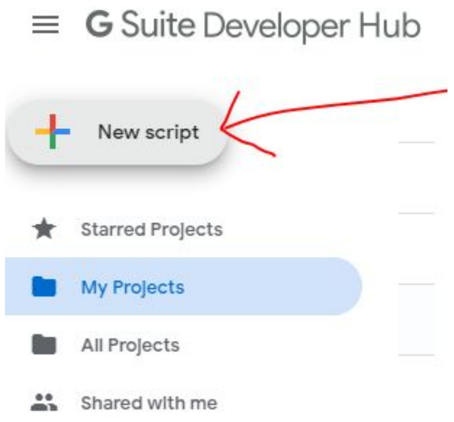
# Google Script

Class CalendarApp - Content Feed



## Class CalendarApp - Write Google Script

Allows a script to read and update the user's Google Calendar. This class provides direct access to the user's default calendar, as well as the ability to retrieve additional calendars that the user owns or is subscribed to.

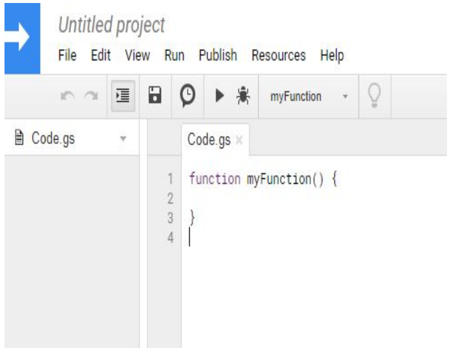
Creating Google Script - GSuite Developer Hub

https://script.google.com/home

**Create Stand alone Script**

Opens Default Online Editor with myFunction()

Code is JavaScript Based



```
function myFunction() {

}
```

## Write Google Script

Open the editor for script.

Add a title for the project

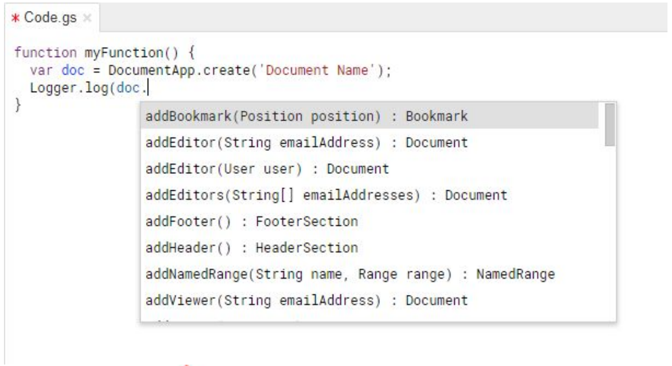
Add lines of code in the default function.

Note using the suggestion tool helps save time and ensure you have the correct object to apply the method to.

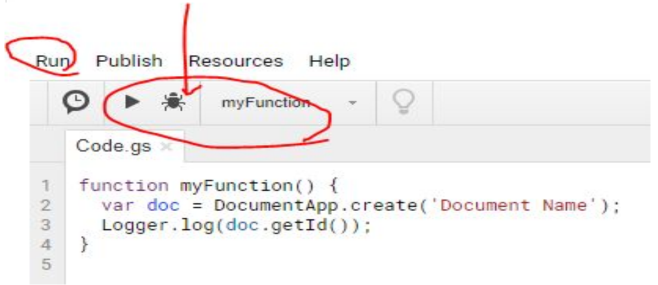
Try Logger.log() to log and debug.

Select Run or Press the Icons on the shortcut menu. Select the function you want to invoke.

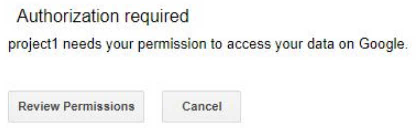
```
function myFunction() {
  var doc = DocumentApp.create('Document Name');
  Logger.log(doc.getId());
}
```



```
function myFunction() {
  var doc = DocumentApp.create('Document Name');
  Logger.log(doc.
}
addBookmark(Position position) : Bookmark
addEditor(String emailAddress) : Document
addEditor(User user) : Document
addEditors(String[] emailAddresses) : Document
addFooter() : FooterSection
addHeader() : HeaderSection
addNamedRange(String name, Range range) : NamedRange
addViewer(String emailAddress) : Document
```

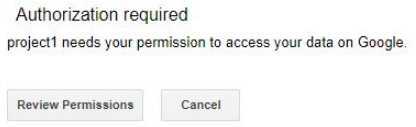


```
function myFunction() {
  var doc = DocumentApp.create('Document Name');
  Logger.log(doc.getId());
}
```



Authorization required

project1 needs your permission to access your data on Google.


Authorization required

project1 needs your permission to access your data on Google.

## Running the Function

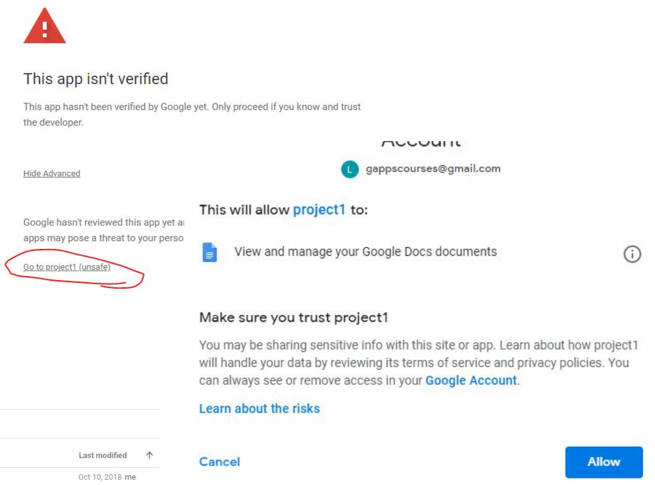
Select to provide permissions in order to run the app and allow it to make changes.

Go to your drive.

[https://drive.google.com/drive/u/0/my-drive](https://drive.google.com/drive/u/0/my-drive)

You will see the new file created there, as well the google script that you just created.

```
function myFunction() {
  var doc = DocumentApp.create('Document Name');
  Logger.log(doc.getId());
}
```


This app isn't verified

This app hasn't been verified by Google yet. Only proceed if you know and trust the developer.

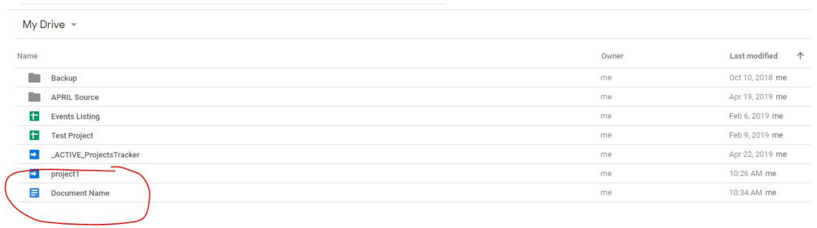
Google hasn't reviewed this app yet a
apps may pose a threat to your perso

This will allow project1 to:

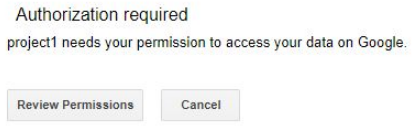
View and manage your Google Docs documents

Make sure you trust project1

You may be sharing sensitive info with this site or app. Learn about how project1 will handle your data by reviewing its terms of service and privacy policies. You can always see or remove access in your Google Account.


My Drive

| Name | Owner | Last modified |
|---|---|---|
| Backup | me | Oct 10, 2018 me |
| APRIL Source | me | Apr 19, 2019 me |
| Events Listing | me | Feb 6, 2019 me |
| Test Project | me | Feb 9, 2019 me |
| _ACTIVE_ProjectsTracker | me | Apr 22, 2019 me |
| project1 | me | 10:26 AM me |
| Document Name | me | 10:34 AM me |



Authorization required

project1 needs your permission to access your data on Google.

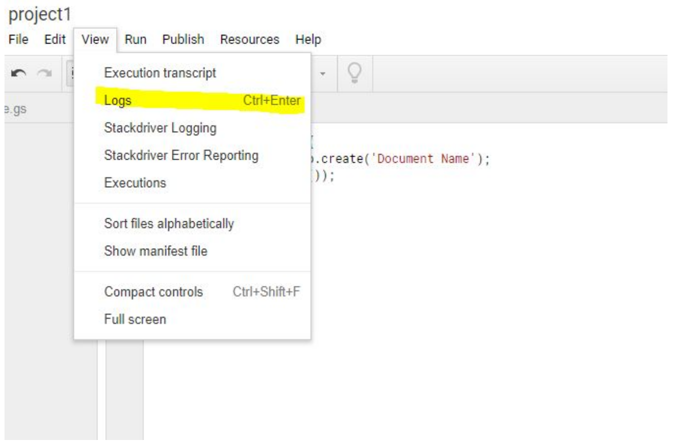
## Checking Logs

Logger.log is similar to console.log in JavaScript - Logs created this way can be viewed by selecting View > Logs in the script editor. These logs are intended for simple checks during development and debugging, and do not persist very long.

