Queries	Solutions
How to apply Array Formula in Column	=ARRAYFORMULA($G$5:$G * $H$5:$H)
This script is for combine multiple sheet data in one sheet	=QUERY({'2025'!A2:I;'2024'!A2:I;'2023'!A2:I;'2022'!A2:I;'2021'!A2:I;'2020'!A2:I;'2019'!A2:I;'2018'!A2:I},"SELECT * WHERE Col1 IS NOT NULL OR Col2 IS NOT NULL",0)

How to Insert data in to sheet .	Array = [];
	for(){
	Array[i] = [];
	Array[i].push(arr[i][0]);
	Array[i].push(arr[i][1]);
	Array[i].push(arr[i][2]);
	
	insert(sheet,Array[i],2);
	}
	
	E.g.
	
	var addNewData = [];
	var reports = fetchsheetinactivefilebyGID(1389793800);
	
	addNewData[indexVal] = [];
	addNewData[indexVal].push(new Date());
	addNewData[indexVal].push(Session.getEffectiveUser().getEmail());
	for(var k = 0 ; k <= sheetAllData[indexVal].length ; k++){
	  if(k!=25 && k!=27){
	    addNewData[indexVal].push(sheetAllData[indexVal][k]);
	  }
	}
	insertRow(reports,addNewData[indexVal],2);
	
	//***for insert data ***//
	function insertRow(sheet, rowData, optIndex) {
	  //var index = optIndex || 1;
	  var index = optIndex ;
	  sheet.insertRowBefore(index).getRange(index, 1, 1, rowData.length).setValues([rowData]);
	}//---End of insertRow()---//

How to add Custom Menu In Sheet	var ss = SpreadsheetApp.getActiveSpreadsheet();
	 var menu = [];
	 menu.push({name:'Add Ingredient Items',functionName : "addNewIngredients"})
	 ss.addMenu("Ingredients",menu);
	 var pmenu = [];
	 pmenu.push({name:'Add Purchase Items',functionName : "addNewPurchase"})
	 ss.addMenu("Purchase",pmenu);

Hide Sheet Code	```var sheet = SpreadsheetApp.getActiveSpreadsheet().getSheets();
Logger.log(sheet);
//var user = Session.getActiveUser().getEmail();
var user = Session.getActiveUser().getEmail();
//Browser.msgBox(user);
Logger.log(user);
for(var ss in sheet){
if(user == "mailuser@testmail.com" && sheet[ss].getSheetId() == "740787435"){
sheet[ss].hideSheet();
} else {
//if(sheet[ss].isSheetHidden()){
sheet[ss].showSheet();
//}
}
}
``
Know Last Edit User	```function onEdit(e){
Logger.log(e);
// Set a comment on the edited cell to indicate when it was changed.
var range = e.range;
range.setNote('Last modified: ' + new Date()+"User:"+Session.getActiveUser().getEmail());
}
``
Automatic Formula copy Down	```function onEdit(e){

   var mainSheet = e.source.getActiveSheet();
   var col = mainSheet.getActiveRange().getColumn();
   var row = mainSheet.getActiveRange().getRow();
   Logger.log(col+" :: "+row);
   if(col==1){
     var fillDown = mainSheet.getRange(2,2,row-1).setFormula("=SQRT(A2)");
     mainSheet.getRange("B2").copyTo(fillDown);
   }

}
``` |
| **Formula Copy down** | ```if(col ==7){
     var fd = mainSheet.getRange(1,8,row).setFormula("=(G1*G1)");
     mainSheet.getRange("H1").copyTo(fd);
   }
``` |
| **Data Filter Or dynamic Drop Down** | `=ARRAYFORMULA(UNIQUE(FILTER($A$3:$A,$D$3:D = $G$3)))` |

Simple Form With Complete Validation

```
//***Function For Add Vendor Data***//
function addVendor() {
  if(SpreadsheetApp.getActiveSheet().getSheetId() == "1769813931"){
    var ss = SpreadsheetApp.getActiveSheet();
    var formSheetData = ss.getDataRange().getValues();
    formSheetData.shift();//Remove heading of Whole Sheet Data
    //Master Sheet Which is contain all data , Here Used For Data Validation//
    var masterSheet = fetchsheetinactivefilebyGID(1384371091);
    var masterSheetData = masterSheet.getDataRange().getValues();
    masterSheetData.shift();
    masterSheetData.shift();
    var errorMsg = "Please Add Unique Data!!! It's Already Exists In Database";
    var proceedMsg = "No duplicate Item/s Found. Confirm that you want to add these new Item/s?";
    var mandMsg = "Vendor Name is Mandatory !!!";
    var errorCodes = "";
    var whichMsg = "";
    var addTheseOnes = "";
    var proceed;
    if(formSheetData.length > 0) {
      for(var i in formSheetData) {
        if(masterSheetData.length > 0 ) {
          for(var j in masterSheetData) {
            if(formSheetData[i][0] == "" || formSheetData[i][0] == undefined) {
              whichMsg = mandMsg;
            }else if(masterSheetData[j][0] == formSheetData[i][0]) {
              var ii = parseInt(i) + parseInt(2);
              ss.getRange(ii,1).setBackground('#65A6AE');
              whichMsg = errorMsg;
              errorCodes += formSheetData[i][0] + "\n";
            } else {}
          }//---End of for(var j in masterSheetData) Inner Loop ---//
        }//---End of if(masterSheetData.length > 0 )---//
        if (whichMsg != errorMsg) {
          addTheseOnes += formSheetData[i][0] + "\n";
          ss.getRange(2,1,ss.getLastRow(),ss.getLastColumn()).clearFormat();
        }
      }//---End of for(var i in formSheetData) Outer Loop ---//
      if (whichMsg == mandMsg){
        Browser.msgBox(mandMsg);
        return;
      } else if((whichMsg == errorMsg)){
        Browser.msgBox(errorMsg + "\n" + errorCodes);
        return;
      } else {
        proceed = Browser.msgBox(proceedMsg, addTheseOnes, Browser.Buttons.YES_NO);
        if (proceed == 'yes') {
          var addNewData = [];
          for(var k in formSheetData){
            addNewData[k] = [];
            addNewData[k].push(formSheetData[k][0]);
            //Data Insert Here//
            masterSheet.insertRowBefore(3).getRange(3,1,1,addNewData[k].length).setValues([addNewData[k]]);
          }//---End of for(var k in formSheetData)---//
          ss.getRange(2,1, ss.getLastRow(),ss.getLastColumn()).clear({ formatOnly: true, contentsOnly: true });
        } else if (proceed == ('no' || 'cancel')) {
          Browser.msgBox("Make Corrections and then Reprocess. See you later !!! ");
        } else {}
      }//---End of Else---//
```

| Calender Date Book With Send Mail | ```
//---Begin---Final code---//
function myFunction(e) {
  /*
  index of field
  0 Timestamp
  1 Email Address
  2 Event
  3 Start Date
  4 STime
  5 End Date
  6 ETime
  7 Location
  8 Description
  */
  Logger.log(e.values);
  var dddt = e.values[3].split('/');
  var edddt = e.values[5].split('/');
  //mm/dd/YYYY
  Logger.log(dddt[0]);
  Logger.log(dddt[1]);
  Logger.log(dddt[2]);
  var tm = e.values[4].split(':');
  var etm = e.values[6].split(':');
  //Logger.log(e.values[4].split(':'))
  Logger.log(tm[0]);
  Logger.log(tm[1]);
  Logger.log(tm[2]);
  var sptAMPM = tm[2].split(" ");
  var esptAMPM = etm[2].split(" ");
  Logger.log(sptAMPM[0]);
  Logger.log(sptAMPM[1]);

