

## Sprint-4

TEAM ID	PNT2022TMID34531
PROJECT NAME	<b>IoT Based Smart Crop Protection System for Agriculture</b>

### Code in the Python IDLE:

```
python modified project code.py - C:/Users/HP/AppData/Local/Programs/Python/Python37/python modified project code.py (3.7.0)
File Edit Format Run Options Window Help
import cv2
import numpy as np
import wiotp.sdk.device
import playsound
import random
import time
import datetime
import ibm_boto3
from ibm_botocore.client import Config, ClientError
#Cloudant DB
from cloudant.client import Cloudant
from cloudant.error import CloudantException
from cloudant.result import Result, ResultByKey
from clarifai_grpc.channel.clarifai_channel import ClarifaiChannel
from clarifai_grpc.grpc.api import service_pb2_grpc
stub=service_pb2_grpc.V2Stub(ClarifaiChannel.get_grpc_channel())
from clarifai_grpc.grpc.api import service_pb2, resources_pb2
from clarifai_grpc.grpc.api.status import status_code_pb2
#This is how you authenticate
metadata = (('authorization', 'Key 3f24cd30d49f6f7ab5d79a5597356fc7'),)#clarifi service credential
COS_ENDPOINT = "https://s3.ap.cloud-object-storage.appdomain.cloud"
COS_API_KEY_ID = "uk89t2Ead9kwv4PKtIvp07UdN57Z8zF095U2_JsMstTv"
COS_AUTH_ENDPOINT = "https://iam.cloud.ibm.com/identity/token"
COS_RESOURCE_CRN = "crn:vl:bluemix:public:cloud-object-storage:global:a/c3c1d2d11b42464d9c706f832c28b007:380fafe0-66e6-44f2-9abb-e989ae484b63::"
clientdb=Cloudant("apikey-v2-17ohzcgg9s8gag385pikw8c0rvg66cm8u9rdhnbj4pb","d2c0c50d290716c4f3a6c163754fd4d2",url= "https://apikey-v2-17ohzcgg9s8gag385pikw8c0rvg66cm8u9rdhnbj4pb")
clientdb.connect()
#create resource
cos = ibm_boto3.resource("s3",
    ibm_api_key_id=COS_API_KEY_ID,
    ibm_service_instance_id=COS_RESOURCE_CRN,
    ibm_auth_endpoint=COS_AUTH_ENDPOINT,
    config=Config(signature_version="oauth"),
    endpoint_url=COS_ENDPOINT
)
def multi_part_upload(bucket_name, item_name, file_path):
    try:
        print("Starting file transfer for {} to bucket:{}".format(item_name, bucket_name))
        part_size = 1024 * 1024 * 5
        file_threshold = 1024 * 1024 * 15
        transfer_config=ibm_boto3.s3.transfer.TransferConfig(
            multipart_threshold=file_threshold,
            multipart_chunksize=part_size
        )
        with open(file_path, "rb") as file_data:
            cos.Object(bucket_name, item_name).upload_fileobj(
                Fileobj=file_data
            )
    except ClientError as err:
```

```
python modified project code.py - C:/Users/HP/AppData/Local/Programs/Python/Python37/python modified project code.py (3.7.0)
File Edit Format Run Options Window Help

    multipart_threshold=file_threshold,
    multipart_chunksize=part_size
)
    with open(file_path, "rb") as file_data:
        cos.Object(bucket_name, item_name).upload_fileobj(
            Fileobj=file_data,
            Config=transfer_config
        )
    print ("Transfer for {} Complete!\n".format(item_name))
except ClientError as be:
    print ("CLIENT ERROR: {}".format(be))
except Exception as e:
    print("Unable to complete multi-part upload: {}".format(e))

def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data)
    command=cmd.data['command']
    print(command)
    if(command=='lighton'):
        print('lighton')
    elif(command=='lightoff'):
        print('lightoff')
    elif(command=='motoron'):
        print('motoron')
    elif(command=='motoroff'):
        print('motoroff')

myConfig = {
    "identity": {
        "orgId": "kc06ni",
        "typeId": "abcd",
        "deviceId": "123"
    },
    "auth": {
        "token": "12345678"
    }
}

client= wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
client.connect()

database_name="sample1"
my_database = clientdb.create_database(database_name)
if my_database.exists():
    print(f'"{database_name}" successfully created.')
    cap=cv2.VideoCapture("garden.mp4")
if (cap.isOpened()==True):
    print('File opened')
    print('File opened')
```

