PYTHON SCRIPT
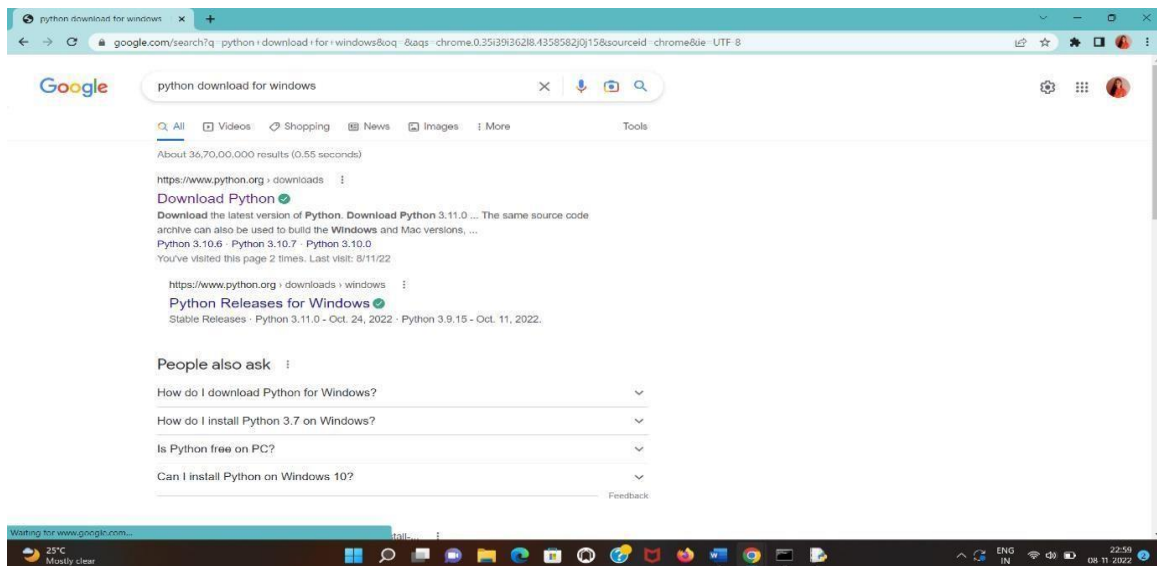
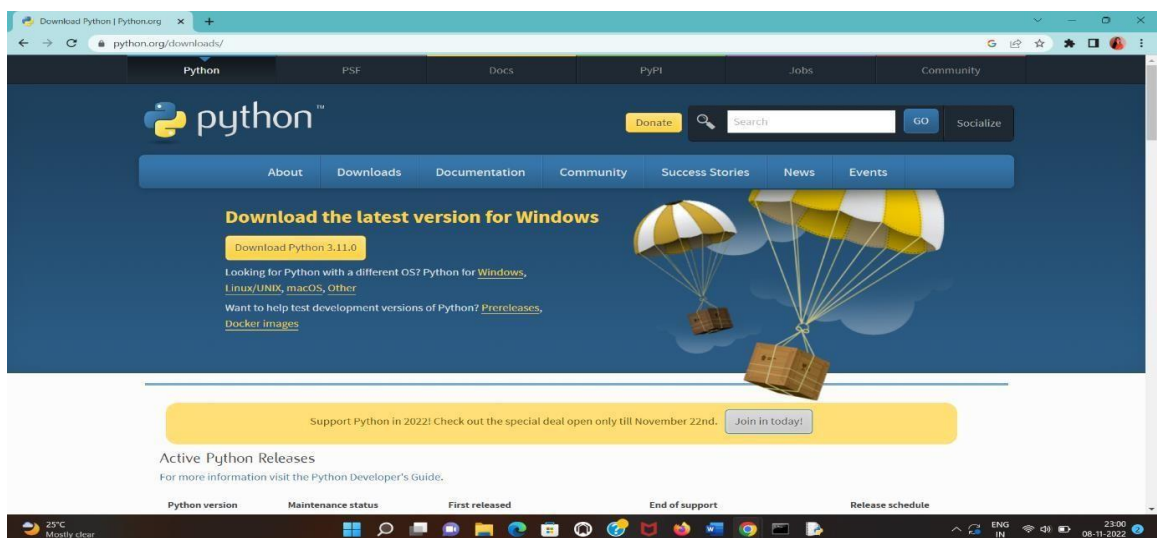| Assignment Date	06 NOVEMBER2022
Team ID	PNT2022TMID32740
Project Name	Gas Leakage Monitoring and Alerting System

AIM:

To install python version 3.9.6 and IBM Watson IoT platform packages in python.