[https://developers.google.com/apps-script/guides/logging](https://developers.google.com/apps-script/guides/logging)

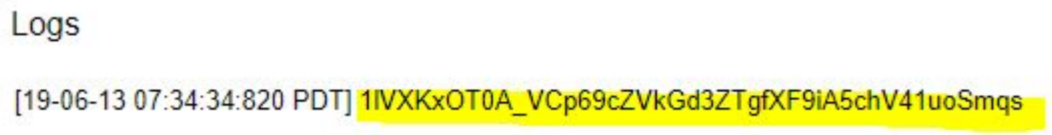
The ID is that of the new document you just created.

```
function myFunction() {
  var doc = DocumentApp.create('Document Name');
  Logger.log(doc.getId());
}
```



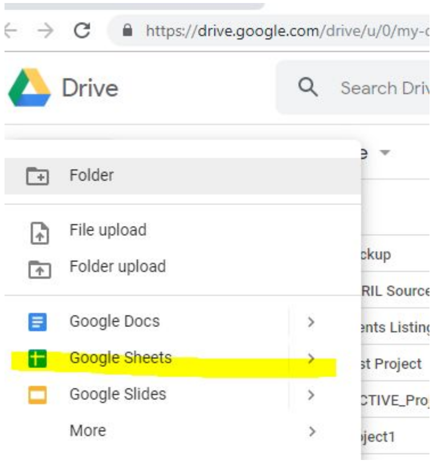
project1
File Edit View Run Publish Resources Help
Execution transcript
Logs Ctrl+Enter
Stackdriver Logging
Stackdriver Error Reporting
Executions
Sort files alphabetically
Show manifest file
Compact controls Ctrl+Shift+F
Full screen
.gs

```
).create('Document Name');
));
```



Logs

```
[19-06-13 07:34:34:820 PDT] 1lVXKxOT0A_VCp69cZVkGd3ZTgfXF9iA5chV41uoSmqs
```



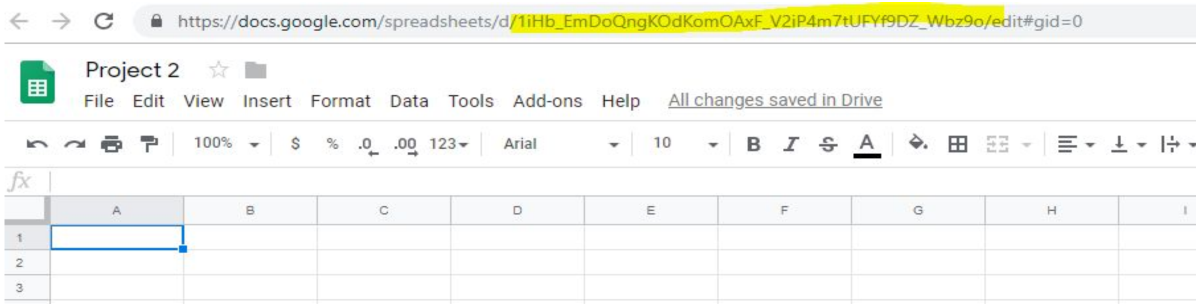
## Google Script Bound Script

Container-bound Scripts - A script is bound to a Google Sheets, Docs, Slides, or Forms file if it was created from that document rather than as a standalone script.  Bound scripts generally behave like standalone scripts except that they do not appear in Google Drive, they cannot be detached from the file they are bound to, and they gain a few special privileges over the parent file.

https://developers.google.com/apps-script/guides/bound

Go to your Google Drive and create a new **Google Sheet.**

Name it project 2 and notice the URL there is a unique ID for the file there.


https://docs.google.com/spreadsheets/d/1iHb_EmDoQngKOdKomOAxF_V2iP4m7tUFYf9DZ_Wbz9o/edit#gid=0
Project 2
File Edit View Insert Format Data Tools Add-ons Help All changes saved in Drive
100% $ % .0 .00 123 Arial 10 B I S A
A B C D E F G H I
1
2
3

## Google Bound Script

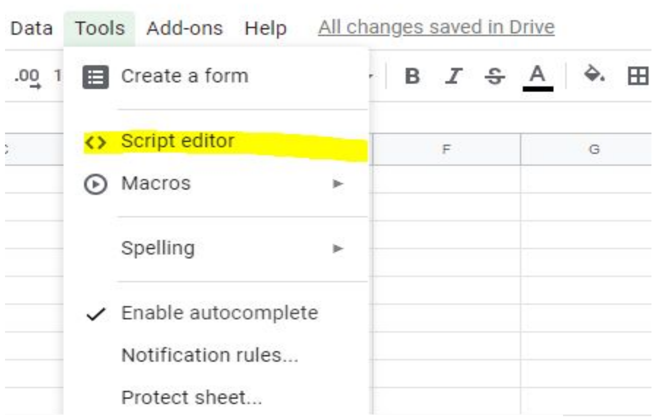
Under tools in the menu bar select **Script editor.**

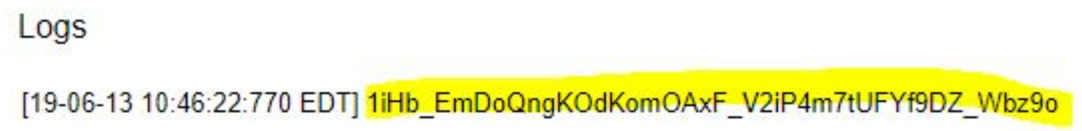
This will open the cloud based editor for Google Script - same as before.

Create the code using getActiveSpreadsheet() and log the id into the Logger.

You'll notice the ID is the same as the id in the URL of the spreadsheet.

```
function myFunction() {
  var sheet =
SpreadsheetApp.getActiveSpreadsheet();
  Logger.log(sheet.getId());
}
```





```
Logs

[19-06-13 10:46:22:770 EDT] 1iHb_EmDoQngKOdKomOAxF_V2iP4m7tUFYf9DZ_Wbz9o
```

## Spreadsheet Service

This service allows scripts to create, access, and modify Google Sheets files

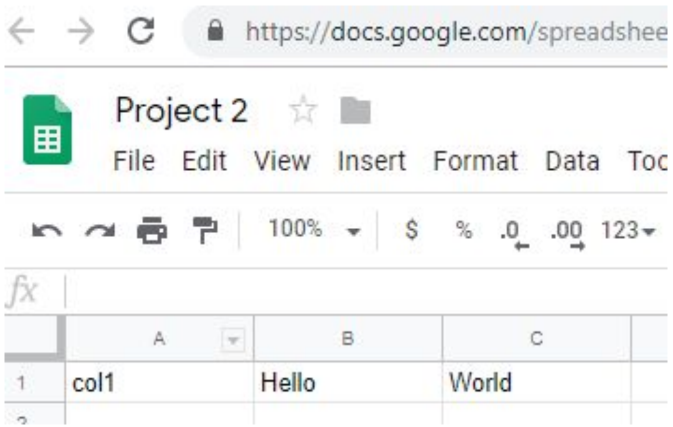
https://developers.google.com/apps-script/reference/spreadsheet/

Select the sheet you want to use.

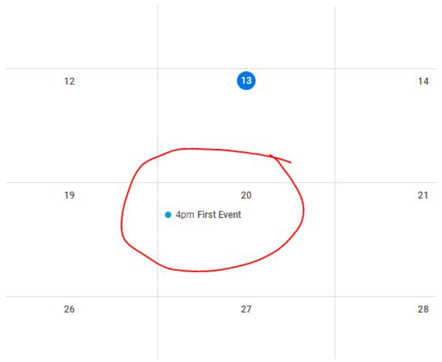
Invoke the function with the code to appendRow using an array.

Go back into the sheet and notice the content contained there.

```
function myFunction() {
  var ss = SpreadsheetApp.getActiveSpreadsheet();
  var sheet = ss.getSheets()[0];
  sheet.appendRow(["col1", "Hello", "World"]);
}
```


https://docs.google.com/spreadshee
Project 2
File Edit View Insert Format Data Toc
100% $ % .0 .00 123
fx

| | A | B | C |
|---|---|---|---|
| 1 | col1 | Hello | World |



## Calendar Service

This service allows a script to access and modify the user's Google Calendar, including additional calendars that the user is subscribed to.