Ln: 102 Col: 55

28°C Cloudy

```
*python modified project code.py - C:/Users/HP/AppData/Local/Programs/Python/Python37/python modified project code.py (3.7.0)*
File Edit Format Run Options Window Help

}
client= wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
client.connect()

database_name="sample1"
my_database = clientdb.create_database(database_name)
if my_database.exists():
    print(f'"{database_name}" successfully created.')
    cap=cv2.VideoCapture("garden.mp4")
if (cap.isOpened()==True):
    print('File opened')
else:
    print ('File not found')

while (cap.isOpened()):
    ret, frame = cap.read()
    gray =cv2.cvtColor(frame ,cv2.COLOR_BGR2GRAY)
    imS=cv2.resize(frame,(960,540))
    cv2.imwrite('ex.jpg',imS)
    with open("ex.jpg","rb") as f:
        file_bytes = f.read()
    request = service_pb2.PostModelOutputsRequest(
        model_id='aaa03c23b3724a16a56b629203edc62c',#"e2eb7c86020f842148cd36bb3a11086a"
        inputs=[resources_pb2.Input(data=resources_pb2.Data(image=resources_pb2.Image(base64=file_bytes))
    ))
    response = stub.PostModelOutputs(request ,metadata=metadata)
    if response.status_code != status_code_pb2.SUCCESS:
        raise Exception("Request failed,status code :"+ str (response.status_code))
    detect=False
    for concept in response.outputs[0].data.concepts:
        if (concept.value>0.98):
            if (concept.name=="animal"):
                print("Alert! Alert! animal detected")
                playsound.playsound('alert.mp3')
                picname=datetime.datetime.now().strftime("%y-%m-%d-%H-%M")
                cv2.imwrite(picname+'.jpg',frame)
                multi_part_upload('adalin',picname+'.jpg',picname+'.jpg')
                json_document={"link":COS_ENDPOINT+'/'+adalin+'/'+picname+'.jpg'}
                new_document = my_database.create_document(json_document)
                if new_document.exists():
                    print (f"Document successfully created.")
                    time.sleep(5)
                detect =True
    moist=random.randint(0,100)
    temp=random.randint(0,100)
    humidity=random.randint(0,100)
```

Ln: 122 Col: 50

28°C Humid

```
*python modified project code.py - C:/Users/HP/AppData/Local/Programs/Python/Python37/python modified project code.py (3.7.0)
File Edit Format Run Options Window Help

while(cap.isOpened()):
    ret, frame = cap.read()
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    imS=cv2.resize(frame, (960,540))
    cv2.imwrite('ex.jpg',imS)
    with open("ex.jpg","rb") as f:
        file_bytes = f.read()
    request = service_pb2.PostModelOutputsRequest(
        model_id='aaa03c23b3724a16a56b629203edc62c',#"e2eb7c86020f842148cd36bb3a11086a"
        inputs=[resources_pb2.Input(data=resources_pb2.Data(image=resources_pb2.Image(base64=file_bytes))
    ))
    response = stub.PostModelOutputs(request,metadata=metadata)
    if response.status_code != status_code_pb2.SUCCESS:
        raise Exception("Request failed,status code :"+ str (response.status_code))
    detect=False
    for concept in response.outputs[0].data.concepts:
        if (concept.value>0.98):
            if (concept.name=="animal"):
                print("Alert! Alert! animal detected")
                playsound.playsound('alert.mp3')
                picname=datetime.datetime.now().strftime("%y-%m-%d-%H-%M")
                cv2.imwrite(picname+'.jpg',frame)
                multi_part_upload('adalin',picname+'.jpg',picname+'.jpg')
                json_document={"link":COS_ENDPOINT+'/'+'adalin'+'/'+picname+'.jpg'}
                new_document = my_database.create_document(json_document)
                if new_document.exists():
                    print ("Document successfully created.")
                    time.sleep(5)
                    detect =True
    moist=random.randint(0,100)
    temp=random.randint(0,100)
    humidity =random.randint(0,100)
    myData={'Animal': detect,'temperature': temp,'moisture':moist,'humidity':humidity}
    print(myData)
    if (humidity!=None):
        client.publishEvent(eventId="status",msgFormat="json", data=myData, qos=0, onPublish=None)
        print("Publish Ok..")
    client.commandCallback = myCommandCallback
    cv2.imshow('frame',imS)
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break
client.disconnect()
cap.release()
cv2.destroyAllWindows()

Ln: 122 Col: 50
```

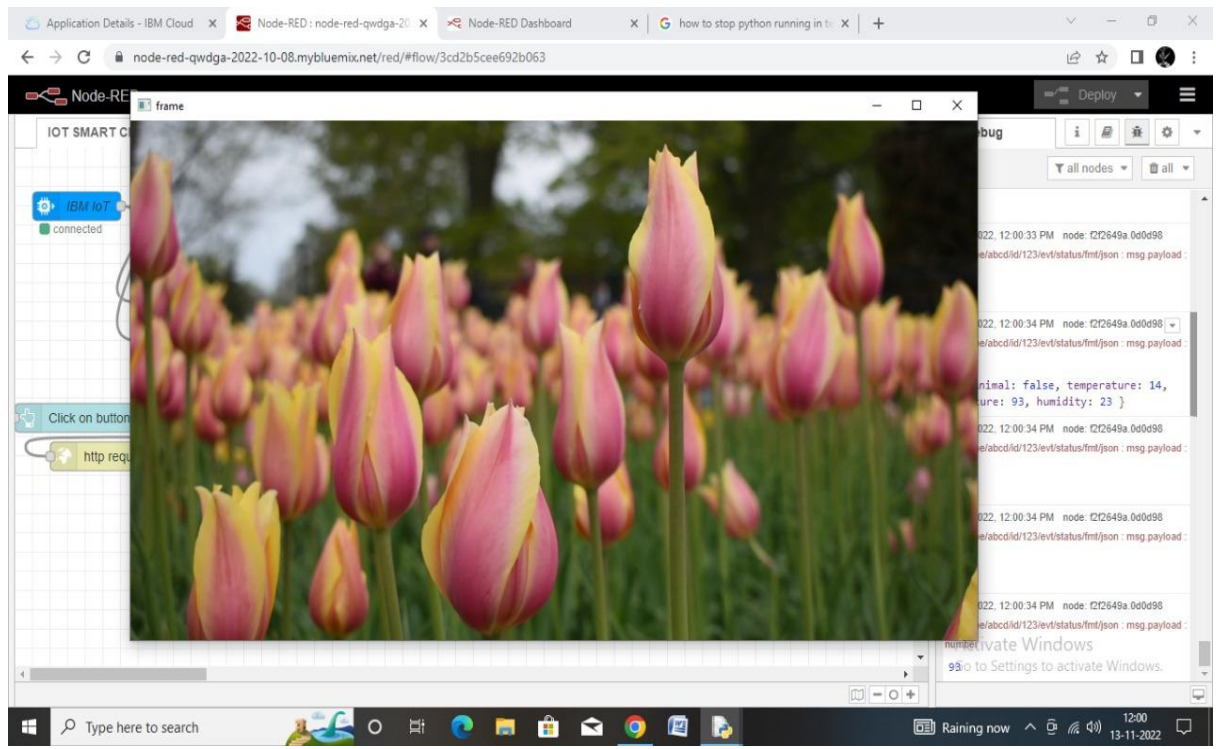
## Output from python:

- When Animal is not detected:

```
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 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\python\Python37\python modified project code.py =====
2022-11-13 12:00:04,390 wiotp.sdk.device.client.DeviceClient INFO Connected successfully: d:kc06
ni:abcd:123
'sample1' successfully created.
File opened
{'Animal': False, 'temperature': 29, 'moisture': 63, 'humidity': 21}
Publish Ok..
{'Animal': False, 'temperature': 80, 'moisture': 24, 'humidity': 9}
Publish Ok..
{'Animal': False, 'temperature': 45, 'moisture': 100, 'humidity': 13}
Publish Ok..
{'Animal': False, 'temperature': 3, 'moisture': 97, 'humidity': 38}
Publish Ok..
{'Animal': False, 'temperature': 5, 'moisture': 80, 'humidity': 4}
Publish Ok..
{'Animal': False, 'temperature': 75, 'moisture': 7, 'humidity': 38}
Publish Ok..
{'Animal': False, 'temperature': 48, 'moisture': 56, 'humidity': 4}
Publish Ok..
{'Animal': False, 'temperature': 86, 'moisture': 15, 'humidity': 46}
Publish Ok..
{'Animal': False, 'temperature': 10, 'moisture': 26, 'humidity': 51}
Publish Ok..
{'Animal': False, 'temperature': 3, 'moisture': 69, 'humidity': 27}
Publish Ok..
{'Animal': False, 'temperature': 19, 'moisture': 46, 'humidity': 85}

Activate Windows
Go to Settings to activate Windows.

Ln: 43 Col: 4
```



- When Animal is Detected:

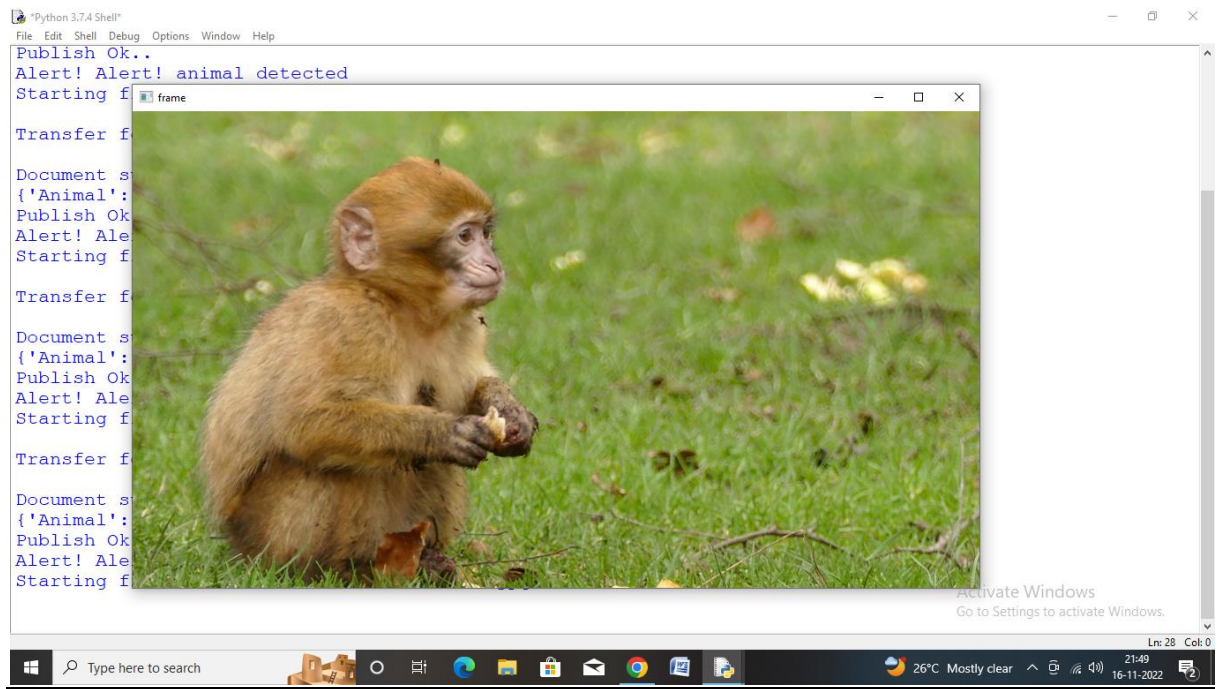
```
Python 3.7.4 Shell
File Edit Shell Debug Options Window Help
Python 3.7.4 (tags/v3.7.4:e09359112e, Jul 8 2019, 20:34:20) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\python\Python37\project modified python code.py =====
2022-11-16 21:48:03,429 wiotp.sdk.device.client.DeviceClient INFO Connected successfully: d:kc06
ni:abcd:123
'sample1' successfully created.
File opened
{'Animal': False, 'temperature': 97, 'moisture': 26, 'humidity': 98}
Publish Ok..
Alert! Alert! animal detected
Starting file transfer for 22-11-16-21-48.jpg to bucket:adalin

Transfer for 22-11-16-21-48.jpg Complete!

Document successfully created.
{'Animal': True, 'temperature': 38, 'moisture': 48, 'humidity': 62}
Publish Ok..
Alert! Alert! animal detected
Starting file transfer for 22-11-16-21-48.jpg to bucket:adalin

|

Activate Windows
Go to Settings to activate Windows.
Ln: 16 Col: 0
```



