DEVELOP THE PYTHON SCRIPT

TEAM ID	**PNT2022TMID14113**
PROJECT TITLE	**IOT BASED SMART CROP PROTECTION FOR AGRICULTURE**

CODE:

```
import cv2

import numpy as np

Import wiotp.sdk.device

import playsound

import randon

import time

Expert datetime

import ibm _boto3

from ibm_botocore.client import config, ClientError

#CloudantDB

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","Keybc85e516574f43f42f6f6a2e9eb87'),)

COS ENDPOINT = "https://s3.jp-tok.cloud-object-storage.appdomain.cloud"
COS_API_KEY_ID = "rep-ct18m07990117XFAE7170cmFLLOQA25"
```

COS AUTH ENDPOINT = https://iam.cloud.ihm.com/identity/token

COS RESOURCE_CRN "ern:vi:bluemia:public:cloud-object-storage:global:/b64a3da97440b00c23ef23d6:199ab165-0d9d-420-842-98d86804368::”

clientdb Cloudant ("apikey-v2-163cpaghhfdi kvpasohverju5v5y3ubs", "bab1191453255bb78e7e2f0e1", url="https://apikey-v2-16aterpkohhaefikvpsschwerp5fvgube:bab115

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 (0) to bucket: (1)\n".format(item_name,

bucket_name))

#set 5 MB chunks

Part_size=1024*1024*5

#set threadhold to 15 MB

File_threshold=1024*1024*15

#set the transfer threshold and chunk size

Transfer_config ibm_boto3.s3.transfer.TransferConfig

(

Multipart_threshold-file_threshold,

Multipart_chunksize-part_size

```
)
#the upload_fileobj method will automatically execute a multi-part upload
#in 5 MB chunks for all files over 15 MB
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 (0) Complete!\n".format(item_name))
except clienterror as be:
print(“CLIENT ERROR:(0)\n”.format(be))
except Exception as be:
 print(“unable to complete multi-part upload:(0)”.format(0))
def myCommandCallback(cmd):
 print("Command received: " cmd.data)
command cnd.data('command')
print (command)
if(command==’Lighton’):
 print (lighton")
elif(command==’lightoff'):
 print (lightoff")
elif(commande=='motoron’):
 print(‘motoren’)
elif(command==’motoroff'):
print('motoroff’)
myConfigI={
```

"identity":{

“orgid:”hj5fmy”,

"typeid": "NodeMCU",

"deviceId": "12345"

},

"auth": {

"token": "12345678"

}

client - wiotp.sdk.device.DeviceClient (config-myConfig, logHandler=none)
client.connect()

database name="sample"

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 opened’)

While(cap.isopened());

Ret,frame = cap.read()

Gray=cv2.cvtcolor(frame,cv2.colour_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.postmodeloutputrequest

(

```
Model_id='aaa03c23b3724a56b629203edc62c',
Inputs={resoursces_pb2.input(data=resources_pb2.data(image=resources_pb2.imag
e(base64=file_bytes)) )})
Response=stub.postmode1outputs(request,metadata=metadata)
If response.ststus.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:
#print('%12s:%,2f'%(concepts.name,concept.value))
If(concept.value>0.98):
If(concepts.name=="animal"):
Print("Alert!Alert!animal detected")
Playsound.playsound('alert.mp3')
Picname+datetime.datetime.now().strftime("%y-%m-%d-%H-%M")
Cv2.imwrite(picname+'.jng',frame)
Multi_part_upload('gnaneshwar',picname+'.jpg",picname+'.jpg')
Json_document=("link";cos_ENDPOINT+'/'+"gnaneshwar'+'/'+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)
Humidity=random,.randint(0,100)
Mydata={'animal':detect''moisture':moist,'humidity':humidity}
Print(mydata)
If(humidity!=none):
```

```
Client.publishevent(evenid=”status”,msgformat=”json”,data=mydata,qos=0,onpublis
h=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()
```