STEPS:

1. Search for python for windows in Google search engine.

2. Click the First link..

3. Search for python 3.9.6 version which is suitable for IBM Watson platform.
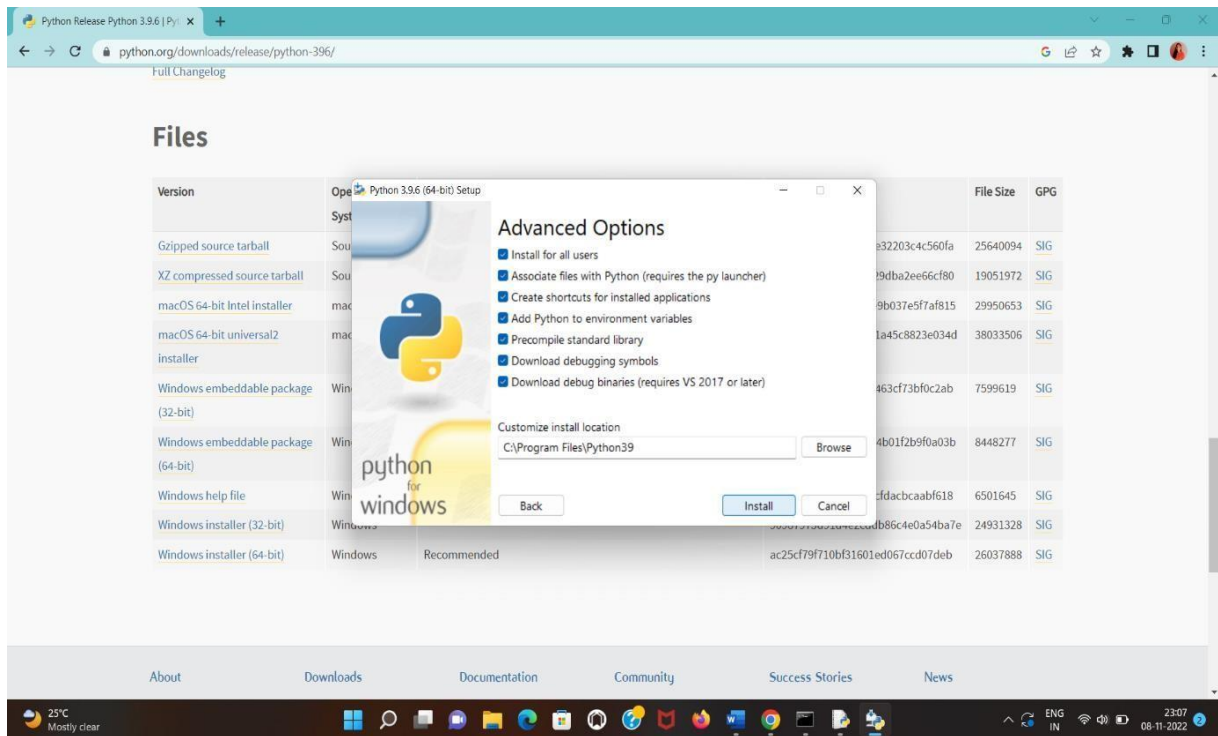

4. Click the python version 3.9.6 and download.

5. Install python 3.9.6 (64-bit) then click customize.
