



FREE eBook

LEARNING

google-spreadsheet

Free unaffiliated eBook created from
Stack Overflow contributors.

#google-

spreadsheet

Table of Contents

About.....	1
Chapter 1: Getting started with google-spreadsheet.....	2
Remarks.....	2
Examples.....	2
Installation or Setup.....	2
The Website.....	2
Chapter 2: Add a Google Form to a web page.....	3
Introduction.....	3
Remarks.....	3
Examples.....	3
Build a Google form.....	3
Embed the Google form.....	3
Add Button and Dialog to html.....	4
Add an event listener to the button.....	4
Manage the dialog box and google form iframe.....	4
Chapter 3: Query Function.....	6
Remarks.....	6
Official Documentation.....	6
Google Docs editors Help.....	6
Google Charts on Google Developers.....	6
Examples.....	6
Introduction into queries.....	6
Sorting with QUERY().....	7
Filtering with QUERY().....	7
Filter a query by an aggregation result.....	8
Chapter 4: Using arrays in Google Sheets.....	9
Syntax.....	9
Parameters.....	9
Remarks.....	9
Overview.....	9

Official Documentation	9
Google Docs editors Help.....	9
Examples.....	9
Array of literals.....	9
Returning a range as an array.....	10
Append column with row numbering.....	10
Credits	12

About

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [google-spreadsheet](#)

It is an unofficial and free google-spreadsheet ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official google-spreadsheet.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

Chapter 1: Getting started with google-spreadsheet

Remarks

Google Spreadsheet or Google Sheets can help you collaborate with teammates! With their built-in chat and many more features, you can edit and complete projects together. With Google Script and functions, mathematical equations can be done within Google Sheets, just like Excel. A cheaper version, and arguably better version, why opt out?

Examples

Installation or Setup

Google Sheets is a spreadsheet application that runs on web browser. It doesn't require any installation or setup just a Google account and a modern web browser.

The Website

Visit <http://sheets.google.com> to try out now! Create a new spreadsheet and start playing around.

Read [Getting started with google-spreadsheet online](https://riptutorial.com/google-spreadsheet/topic/4910/getting-started-with-google-spreadsheet): <https://riptutorial.com/google-spreadsheet/topic/4910/getting-started-with-google-spreadsheet>

Chapter 2: Add a Google Form to a web page

Introduction

Google Spreadsheets has a powerful add on called [Google Forms](#) that allows a web developer to add simple forms easily to web sites in order to collect data from users.

This article discusses the way to embed these into a web application.

I've also created a [Youtube video](#) with a running commentary, screenshots and so on.

Remarks

The examples above are adapted from a fully functional site and this article assumes a reasonable existing knowledge of HTML/Javascript/CSS in order to use these code snippets.

Examples

Build a Google form

Log into a Google Account and click New > More > Google Forms.

Build the form fields required using the editor.

If the form was built with an account that is part of an organisation then click on the cog and unselect the option that only members can complete the form.

Set the form to save the responses to a spreadsheet by clicking on the Responses tab, and click the spreadsheet icon. The popup provides the option to save this form data to a new or existing spreadsheet. By selecting existing it allows multiple forms per spreadsheet. Follow the prompts to complete this task. This is a good time to save some test data to make sure it is all working.

Optionally the web app may wish to set some pre-filled responses in the fields. If that is the case go back to the form and click on the three dots dropdown menu, then click `Get pre-filled link`. This will load the form in a special mode where fields can be completed without submitting the data. When completing the fields use the label name as the prefilled value. Then save the URL which will have parameters similar to `entry.123=labelname1&entry.456=labelname2`. Save a copy of that URL for later.

Embed the Google form

This is done by adding a button, dialog box and iframe as explained below.

The examples below use [MDL](#) for look and feel because it is used by Google forms and so it makes the additional elements look fairly seamless.

Dialog boxes may require a [polyfill](#) if you plan to support older browsers.

Add Button and Dialog to html

```
<button id="googleFormButton" class="mdl-button mdl-js-button mdl-button--raised">
  Load Form
</button>
```

```
<dialog id="googleFormsDialog" class="mdl-dialog">
  <!-- <h4 class="mdl-dialog__title">Google Form</h4> -->
  <div id="googleformparent" class="mdl-dialog__content">
    <div id="googleFormsDialogIFrameLoading">Loading...</div>
    <!-- IFrame element googleFormsDialogIFrame is added dynamically due to google forms popup
  issue. -->
  </div>
  <div class="mdl-dialog__actions">
    <button id="dialogclose" type="button" class="mdl-button">Close</button>
  </div>
</dialog>
```

Add an event listener to the button.

The value of GOOGLE-FORM-PREFILLED-URL should look something like this:

https://docs.google.com/forms/.../?usp=pp_url&entry.1739003583=labelname1

```
jQuery('#googleFormButton').click(showGoogleForm)
jQuery('#googleFormButton').attr('googleFormsURL', 'GOOGLE-FORM-PREFILLED-URL')
```

Manage the dialog box and google form iframe

Add a new function called showGoogleForm and adapt the follow code to suit. Note for simplicity this example does not contain any error checking which should be added in a production environment.