## Output from IoT Watson:

A screenshot of the IBM Watson IoT Platform web interface. The browser tabs include 'Service Details - IBM', 'IBM Watson IoT Platform', '(1) WhatsApp', 'wiotp.sdk.device.client', 'Application Details', and 'Node-RED: node-red'. The address bar shows the URL 'kc06ni.internetofthings.ibmcloud.com/dashboard/devices/drilldown/abcd:123?returnTo=/devices/browse'. The page title is 'Device Drilldown - 123'. On the left, a sidebar contains navigation links: 'Back', 'Connection Information', 'Recent Events' (selected), 'State', 'Device Information', 'Metadata', 'Diagnostics', 'Connection Logs', and 'Device Actions'. The main content area is divided into two sections. The 'Recent Events' section has a subtitle 'The recent events listed show the live stream of data that is coming and going from this device.' and contains a table with three rows of event data. The 'State' section has a subtitle 'This table shows a list of data points that are reported by this device.' and currently displays 'Showing Raw Data | No Interfaces Available'. The top right of the interface shows a user profile for '961819106003@smartintranz.com' with ID 'kc06ni'. The Windows taskbar at the bottom shows 'Type here to search', application icons, and system tray information including '26°C Cloudy', '19:56', and '14-11-2022'. An 'Activate Windows' watermark is visible in the bottom right corner.



IBM Watson IoT Platform

Device Drilldown - 123

Recent Events

The recent events listed show the live stream of data that is coming and going from this device.

Event	Value	Format	Last Received
status	{"Animal":false,"temperature":33,"moisture":87,"..."}	json	a few seconds ago
status	{"Animal":false,"temperature":75,"moisture":89,"..."}	json	a few seconds ago
status	{"Animal":false,"temperature":92,"moisture":54,"..."}	json	a few seconds ago
status	{"Animal":false,"temperature":87,"moisture":37,"..."}	json	a few seconds ago
status	{"Animal":false,"temperature":4,"moisture":8,"hu..."}	json	a few seconds ago

Activate Windows  
Go to Settings to activate Windows.

## Output from Node-RED:

Node-RED

IOT SMART CROP PROTECT

Flow 1

debug

11/13/2022, 12:11:29 PM node: f2f2649a-0d0d98  
iot-2/type/abcd/123/evt/status/fmt/json : msg payload : number

21

11/13/2022, 12:11:34 PM node: f2f2649a-0d0d98  
iot-2/type/abcd/123/evt/status/fmt/json : msg payload : Object

{ Animal: true, temperature: 64, moisture: 66, humidity: 16 }

11/13/2022, 12:11:34 PM node: f2f2649a-0d0d98  
iot-2/type/abcd/123/evt/status/fmt/json : msg payload : number