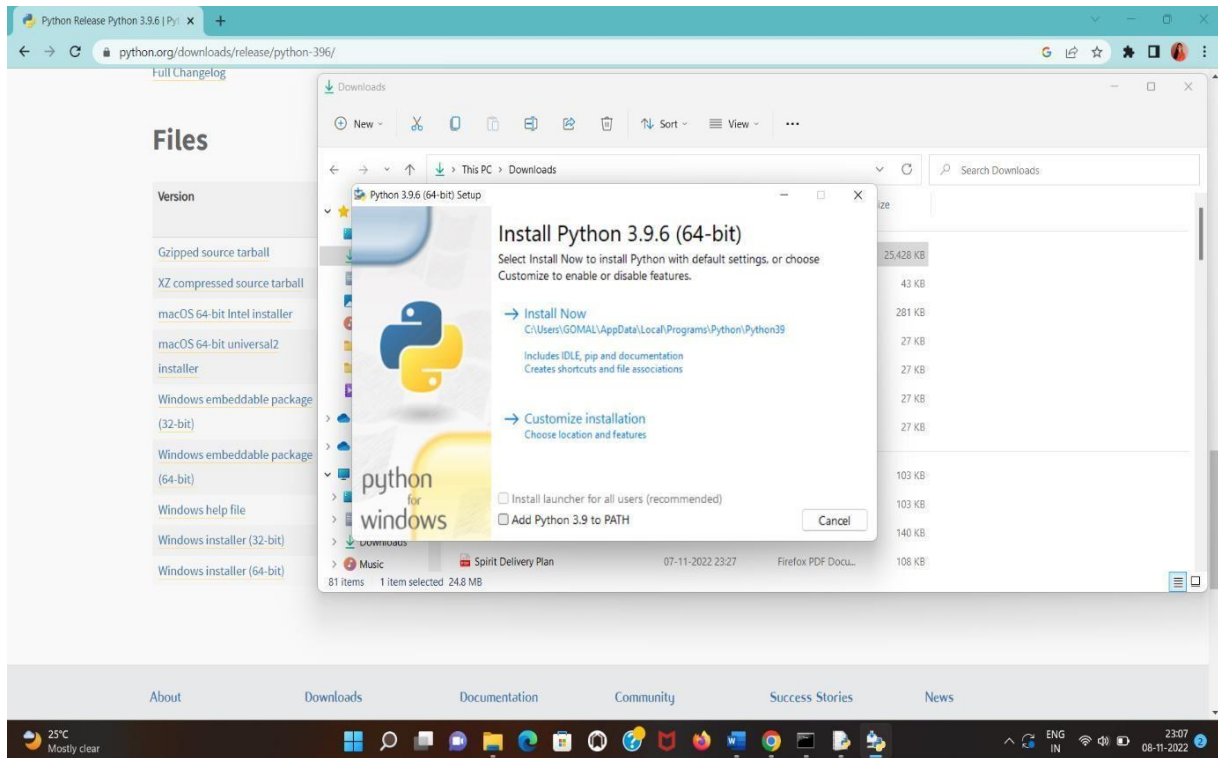
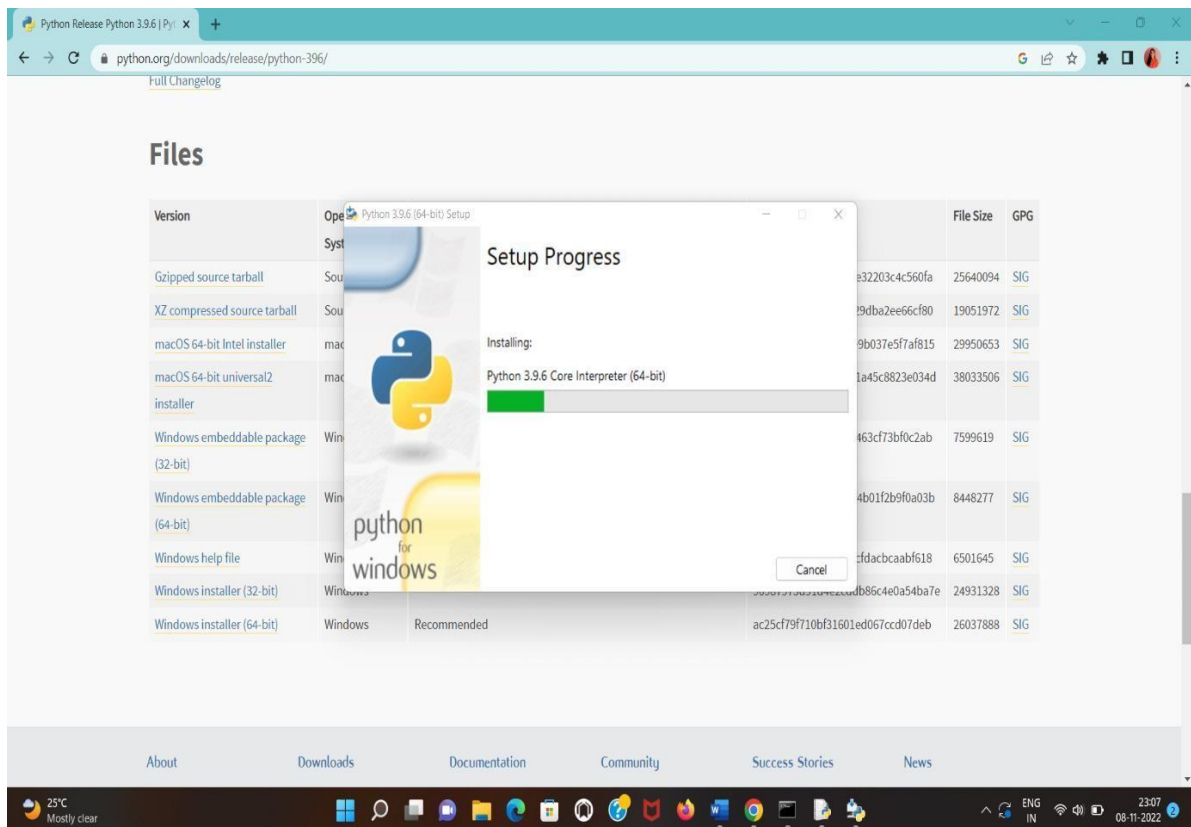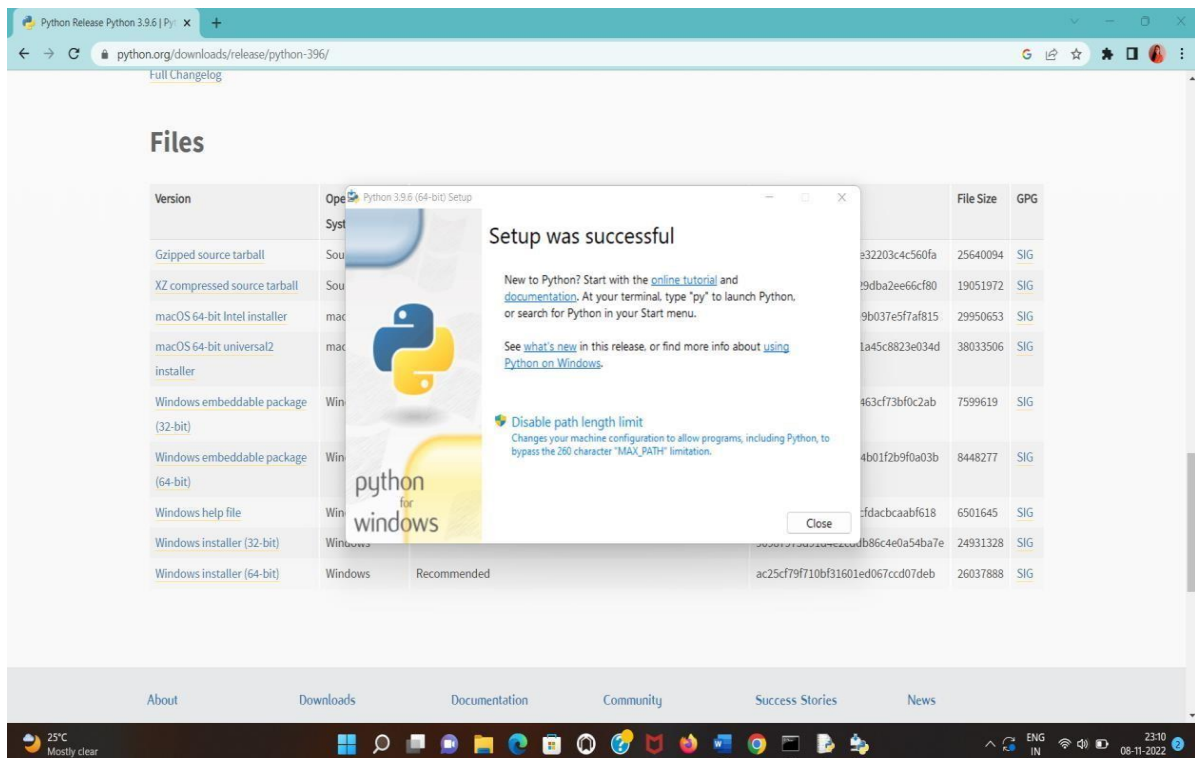
