

# Instructions to send email automatically in Google Sheet with Google Script

Google Script is a powerful tool used to automate many tasks in Google such as Sheets and Docs. And now the ability to automatically send emails makes it a more powerful tool.

Google Script is a powerful tool used to automate many tasks in Google such as Sheets and Docs. And now the ability to automatically send emails makes it a more powerful tool.

You are a monthly employee who has to send dozens of emails to your boss using automatic formulas in Google Sheets or you are a manager who has to spend hours emailing team members, so the Google Script function will Help you complete this task and more.

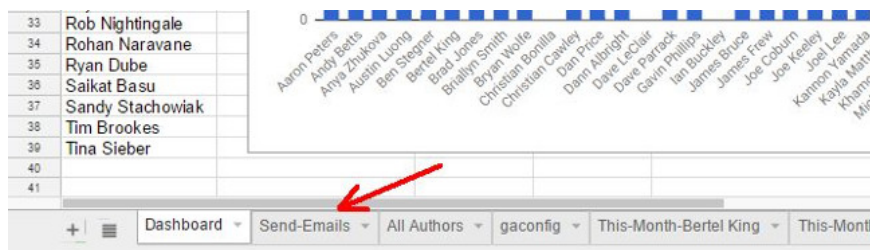
Just take a moment to follow and follow the instructions in this article, you will never have to send data manually again. Let Google Script do 'personal assistant' to do all the work for you.

## Create a sheet to send Email

The first step is to set up a worksheet that includes your name, email address and messages to the people you want to email.

For example, you create a message board to automatically import all articles published in the month from the Google Analytics tool. Then create a sheet "All Authors" that synthesizes published articles for each author, and now sends each author an email with their number of articles.

To get started, create your special email worksheet by adding a new worksheet to the current worksheet and name it "Send emails."



In this new Send emails page, you will create a title. Each row in this spreadsheet will represent a personal email that will be sent. In the above example, the picture includes the recipient's name, email address and message placed inside the script.

	A	B	C	D	E	F	G
1	Name	Email Address	Message 1	Parameter 1	Message 2	Parameter 2	
2							
3							
4							
5							
6							
7							
8							

You can create an email with any size and structure by using columns to merge fixed information and change information.

	B	C	D	E	F	G	H
Address		Message 1	Message 2	Parameter 2	Message 3		
e@makeuseof.com		Here is your monthly title count this month.	You wrote a total of		1 article(s).		
e@makeuseof.com		Here is your monthly title count this month.	You wrote a total of		1 article(s).		
e@makeuseof.com		Here is your monthly title count this month.	You wrote a total of		2 article(s).		

Fixed information is just the text entered in the box and does not change every month. It is only part of an email that is always preserved. However, for the data you want to change each month, you can insert relevant data import functions from any other worksheet in your spreadsheet.

	B	C	D	E	F	G	H
Address		Message 1	Message 2	Parameter 2	Message 3		
akeuseof.com		Here is your monthly title count this month.	You wrote a total of	=This-Month-Aaron Peters'!B4			
akeuseof.com		Here is your monthly title count this month.	You wrote a total of		1 article(s).		
akeuseof.com		Here is your monthly title count this month.	You wrote a total of		2 article(s).		

What we want here is, whenever you open the worksheet to review the report automatically, you can click on the menu item to email it via Google Script and send the results to everyone. You will see the new menu item customization instructions at the end of the article.

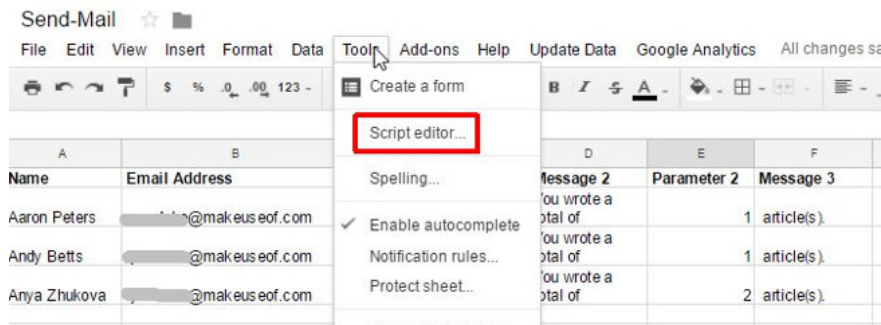
When finished creating the sheet, it will look like the image below:

	A	B	C	D	E	F
1	Name	Email Address	Message 1	Message 2	Parameter 2	Message 3
2	Aaron Peters	@makeuseof.com	Here is your monthly title count this month.	You wrote a total of		1 article(s).
3	Andy Betts	@makeuseof.com	Here is your monthly title count this month.	You wrote a total of		1 article(s).
4	Anya Zhukova	@makeuseof.com	Here is your monthly title count this month.	You wrote a total of		2 article(s).
5						
6						
7						
8						