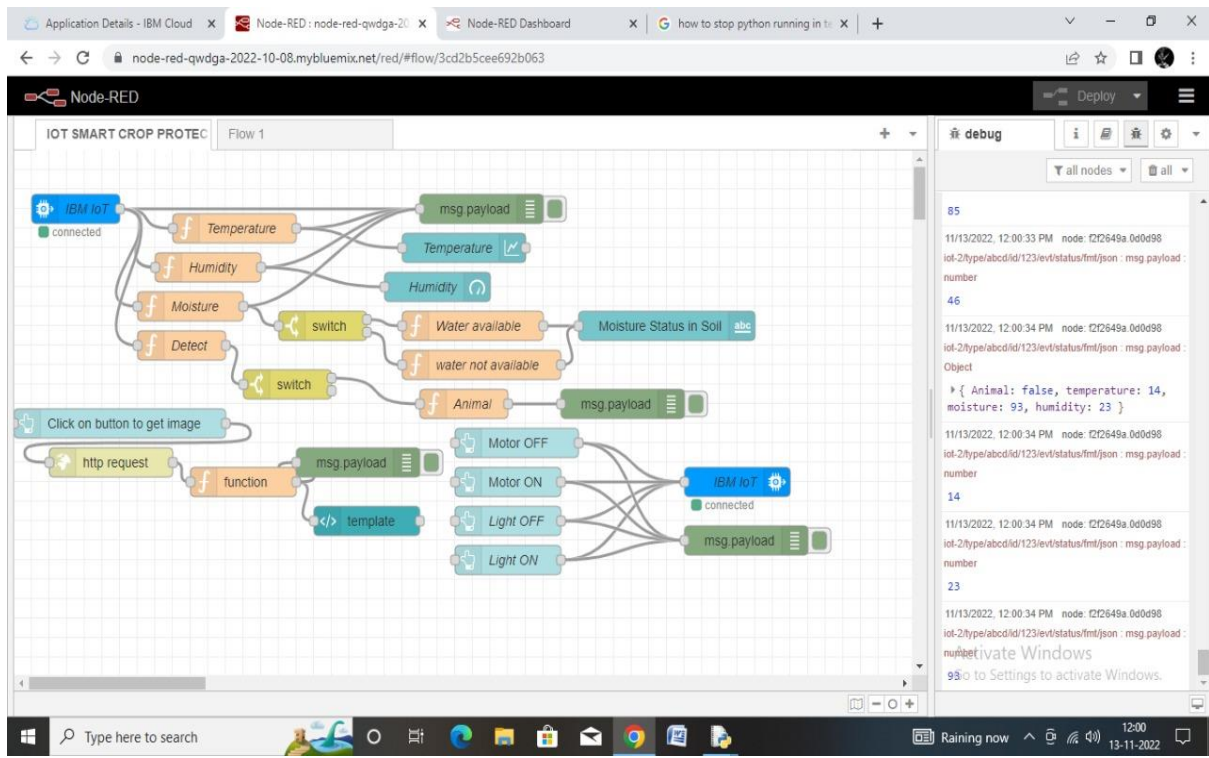
64

11/13/2022, 12:11:35 PM node: f2f2649a-0d0d98  
iot-2/type/abcd/123/evt/status/fmt/json : msg payload : number