  Logger.log("==");
  Logger.log(dddt[2]);
  Logger.log(edddt[2]);
  Logger.log("==");
  var yr = dddt[2];
  var eyr = edddt[2];
  var start_date = new Date(yr,dddt[0]-1,dddt[1],tm[0],tm[1],sptAMPM[0],0);
  var end_date = new Date(eyr,edddt[0]-1,edddt[1],etm[0],etm[1],esptAMPM[0],0);
  //Logger.log(yr+"**"+dddt[0]-1+"**"+dddt[1]+"**"+tm[0]+"**"+tm[1]+"**"+sptAMPM[0]+"&&&&&"+eyr+yr);
  Logger.log(start_date);
  Logger.log(end_date);

  //CalendarApp.getDefaultCalendar().createAllDayEvent(e.values[2],start_date,end_date,{location:e.values[7],sendInvites:true,
guests:'kk@gmail.com'});
  CalendarApp.getDefaultCalendar().createEvent(e.values[2],start_date,end_date,{location:e.values[7],sendInvites:true,
guests:'kk@gmail.com'});
  MailApp.sendEmail({
    to:"kk@gmail.com",
    subject:"Calender Invitaion ",
    htmlBody:"What you want , which is show Here"
  });

}
//---End---Final code---//

function myFunctiontest() {
  var cal = CalendarApp.getDefaultCalendar().createAllDayEvent("TEST KRSDHSNA A JUYO",new Date('Nov 23,2017'),new Date
``` |

| How to Make Custom Menu ? | ```javascript
function onOpen() {
 var ui = SpreadsheetApp.getUi();
 // Or DocumentApp or FormApp.
 ui.createMenu('Custom Menu One')
    .addItem('First item', 'menuItem1')
    .addSeparator()
    .addSubMenu(ui.createMenu('Sub-menu')
       .addItem('Second item', 'menuItem2'))
  .addToUi();

 ui.createMenu('Custom Menu Two')
    .addItem('Option1-DialogBox','Option1function')
    .addSeparator()
    .addSubMenu(ui.createMenu('Option1-Sub')
       .addItem('Third item-Sidebar','Option2function'))
    .addToUi();

}

function menuItem1() {
  SpreadsheetApp.getUi() // Or DocumentApp or FormApp.
     .alert('You clicked the first menu item!');
}

function menuItem2() {
  SpreadsheetApp.getUi() // Or DocumentApp or FormApp.
     .alert('You clicked the second menu item!');
}

function Option1function(){
 //SpreadsheetApp.getUi()
 //.alert('Main Menu Click !!!! ');
 //Begin for Custom Dialog Box
 var htmldt = HtmlService.createHtmlOutputFromFile('demoFile')
 .setHeight(500)
 .setWidth(700);
    SpreadsheetApp.getUi()
    .showModalDialog(htmldt, 'This is Only For Demo Purpose !!!');
 //End for Custom Dialog Box

}

function Option2function(){
 //SpreadsheetApp.getUi()
 //.alert('Sub Menu Click !!!! ');
 var htmlside = HtmlService.createHtmlOutputFromFile('demoFile')
 .setTitle('This is Side Bar !!! ')
 .setWidth(300);
 SpreadsheetApp.getUi()
 .showSidebar(htmlside);

}
``` |

| How to Error Handle in Script? | ```javascript
// for Error Handling
function myFunction() {
 try{
  // for Error Generate --- dfjdjf ---
  var ui = SpreadsheetApp.getUi();
  var userChoice = ui.alert(
   'Please confirm',
   'Are you sure you want to continue?',
   ui.ButtonSet.YES_NO);
  if(userChoice == "YES") {
   Browser.msgBox("OK!!!");
  } else{
   Browser.msgBox("Cansel !!! ");
  }
 } catch (e){
  Logger.log(e.toString());
  MailApp.sendEmail(Session.getActiveUser().getEmail(), "Error Found !!! ", "Contact to Admin !!! ");
 }
}
``` |
|---|---|
| get & Set Formula Automatic | ```javascript
function onEdit(e){

  var mainSheet = e.source.getActiveSheet();
  var col = mainSheet.getActiveRange().getColumn();
  var row = mainSheet.getActiveRange().getRow();
  Logger.log(col+" :: "+row);
  if(col==1){
    var fillDown = mainSheet.getRange(2,2,row-1).setFormula("=SQRT(A2)");
    mainSheet.getRange("B2").copyTo(fillDown);
  }
 if(col ==7){
  var fd = mainSheet.getRange(1,8,row).setFormula("=(G1*G1)");
  mainSheet.getRange("H1").copyTo(fd);
 }

}

function aabb(){

  var ss = SpreadsheetApp.getActiveSheet();
  var col = ss.getActiveRange().getColumn();
  var row = ss.getActiveRange().getRow();
  Logger.log(col+" :: "+row);
  Logger.log(ss.getLastColumn());
  Logger.log(ss.getLastRow()-1);
  //Logger.log(ss.getDataRange().getValues());
  var aa = ss.getActiveCell().getColumn();
  var bb = ss.getActiveRange().getColumn();
   Logger.log(bb);

}
```
https://docs.google.com/spreadsheets/d/13AJJmRJpAiS3_1bMyHZPmYieeP15K0L-KbENi02-Xf0/edit#gid=0 |

| | |
|---|---|
| **Query Function IT's Power Full Function Of Google sheet** | =QUERY(Ingredient_Master,"SELECT A WHERE D='Dairy' ORDER BY A ASC ",0) |
| **Dependable drop down** | //*** This is for Dynamic Drop Down purpose ***//<br>function depDrop_(range, sourceRange){<br> var rule = SpreadsheetApp.newDataValidation().requireValueInRange(sourceRange, true).build();<br> range.setDataValidation(rule);<br>}//---End of depDrop_(range, sourceRange)---//<br><br>function onEdit(){<br> //---Begin For Ingredient Form---//<br> if(SpreadsheetApp.getActiveSheet().getSheetId() == "276836978" ){<br>  //Use in Purchase Form for getting Ingredient Data//<br>  var aCell = SpreadsheetApp.getActiveSheet().getActiveCell();<br>  var aRow = aCell.getRow();<br>  var aColumn = aCell.getColumn();<br>  //For Ingredient Form<br>  if (aColumn == 1 && SpreadsheetApp.getActiveSheet()){<br>   var range = SpreadsheetApp.getActiveSheet().getRange(aCell.getRow(), aColumn + 1);<br>   var sourceRange = SpreadsheetApp.getActiveSpreadsheet().getRangeByName(aCell.getValue());<br>   depDrop_(range, sourceRange);<br>  }<br> }<br>} |
| **Formula Set With Condition using Back Slash** | addStockData[i].push("=SUM(QUERY(Purchase_Master,\"SELECT H WHERE E = \"&C3&\" \",0))");<br>      addStockData[i].push("=SUM(QUERY(Purchase_Master,\"SELECT I WHERE E = \"&C3&\" \",0))");<br>      //updateEmpMasterPart1.push("=if((today()-C2)>'D: Fixed Data'!$B$3,\"Yes\",\"Not yet\")")"; |
| **Color Code With Conditional** | AND($A$1:$A="Handed Over to Student",ISBLANK($B$1:$B)=FALSE) |
| **How to use of Filter Function** | =FILTER($A$1:$B9,$A$1:$A9=$E$1) |
| **auto provide $ during range selection** | Press F4 key Window OS |
| **Multiple sheet Open Using Menu bar** | function one(){<br> /*<br> var ss = SpreadsheetApp.getActiveSpreadsheet();<br> SpreadsheetApp.setActiveSheet(ss.getSheets()[0]);<br> */<br> sheetData(0);<br>}<br><br>function two(){<br> sheetData(87572974);<br>}<br><br>function sheetData(SheetID){<br> var ss = SpreadsheetApp.getActiveSpreadsheet().getSheets();<br> for(var i in ss){<br>  if(ss[i].getSheetId() == SheetID){<br>   return SpreadsheetApp.setActiveSheet(ss[i]);<br>   break;<br>  }<br> }<br>} |