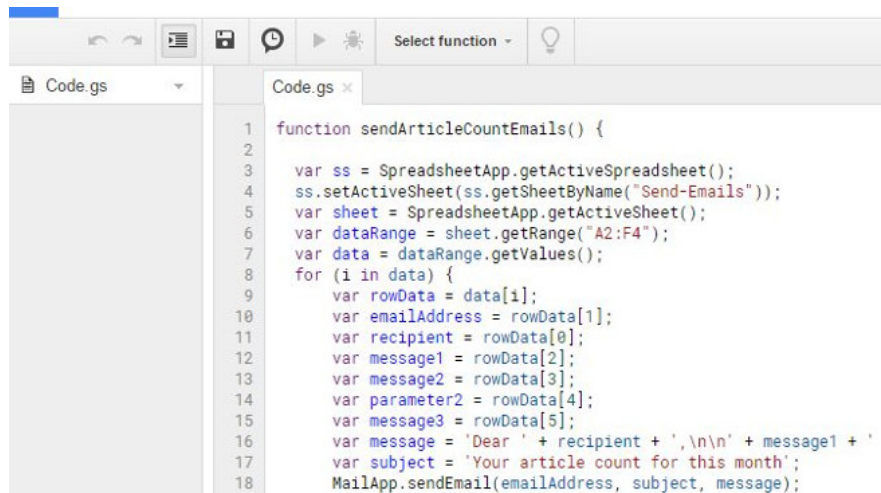
After creating a worksheet with all your personal emails, the next step is to write a script!

## Write automated email scripts

To write script, you need to use the script editor. Select **Tools> Script editor**.



Script editor as shown below. Choose to create a new function at the top right.



Copy and paste this script into the script writing section.

```
function sendArticleCountEmails ( ) { var ss = SpreadsheetApp . getActiveSpreadsheet();
```

This script seems quite complicated, but it's not really all. Let's "dissect" it to know exactly what each line says.

## Split Code

In order for this function to work properly, you need to make sure that the worksheet stores all of your email information as the current sheet. If not, the following steps cannot be performed. That's why these two lines of code appear:

```
var ss = SpreadsheetApp . getActiveSheet ( ) ; ss . setActiveSheet ( ss . getSheetByName ( 'Sheet1' ) ) ;
```

Next, extract all the data from the worksheet. The **sheet.GetRange ()** function will retrieve information from any range that you include in quotation marks. Next, the **dataRange.getValues ()** function will extract the values and store them into a two-dimensional array called **data**.

```
var sheet = SpreadsheetApp . getActiveSheet ( ) ; var dataRange = sheet . getRange ( 'A1:B10' ) ; var data = dataRange . getValues ( ) ;
```

Now all data is stored in an array, you can repeat using a for loop. Each element of the array is a row containing a one-dimensional array of column elements. You can reference each column by number.

```
var rowData = data [ i ] ; var emailAddress = rowData [ 1 ] ; var recipient = rowData [ 2 ] ;
```

As you can see above, the column elements are pulled into a one-dimensional array called **rowData**. Then, go to the second column (to get the email address) by referring to **rowData [1]**, the second element of the array (the first element of the array is always zero).

The next step is to merge all the segments of the message into the content of the email. You can edit the format of the email content by using the letter **n** to return to the beginning of the line. This is the line of code to merge the message segments:

```
var message = 'Dear ' + recipient + ',\n\n' + message1 + ' ' + message2 + ' ' + parameter ;
```

The **+** character is a join command. You place static text inside single quotes. Therefore, this message is put together by placing the word **"Dear"** in front of the recipient's variable (note **"Dear"** has a space).

Next, you add two **n** characters at the end to make the message into two segments. Next, add the first message segment, space, second message segment, space, and variable that will be included in the message each month and finally the last part of the message.

You can have more messages by using more columns to create the entire message section by section.

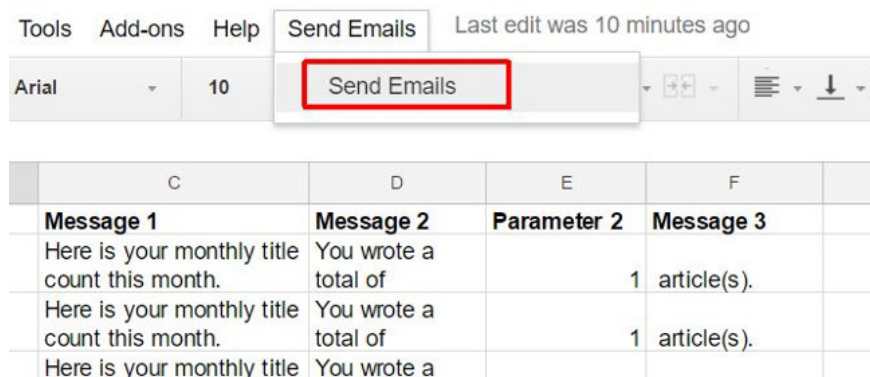
The last two lines of code to title the email (can include data from the spreadsheet if you want), and finally **sendEmail ()**.

## How to activate the script

Go back to the Script Editor window, just search for the **"onOpen"** function. If this function is available, you will see a series of code inside parentheses. Scroll to the bottom of the code and insert the following lines:

```
var spreadsheet = SpreadsheetApp . getActiveSheet ( ) ; var menuItems = [ { name : 'Send Email' } ] ;
```

Just be sure to type the name of the function correctly in single quotes for the **functionName** parameter. This will load the custom menu item into the spreadsheet every time you open the page.



Now, when you want to open the monthly spreadsheet to review all data, you just need to click on the item menu to send an email. This will run the new function and send all those emails to you. A job that you can take several hours but now only needs a single click!

## Google Script saves you time

This is just one of many ways you can use Google Script to automate things. Although the initial automation setting is a bit time-consuming, every minute you spend setting it saves you hours later. Just think about the great things that you can do with those hours, and be reluctant to start trying today!

You finished reading the article "**Instructions to send email automatically in Google Sheet with Google Script**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.