01/22 To link the nodes and create...
Notes

The right pane shows the following text:

15 November 2022

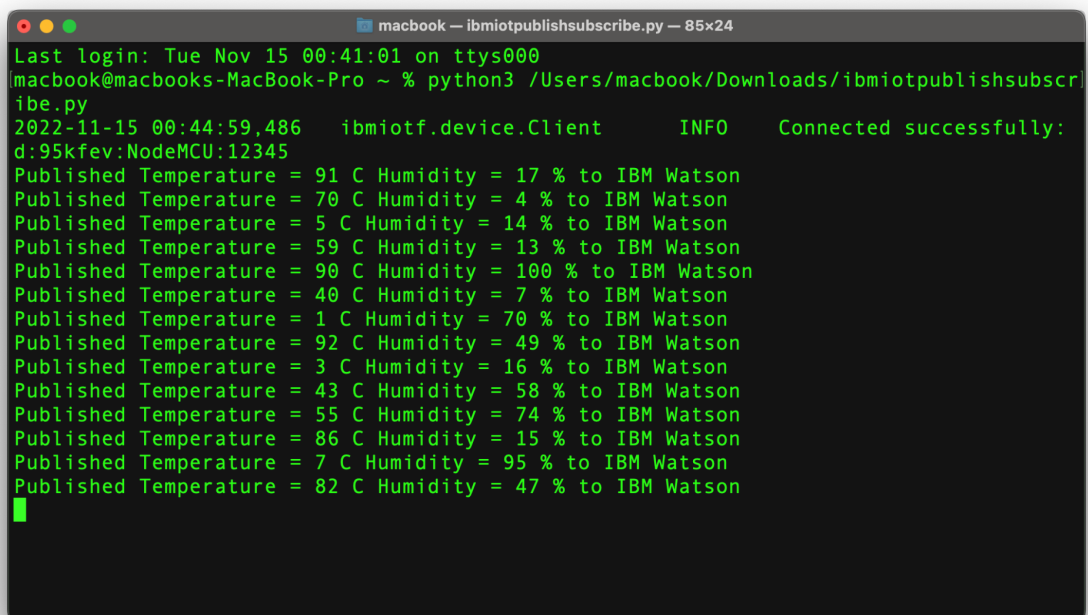
Organization ID : 95kfev
Device Type : NodeMCU
Device ID : 12345
Authentication Method : use-token-auth
Authentication Token : 12345678

Step 5 : Run the python script



```
macbook — zsh — 80x24
macbook@macbooks-MacBook-Pro ~ % python3 /Users/macbook/Downloads/ibmiotpublishsubscribe.py
```

Step 6 : View the output in Terminal/ Command Prompt



```
macbook — ibmiotpublishsubscribe.py — 85x24
Last login: Tue Nov 15 00:41:01 on ttys000
macbook@macbooks-MacBook-Pro ~ % python3 /Users/macbook/Downloads/ibmiotpublishsubscribe.py
2022-11-15 00:44:59.486 ibmiotf.device.Client INFO Connected successfully:
d:95kfev:NodeMCU:12345
Published Temperature = 91 C Humidity = 17 % to IBM Watson
Published Temperature = 70 C Humidity = 4 % to IBM Watson
Published Temperature = 5 C Humidity = 14 % to IBM Watson
Published Temperature = 59 C Humidity = 13 % to IBM Watson
Published Temperature = 90 C Humidity = 100 % to IBM Watson
Published Temperature = 40 C Humidity = 7 % to IBM Watson
Published Temperature = 1 C Humidity = 70 % to IBM Watson
Published Temperature = 92 C Humidity = 49 % to IBM Watson
Published Temperature = 3 C Humidity = 16 % to IBM Watson
Published Temperature = 43 C Humidity = 58 % to IBM Watson
Published Temperature = 55 C Humidity = 74 % to IBM Watson
Published Temperature = 86 C Humidity = 15 % to IBM Watson
Published Temperature = 7 C Humidity = 95 % to IBM Watson
Published Temperature = 82 C Humidity = 47 % to IBM Watson
```

Step 7 : View the Random Data Values in Watson IBM Platform

The screenshot shows the IBM Watson IoT Platform interface. The top navigation bar includes 'Browse', 'Action', 'Device Types', and 'Interfaces'. A search bar is present with the text 'Search by Device ID'. The main content area displays a table of devices. The first device, ID 12345, is highlighted. Below the device list, a section titled 'Recent Events' shows a stream of data from an IoT sensor. The events are listed in a table with columns for Event, Value, Format, and Last Received. The values are JSON objects containing temperature and humidity data. A status box at the bottom right indicates '1 Simulation running'.

Event	Value	Format	Last Received
IoTSensor	{"temp":5,"Humid":20}	json	a few seconds ago
IoTSensor	{"temp":99,"Humid":33}	json	a few seconds ago
IoTSensor	{"temp":85,"Humid":46}	json	a few seconds ago
IoTSensor	{"temp":84,"Humid":77}	json	a few seconds ago
IoTSensor	{"temp":9,"Humid":5}	json	a few seconds ago

Step 8 : Create a board in Watson IOT platform to Visualise the Data

The screenshot shows the IBM Watson IoT Platform interface for managing boards. The top navigation bar includes 'Your boards' and 'Public boards'. A 'Create New Board' button is visible. The main content area displays a grid of boards. The first board, titled 'RANDOM SENSOR DATA VISUALIZER', is highlighted with a red circle and contains one card. The other two boards, 'USAGE OVERVIEW' and 'RISK AND SECURITY OVERVIEW', contain three and four cards respectively. A large plus sign icon is visible in the bottom right corner, indicating the option to add a new board.

Step 9 : Create a Card inside the Newly Created Board for Visualising in a Graphical Way