| | |
|---|---|
| **Spreadsheet Print Script** | ```function printPdf() {``` |

```
function printPdf() {
    SpreadsheetApp.flush();
    var ss = SpreadsheetApp.getActiveSpreadsheet();
    var sheet = ss.getActiveSheet();
    var gid = sheet.getSheetId();
    //var pdfOpts = '&size=A4&fzr=false&portrait=false&fitw=true&gridlines=false&printtitle=false&sheetnames=false&pagenum=UNDEFINED&attachment=false&gid='+gid;
    var pdfOpts = '&size=A4&fzr=false&portrait=false&fitw=true&gridlines=true&printtitle=false&sheetnames=false&pagenum=UNDEFINED&attachment=false&gid='+gid;
    var Avals = sheet.getRange("B1:B").getValues();
    var entryRows = Avals.filter(String).length;
    var row2 = entryRows;
    var printRange = '&c1=0' + '&r1=0' + '&c2=5' + '&r2='+row2; // B2:APn
    var url = ss.getUrl().replace(/edit$/, '') + 'export?format=pdf' + pdfOpts + printRange;
    var app = UiApp.createApplication().setWidth(500).setHeight(250);
    app.setTitle('Print this sheet');
    var link = app.createAnchor('Download PDF', url).setTarget('_new');
    app.add(link);
    ss.show(app);
}
```

conditional count & Sum	`"=SUMIFS(H1:H10,H1:H10,"3",G1:G10,"11")","=COUNTIFS(G1:G10,"11",H1:H10,"3")",""`
Auto Back up google sheet	`function makeCopy() {`

```
function makeCopy() {

// generates the timestamp and stores in variable formattedDate as year-month-date hour-minute-second
var formattedDate = Utilities.formatDate(new Date(), "GMT", "yyyy-MM-dd' 'HH:mm:ss");

// gets the name of the original file and appends the word "copy" followed by the timestamp stored in formattedDate
var name = SpreadsheetApp.getActiveSpreadsheet().getName() + " Copy " + formattedDate;

// gets the destination folder by their ID. REPLACE xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx with your folder's ID that you can get by opening the folder in Google Drive and checking the URL in the browser's address bar
var destination = DriveApp.getFolderById("xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx");

// gets the current Google Sheet file
var file = DriveApp.getFileById(SpreadsheetApp.getActiveSpreadsheet().getId())

// makes copy of "file" with "name" at the "destination"
file.makeCopy(name, destination);
}
```

| Google spreadsheet send pdf mail | ```javascript
/* Email Google Spreadsheet as PDF */
function emailGoogleSpreadsheetAsPDF() {

  // Send the PDF of the spreadsheet to this email address
  var email = Session.getActiveUser().getEmail()

  // Get the currently active spreadsheet URL (link)
  var ss = SpreadsheetApp.getActiveSpreadsheet();

  // Subject of email message
  var subject = "PDF generated from spreadsheet " + ss.getName();

  // Email Body can  be HTML too
  var body = "Install the <a href='http://www.labnol.org/email-sheet'>Email Spreadsheet add-on</a> for one-click conversion.";

  var blob = DriveApp.getFileById(ss.getId()).getAs("application/pdf");

  blob.setName(ss.getName() + ".pdf");

  // If allowed to send emails, send the email with the PDF attachment
  if (MailApp.getRemainingDailyQuota() > 0)
    GmailApp.sendEmail(email, subject, body, {
      htmlBody: body,
      attachments:[blob]
    });
}
``` |

E COMMERCE WITH GOOGLE DRIVE	
| | ```
/* PayPal Shop with Apps Script */

/* Written by Amit Agarwal - ctrlq.org */


PAYPAL = [
  ["product-001", "useful-websites-book.pdf"],
  ["product-002", "linux-training-course.mp4"],
  ["product-003", "steve-audio-book.mp3"],
  ["product-004", "presentation-template.ppt"]
];

function PayPal() {
  ScriptApp.newTrigger("myShop").timeBased().everyMinutes(5).create();
}

function myShop() {

  var file, size, files, threads;

  for (var p in PAYPAL) {

    threads = GmailApp.search("is:unread from:paypal " + PAYPAL[p][0]);

    if (threads.length > 0) {

      files = DriveApp.searchFiles('title contains "' + PAYPAL[p][1] + '"');

      if (files.hasNext()) {

        file = files.next();
        size = file.getSize()/(1024*1024);

        for (var i=0; i<threads.length; i++) {

          var buyer   = threads[i].getMessages()[0].getReplyTo();
          var subject = "Thank you for your purchase";
          var body    = "Please download the file using the link below.\n\n";

          if (size > 20) {
            file.addViewer(buyer);
            GmailApp.sendEmail(buyer, subject, body + file.getUrl());
          } else {
            GmailApp.sendEmail(buyer, subject, body, {attachments: file.getBlob()});
          }

          threads[i].markRead().moveToArchive();
        }
      }
    }
  }
}
``` |

| **Send mail with Attachment** | ```
function myFunction() {
var revisionSSFileID = "1kt8fiHAV_TbP85px-LGQTqnZIPJVGfJ56Vu1cT2GN94";
var newSheetGID = 0 ;
var arrearsGID = 196794054;
var newSheetName = "kkindia";
 var url = "https://docs.google.com/feeds/download/spreadsheets/Export?key=" + revisionSSFileID + "&exportFormat=xlsx&gid="+ newSheetGID;
 var url2 = "https://docs.google.com/feeds/download/spreadsheets/Export?key=" + revisionSSFileID + "&exportFormat=xlsx&gid="+ arrearsGID ;

 var params = {
    method     : "get",
    headers    : {"Authorization": "Bearer " + ScriptApp.getOAuthToken()},
    muteHttpExceptions: true
 };

 var blob = UrlFetchApp.fetch(url, params).getBlob();
 blob.setName("Salary Revision File generated on " + newSheetName + ".xlsx");
 var blob2 = UrlFetchApp.fetch(url2, params).getBlob();
 blob2.setName("Arrears Effective from file.xlsx");
 MailApp.sendEmail(Session.getActiveUser().getEmail(), "Google Sheet to Excel Format for Payroll (Greytip) Upload", "The XLSX file is attached.", {attachments: [blob,blob2]});

}
``` |
| **Dynamic Sub menu creating** | ```
function onOpen() {
  var ui = SpreadsheetApp.getUi();
  var ss = SpreadsheetApp.getActiveSpreadsheet();
  var sheets = SpreadsheetApp.getActiveSpreadsheet().getSheets();
  var menu = ui.createMenu('Grade 8 ')
  var subMenu = ui.createMenu('Maths');
  for(i=0; i < sheets.length; i++){
    subMenu.addItem(sheets[i].getName(), "test"+i);
  }
  menu.addSubMenu(subMenu).addToUi();
}

function test0(){ Browser.msgBox("ok");}
function test1(){ Browser.msgBox("ok1");}
function test2(){ Browser.msgBox("ok2");}
function test3(){ Browser.msgBox("ok3");}
function test4(){ Browser.msgBox("ok4");}
``` |

| Cloud printing script | ```javascript
function test() {
  Logger.log(search())
  //Logger.log(printGoogleDoc("<docidhere>", "<printeridhere>"))
}
/*Find the printerid @ https://www.google.com/cloudprint/#printers
 *select the printer, click details, and look under Advanced Details for the "printerid"
 */