[https://developers.google.com/apps-script/reference/calendar/](https://developers.google.com/apps-script/reference/calendar/)

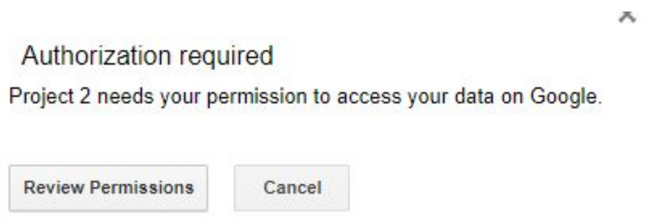
Invoke the new function addEvent()

Allow permissions.

Check your calendar for the date the you entered in the new Date object. You will see an event there.

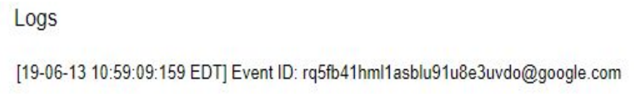
Logs will have the event id.

```
function addEvent(){
  var event =
CalendarApp.getDefaultCalendar().createEvent('First
Event',
  new Date('June 20, 2019 20:00:00 UTC'),
  new Date('June 20, 2019 21:00:00 UTC'));
  Logger.log('Event ID: ' + event.getId());
}
```



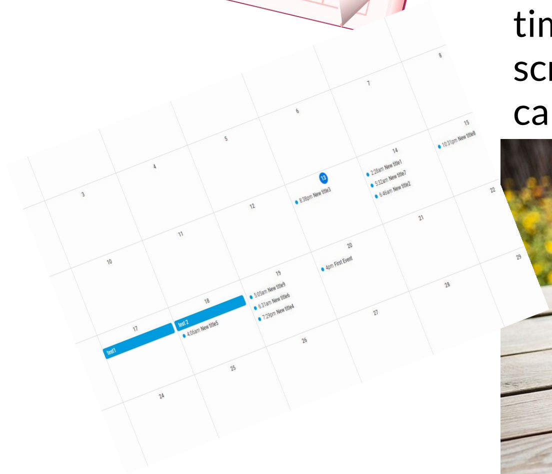
Authorization required

Project 2 needs your permission to access your data on Google.



Logs

```
[19-06-13 10:59:09:159 EDT] Event ID: rq5fb41hml1asblu91u8e3uvdo@google.com
```

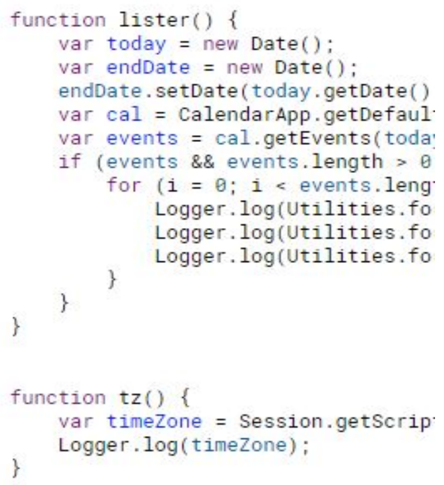


## Populate Calendar

Warning this will add random events to your calendar for the next week, we need some data to use in the calendar.

Generate random start time and end time. Create new event title. Run the script to add some events which we can use later as our feed of events.

```
function addEvents(){
 var cal = CalendarApp.getDefaultCalendar()
 for(var x=0;x<10;x++){
   var startTime = randomDate(new Date(), new Date('June 20, 2019'));
   var endTime = new Date(startTime.getTime() + (2*1000*60*60));
   Logger.log(startTime);
   Logger.log(endTime);
   cal.createEvent('New title'+x, startTime, endTime)
 }

}

function randomDate(start, end) {
   return new Date(start.getTime() + Math.random() * (end.getTime() - start.getTime()));
}
```



```
function lister() {
    var today = new Date();
    var endDate = new Date();
    endDate.setDate(today.getDate()
    var cal = CalendarApp.getDefaul
    var events = cal.getEvents(toda
    if (events && events.length > 0
        for (i = 0; i < events.leng
            Logger.log(Utilities.fo
            Logger.log(Utilities.fo
            Logger.log(Utilities.fo
        }
    }
}

function tz() {
    var timeZone = Session.getScrip
    Logger.log(timeZone);
}
```

## List Events

Create a list of events from your calendar - use a start and end date/time.

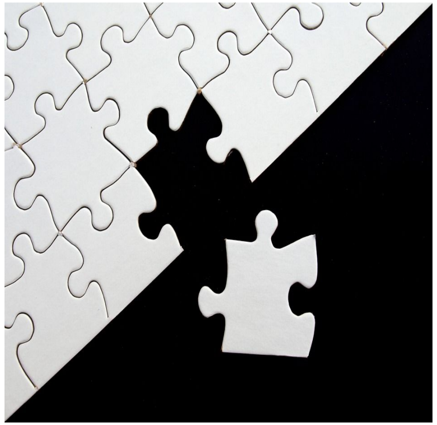
Create an array of data from the event objects.

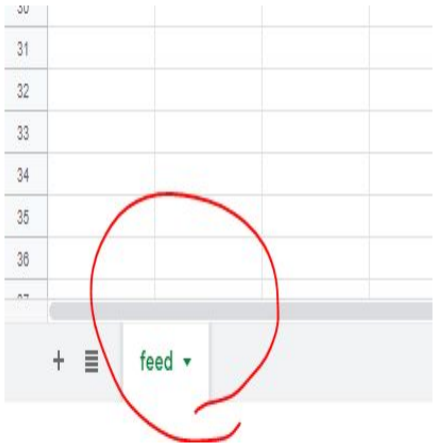
getScriptTimeZone() - Gets the time zone of the script.

https://developers.google.com/apps-script/reference/base/session#getScriptTimeZone()

```
function tz() {
    var timeZone = Session.getScriptTimeZone();
    Logger.log(timeZone);
}

function lister() {
    var today = new Date();
    var endDate = new Date();
    endDate.setDate(today.getDate() + 7);
    var cal = CalendarApp.getDefaultCalendar();
    var events = cal.getEvents(today, endDate);
    if (events && events.length > 0) {
        for (i = 0; i < events.length; i++) {

Logger.log(Utilities.formatDate(events[i].getStartTim
e(), Session.getScriptTimeZone(), "HH:mm"));

Logger.log(Utilities.formatDate(events[i].getEndTime
(), Session.getScriptTimeZone(), "HH:mm"));

Logger.log(Utilities.formatDate(events[i].getStartTim
e(), Session.getScriptTimeZone(), "MMM dd yyyy"));
        }
    }
}
```

```
function listEvents() {
   var today = new Date();
   var endDate = new Date();
   endDate.setDate(today.getDate() + 7);
   var cal = CalendarApp.getDefaultCalendar();
   var events = cal.getEvents(today, endDate);
   var data = [];
   var days = [];
   data.push("Events for today " + Utilities.formatDate(today, Session.getScriptTimeZone(), "MMM dd yyyy"));
   if (events && events.length > 0) {
      for (i = 0; i < events.length; i++) {
         var obj = {
            start: Utilities.formatDate(events[i].getStartTime(), Session.getScriptTimeZone(), "HH:mm")
            , end: Utilities.formatDate(events[i].getEndTime(), Session.getScriptTimeZone(), "HH:mm")
            , date: Utilities.formatDate(events[i].getStartTime(), Session.getScriptTimeZone(), "MMM dd yyyy")
            , title: events[i].getTitle()
            , location: events[i].getLocation()
            , desc: events[i].getDescription()
            , timestamp: Utilities.formatDate(events[i].getStartTime(), Session.getScriptTimeZone(), "yyyy-MM-dd'T'HH:mm:ss'Z'")
         };
         days.push(obj);
         data.push(events[i].getTitle() + ' : ' + Utilities.formatDate(events[i].getStartTime(), Session.getScriptTimeZone(), "HH:mm") + ' - ' +
Utilities.formatDate(events[i].getEndTime(), Session.getScriptTimeZone(), "HH:mm"))
      }
      return days;
   }
   else {
      return ['No events found', '', ''];
   }
}
```





## Add to Spreadsheet

1. Create a sheet named ‘feed’
2. Clear the sheet of existing data
3. Add a row for the header
4. Get the date range and events list
5. Iterate thru the events and get start and end times, add into an array
6. Add the array or row of data to the spreadsheet.

```
function addToSheet() {
    var ss = SpreadsheetApp.getActiveSpreadsheet();
    var sheet = ss.getSheetByName("feed");
  sheet.deleteRows(1, 100);
ss.appendRow(['Start','End','Date','Title','Location','Desc','F
ull']);
    var today = new Date();
    var endDate = new Date();
    endDate.setDate(today.getDate() + 7);
    var cal = CalendarApp.getDefaultCalendar();
    var events = cal.getEvents(today, endDate);
    var data = [];
    var days = [];
    for (i = 0; i < events.length; i++) {
        var rep =
[Utilities.formatDate(events[i].getStartTime(),
Session.getScriptTimeZone(), "HH:mm"),
Utilities.formatDate(events[i].getEndTime(),
Session.getScriptTimeZone(), "HH:mm"),
Utilities.formatDate(events[i].getStartTime(),
Session.getScriptTimeZone(), "MMM dd yyyy"),
events[i].getTitle(), events[i].getLocation(),
events[i].getDescription(),
Utilities.formatDate(events[i].getStartTime(),
Session.getScriptTimeZone(),
"yyyy-MM-dd'T'HH:mm:ss'Z'")];
        var added = ss.appendRow(rep);
    }
}
```

## Create a Trigger

Class ScriptApp - Access and manipulate script publishing and triggers. This class allows users to create script triggers and control publishing the script as a service.