The url should look something like this:

https://docs.google.com/forms/.../?usp=pp_url&entry.1739103583=labelname1

```
var showGoogleForm = function (e) {
  var url = e.currentTarget.googleFormsURL
  url = url.replace('labelname1', 'Some prefilled value')
  url = url.replace('labelname2', 'Another prefilled value')

  // Add the iFrame dynamically to avoid popup issue
  jQuery('<iframe id="#googleform" src="" width="100%" height="100%" frameborder="0"
marginheight="0" marginwidth="0">Loading...</iframe>').appendTo('#googleformparent')

  // Set the prefilled url as the iFrame source
  jQuery('#googleform').attr('src', url)

  // Remove the iframe element when the user closes the dialog to avoid the popup if the
  user did not submit the form.
  jQuery('#dialogclose').click(function(e) {
    jQuery('#googleform').remove()
  })
}
```

```
})  
}
```

Read Add a Google Form to a web page online: <https://riptutorial.com/google-spreadsheet/topic/9909/add-a-google-form-to-a-web-page>

Chapter 3: Query Function

Remarks

Official Documentation

Google Docs editors Help

- [QUERY](#)

Google Charts on Google Developers

- [Query Language Reference \(Version 0.7\)](#)

Examples

Introduction into queries

Source table

Row	A	B	C	D
1	Code	Product	Colour	Price
2	1	pen	red	500
3	2	pen	blue	-50
4	3	pen	red	0
5	4	pencil	blue	17
6	5	pencil	green	-1.5

to select all:

```
= QUERY(A1:D5, "select *")
```

or

```
= QUERY(A1:D5, "select A, B, C, D")
```

or convert data range into array and use this formula:

```
= QUERY({A1:D5}, "select Col1, Col2, Col3, Col4")
```

Sorting with QUERY()

A	B	C	D
1	pen	red	500
2	pen	blue	-50
3	pen	red	0
4	pencil	blue	17
5	pencil	green	-1.5

To sort by column D with "order by":

```
=QUERY("A1:D6","select * order by D desc",1)
```

Filtering with QUERY()

A	B	C	D
1	pen	red	500
2	pen	blue	-50
3	pen	red	0
4	pencil	blue	17
5	pencil	green	-1.5

To only return "pencil" data:

```
=QUERY("A1:D6","select * where B='pencil' ",1)
```

To only return rows that contain "pen" (all rows):

```
=QUERY("A1:D6","select * where B contains 'pen' ",1)
```

To only return rows where the price is greater than 0:

```
=QUERY("A1:D6","select * where D>0 ",1)
```

Note that text strings require apostrophes while numerical values do not.

Filter a query by an aggregation result

```
=QUERY(QUERY(A1:D6,"select C,SUM(D) group by C",1),"select Col2>0",1)
```

Read Query Function online: <https://riptutorial.com/google-spreadsheet/topic/5014/query-function>

Chapter 4: Using arrays in Google Sheets

Syntax

- `={item1,item2}`
- `={item2,item2;item3,item4}`
- `={{item1;item2},{item3,item4;item5,item6}}`

Parameters

Parameter	Details
itemN	It could be a value, a cell reference, a range reference or a function

Remarks

Overview

An array of literals is written between curly brackets. Separators depends on the spreadsheet's regional configuration settings.

- To separate columns, if the decimal separator is `.` use `,` but if the decimal separator is `,` then use `\`.
- To separate rows use `;`.

Official Documentation

Google Docs editors Help

- [Using arrays in Google Sheets](#)

Examples

Array of literals

Formula in A1

```
={"Item name","Quantity";"Apples",2;"Blueberries",5}
```

Important: In certain countries the comma is used as a decimal separator (e.g: €1,00). If that's

your case, you would need to use backslashes (\) instead: ([Docs](#))

```
={"Item name"\ "Quantity"; "Apples"\2; "Blueberries"\5}
```

Result

Row	A	B
1	Item name	Quantity
2	Apples	2
3	Blueberries	5

Returning a range as an array

Row	A	B
1	Fruit	Apple
2	Weekday	Monday
3	Animal	Dog

Formula on C1

```
={A1:A3}
```

Result

Row	C
1	Fruit
2	Weekday
3	Dog

Alternative formula

```
=ARRAYFORMULA(A1:A3)
```

Append column with row numbering

A1:A4 have A,B,C,D.

B1 have the following formula:

```
=ARRAYFORMULA({A1:A4,ROW(A1:A4)})
```

Result

	A	B	C
1	A	A	1
2	B	B	2
3	C	C	3
4	D	D	4

Read Using arrays in Google Sheets online: <https://riptutorial.com/google-spreadsheet/topic/5283/using-arrays-in-google-sheets>

Credits

S. No	Chapters	Contributors
1	Getting started with google-spreadsheet	Community , fourjr , Rubén
2	Add a Google Form to a web page	Simon Hutchison
3	Query Function	Max Makhrov , Rubén , Samantha , Sandy Good
4	Using arrays in Google Sheets	Alfro , Rubén