/*This function will print a normal google doc by fileid
*/
function printGoogleDoc(docid, printerid, optDocName) {
  var doc = DriveApp.getFileById(docid);

  var title = optDocName || doc.getName();
  var ticket = {
    version: "1.0",
    print: {
      color: {
        type: "STANDARD_COLOR",
        vendor_id: "Color"
      },
      duplex: {
        type: "NO_DUPLEX"
      }
    }
  }
  var content = doc.getBlob();
  var optContentType = "application/pdf";
  var optTag = "";

  return submit(printerid, title, ticket, content, optContentType, optTag)
}
``` |
|---|---|

Send mail pdf file	
| | ```
function get_purchaseReport_SendMail(){
 auto_date_input();
 var vendor_sheet = fetchsheetinactivefilebyGID(1384371091);
 var v_value = vendor_sheet.getDataRange().getValues(); v_value.shift(); v_value.shift();
 var main_sheet = fetchsheetinactivefilebyGID(1125044248);
 var main_sheet_value = main_sheet.getRange("A1:G4").getValues();
 var startDate = main_sheet_value[1][1];
 var endDate = main_sheet_value[1][6];
 var bill_data = main_sheet.getRange("A5").getValue();
 var key='1Ewqqx6IACdXsUTf7HSRtFBfowx0Ldx7u9voK85LsEe4';
 var subject = "Kitchen Purchase Report - "+startDate+" To "+endDate;
 var emailBody = "This email contains the Attachment of vendor wise Purchase Report";
 var params = {
  method    : "get",
  headers    : {"Authorization": "Bearer " + ScriptApp.getOAuthToken()},
  muteHttpExceptions: true
 };
 var attachArray = [];
 for(var i in v_value){
  account_purchase_report(v_value[i][0]);
  var bill_data = main_sheet.getRange("A5").getValue();
  if(bill_data !=""){
   var url = "https://docs.google.com/feeds/download/spreadsheets/Export?key=" + key + "&exportFormat=pdf&gid=1125044248";
   var blob = UrlFetchApp.fetch(url, params).getBlob();
   blob.setName("Purchase Report of "+v_value[i][0]+".pdf");
   attachArray.push(blob);
  }
 }
 MailApp.sendEmail({
  to: "mailuser@testmail.com",
  subject: subject,
  htmlBody: emailBody,
  attachments: attachArray
 });
}//---End of totalVendorList()---//
``` |

Mail Advanced Script	

```
//***First Load onOpen() Function***//
function onOpen(){
 var ss = SpreadsheetApp.getActiveSpreadsheet();
 var menu = [];
 menu.push({name:'Send Mail',functionName : "sendMailFunction"})
 ss.addMenu("Operations",menu);
}//---End of onOpen()---//

function sendMailFunction(){
 try{
  var ss = SpreadsheetApp.getActiveSheet();
  if(ss.getSheetId() == 0){
   var subject_line = "Registration Confirmation - Surat Iyenger Yoga Event";
   var allData = ss.getRange(2,1,ss.getLastRow()-1,6).getValues();
   Logger.log(allData);
   var url_link = "http://nucleus.testmail.com";
   var confirm = "Are you sure to send bulk mail?";
   var j;
   if(Browser.msgBox("Mail Sending Process...", confirm, Browser.Buttons.YES_NO_CANCEL) == "yes"){
    for(var i in allData) {
     j = parseInt(i) + parseInt(2);
     Logger.log(j);
     if( allData[i][0]!="" && allData[i][4]!="" && ( allData[i][5] !="Sent" && allData[i][5]!="Pending" ) && validateEmail(allData[i][4]) ==
true &&  ( allData[i][0] == "Grade 1" || allData[i][0] == "Grade 2" || allData[i][0] == "Grade 3" || allData[i][0] == "Grade 4") && MailApp.
getRemainingDailyQuota() > 1){
       MailApp.sendEmail({
        to:allData[i][4],
        subject:"AS/PS Consent 2018-19",
        replyTo:"Surat@testmail.com",
        htmlBody:"<p>Dear Parent / Student,</p><p>You haven't yet marked your selection for AS & PS for the next year. Please
do so by logging into nucleus <a href="+url_link+"> login </a> from your parent account. If you don't mark your AS / PS preference
by 26th April, then we will choose one for you. For any further queries, kindly email us on Surat@testmail.com</p>"
       });
       ss.getRange("F"+j).setValue("Sent");
       Logger.log("Grade upto 4 %s",j);
     } else if( allData[i][0]!="" && ( allData[i][3]!="" || allData[i][4]!="" ) && ( allData[i][5] !="Sent" && allData[i][5]!="Pending" ) && (
validateEmail(allData[i][3]) == true || validateEmail(allData[i][4]) == true ) && ( allData[i][0] == "Grade 5" || allData[i][0] == "Grade 6" ||
allData[i][0] == "Grade 7" || allData[i][0] == "Grade 8" || allData[i][0] == "Grade 9" || allData[i][0] == "Grade 10" || allData[i][0] ==
"Grade 11" || allData[i][0] == "Grade 12" ) && MailApp.getRemainingDailyQuota() > 1){
       MailApp.sendEmail({
        to:allData[i][3]+","+allData[i][4],
        subject:"AS/PS Consent 2018-19",
        replyTo:"Surat@testmail.com",
        htmlBody:"<p>Dear Parent / Student,</p><p>You haven't yet marked your selection for AS & PS for the next year. Please
do so by logging into nucleus <a href="+url_link+"> login </a> from your parent account. If you don't mark your AS / PS preference
by 26th April, then we will choose one for you. For any further queries, kindly email us on Surat@testmail.com</p>"
       });
       ss.getRange("F"+j).setValue("Sent");
       Logger.log("Grade 5 to 12 %s",j);
     }
    }//---End of for(var i in allData) ---//
   } else{
    Browser.msgBox("Oops!!!Again Reprocess!!!");
   }
  } else{
   Browser.msgBox("Oops!!! It's wrong sheet , Please Open proper sheet and again Reprocess!!!");
  }
 } catch(e){
  Logger.log(e.toString());
 }
```

Send Mail doc to pdf	```unction sendDoc(){` ` var subject = "Kitchen Purchase Report - ";` ` var emailBody = "This email contains the Attachment of vendor wise Purchase Report";` ` var params = {` ` method : "get",` ` headers : {"Authorization": "Bearer " + ScriptApp.getOAuthToken()},` ` muteHttpExceptions: true` ` };`  ` var key = "1Am8xzen42S35zsTat72jYKSpTV7Xs2MqXnzGztctkt8";` ` var url = "https://docs.google.com/feeds/download/documents/export/Export?id="+key+"&exportFormat=pdf";` ` var blob = UrlFetchApp.fetch(url, params).getBlob();` ` blob.setName("Purchase Report.pdf");`  ` Logger.log(blob);`   ` MailApp.sendEmail({` ` to: "mailuser@testmail.com",` ` subject: subject,` ` htmlBody: emailBody,` ` attachments: blob` ` });`  `}```
Provide double quotes	`var text = "\"" + response[key].imagePath + "\"";`

download xlxs file Link	`var url = "https://docs.google.com/feeds/download/spreadsheets/Export?key=" + revisionSSFileID + "&exportFormat=xlsx&gid="+ newSheetGID;`
	
