

# How to create custom functions in Google Sheets

If you need to go beyond what Google Sheets integrates, creating a custom function would be a useful solution. Custom functions are pieces of code that perform actions on the worksheet.

Google Sheets has a number of useful features for handling numerical calculations, searching, and manipulating strings. If your sheets are more advanced, you may find you need to build complex formulas to get the job done.

If you need to go beyond what Google Sheets integrates, creating a custom function would be a useful solution. Custom functions are pieces of code that perform actions on the worksheet. When writing them, you can name and call them multiple times. This saves a lot of time.

Let's look at how to create a custom function in Google Sheets, using the Google script.

## How to create custom functions in Google Sheets?

1. Functions in Google Sheets
2. Create custom functions in Google Sheets
  1. Open Script Editor
  2. Create function
  3. Save function
3. Use custom functions
4. Use the function again

## Functions in Google Sheets

Google Sheets has quite powerful functions built in, such as **Sum** or **Average**:

fx   =sum(B2:B6)			
	A	B	C
1		<b>Price</b>	
2		\$4.50	
3		\$3.50	
4		\$12.75	
5		\$8.00	
6		\$9.20	
7	<b>Total</b>	\$37.95	
8			

fx   =AVERAGE(B2:B6)			
	A	B	C
1		<b>Price</b>	
2		\$4.50	
3		\$3.50	
4		\$12.75	
5		\$8.00	
6		\$9.20	
7	<b>Total</b>	\$37.95	
8	<b>Average</b>	\$7.59	
9			

Google Sheets has quite powerful functions already built in

What if you wanted to perform a calculation that is not included in the standard functions? Consider a situation in which you want to add sales tax to the price of an item. Because tax rates vary by location, you'll need to build a function with a long list of nested logical functions. It should look like the following:

```
'=if(A2="PA",B2*0.06,if(A2="CA",B2*0.0625,B2*0))'
```

Now imagine if you must add multiple conditions to this statement for each state. Everything will get out of control!

A custom Google Sheets function can handle all these jobs. You can put all the complex code into a script, name it and call the function. There is no bulky code in Google Sheet, it's just a simple function like Sum.

Learning how to create custom functions opens up a whole new world. So let's start right away!

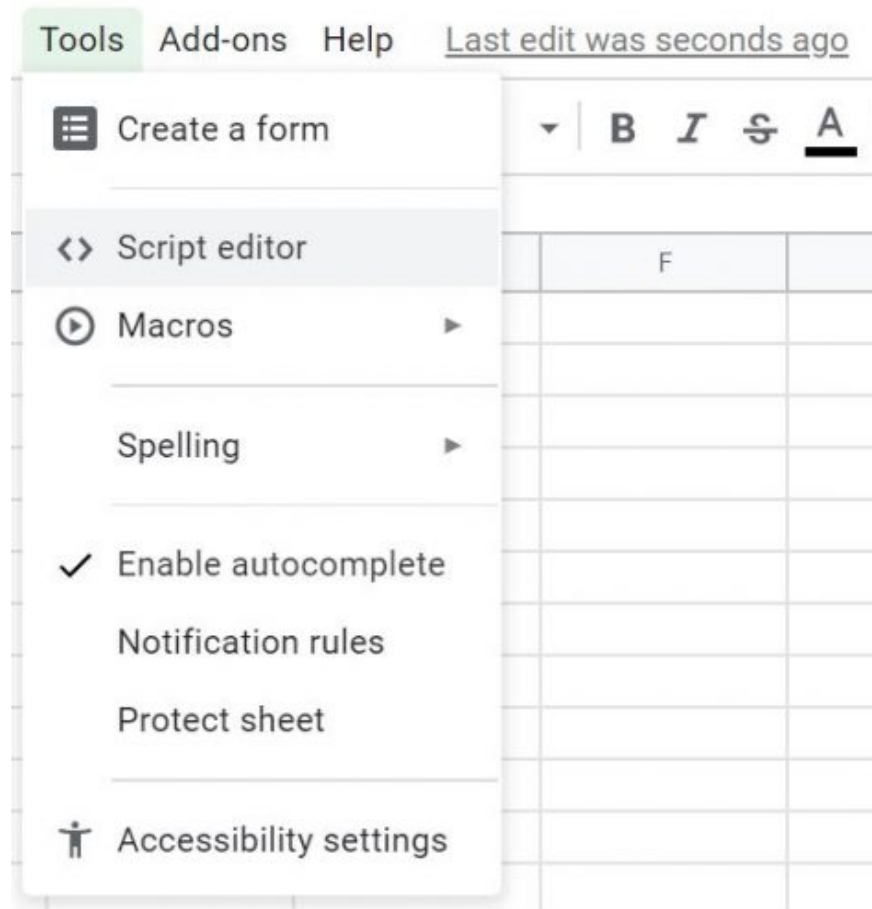
## Create custom functions in Google Sheets

If you are new to script creation, don't be afraid! Everything is very easy

The custom functions for Google Sheets are written in JavaScript code. If you are a JavaScript expert, you will feel very comfortable. If not, then this is one of the simple programming languages ??that you can start learning right now.

## Open Script Editor

Open **Google Sheets** and select **Tools> Script Editor** .



Open Script Editor

## Create function

You will want to give your function a simple, but specify name of what the function will do.

The inputs you want to use will be in parentheses as variables. This will be the cell value you want to work with. If there is more than one cell value, you can separate them with a comma.

