/\*

Essentially ESP32 will send Data over using HTTP POST Request

POST Request by ESP32 will look smth like this: \*\*\*\*Refer to BOTTOM for the test POST REQUEST URL

https://script.google.com/macros/s/AKfycbx5a0hyg6avjI0sBRZJsHyf1Q6wZheKZsYZLOU\_8hFm5-z8SsIBDHImBCcKln3I2qLB/exec?myFoldername=ESP32-CAM&myFilename=ESP32-CAM.jpg&myFile=

As you can see from above, there are 3 Data parameters being sent

myFoldername -> Name of the folder in Google Drive which stores the images

myFilename -> Base name of the image files to be stored in Google Drive

myFile -> Contains infomation on 1) data type (data:image/jpeg) 2) encoded as Base64 3) Encoded Base64

\*/

//doPost(e) can only be "activated" when it receives a POST Request; the "e" contains all the data received during the request

function doPost(e) {

// storing parameter data into variables

var myFoldername = e.parameter.myFoldername;

var myFilename = Utilities.formatDate(new Date(), "GMT+8", "yyyy-MM-dd\_HH\_mm\_ss")+"-"+e.parameter.myFilename; //Each imagefile uploaded will have a timestamp attached to it so that every image name is unique and timestamped

//var myFilename = e.parameter.myFilename; //Uncomment this for all images having base name only

//myFile stores the infomation on data type and the encoded base64 data

//myFile is in this format "myFile=....."

var myFile = e.parameter.myFile;

var contentType = myFile.substring(myFile.indexOf(":")+1, myFile.indexOf(";")); //content type "image/jpeg" is defined by "data:image/jpeg;" which is inbetween ":" and ";" so by using substring within the index position of ":" and ";" we are able to extract out "image/jpeg"

var data = myFile.substring(myFile.indexOf(",")+1); //Based on the format of myFile, any string after "," is the Base64 code which is stored as data

//we need to decode the base64 code back into an Image using Utilities

data = Utilities.base64Decode(data); //decoding base64

var blob = Utilities.newBlob(data, contentType, myFilename); //creating a blob as a placeholder for the decoded base64 image data

//in arduino the image captured is converted into base64 so when we receive the data in apps script we need to decode it first

// This Section saves the image to Google Drive.

// check if the folder exists in your drive if not create a new folder

var folder, folders = DriveApp.getFoldersByName(myFoldername);

if (folders.hasNext()) {

//if folder exists set folder

folder = folders.next();

} else {

//folder does not exist so create a new folder

folder = DriveApp.createFolder(myFoldername);

}

//create an new image file in the folder

var file = folder.createFile(blob);

file.setDescription("Uploaded by " + myFilename);

//Creates a url for the image for viewing

var imageID = file.getUrl().substring(file.getUrl().indexOf("/d/")+3,file.getUrl().indexOf("view")-1);

var imageUrl = "https://drive.google.com/uc?authuser=0&id="+imageID;

// Outputs the Foldername and filename followed by drive image url

return ContentService.createTextOutput(myFoldername+"/"+myFilename+"\n"+imageUrl+"\n");

}

/\* HOW TO TEST POST REQUEST URL

Test POST Request (Replace with your deployment ID):

https://script.google.com/macros/s/YOURDEPLOYMENTID/exec?myFoldername=test&myFilename=APPS\_SCRIPT\_TEST.jpg&myFile=data:image/jpeg;base64,

Go to this website and test your post request url:

https://www.codepunker.com/tools/http-requests

\*/