	`var url2 = "https://docs.google.com/feeds/download/spreadsheets/Export?key=" + revisionSSFileID + "&exportFormat=xlsx&gid="+ arrearsGID ;`

| Convert Spreadsheet to JSON | ```javascript
function doGet(e){

 // Change Spread Sheet url
 var ss = SpreadsheetApp.openByUrl("Your Spread Sheet URL");

// Sheet Name, Chnage Sheet1 to Users in Spread Sheet. Or any other name as you wish
 var sheet = ss.getSheetByName("Users");

 return getUsers(sheet);

}


function getUsers(sheet){
  var jo = {};
  var dataArray = [];

// collecting data from 2nd Row , 1st column to last row and last column
  var rows = sheet.getRange(2,1,sheet.getLastRow()-1, sheet.getLastColumn()).getValues();

  for(var i = 0, l= rows.length; i<l ; i++){
    var dataRow = rows[i];
    var record = {};
    record['id'] = dataRow[0];
    record['name'] = dataRow[1];

    dataArray.push(record);

  }

  jo.user = dataArray;

  var result = JSON.stringify(jo);

  return ContentService.createTextOutput(result).setMimeType(ContentService.MimeType.JSON);

}
``` |

| **String,_array_date_number etc. Various Validation Function** | ```js
function isString (value) {
  return typeof value === 'string' || value instanceof String;
}

function isNumber (value) {
  return typeof value === 'number' && isFinite(value);
}

function isArray (value) {
  return value && typeof value === 'object' && value.constructor === Array;
}

function isFunction (value) {
  return typeof value === 'function';
}

function isObject (value) {
  return value && typeof value === 'object' && value.constructor === Object;
}

function isNull (value) {
  return value === null;
}

function isUndefined (value) {
  return typeof value === 'undefined';
}

function isBoolean (value) {
  return typeof value === 'boolean';
}

function isSymbol (value) {
  return typeof value === 'symbol';
}

function isDate (value) {
  return value instanceof Date;
}

function isError (value) {
  return value instanceof Error && typeof value.message !== 'undefined';
}

function isRegExp (value) {
  return value && typeof value === 'object' && value.constructor === RegExp;
}
``` |

| **JavaScript Object References** | ```javascript
var rect = { width: 20, height: 10 };
// add new object
rect.color = { red: 0, green: 255, blue: 128 };
// add new method
rect.getArea = function() {
   return this.width * this.height;
};
alert(rect.color.red);          // => 0
alert(rect.color.green);         // => 255
alert(rect.color.blue);         // => 128
alert(rect.getArea());          // => 200
``` |
| **Complete Tutorial for Javascript** | |
| **Error handling in GAS** | ```javascript
try{
 // code
 } catch(e){
 //code
 Logger.log(e.toString());
 }
``` |
| **Fetch Same sheet data in google spreadsheet** | ```javascript
function fetchsheetinactivefilebyGID (GID) {
 var ss = SpreadsheetApp.getActive();
 var allsheets = SpreadsheetApp.getActive().getSheets();
  for(var sheetnumber in allsheets){ // iterate all sheets and compare ids and fetch the one with the ID hardcoded
    if(allsheets[sheetnumber].getSheetId()==GID){break}
    }
 var sheet = ss.getSheets()[sheetnumber];
 return sheet;
}//---End of fetchsheetinactivefilebyGID()---//
``` |
| **Fetch Diff sheet data in google spreadsheet** | ```javascript
function fetchsheetinGivenfilebyGID (ss, GID) {
 var allsheets = ss.getSheets();
  for(var sheetnumber in allsheets){ // iterate all sheets and compare ids and fetch the one with the ID hardcoded
    if(allsheets[sheetnumber].getSheetId()==GID){break}
    }
 var sheet = ss.getSheets()[sheetnumber];
 return sheet;
}//---End of fetchsheetinGivenfilebyGID()---//
``` |
| **dd-MM-YYYY Formate date(01 Jan 2018)** | ```javascript
function ISTdate (z) {
 // Logger.log(z);
 // Logger.log(IsNull(z));
 if (IsNull(z)) {}
 else {z = Utilities.formatDate(z, "IST", "dd MMM yyyy");}
 return z
}
``` |

Traspose Array	```
	function transpose(a)
	{
	  return a[0].map(function (_, c) { return a.map(function (r) { return r[c]; }); });
	  // or in more modern dialect
	  // return a[0].map((_, c) => a.map(r => r[c]));
	}
	
	function specialTranspose(array) {
	
	  var returnArray = [];
	  var length = array.length;
	  var width = array[0].length;
	  for (i = 0; i < length; i++) {
	    for (j = 0; j < width; j++) {
	      returnArray.push(array[i][j]);
	    }
	  }
	
	// Logger.log(returnArray.length);
	
	  var flatArray = [];
	  flatArray[0] = [];
	  for (k = 0; k < returnArray.length; k++) {
	    flatArray[0].push(returnArray[k]);
	  }
	  return flatArray;
	}
	``

Get Google Form ID (Component - Item)	```
	var form = FormApp.getActiveForm();
	      var items = form.getItems();
	      for (var i in items) {
	        Logger.log(items[i].getTitle() + ': ' + items[i].getId());
	
	      }
	``

HTML Table Format Email Send 2*2 Matrix etc.	

```
function abc(){
var TestArrayS = [
      ["Self Review - Scorecard","Team who had filed up the form","Rules Knowledge and Use","Fouls and Body Contact","Fair-
Mindedness","Positive Attitude and Self-Control","Communication","Total Score - Self Review"],
      ["",content[1],content[9],content[10],content[11],content[12],content[13],totPointSelf]
   ];

   var ColWidthArrayS = [350,200];
   var EmailbodyS = MakeHTMLTable(TestArrayS,ColWidthArrayS);

 MailApp.sendEmail({});
}

function MakeHTMLTable(TwoDArray, ColWidthArray) {
   var endofCell= "</td>";
   Logger.log("col length : "+ColWidthArray.length);
   var FirstColWidth = ColWidthArray[0];
   var SecondColWidth = ColWidthArray[1];

   var ColWidthOpener = "<col width=\"";
   var ColWidthCloser = "\">";

   var TableHeader = "<!DOCTYPE html><html><head><meta charset=\"UTF-8\"><title>Table created using Google
Spreadsheet</title></head><body><!DOCTYPE html><html><head><meta charset=\"UTF-8\"><title>HTML Table for Email on the
fly</title></head><body><table cellspacing=\"0\" cellpadding=\"0\" dir=\"ltr\" border=\"1\" style=\"table-layout:fixed;font-size:10pt;font-
family:arial,sans,sans-serif;width:0px;border-collapse:collapse;border:none\"><colgroup>";
   Logger.log(TableHeader);

   var DefiningColumns = "";
   for(var k = 0; k < ColWidthArray.length; k++) {
     Logger.log(ColWidthArray[k]);
     DefiningColumns += ColWidthOpener + ColWidthArray[k] + ColWidthCloser; }
   Logger.log(DefiningColumns);

   TableHeader += DefiningColumns;
   Logger.log(TableHeader);