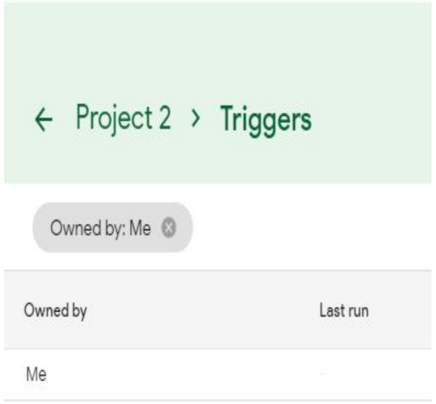
https://developers.google.com/apps-script/reference/script/script-app

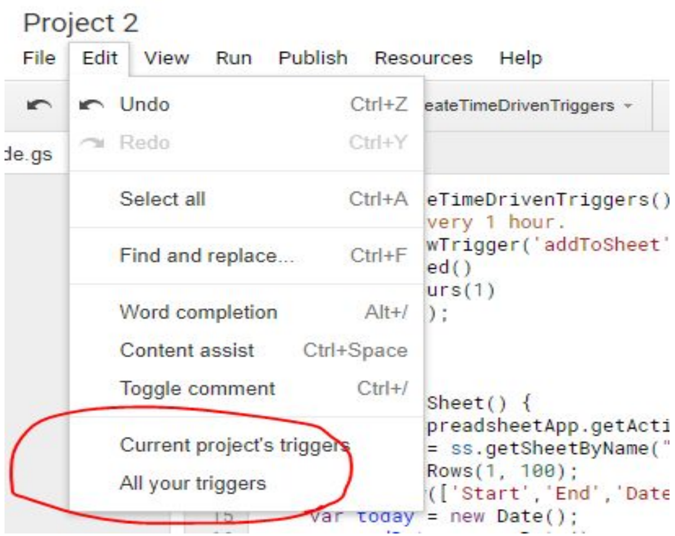
You can see all project triggers under Edit *Current project's triggers* or *All your triggers.*

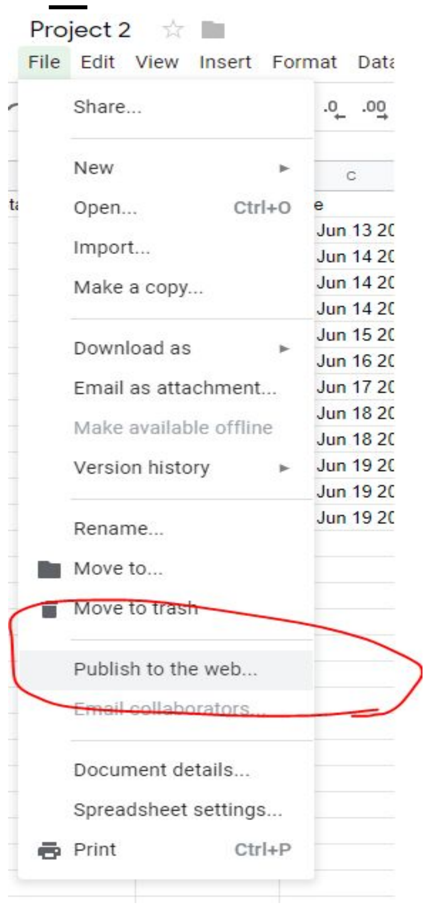
https://script.google.com/u/0/home/triggers

**The function will run every hour.**

```
function createTimeDrivenTriggers() {
  // Trigger every 1 hour.
  ScriptApp.newTrigger('addToSheet')
      .timeBased()
      .everyHours(1)
      .create();
}
```







## Spreadsheet as JSON

Publish your Spreadsheet so it can be used as a JSON data feed.

Open the spreadsheet select in the File menu, go down to ***publish to the web*** option.

Select **Publish** Button

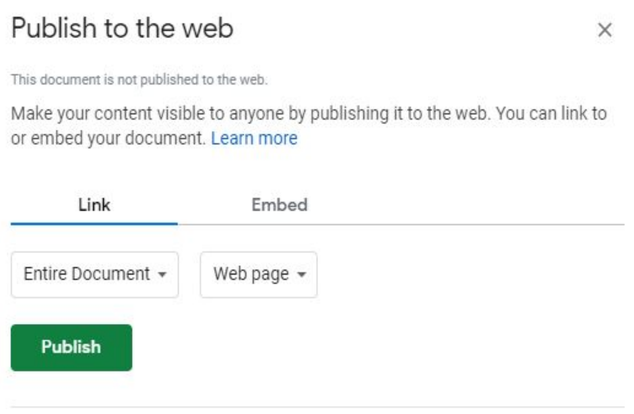
Using your Spreadsheet ID update the below URL with **YOUR ID**.

**https://spreadsheets.google.com/feeds/list/YOURID/1/public/values?alt=json**

**Open in browser the url from above with your ID.**

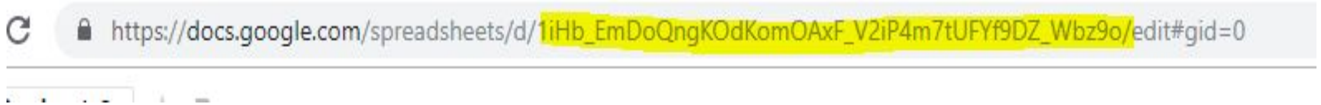
**Get the ID from the spreadsheet URL**

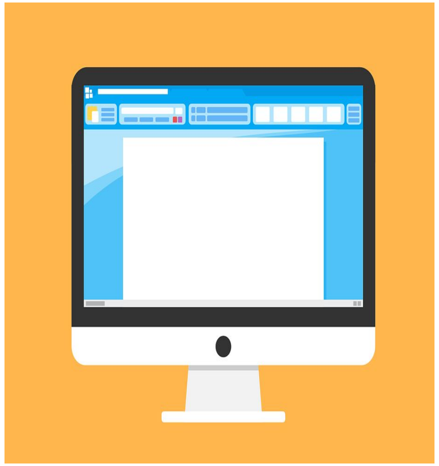
```
function createTimeDrivenTriggers() {
  // Trigger every 1 hour.
  ScriptApp.newTrigger('addToSheet')
      .timeBased()
      .everyHours(1)
      .create();
}
```


Publish to the web

This document is not published to the web.

Make your content visible to anyone by publishing it to the web. You can link to or embed your document. Learn more


https://docs.google.com/spreadsheets/d/1iHb_EmDoQngKOdKomOAxF_V2iP4m7tUFYf9DZ_Wbz9o/edit#gid=0



## JSON data

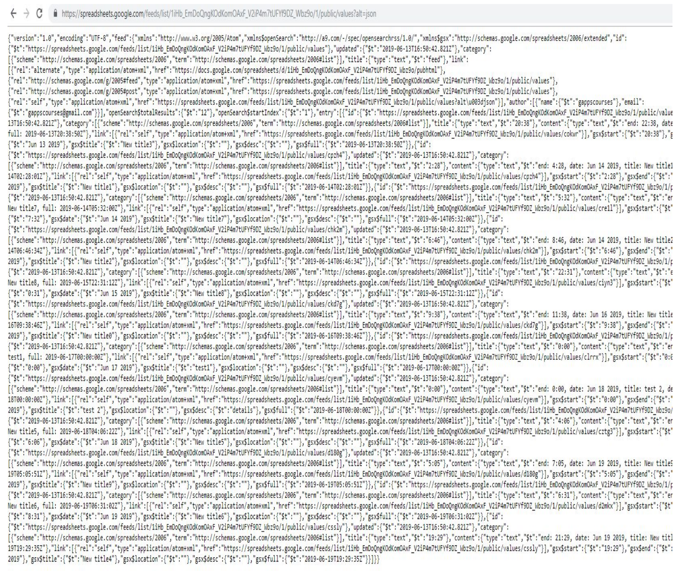
**https://spreadsheets.google.com/feeds/list/YOURID/1/public/values?alt=json**

**Open in browser the url from above with your ID.**

You should be able to go to the URL and see a bunch of JSON data.