6.Click the next option.
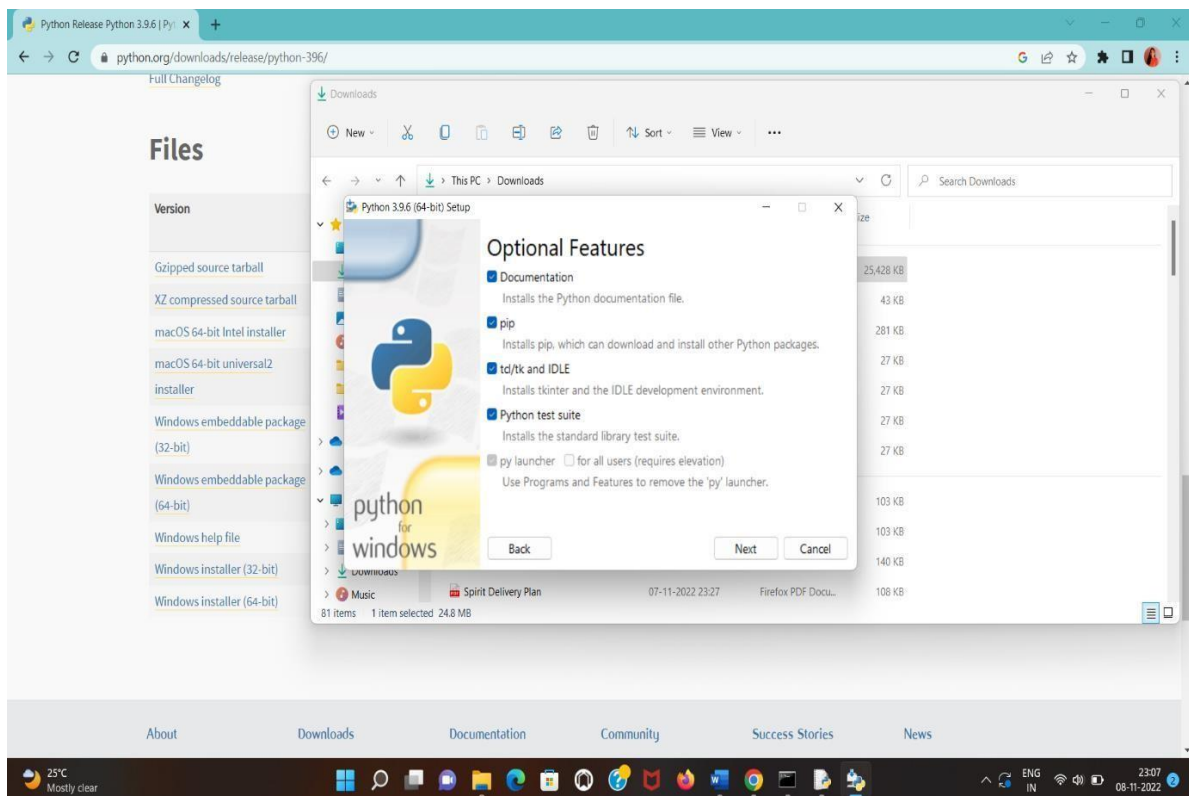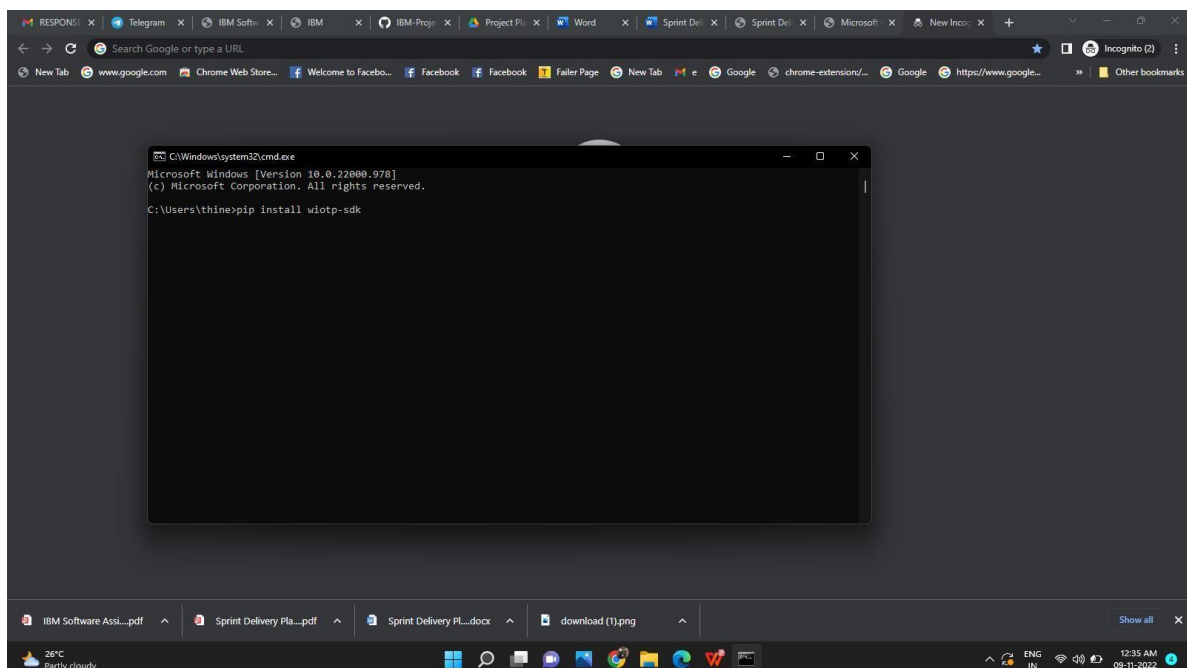
7.Select all the options and install the app.