   //TableHeader += FirstColWidth + "\"><col width=\"" + SecondColWidth + "\"></colgroup><tbody><tr style=\"height:21px\"><td
style=\"overflow:hidden;padding:2px 3px;vertical-align:bottom;background-color:rgb(139,195,74);font-weight:bold\">";
   TableHeader += "</colgroup><tbody><tr style=\"height:21px\">";
   Logger.log(TableHeader);

   var HeaderRowStyle = "<td style=\"overflow:hidden;padding:2px 3px;vertical-align:bottom;background-color:rgb(139,195,74);font-
weight:bold\">";
   var HeaderCols = "";
   for(var m = 0; m < TwoDArray.length; m++) {
      Logger.log(TwoDArray[m][0]);
     HeaderCols += HeaderRowStyle + TwoDArray[m][0] + endofCell; }
   Logger.log(HeaderCols);
   TableHeader += HeaderCols;
   Logger.log(TableHeader);

   var endofRow= "</tr>";
   var startofRow= "<tr style=\"height:21px\">";
   var whiteRowCol= "<td style=\"overflow:hidden;padding:2px 3px;vertical-align:bottom\">";
   var greenRowCol= "<td style=\"overflow:hidden;padding:2px 3px;vertical-align:bottom;background-color:rgb(238,247,227)\">";
   var TableEnder = "</tbody></table></body></html></body></html>'>";

   var FullHTMLTable = TableHeader;
```

Numbers to Indian Decimal conversion	```function numberToIndianNoDec(number){```
	 ``` var parts = number.toString().split(".");```
	 ``` parts[0] = parts[0].replace(/\B(?=(\d{3})+(?!\d))/g, ",");```
	 ``` parts[1] = ((parts[1] || "") + "00").slice(0,0);```
	 ``` return parts[0];```
	 ```}```
	
	 ```function numberToIndian2Dec(number){```
	 ``` var parts = number.toString().split(".");```
	 ``` parts[0] = parts[0].replace(/\B(?=(\d{3})+(?!\d))/g, ",");```
	 ``` parts[1] = ((parts[1] || "") + "00").slice(0,2);```
	 ``` return parts.join(".");```
	 ```}```

Only One column Data Length & Data Fetch	```var Avals = itemInputSheet.getRange("E1:E").getValues();``` ```var Alast = Avals.filter(String).length;```

Confirmation Like confirm button(Javascript) in GAS	```var cName = "Text....";```
	 ```var proceed = Browser.msgBox("Add to the master list?",```
	 ``` cName , Browser.Buttons.YES_NO);```
	 ``` if (proceed == 'yes') {```
	 ``` //Code```
	 ``` } else {```
	 ``` Browser.msgBox("Make desired changes require...& reprocess")};```
	 ``` }```

push() pop() shift() unshift() Function Javascript	Unshift and shift make the whole array shift sideways (by adding and removing items from the beginning).
	Push and pop do NOT make the array shift sideways (because they add and remove items at the end).
	Shift() :
	This method is similar to pop() method, the only difference is that it works at the beginning of the array. It delete the first element from the given array and returns it.
	Example :

	// Creating a test array
	var data = ["A", "B", "C"];
	// Delete the element from the beginning of the array.
	 data.shift();
	Unshift() :

	This method is similar to push() method, the only difference is that it works at the beginning of the array. It add one or more elements to the beginning of an array.
	Example :

	// Creating a test array
	var data = ["X"];
	// Add the element at the beginning of an array.
	data.unshift("B", "C");
	Push() :

	This method can add one or more elements at the end of an array.
	Example :

	// Creating a test array
	var data = ["X"];
	// Add the element at the end of an array.
	data.push("B", "C");
	Pop() :

	This method removes the last element from the given array and returns it.
	Example :

	// Creating a test array
	var data = ["A", "B", "C"];
	// Delete the element from the end of an array.
	data.pop();

Provide Data validation in google sheet	var sourceRange = fromIngredientMaster.getRange("A3:A"+noofrow).getValues();
	  var rule = SpreadsheetApp.newDataValidation();
	  rule.setAllowInvalid(false);
	  rule.setHelpText("Select From List!!!");
	  rule.requireValueInList(sourceRange,true);
	  Tosheet.getRange("B4:B").setDataValidation(rule);

Set Formula in range Range	stockFormsheet.getRange(4,3).setFormula("ARRAYFORMULA(if(ISBLANK($B$4:$B),\"\",VLOOKUP($B$4:$B,Ingredient_Master, 5,0)))");

Set Google sheet Validation with nameRange & Remove	var range = issueSheet.getRange("A2");
	var sourceRange = SpreadsheetApp.getActiveSpreadsheet().getRangeByName(comm_val);
	 var rule = SpreadsheetApp.newDataValidation().requireValueInRange(sourceRange, true).build();
	 range.setDataValidation(rule);
	
	 issue_Form.getRange("B4:B").clearDataValidations();

Sheet Data ASC & DESC	stockMasterSheet.getRange(3,1,stockMasterSheet.getLastRow(),stockMasterSheet.getLastColumn()).sort({column: 3, ascending: true});
	
	dailyCostsheet.getRange(3,1,dailyCostsheet.getLastRow(),dailyCostsheet.getLastColumn()).sort({column: 1, descending: true});
	     issueRecMaster.getRange(3,1,issueRecMaster.getLastRow(),issueRecMaster.getLastColumn()).sort({column: 1, descending: true});

Multi Dimensional Array GAS	function myFunction() {
	 var ss = fetchsheetinactivefilebyGID(0);
	 //var multidarray = [[["KK"],["aa"],["11"]],[["Surat"],["bb"],["22"]],[["India"],["cc"],["33"]]];
	// var multidarray = [[[["KK"],["AA"],["LL"]]],[[[["123"],[["k"],[["i"],["p"],["l"]],["qwerty"]],["uu"]],["22"],["33"]]]];
	 var multidarray =
	 [
	  [
	   [
	    [
	     [],
	     [
	      [],
	      ["kkindia"],
	      []
	     ],
	     []
	    ],
	    [],
	    []
	   ],
	   []
	  ],
	  [[[[],[[],[],[]],[]],[],[],[]]
	];
	 Logger.log(multidarray[0][0][0][1][1]);
	 //ss.getRange("A1").setValue(multidarray);
	}

Open Specific Sheet Using Sheet id - creating / use with menubar	//---Purpose:Get Sheet used for forms/master Sheet open by menubar/button Date:22-01-2018:kkindia---//
	function open_sheet_FromID(clickSheetId){
	 var ss = SpreadsheetApp.getActiveSpreadsheet().getSheets();
	 for(var i in ss){
	  if(ss[i].getSheetId() == clickSheetId){
	   return SpreadsheetApp.setActiveSheet(ss[i]);
	   break;
	  }
	 }//---End of For()---//
	}//---End of open_sheet_FromID()---//

Creating pdf file (From Google Sheet) & Saving in Google Drive Specific Folder	function pantry_sales_report_PDF(from_date,to_date) {
	 try{
	 var foldersave = DriveApp.getFolderById('Folder ID');
	 //var folderURL = foldersave.getUrl();
	 var reportGID = 1242928613; //ISTdate
	 SpreadsheetApp.flush();
	 var request = {
	  "method": "GET",
	  "headers":{"Authorization": "Bearer "+ScriptApp.getOAuthToken()},
	  "muteHttpExceptions": true
	 };
	 var key='Sheet KEY';
	 var fetch='https://docs.google.com/spreadsheets/d/'+key+'/export?format=pdf&size=A4&portrait=true&gid='+ reportGID;
	 var name = "Pantry sales Report for FS : "+ISTdate(from_date)+" to "+ISTdate(to_date);
	 var pdf = UrlFetchApp.fetch(fetch, request);
	 pdf = pdf.getBlob().setName(name);
	 var file = foldersave.createFile(pdf);
	 //Logger.log(file.getUrl());
	 var emailBody = "Pantry sales Report for FS";
	 var d = new Date();
	 var subject = "Pantry sales Report for FS: "+ISTdate(from_date)+" to "+ISTdate(to_date);
	 MailApp.sendEmail({
	  to: "mail id Shown HERE",
	  subject: subject,
	  htmlBody: emailBody,
	  attachments: [pdf]
	 });
	 }catch(e){
	  Logger.log(e.toString());
	 }
	}//---End of pantry_sales_report_PDF(from_date,to_date)---//

Multiple Attachment pdf File Creating & Save google drive	```function get_purchaseReport_SendMail(){
  try{
    auto_date_input();
    var vendor_sheet = fetchsheetinactivefilebyGID(1384371091);
    var v_value = vendor_sheet.getDataRange().getValues(); v_value.shift(); v_value.shift();
    var main_sheet = fetchsheetinactivefilebyGID(1432422303);
    var main_sheet_value = main_sheet.getRange("A1:G6").getValues();
    var startDate = ISTdate(main_sheet_value[1][1]);
    var endDate = ISTdate(main_sheet_value[1][6]);
    var foldersave = DriveApp.getFolderById('folder id');
    var key='key show here';
    var subject = "Kitchen Purchase Report - "+startDate+" to "+endDate;
    var emailBody = "This email contains attachment of The Purchase Report of FS Kitchen";
    var params = {
      method   : "get",
      headers   : {"Authorization": "Bearer " + ScriptApp.getOAuthToken()},
      muteHttpExceptions: true
    };
    var attachArray = [];
    for(var i in v_value){
      account_purchase_report(v_value[i][0]);
      var bill_data = main_sheet.getRange("A7").getValue();
      if(bill_data !=""){
        var url = "https://docs.google.com/feeds/download/spreadsheets/Export?key=" + key + "&exportFormat=pdf&gid=1432422303";
        var blob = UrlFetchApp.fetch(url, params).getBlob();
        blob.setName("Purchase Report of "+v_value[i][0]+".pdf");
        foldersave.createFile(blob.setName("Purchase Report of "+v_value[i][0]+".pdf"));
        attachArray.push(blob);
      }
    }

    MailApp.sendEmail({
      to: "Mail id put here",
      subject: subject,
      htmlBody: emailBody,
      attachments: attachArray
    });
  }catch(e){
    Logger.log(e.toString());
  }
}//---End of totalVendorList()---//``` |
| **Set/Remove Dynamic border & ASC _ DESC** | ```main_sheet.getRange("G"+addr).setBorder(true,true,true,true, true, true);
    main_sheet.getRange(7,1,filter_array.length,filter_array[0].length).sort([{column: 1, ascending: true },{column: 2, ascending: true}]);
    SpreadsheetApp.flush();

main_sheet.getRange("A7:G1000").setBorder(false,false,false,false,false,false);//for border``` |

| Javascript date function With Month names | ```
function auto_date_input(){

    var main_sheet = fetchsheetinactivefilebyGID(1432422303);
    var today = new Date();
    var dd = today.getDate();
    var mm = today.getMonth()+1; //January is 0!
    var yyyy = today.getFullYear();
    if(dd<10){
      dd='0'+dd;
    }
    if(mm<10){
      mm='0'+mm;
    }
    var today = dd+'/'+mm+'/'+yyyy;
    mm=mm;
    //------------------------update : 18-Apr-2018 :kkindia------------------------//
    //var m_names = new Array("Jan", "Feb", "Mar","Apr", "May", "Jun", "Jul", "Aug", "Sep","Oct", "Nov", "Dec");
    //JanFebMarAprMayJunJulAugSepOctNovDec



}
``` |