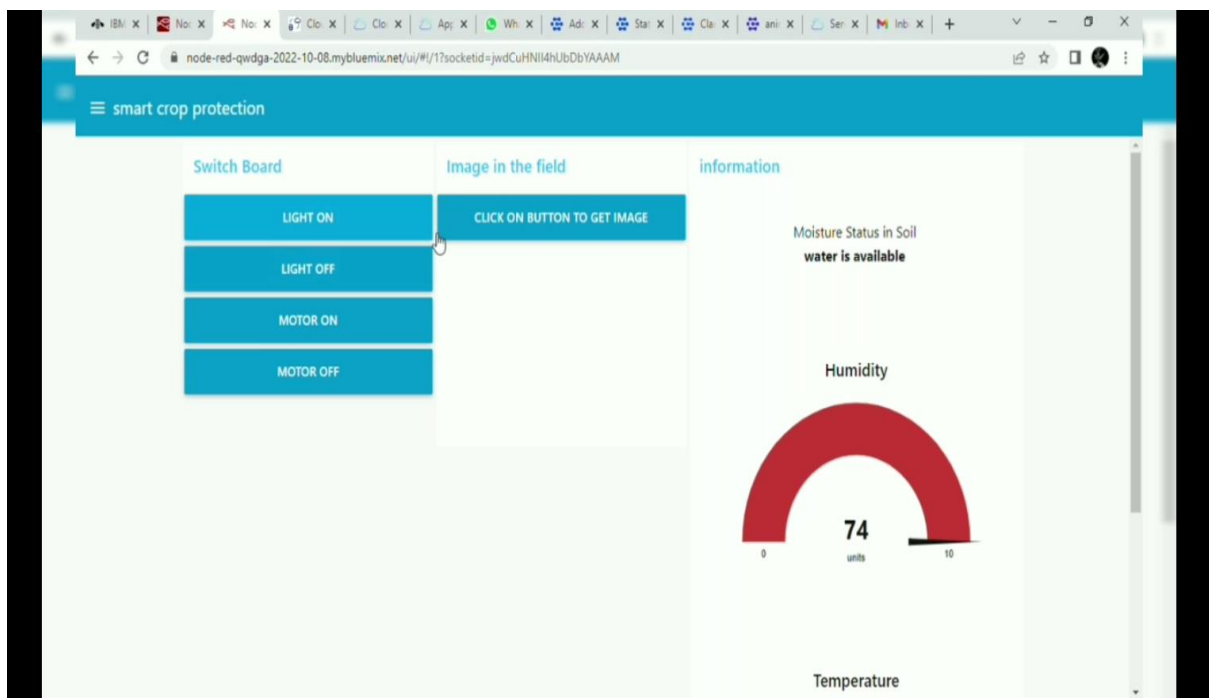
16

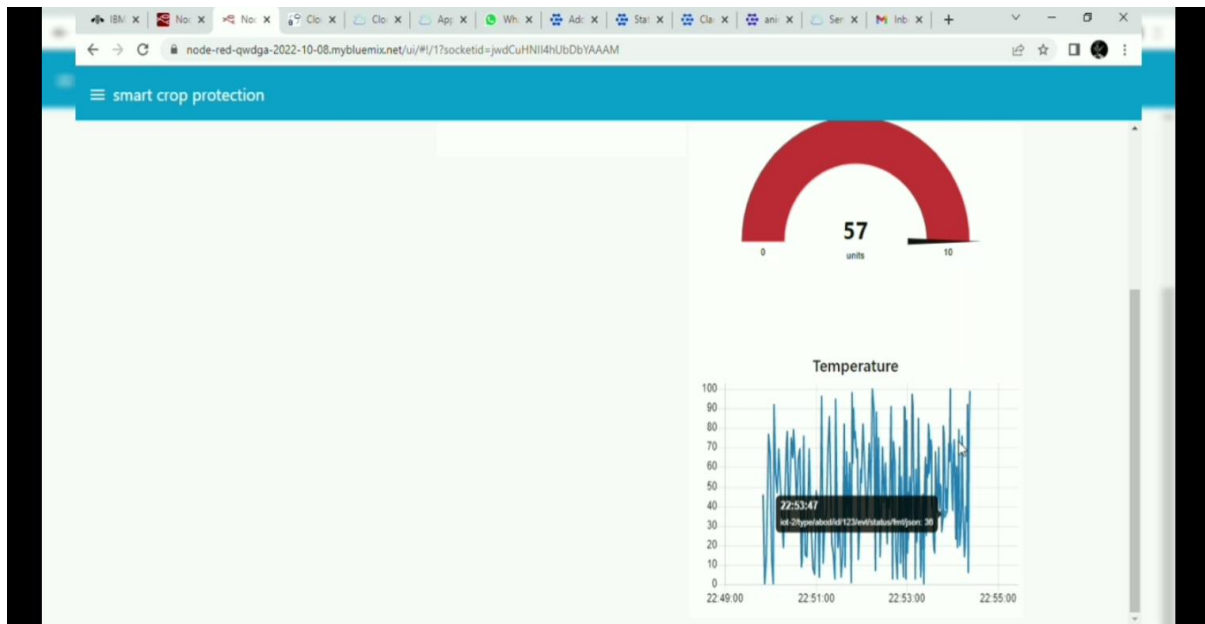
11/13/2022, 12:11:36 PM node: f2f2649a-0d0d98  
iot-2/type/abcd/123/evt/status/fmt/json : msg payload : number

66



## Output from Node-RED Dashboard:





## Output for Commands from Node-RED to Python:

```
Command received: {'command': 'lightoff'}
lightoff
lightoff
{'Animal': True, 'temperature': 45, 'moisture': 48, 'humidity': 30}
Publish Ok..
Alert! Alert! animal detected
Starting file transfer for 22-11-15-22-27.jpg to bucket:adalin

Transfer for 22-11-15-22-27.jpg Complete!

Document successfully created.
{'Animal': True, 'temperature': 5, 'moisture': 48, 'humidity': 38}
Publish Ok..
Alert! Alert! animal detected
Starting file transfer for 22-11-15-22-28.jpg to bucket:adalin

Transfer for 22-11-15-22-28.jpg Complete!

Document successfully created.
Command received: {'command': 'motor on'}
motor on
Command received: {'command': 'motor on'}
motor on
{'Animal': True, 'temperature': 44, 'moisture': 52, 'humidity': 1}
Publish Ok..
Alert! Alert! animal detected
Starting file transfer for 22-11-15-22-28.jpg to bucket:adalin
```



```

Transfer for 22-11-15-22-27.jpg Complete!

Document successfully created.
{'Animal': True, 'temperature': 65, 'moisture': 74, 'humidity': 67}
Publish Ok..
Alert! Alert! animal detected
Starting file transfer for 22-11-15-22-27.jpg to bucket:adalin

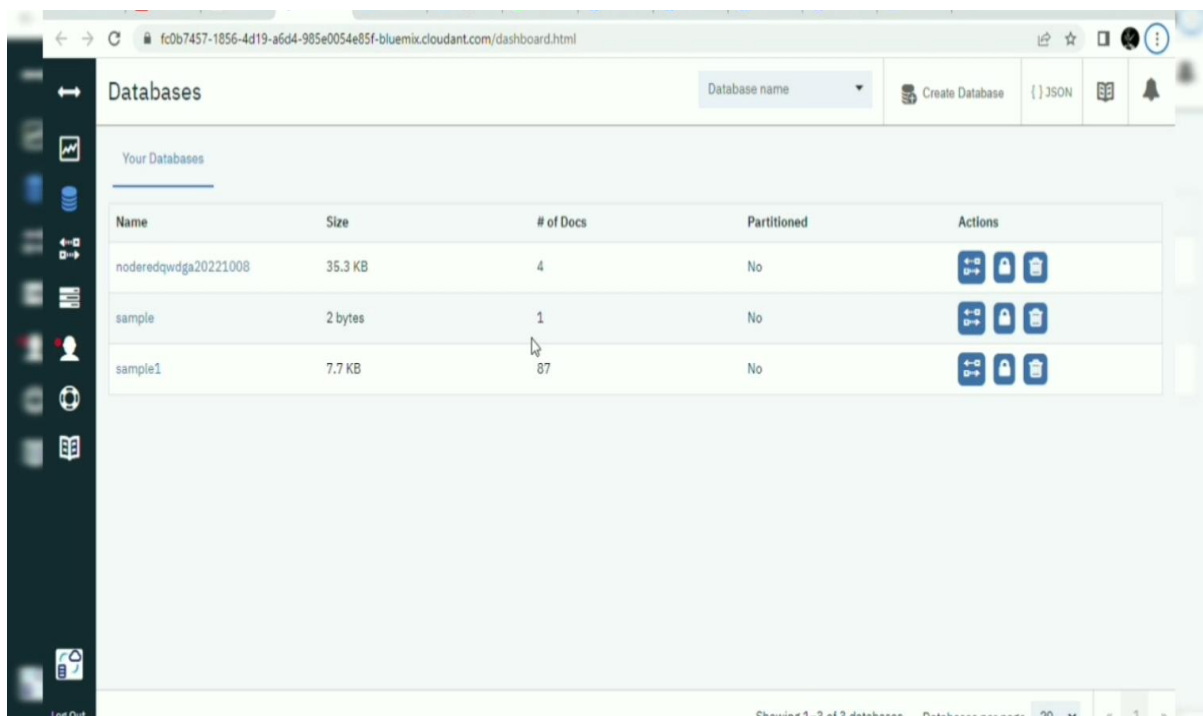
Command received: {'command': 'lighton'}
lighton
lighton
Transfer for 22-11-15-22-27.jpg Complete!

Command received: {'command': 'lighton'}
lighton
lighton
Command received: {'command': 'lightoff'}Document successfully created.

lightoff
lightoff
Command received: {'command': 'lightoff'}
lightoff
lightoff
{'Animal': True, 'temperature': 45, 'moisture': 48, 'humidity': 30}
Publish Ok..
Alert! Alert! animal detected
Starting file transfer for 22-11-15-22-27.jpg to bucket:adalin