To use this tax example, you can copy and paste this code into the Script Editor:

```
function tax(input, location) { var rate = 0 ; switch (location) { case 'PA': ra
```

This function called **tax** will calculate the tax rate based on the price, depending on the location you enter the function. This is the assumed tax rate.

```
* Code.gs x
1 function tax(input, location) {
2   var rate = 0 ;
3   switch (location) {
4     case 'PA':
5       rate = 0.06;
6       break;
7     case 'CA':
8       rate = 0.0625;
9       break;
10    default:
11      rate = 0;
12    }
13    return (input * rate);
14  }
15
16
```

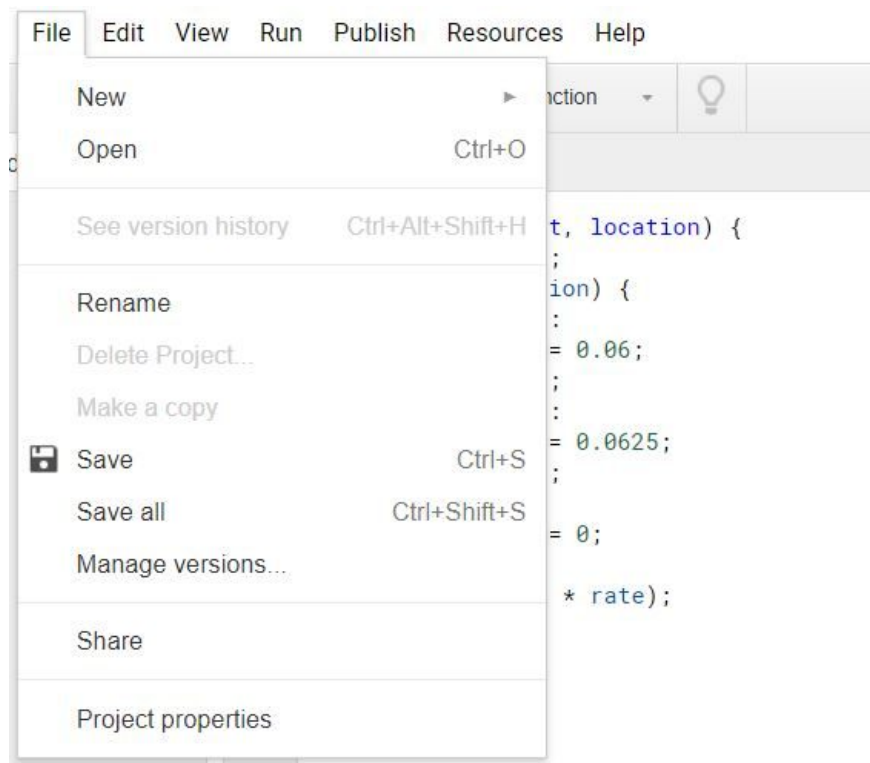
### Presumptive tax rate

Script will occupy 2 cells. One cell is assigned to **input**, the other is **location**. The code will run to determine the status you want to calculate and return the tax amount.

The article only covers 2 locations in this example to give you an overview. You can add more by adding additional lines with the positions you need.

### Save function

Choose **File**> **Save** , name the project and click **OK**.



Save function

## Use custom functions

After creating your function, you can use it just like the built-in functions. In the cell where you want your calculation to appear, enter an = sign followed by the function name.

For example, in this tax example, we are using 2 **inputs** . **Location** will determine the tax rate and price of the product to which it is applied:

=tax(B2, A2)

Where **B2** is the price of the product and **A2** is the taxable position.

	A	B	C	D
1	<b>Location</b>	<b>Price</b>	<b>Tax</b>	
2	PA	\$4.50	\$0.27	
3	CA	\$3.50		
4	PA	\$12.75		
5		\$8.00		
6		\$9.20		
7	<b>Total</b>	\$37.95		
8	<b>Average</b>	\$7.59		
9				

For example

You can use AutoFill like Excel to drag and drop your function into all rows, the same way you use an integrated function:

	A	B	C
1	<b>Location</b>	<b>Price</b>	<b>Tax</b>
2	PA	\$4.50	\$0.27
3	CA	\$3.50	\$0.22
4	PA	\$12.75	\$0.77
5		\$8.00	
6		\$9.20	
7	<b>Total</b>	\$37.95	
8	<b>Average</b>	\$7.59	
9			

You can use AutoFill just like Excel

After creating your first custom function, you can get a few more things you want. Adding code to the script is easy. Follow the steps above to create a new function the same way and add them below the existing code.

```
Code.gs x
1 function tax(input, location) {
2   var rate = 0 ;
3   switch (location) {
4     case 'PA':
5       rate = 0.06;
6       break;
7     case 'CA':
8       rate = 0.0625;
9       break;
10    default:
11      rate = 0;
12    }
13    return (input * rate);
14  }
15
16  function getUrl(){
17    return 'https://makeuseof.com'
18  }
19
```

Adding code to the script is easy  
Here is the result of the new script:

<i>fx</i>	=getUrl()		
	D	E	F
1	MakeUseOf		
2	<a href="https://makeuseof.com">https://makeuseof.com</a>		
3			
4			

Results of the new script

## Use the function again

After an effort to create a custom function, you can reuse this function later. If you create a script to solve a common problem, you can save a lot of time.

Even if you don't need all of them on future sheets, you should still know how to save them, in case something goes wrong.

There are several ways to reuse functions:

1. **Save the function in a blank page and use it as a template** , using a copy for all future sheets.
2. **Copy functions from one page to another** . This is quite tedious, but it will work. Open the Script Editor and copy all the code from one worksheet, open the script editor in another sheet and paste the code into it.
3. **Save the worksheet to the Google template library** . Keep in mind this will make your document accessible to others. You can restrict this to domain members, if you have a Google Apps for Work subscription. If you haven't used the template collection before, it's worth a try now.

You finished reading the article "**How to create custom functions in Google Sheets**" 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.

---