| How to Get, Parse, Query, and Return JSON | ```javascript
// make a GET request to the API
var chuckNorrisJSON = UrlFetchApp.fetch("http://api.icndb.com/jokes/random");

// log the text that you retrieved
Logger.log(chuckNorrisJSON)

function doPost(request){

  // get the sting value of the POST data
  var postJSON = request.postData.getDataAsString();

  // save that JSON to a file
  DriveApp.createFile('post.json', postJSON);

}

// define a file iterator
  // that will iterate through all files with the name 'post.json'
  var iter = DriveApp.getFilesByName("post.json");

  // iterate through all the files named 'post.json
  while (iter.hasNext()) {

    // define a File object variable and set the Media Tyep
    var file = iter.next();
    var jsonFile = file.getAs('application/json')

    // log the contents of the file
    Logger.log(jsonFile.getDataAsString());

  }

var myDog = {"name": "Rhino", "breed": "pug", "age": 8}

var myJSON = JSON.stringify(myDog);

function doGet(request){

  // return JSON text with the appropriate Media Type
  return ContentService.createTextOutput(myJSON).setMimeType(ContentService.MimeType.JSON);

}
``` |

| Google apps script doGet & doPost | ```javascript
function doGet(){

 var appData = {
 "heading": "Hello World!",
 "body": "Welcome to the web app."
 };

 var JSONString = JSON.stringify(appData);
 var JSONOutput = ContentService.createTextOutput(JSONString);
 JSONOutput.setMimeType(ContentService.MimeType.JSON);
 return JSONOutput
}

function doPost(e){

 // Get the "text" query string value
 // eg. &text=Your%20Name
 var name = e.parameter.text

 // Get the Chuck Norris Joke
 var chuckQuote = postChuckNorris(name);

 // Return plain text Output
 return ContentService.createTextOutput(chuckQuote);
}


// Make a call to the Chuck Norris joke API
// parse the JSON and return only the joke
function postChuckNorris(name){

 var queryString = makeQueryString(name)

 var chuckData = UrlFetchApp.fetch("http://api.icndb.com/jokes/random/" + queryString);

 var chuckJSON = JSON.parse(chuckData.getContentText());
 var chuckQuote = chuckJSON.value.joke;

 return chuckQuote
}


// Helper function to assemble the query string for
// calling the Chuck Norris API from a given name
function makeQueryString(name){
 var query = ""

 if (name !== undefined){

 var names = name.split(" ");
 query = "?firstName=" + names[0];

 if (names[1] !== undefined){
  query += '&lastName=' + names[1];
 }

 }

 return query
``` |

JSON Convert	`const myObj = {`
	`  name: 'Skip',`
	`  age: 2,`
	`  favoriteFood: 'Steak'`
	`};`
	`const myObjStr = JSON.stringify(myObj);`
	`console.log(myObjStr);`
	`// "{"name":"Skip","age":2,"favoriteFood":"Steak"}"`
	`console.log(JSON.parse(myObjStr));`
	`// Object {name:"Skip",age:2,favoriteFood:"Steak"}"`
Google sheet to JSON output data	https://script.google.com/a/macros/testmail.com/s/AKfycbygukdW3tt8sCPcFDlkMnMuNu9bH5fpt7bKV50p2bM/exec?id=SheetID&sheet=Sheet1

| **Publish Google Spreadsheets as JSON with Apps Script** | ```
/* Source: https://gist.github.com/daichan4649/8877801 */
function doGet(e) {

  var sheetName = "Sheet 1";
  var sheetId   = "1234...";

  var book = SpreadsheetApp.openById(sheetId);
  var sheet = book.getSheetByName(sheetName);

  var json = convertSheet2JsonText(sheet);

  return ContentService
        .createTextOutput(JSON.stringify(json))
        .setMimeType(ContentService.MimeType.JSON);
}

function convertSheet2JsonText(sheet) {
 // first line(title)
 var colStartIndex = 1;
 var rowNum = 1;
 var firstRange = sheet.getRange(1, 1, 1, sheet.getLastColumn());
 var firstRowValues = firstRange.getValues();
 var titleColumns = firstRowValues[0];