```

## Output from Cloudant Database:



The screenshot shows the Cloudant Databases dashboard. At the top, there's a header with 'Databases' and a search bar. Below it, a table lists the databases. The table has columns for Name, Size, # of Docs, Partitioned, and Actions. There are three databases listed: 'noderedqwdga20221008', 'sample', and 'sample1'. Each database has a set of action icons (add, edit, delete, etc.).

Name	Size	# of Docs	Partitioned	Actions
noderedqwdga20221008	35.3 KB	4	No	[Add] [Edit] [Delete]
sample	2 bytes	1	No	[Add] [Edit] [Delete]
sample1	7.7 KB	87	No	[Add] [Edit] [Delete]

fc0b7457-1856-4d19-a6d4-985e0054e85f-bluemix.cloudant.com/dashboard.html#database/sample1/\_all\_docs

sample1

Document ID

Options

{ } JSON

Create Document

Table Metadata { } JSON

	id	key	value
<input type="checkbox"/>	014263adb47bf005e8c67df4589b0...	014263adb47bf005e8c67df4589b0...	{ "rev": "1-a46597adb787ae773fdbc...
<input type="checkbox"/>	014263adb47bf005e8c67df4589f98...	014263adb47bf005e8c67df4589f98...	{ "rev": "1-a46597adb787ae773fdbc...
<input type="checkbox"/>	09d623fdae63eed6273c7590fc227...	09d623fdae63eed6273c7590fc227...	{ "rev": "1-75521a01a8da352f7c098...
<input type="checkbox"/>	09d623fdae63eed6273c7590fc284c...	09d623fdae63eed6273c7590fc284c...	{ "rev": "1-75521a01a8da352f7c098...
<input type="checkbox"/>	102fa77c77f348342f30072e7cc042...	102fa77c77f348342f30072e7cc042...	{ "rev": "1-5e7849ffa09bee405afe44...
<input type="checkbox"/>	103e43a697c7e373514d2ad30feaf...	103e43a697c7e373514d2ad30feaf...	{ "rev": "1-1de3872cf13c96fa1bd62...
<input type="checkbox"/>	1672cc4d3da6862dc981e2752adc2...	1672cc4d3da6862dc981e2752adc2...	{ "rev": "1-f8d76894991ce64619818...
<input type="checkbox"/>	1b4c6578000bf6b73b72a25654f54...	1b4c6578000bf6b73b72a25654f54...	{ "rev": "1-b6e1ee0b729212cf5dd7b...
<input type="checkbox"/>	1ca9d53fb8e11e3d3100ff20538b3ee	1ca9d53fb8e11e3d3100ff20538b3ee	{ "rev": "1-abda89b250b9f60a97...
<input type="checkbox"/>	1ca9d53fb8e11e3d3100ff2054f5860	1ca9d53fb8e11e3d3100ff2054f5860	{ "rev": "1-d5a8af40f184a68ce9941...
<input type="checkbox"/>	1d89bb917bd5f946eddf51019d655...	1d89bb917bd5f946eddf51019d655...	{ "rev": "1-dfd049de9e405a2967186...