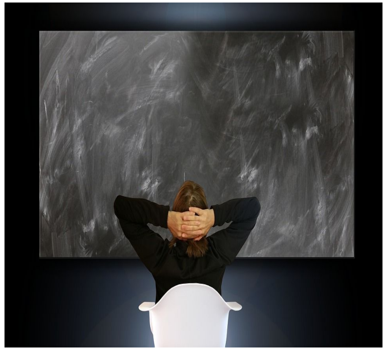
Open your editor and we will make an AJAX request to this URL for the data.

The editor that I use in the course is brackets.io - free open source Adobe product.



https://spreadsheets.google.com/feeds/list/1iHb_EmDoQngKOdKomOAxF_V2iP4m7tUFYf9DZ_Wbz9o/1/public/values?alt=json

```
{"version":"1.0","encoding":"UTF-8","feed":{"xmlns":"http://www.w3.org/2005/Atom","xmlns$openSearch":"http://a9.com/-/spec/opensearchrss/1.0/","xmlns$gsx":"http://schemas.google.com/spreadsheets/2006/extended","id":
{"$t":"https://spreadsheets.google.com/feeds/list/1iHb_EmDoQngKOdKomOAxF_V2iP4m7tUFYf9DZ_Wbz9o/1/public/values"},"updated":{"$t":"2019-06-13T16:50:42.821Z"},"category":
[{"scheme":"http://schemas.google.com/spreadsheets/2006","term":"http://schemas.google.com/spreadsheets/2006#list"}],"title":{"type":"text","$t":"feed"},"link":
[{"rel":"alternate","type":"application/atom+xml","href":"https://docs.google.com/spreadsheets/d/1iHb_EmDoQngKOdKomOAxF_V2iP4m7tUFYf9DZ_Wbz9o/pubhtml"},
{"rel":"http://schemas.google.com/g/2005#feed","type":"application/atom+xml","href":"https://spreadsheets.google.com/feeds/list/1iHb_EmDoQngKOdKomOAxF_V2iP4m7tUFYf9DZ_Wbz9o/1/public/values"},
{"rel":"http://schemas.google.com/g/2005#post","type":"application/atom+xml","href":"https://spreadsheets.google.com/feeds/list/1iHb_EmDoQngKOdKomOAxF_V2iP4m7tUFYf9DZ_Wbz9o/1/public/values"},
{"rel":"self","type":"application/atom+xml","href":"https://spreadsheets.google.com/feeds/list/1iHb_EmDoQngKOdKomOAxF_V2iP4m7tUFYf9DZ_Wbz9o/1/public/values?alt\u003djson"}],"author":[{"name":{"$t":"gappscourses"},"email":
{"$t":"gappscourses@gmail.com"}}],"openSearch$totalResults":{"$t":"12"},"openSearch$startIndex":{"$t":"1"},"entry":[{"id":{"$t":"https://spreadsheets.google.com/feeds/list/1iHb_EmDoQngKOdKomOAxF_V2iP4m7tUFYf9DZ_Wbz9o/1/public/value
13T16:50:42.821Z"},"category":[{"scheme":"http://schemas.google.com/spreadsheets/2006","term":"http://schemas.google.com/spreadsheets/2006#list"}],"title":{"type":"text","$t":"20:38"},"content":{"type":"text","$t":"end: 22:38, date
full: 2019-06-13T20:38:50Z"},"link":[{"rel":"self","type":"application/atom+xml","href":"https://spreadsheets.google.com/feeds/list/1iHb_EmDoQngKOdKomOAxF_V2iP4m7tUFYf9DZ_Wbz9o/1/public/values/cokwr"}],"gsx$start":{"$t":"20:38"},"g
{"$t":"Jun 13 2019"},"gsx$title":{"$t":"New title3"},"gsx$location":{"$t":""},"gsx$desc":{"$t":""},"gsx$full":{"$t":"2019-06-13T20:38:50Z"}},{"id":
{"$t":"https://spreadsheets.google.com/feeds/list/1iHb_EmDoQngKOdKomOAxF_V2iP4m7tUFYf9DZ_Wbz9o/1/public/values/cpzh4"},"updated":{"$t":"2019-06-13T16:50:42.821Z"},"category":
[{"scheme":"http://schemas.google.com/spreadsheets/2006","term":"http://schemas.google.com/spreadsheets/2006#list"}],"title":{"type":"text","$t":"2:28"},"content":{"type":"text","$t":"end: 4:28, date: Jun 14 2019, title: New title1
14T02:28:01Z"},"link":[{"rel":"self","type":"application/atom+xml","href":"https://spreadsheets.google.com/feeds/list/1iHb_EmDoQngKOdKomOAxF_V2iP4m7tUFYf9DZ_Wbz9o/1/public/values/cpzh4"}],"gsx$start":{"$t":"2:28"},"gsx$end":{"$t":"
2019"},"gsx$title":{"$t":"New title1"},"gsx$location":{"$t":""},"gsx$desc":{"$t":""},"gsx$full":{"$t":"2019-06-14T02:28:01Z"}},{"id":{"$t":"https://spreadsheets.google.com/feeds/list/1iHb_EmDoQngKOdKomOAxF_V2iP4m7tUFYf9DZ_Wbz9o/1/p
{"$t":"2019-06-13T16:50:42.821Z"},"category":[{"scheme":"http://schemas.google.com/spreadsheets/2006","term":"http://schemas.google.com/spreadsheets/2006#list"}],"title":{"type":"text","$t":"5:32"},"content":{"type":"text","$t":"er
New title7, full: 2019-06-14T05:32:00Z"},"link":[{"rel":"self","type":"application/atom+xml","href":"https://spreadsheets.google.com/feeds/list/1iHb_EmDoQngKOdKomOAxF_V2iP4m7tUFYf9DZ_Wbz9o/1/public/values/cre1l"}],"gsx$start":{"$t"
{"$t":"7:32"},"gsx$date":{"$t":"Jun 14 2019"},"gsx$title":{"$t":"New title7"},"gsx$location":{"$t":""},"gsx$desc":{"$t":""},"gsx$full":{"$t":"2019-06-14T05:32:00Z"}},{"id":
{"$t":"https://spreadsheets.google.com/feeds/list/1iHb_EmDoQngKOdKomOAxF_V2iP4m7tUFYf9DZ_Wbz9o/1/public/values/chk2m"},"updated":{"$t":"2019-06-13T16:50:42.821Z"},"category":
[{"scheme":"http://schemas.google.com/spreadsheets/2006","term":"http://schemas.google.com/spreadsheets/2006#list"}],"title":{"type":"text","$t":"6:46"},"content":{"type":"text","$t":"end: 8:46, date: Jun 14 2019, title: New title2
14T06:46:34Z"},"link":[{"rel":"self","type":"application/atom+xml","href":"https://spreadsheets.google.com/feeds/list/1iHb_EmDoQngKOdKomOAxF_V2iP4m7tUFYf9DZ_Wbz9o/1/public/values/chk2m"}],"gsx$start":{"$t":"6:46"},"gsx$end":{"$t":"
2019"},"gsx$title":{"$t":"New title2"},"gsx$location":{"$t":""},"gsx$desc":{"$t":""},"gsx$full":{"$t":"2019-06-14T06:46:34Z"}},{"id":{"$t":"https://spreadsheets.google.com/feeds/list/1iHb_EmDoQngKOdKomOAxF_V2iP4m7tUFYf9DZ_Wbz9o/1/p
{"$t":"2019-06-13T16:50:42.821Z"},"category":[{"scheme":"http://schemas.google.com/spreadsheets/2006","term":"http://schemas.google.com/spreadsheets/2006#list"}],"title":{"type":"text","$t":"22:31"},"content":{"type":"text","$t":"e
New title8, full: 2019-06-15T22:31:12Z"},"link":[{"rel":"self","type":"application/atom+xml","href":"https://spreadsheets.google.com/feeds/list/1iHb_EmDoQngKOdKomOAxF_V2iP4m7tUFYf9DZ_Wbz9o/1/public/values/ciyn3"}],"gsx$start":{"$t"
{"$t":"0:31"},"gsx$date":{"$t":"Jun 15 2019"},"gsx$title":{"$t":"New title8"},"gsx$location":{"$t":""},"gsx$desc":{"$t":""},"gsx$full":{"$t":"2019-06-15T22:31:12Z"}},{"id":
{"$t":"https://spreadsheets.google.com/feeds/list/1iHb_EmDoQngKOdKomOAxF_V2iP4m7tUFYf9DZ_Wbz9o/1/public/values/ckd7g"},"updated":{"$t":"2019-06-13T16:50:42.821Z"},"category":