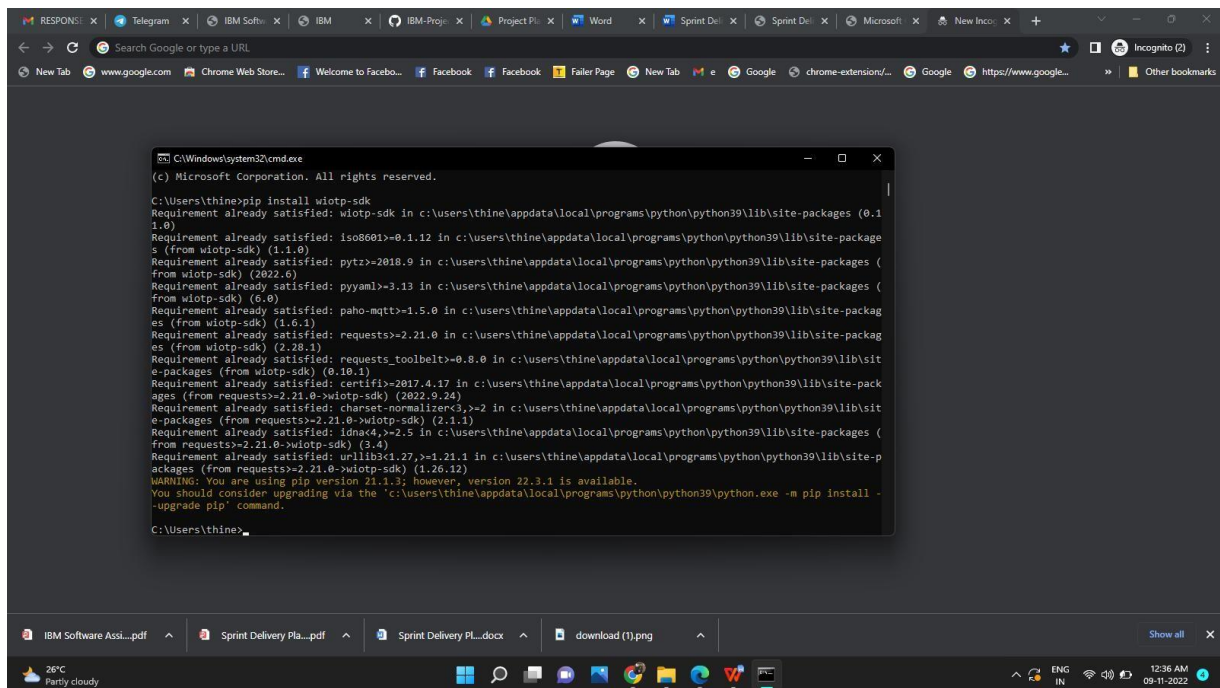8.Set up process will begin after you give install.