Showing document 1 - 20. Documents per page: 20

sample1 > 014263adb47bf005e8c67df4589b00f4

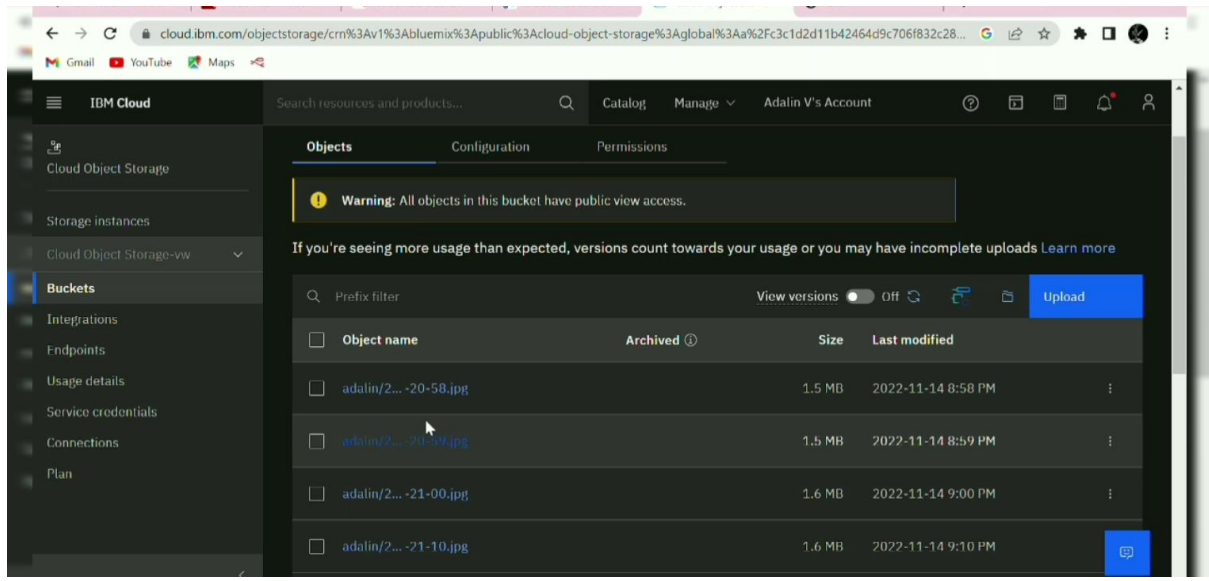
{ } JSON

Save Changes Cancel

Upload Attachment Clone Document Delete

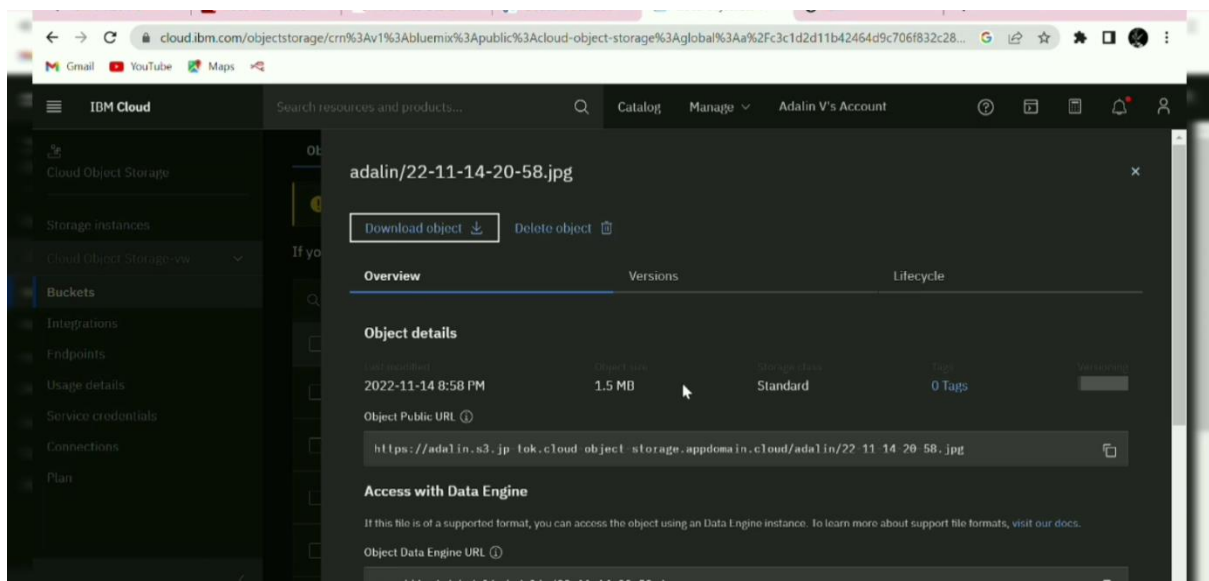
```
1 {
2   "_id": "014263adb47bf005e8c67df4589b00f4",
3   "_rev": "1-a46597adb787ae773fdbcba6e4ba8f30d",
4   "link": "https://control.cloud-object-storage.cloud.ibm.com/v2/endpoints/adalin/22-11-14-20-36.jpg"
5 }
```

## Output from Cloud Object Storage:



The screenshot shows the IBM Cloud Object Storage console. The left sidebar contains navigation links: Cloud Object Storage, Storage instances, Cloud Object Storage-vw, Buckets, Integrations, Endpoints, Usage details, Service credentials, Connections, and Plan. The main area is titled 'Objects' and includes a search bar, a 'View versions' toggle (set to Off), and an 'Upload' button. A warning message states: 'Warning: All objects in this bucket have public view access.' Below this, a table lists objects with columns for Object name, Archived status, Size, and Last modified. The objects listed are:

Object name	Archived	Size	Last modified
adalín/2...-20-58.jpg		1.5 MB	2022-11-14 8:58 PM
adalín/2...-20-59.jpg		1.5 MB	2022-11-14 8:59 PM
adalín/2...-21-00.jpg		1.6 MB	2022-11-14 9:00 PM
adalín/2...-21-10.jpg		1.6 MB	2022-11-14 9:10 PM



The screenshot shows the details view for the object 'adalín/22-11-14-20-58.jpg'. The main area is titled 'adalín/22-11-14-20-58.jpg' and includes a 'Download object' button and a 'Delete object' button. Below this, the 'Overview' tab is selected, showing the following details:

Object details	
Created	2022-11-14 8:58 PM
Object size	1.5 MB
Storage class	Standard
Tags	0 Tags

The 'Object Public URL' is displayed as: `https://adalín.s3.jp_tok.cloud-object-storage.appdomain.cloud/adalín/22-11-14-20-58.jpg`. Below this, the 'Access with Data Engine' section provides information on how to access the object using a Data Engine instance.