[{"scheme":"http://schemas.google.com/spreadsheets/2006","term":"http://schemas.google.com/spreadsheets/2006#list"}],"title":{"type":"text","$t":"9:38"},"content":{"type":"text","$t":"end: 11:38, date: Jun 16 2019, title: New title
16T09:38:46Z"},"link":[{"rel":"self","type":"application/atom+xml","href":"https://spreadsheets.google.com/feeds/list/1iHb_EmDoQngKOdKomOAxF_V2iP4m7tUFYf9DZ_Wbz9o/1/public/values/ckd7g"}],"gsx$start":{"$t":"9:38"},"gsx$end":{"$t":"
2019"},"gsx$title":{"$t":"New title0"},"gsx$location":{"$t":""},"gsx$desc":{"$t":""},"gsx$full":{"$t":"2019-06-16T09:38:46Z"}},{"id":{"$t":"https://spreadsheets.google.com/feeds/list/1iHb_EmDoQngKOdKomOAxF_V2iP4m7tUFYf9DZ_Wbz9o/1/p
{"$t":"2019-06-13T16:50:42.821Z"},"category":[{"scheme":"http://schemas.google.com/spreadsheets/2006","term":"http://schemas.google.com/spreadsheets/2006#list"}],"title":{"type":"text","$t":"0:00"},"content":{"type":"text","$t":"er
test1, full: 2019-06-17T00:00:00Z"},"link":[{"rel":"self","type":"application/atom+xml","href":"https://spreadsheets.google.com/feeds/list/1iHb_EmDoQngKOdKomOAxF_V2iP4m7tUFYf9DZ_Wbz9o/1/public/values/clrrx"}],"gsx$start":{"$t":"0:0
{"$t":"0:00"},"gsx$date":{"$t":"Jun 17 2019"},"gsx$title":{"$t":"test1"},"gsx$location":{"$t":""},"gsx$desc":{"$t":""},"gsx$full":{"$t":"2019-06-17T00:00:00Z"}},{"id":
{"$t":"https://spreadsheets.google.com/feeds/list/1iHb_EmDoQngKOdKomOAxF_V2iP4m7tUFYf9DZ_Wbz9o/1/public/values/cyevm"},"updated":{"$t":"2019-06-13T16:50:42.821Z"},"category":
[{"scheme":"http://schemas.google.com/spreadsheets/2006","term":"http://schemas.google.com/spreadsheets/2006#list"}],"title":{"type":"text","$t":"0:00"},"content":{"type":"text","$t":"end: 0:00, date: Jun 18 2019, title: test 2, de
18T00:00:00Z"},"link":[{"rel":"self","type":"application/atom+xml","href":"https://spreadsheets.google.com/feeds/list/1iHb_EmDoQngKOdKomOAxF_V2iP4m7tUFYf9DZ_Wbz9o/1/public/values/cyevm"}],"gsx$start":{"$t":"0:00"},"gsx$end":{"$t":"
2019"},"gsx$title":{"$t":"test 2"},"gsx$location":{"$t":""},"gsx$desc":{"$t":"details"},"gsx$full":{"$t":"2019-06-18T00:00:00Z"}},{"id":{"$t":"https://spreadsheets.google.com/feeds/list/1iHb_EmDoQngKOdKomOAxF_V2iP4m7tUFYf9DZ_Wbz9o/
{"$t":"2019-06-13T16:50:42.821Z"},"category":[{"scheme":"http://schemas.google.com/spreadsheets/2006","term":"http://schemas.google.com/spreadsheets/2006#list"}],"title":{"type":"text","$t":"4:06"},"content":{"type":"text","$t":"er
New title5, full: 2019-06-18T04:06:22Z"},"link":[{"rel":"self","type":"application/atom+xml","href":"https://spreadsheets.google.com/feeds/list/1iHb_EmDoQngKOdKomOAxF_V2iP4m7tUFYf9DZ_Wbz9o/1/public/values/cztg3"}],"gsx$start":{"$t"
{"$t":"6:06"},"gsx$date":{"$t":"Jun 18 2019"},"gsx$title":{"$t":"New title5"},"gsx$location":{"$t":""},"gsx$desc":{"$t":""},"gsx$full":{"$t":"2019-06-18T04:06:22Z"}},{"id":
{"$t":"https://spreadsheets.google.com/feeds/list/1iHb_EmDoQngKOdKomOAxF_V2iP4m7tUFYf9DZ_Wbz9o/1/public/values/d180g"},"updated":{"$t":"2019-06-13T16:50:42.821Z"},"category":
[{"scheme":"http://schemas.google.com/spreadsheets/2006","term":"http://schemas.google.com/spreadsheets/2006#list"}],"title":{"type":"text","$t":"5:05"},"content":{"type":"text","$t":"end: 7:05, date: Jun 19 2019, title: New title9
19T05:05:51Z"},"link":[{"rel":"self","type":"application/atom+xml","href":"https://spreadsheets.google.com/feeds/list/1iHb_EmDoQngKOdKomOAxF_V2iP4m7tUFYf9DZ_Wbz9o/1/public/values/d180g"}],"gsx$start":{"$t":"5:05"},"gsx$end":{"$t":"
2019"},"gsx$title":{"$t":"New title9"},"gsx$location":{"$t":""},"gsx$desc":{"$t":""},"gsx$full":{"$t":"2019-06-19T05:05:51Z"}},{"id":{"$t":"https://spreadsheets.google.com/feeds/list/1iHb_EmDoQngKOdKomOAxF_V2iP4m7tUFYf9DZ_Wbz9o/1/p
{"$t":"2019-06-13T16:50:42.821Z"},"category":[{"scheme":"http://schemas.google.com/spreadsheets/2006","term":"http://schemas.google.com/spreadsheets/2006#list"}],"title":{"type":"text","$t":"6:31"},"content":{"type":"text","$t":"er
New title6, full: 2019-06-19T06:31:02Z"},"link":[{"rel":"self","type":"application/atom+xml","href":"https://spreadsheets.google.com/feeds/list/1iHb_EmDoQngKOdKomOAxF_V2iP4m7tUFYf9DZ_Wbz9o/1/public/values/d2mkx"}],"gsx$start":{"$t"
{"$t":"8:31"},"gsx$date":{"$t":"Jun 19 2019"},"gsx$title":{"$t":"New title6"},"gsx$location":{"$t":""},"gsx$desc":{"$t":""},"gsx$full":{"$t":"2019-06-19T06:31:02Z"}},{"id":
{"$t":"https://spreadsheets.google.com/feeds/list/1iHb_EmDoQngKOdKomOAxF_V2iP4m7tUFYf9DZ_Wbz9o/1/public/values/cssly"},"updated":{"$t":"2019-06-13T16:50:42.821Z"},"category":
[{"scheme":"http://schemas.google.com/spreadsheets/2006","term":"http://schemas.google.com/spreadsheets/2006#list"}],"title":{"type":"text","$t":"19:29"},"content":{"type":"text","$t":"end: 21:29, date: Jun 19 2019, title: New titl
19T19:29:35Z"},"link":[{"rel":"self","type":"application/atom+xml","href":"https://spreadsheets.google.com/feeds/list/1iHb_EmDoQngKOdKomOAxF_V2iP4m7tUFYf9DZ_Wbz9o/1/public/values/cssly"}],"gsx$start":{"$t":"19:29"},"gsx$end":{"$t":
2019"},"gsx$title":{"$t":"New title4"},"gsx$location":{"$t":""},"gsx$desc":{"$t":""},"gsx$full":{"$t":"2019-06-19T19:29:35Z"}}]}}
```



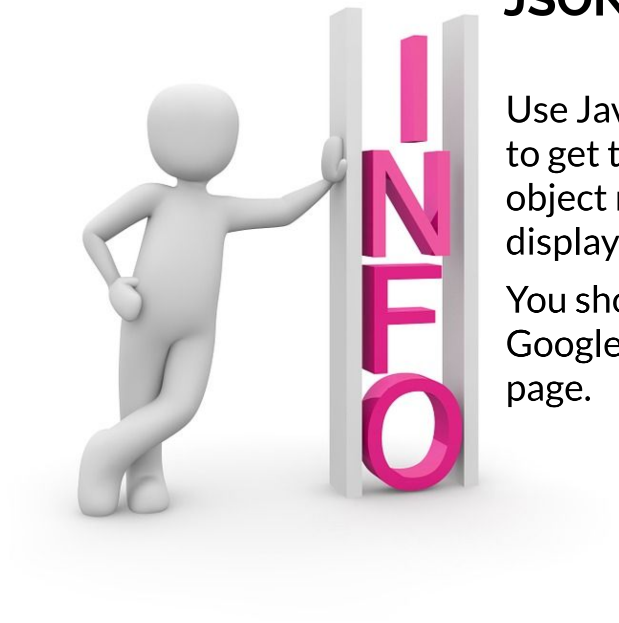
## Setup HTML - AJAX request

JavaScript to make AJAX request to URL with JSON data.

```
<div id="ev"> </div>
<script>
const feedID =
"1wt2Xj5LLYtFa_8PwmbDEpLuUOMz_qlgDDHxF_8wjaYc";
const ev = document.getElementById('ev');
      document.addEventListener('DOMContentLoaded',
test);

function test() {
         const url =
"https://spreadsheets.google.com/feeds/list/" + feedID +
"/1/public/values?alt=json";
         fetch(url).then(function (res) {
            return res.json()
         }).then(function (data) {
            console.log(data);
         })
 }
</script>
```

Update function to parse JSON data.