9. Python 3.9.6 version is installed successfully.

10.   Now install the IBM Watson Platform package through command prompt.
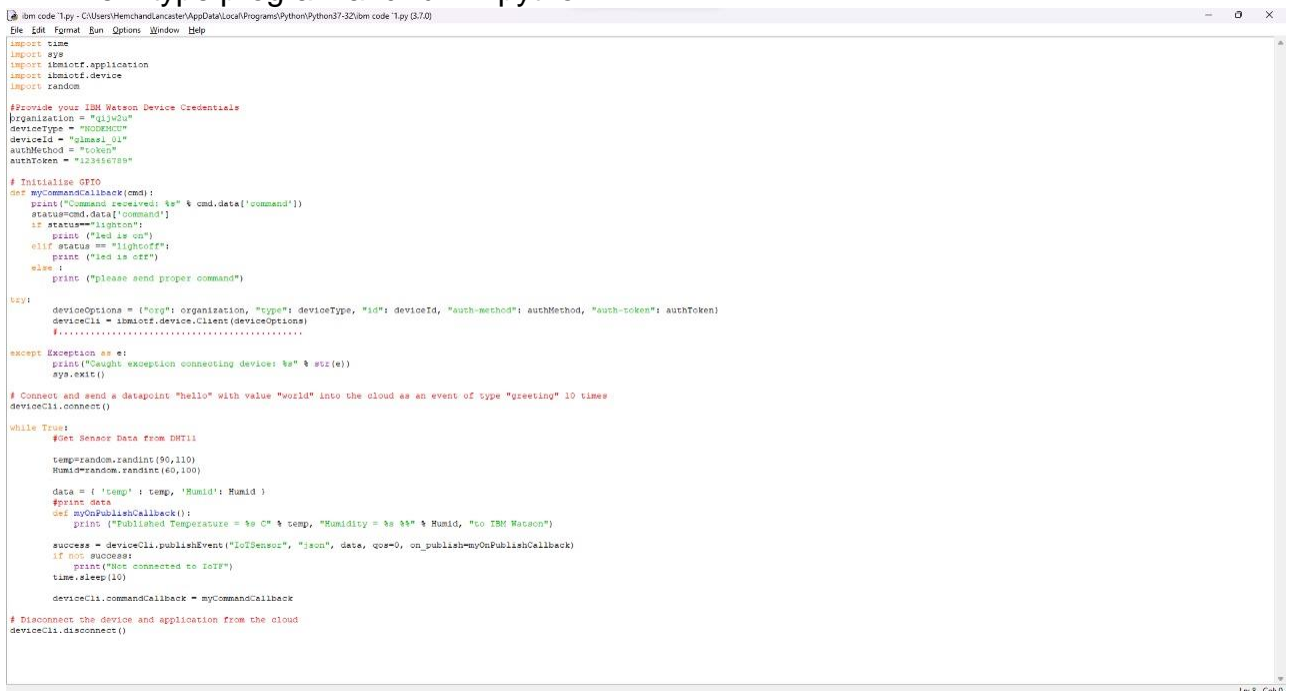
11. The package will be installed.

12.Then type program and run in python.

RESULT:

The python version 3.9.6 and IBM Watson IoT platform package are installed successfully.

Python Code:

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

#Provide your IBM Watson Device Credentials
organization = "qijw2u"
deviceType = "NODEMCU"
deviceId = "glmas1_01"
authMethod = "token"
authToken = "123456789"

# Initialize GPIO
def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status=="lighton":
        print ("led is on")
    elif status == "lightoff":
        print ("led is off")
    else :
        print ("please send proper command")

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(90,110)
    Humid=random.randint(60,100)

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

    success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0,
on_publish=myOnPublishCallback)
    if not success:
        print("Not connected to IoTF")
    time.sleep(10)
```

deviceCli.commandCallback = myCommandCallback

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