 // after the second line(data)
 var lastRow = sheet.getLastRow();
 var rowValues = [];
 for(var rowIndex=2; rowIndex<=lastRow; rowIndex++) {
   var colStartIndex = 1;
   var rowNum = 1;
   var range = sheet.getRange(rowIndex, colStartIndex, rowNum, sheet.getLastColumn());
   var values = range.getValues();
   rowValues.push(values[0]);
 }

 // create json
 var jsonArray = [];
 for(var i=0; i<rowValues.length; i++) {
   var line = rowValues[i];
   var json = new Object();
   for(var j=0; j<titleColumns.length; j++) {
     json[titleColumns[j]] = line[j];
   }
   jsonArray.push(json);
 }
 return jsonArray;
}
``` |

| Google App Script To Fetch Data From JSON Webservice and Write them to google spreadsheet. | ```function pullJSON() {
  var ss = SpreadsheetApp.getActiveSpreadsheet();
  var sheets = ss.getSheets();
  var sheet = ss.getActiveSheet();

  var url="http://example.com/feeds?type=json"; // Paste your JSON URL here

   var response = UrlFetchApp.fetch(url); // get feed
  var dataAll = JSON.parse(response.getContentText()); //
  var dataSet = dataAll;

  var rows = [],
     data;

  for (i = 0; i < dataSet.length; i++) {
    data = dataSet[i];
    rows.push([data.id, data.name,data.email]); //your JSON entities here
  }

  dataRange = sheet.getRange(1, 1, rows.length, 3); // 3 Denotes total number of entites
  dataRange.setValues(rows);

}``` |
|---|---|
| Delete Multiple Rows At a same time using script | ```function deleteRows() {
  var sheet = SpreadsheetApp.getActiveSheet();
  var rows = sheet.getDataRange();
  var numRows = rows.getNumRows();
  var values = rows.getValues();

  var rowsDeleted = 0;
  for (var i = 0; i <= numRows - 1; i++) {
    var row = values[i];
    if (row[0] == 'delete' || row[0] == '') { // This searches all cells in columns A (change to row[1] for columns B and so on) and deletes row if cell is empty or has value 'delete'.
      sheet.deleteRow((parseInt(i)+1) - rowsDeleted);
      rowsDeleted++;
    }
  }
};``` |

| Google form Creation using Google sheet | ```
function myFunction() {
 var ss = SpreadsheetApp.getActiveSpreadsheet().getSheetByName('Sheet1');
 var range = ss.getDataRange();
 var data = range.getValues();
 var numberRows = range.getNumRows();
 var numberColumns = range.getNumColumns();
 var firstRow = 1;
 var form = FormApp.openById('PUT FORM URL NAME HERE');

 for(var i=0;i<numberRows;i++){
  var questionType = data[i][0];
  if (questionType==''){
    continue;
  }
  else if(questionType=='TEXT'){
   form.addTextItem()
    .setTitle(data[i][1])
    .setHelpText(data[i][2])
    .setRequired(true);
  }
  else if(questionType=='PARAGRAPH'){
   form.addParagraphTextItem()
    .setTitle(data[i][1])
    .setHelpText(data[i][2])
    .setRequired(true);
  }
  else if(questionType=='CHOICE'){
  var rowLength = data[i].length;
  var currentRow = firstRow+i;
  var currentRangeValues = SpreadsheetApp.getActiveSpreadsheet().getSheetByName('Sheet1').getRange(currentRow,1,1,
rowLength).getValues();
  var getSheetRange = SpreadsheetApp.getActiveSpreadsheet().getSheetByName('Sheet1').getDataRange();
  var numberOfColumnsSheet = getSheetRange.getNumColumns();
  var numberOfOptionsInCurrentRow = numberOfColumnsSheet;
  var lastColumnInRange = String.fromCharCode(64 + (numberOfOptionsInCurrentRow));
  var range_string = 'E' + currentRow + ":" + lastColumnInRange + currentRow;
  var optionsArray = SpreadsheetApp.getActiveSpreadsheet().getSheetByName('Sheet1').getRange(range_string).getValues();
  var choicesForQuestion =[];
    for (var j=0;j<optionsArray[0].length;j++){
       choicesForQuestion.push(optionsArray[0][j]);
       }
  form.addMultipleChoiceItem()
    .setTitle(data[i][1])
    .setHelpText(data[i][2])
    .setChoiceValues(choicesForQuestion)
    .setRequired(true);
  }
  else if(questionType=='CHECKBOX'){
  var rowLength = data[i].length;
  var currentRow = firstRow+i;
  var currentRangeValues = SpreadsheetApp.getActiveSpreadsheet().getSheetByName('Sheet1').getRange(currentRow,1,1,
rowLength).getValues();
  var getSheetRange = SpreadsheetApp.getActiveSpreadsheet().getSheetByName('Sheet1').getDataRange();
  var numberOfColumnsSheet = getSheetRange.getNumColumns();
  var numberOfOptionsInCurrentRow = numberOfColumnsSheet;
  var lastColumnInRange = String.fromCharCode(64 + (numberOfOptionsInCurrentRow));
  var range_string = 'E' + currentRow + ":" + lastColumnInRange + currentRow;
  var optionsArray = SpreadsheetApp.getActiveSpreadsheet().getSheetByName('Sheet1').getRange(range_string).getValues();
  var choicesForQuestion =[];
``` |

| Array Unique | ```function uniqueElements(array) {
  return array.filter(function (value, index, self) {
    return self.indexOf(value) === index;
  });
}``` |
|---|---|
| get row no using datarange | ```var fullEntryData = entrySheet.getDataRange().getValues();
  var entryData = [];
  for (var i = 0; i < fullEntryData.length; i++) {
      if (fullEntryData[i][1] == "") {break;}
    }
  var entryRows = i;
  var entryCols = entrySheet.getLastColumn();
  var entryData = entrySheet.getRange(3,1,entryRows-2,entryCols).getValues();``` |
| Download google apps Script | https://script.google.com/feeds/download/export?format=json&id=[fileId] |
| Folder Creation Through Script | ```function drive_folder_test(){
 var dr = DriveApp.getFolderById("1a2Ysj-EgcmNtk4_jwSwjgeVaX5cKJUQP");
 var files = dr.getFolders();
 var fl = "11K_TEST";
 var data = "";
 var cond = false;
 while(files.hasNext()){
   data = files.next();
   if(fl == data.getName()){
     cond = true;
     break;
   }
 }
 if(!cond){
   DriveApp.getFolderById("1a2Ysj-EgcmNtk4_jwSwjgeVaX5cKJUQP").createFolder(fl);
 }
}``` |
| PDF GENERATING FROM DOC FILE FORMAT FILE | ```var driveFolder = DriveApp.getFolderById("1lDG_l7uX95tUNHZirGBA8Hkhg8IX-XZQ");
var htmlBody,blob;
htmlBody = "",blob ="";
htmlBody = HtmlService.createHtmlOutput(HTML CODE GOES HERE).getContent();
blob = Utilities.newBlob(htmlBody, 'text/html').getAs('application/pdf').setName("PDF File Name");
driveFolder.createFile(blob);``` |
| Image show in pdf file | ```var image = UrlFetchApp.fetch('img.png');
    var base64 = Utilities.base64Encode(image.getBlob().getBytes());
    var img_view = '';``` |
| | |