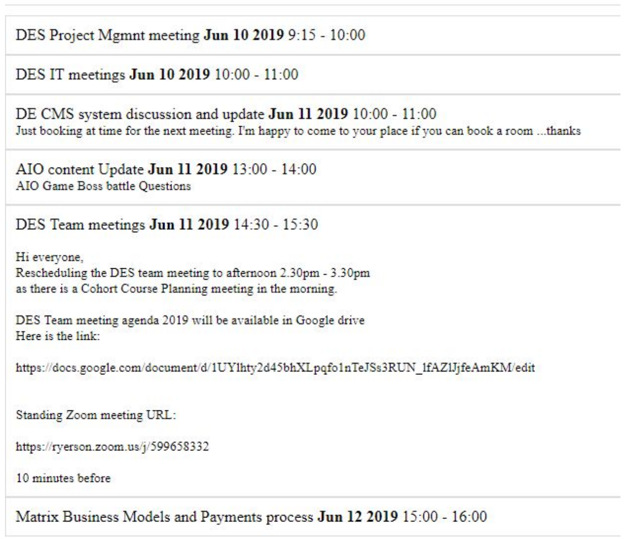
```
function test() {
   const url = "https://spreadsheets.google.com/feeds/list/" +
feedID + "/1/public/values?alt=json";
   fetch(url).then(function (res) {
      return res.json()
   }).then(function (data) {
      let tempArray = [];
      let sheetName = data.feed.title.$t;
      data.feed.entry.forEach(function (element) {
         let holder = {};
         for (let key in element) {
            if (key.substring(0, 3) == "gsx") {
               holder[key.split('$')[1]] = element[key].$t;
            };
         }
         tempArray.push(holder);
      })
      console.log(tempArray);
   })
}
```



## JSON data

Use JavaScript client side in the editor to get the data, then using document object model create elements that can display the data from the JSON feed.

You should see the data from your Google Calendar displayed in the page.



DES Project Mgmnt meeting **Jun 10 2019** 9:15 - 10:00

DES IT meetings **Jun 10 2019** 10:00 - 11:00

DE CMS system discussion and update **Jun 11 2019** 10:00 - 11:00
Just booking at time for the next meeting. I'm happy to come to your place if you can book a room ...thanks

AIO content Update **Jun 11 2019** 13:00 - 14:00
AIO Game Boss battle Questions

DES Team meetings **Jun 11 2019** 14:30 - 15:30

Hi everyone,
Rescheduling the DES team meeting to afternoon 2.30pm - 3.30pm
as there is a Cohort Course Planning meeting in the morning.

DES Team meeting agenda 2019 will be available in Google drive
Here is the link:

https://docs.google.com/document/d/1UY1hty2d45bhXLpqfo1nTeJSs3RUN_lfAZ1JjfeAmKM/edit

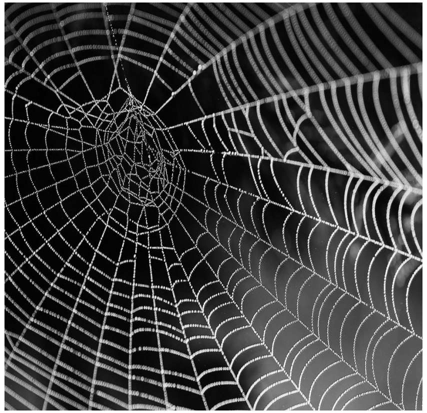
Standing Zoom meeting URL:

https://ryerson.zoom.us/j/599658332

10 minutes before

Matrix Business Models and Payments process **Jun 12 2019** 15:00 - 16:00

```
function loadOutput(myData) {
    console.log(myData);
    for (var i = 1; i < myData.length; i++) {
        console.log(myData[i]);
        const div = document.createElement('div');
        div.innerHTML = myData[i].title + ' <b>' + myData[i].date + ' </b> ' + myData[i].start + ' - ' + myData[i].end + '<br><small>' + myData[i].desc + '</small>';
        ev.appendChild(div);
    }
}

function loadJSON() {
    const url = "https://spreadsheets.google.com/feeds/list/" + feedID + "/1/public/values?alt=json";
    fetch(url).then(function (res) {
        return res.json()
    }).then(function (data) {
        let tempArray = [];
        let sheetName = data.feed.title.$t;
        data.feed.entry.forEach(function (element) {
            let holder = {};
            for (let key in element) {
                if (key.substring(0, 3) == "gsx") {
                    holder[key.split('$')[1]] = element[key].$t;
                };
            }
            tempArray.push(holder);
        })
        loadOutput(tempArray);
    })
}
```



## Google Script WebApp

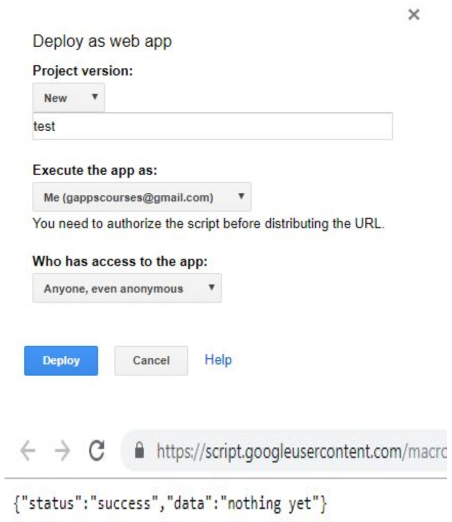
If you build a user interface for a script, you can publish the script as a web app.

https://developers.google.com/apps-script/guides/web

Contains a **doGet(e)** or **doPost(e)** function.

The function returns an HTML service HtmlOutput object or a Content service TextOutput object.

```
function doGet(e) {
   var output = JSON.stringify({
      status: 'success'
      , data: 'nothing yet'
   , });
   return
ContentService.createTextOutput(output).setMimeType(C
ontentService.MimeType.JSON);
}
```


Deploy as web app

Project version: New

test

Execute the app as: Me (gappscourses@gmail.com)

You need to authorize the script before distributing the URL.

Who has access to the app: Anyone, even anonymous

https://script.googleusercontent.com/macro

```
{"status":"success","data":"nothing yet"}
```

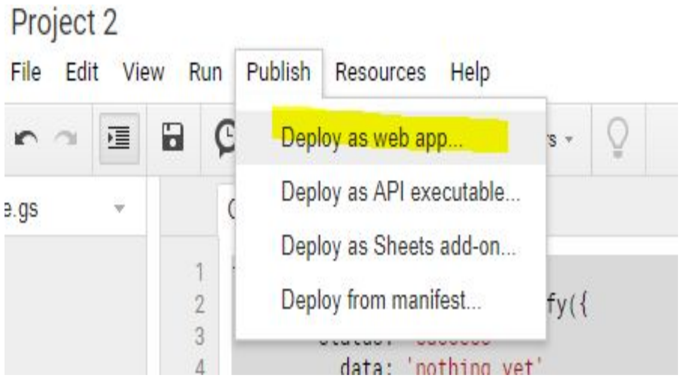
## Deploy WebApp

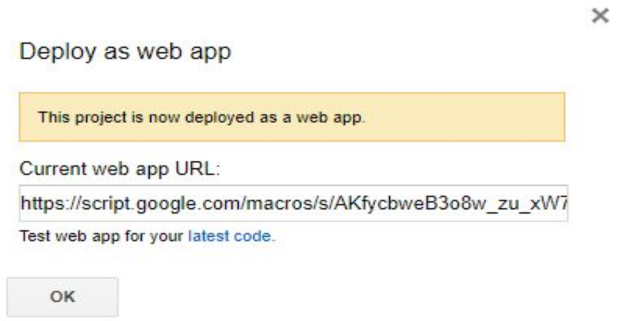
Select the ***deploy as web app*** option under the publish tab.

Add a name and select to execute as your account, and allow access to Anyone even anonymous.

Click **DEPLOY**

Select the app URL to get the exec web app location, while developing you can use the Test web app for your latest code link. This will update immediately once code changes whereas the App URL only when its redeployed.



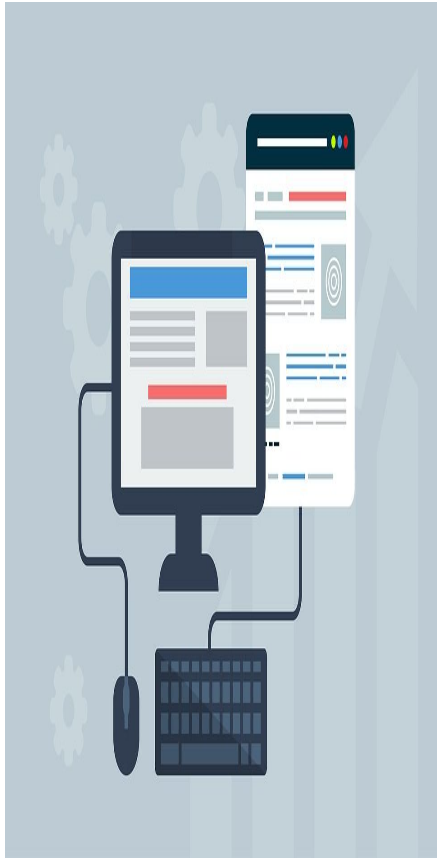

Deploy as web app

This project is now deployed as a web app.

Current web app URL:

https://script.google.com/macros/s/AKfycbweB3o8w_zu_xW7

Test web app for your latest code.



## Deploy WebApp

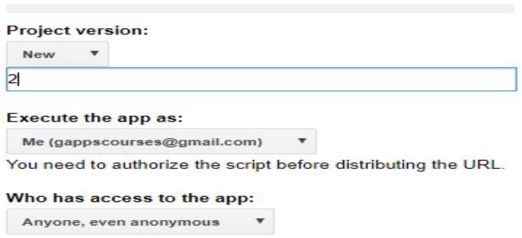
Update data to listEvents() to return the events from the calendar.

Should list all the calendar events from the range as JSON data.

Try the code in the dev url, then publish a new version of the app try it in the published app version.

https://script.google.com/macros/s/**id**/exec

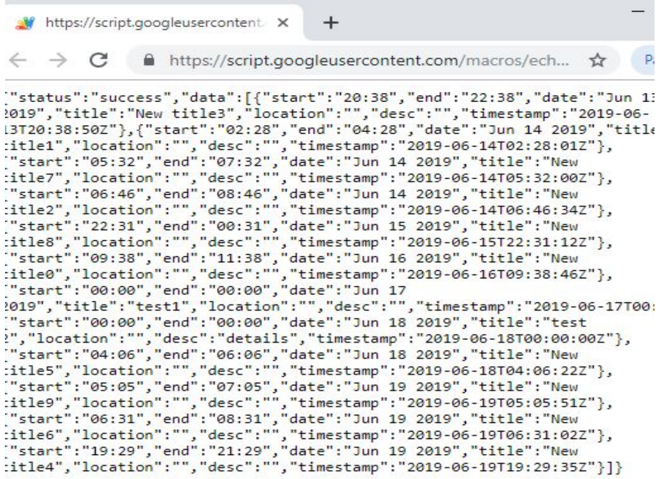
```
function doGet(e) {
  var output = JSON.stringify({
    status: 'success'
    , data: listEvents()
  , });
  return
ContentService.createTextOutput(output).setMimeType(C
ontentService.MimeType.JSON);
}
```



Project version:
New
2

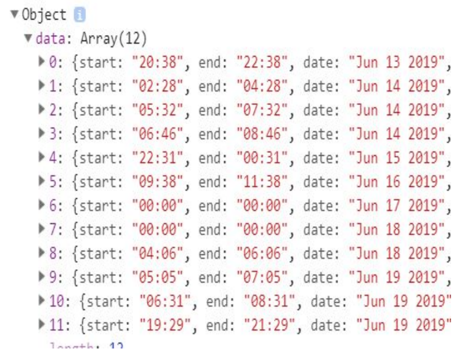
Execute the app as:
Me (gappscourses@gmail.com)
You need to authorize the script before distributing the URL.

Who has access to the app:
Anyone, even anonymous


https://script.googleusercontent
https://script.googleusercontent.com/macros/ech...

```
"status":"success","data":[{"start":"20:38","end":"22:38","date":"Jun 1
019","title":"New title3","location":"","desc":"","timestamp":"2019-06-
3T20:38:50Z"},{"start":"02:28","end":"04:28","date":"Jun 14 2019","titl
itle1","location":"","desc":"","timestamp":"2019-06-14T02:28:01Z"},
"start":"05:32","end":"07:32","date":"Jun 14 2019","title":"New
itle7","location":"","desc":"","timestamp":"2019-06-14T05:32:00Z"},
"start":"06:46","end":"08:46","date":"Jun 14 2019","title":"New
itle2","location":"","desc":"","timestamp":"2019-06-14T06:46:34Z"},
"start":"22:31","end":"00:31","date":"Jun 15 2019","title":"New
itle8","location":"","desc":"","timestamp":"2019-06-15T22:31:12Z"},
"start":"09:38","end":"11:38","date":"Jun 16 2019","title":"New
itle0","location":"","desc":"","timestamp":"2019-06-16T09:38:46Z"},
"start":"00:00","end":"00:00","date":"Jun 17
019","title":"test1","location":"","desc":"","timestamp":"2019-06-17T00
"start":"00:00","end":"00:00","date":"Jun 18 2019","title":"test
","location":"","desc":"details","timestamp":"2019-06-18T00:00:00Z"},
"start":"04:06","end":"06:06","date":"Jun 18 2019","title":"New
itle5","location":"","desc":"","timestamp":"2019-06-18T04:06:22Z"},
"start":"05:05","end":"07:05","date":"Jun 19 2019","title":"New
itle9","location":"","desc":"","timestamp":"2019-06-19T05:05:51Z"},
"start":"06:31","end":"08:31","date":"Jun 19 2019","title":"New
itle6","location":"","desc":"","timestamp":"2019-06-19T06:31:02Z"},
"start":"19:29","end":"21:29","date":"Jun 19 2019","title":"New
itle4","location":"","desc":"","timestamp":"2019-06-19T19:29:35Z"}]}
```

## Client Side get JSON data



```
▼Object
 ▼data: Array(12)
  ▶0: {start: "20:38", end: "22:38", date: "Jun 13 2019",
  ▶1: {start: "02:28", end: "04:28", date: "Jun 14 2019",
  ▶2: {start: "05:32", end: "07:32", date: "Jun 14 2019",
  ▶3: {start: "06:46", end: "08:46", date: "Jun 14 2019",
  ▶4: {start: "22:31", end: "00:31", date: "Jun 15 2019",
  ▶5: {start: "09:38", end: "11:38", date: "Jun 16 2019",
  ▶6: {start: "00:00", end: "00:00", date: "Jun 17 2019",
  ▶7: {start: "00:00", end: "00:00", date: "Jun 18 2019",
  ▶8: {start: "04:06", end: "06:06", date: "Jun 18 2019",
  ▶9: {start: "05:05", end: "07:05", date: "Jun 19 2019",
  ▶10: {start: "06:31", end: "08:31", date: "Jun 19 2019"
  ▶11: {start: "19:29", end: "21:29", date: "Jun 19 2019"
   length: 12
```

Open your editor, create a fetch request to the webapp URL.

Return the data from the webapp.

This is live data which queries the calendar event every time the request is made.

You can parse this object easier than the spreadsheet data and use it to update content on your webpage.

```
const appID =
"AKfycbweB3o8w_zu_xW7DrLAnZ_FCTSyMHMyUO
mbF16tsINXrSWixtTN";
document.addEventListener('DOMContentLoaded',
api);

function api() {
    const url =
'https://script.google.com/macros/s//exec';
    fetch(url).then(function (res) {
        return res.json()
    }).then(function (data) {
        console.log(data);
    })
}
```

# Congratulations on completing the course!

## Thank you for your support

Check out more about JavaScript at MDN. https://developer.mozilla.org/en-US/docs/Web/JavaScript

Find out more about my courses at http://discoveryvip.com/

Course instructor : Laurence Svekis - providing online training to over 500,000 students across hundreds of courses and many platforms.