

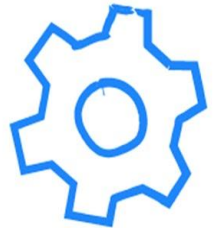
Power Up with Google Apps Script

Empower Your Team With Google Apps automate and more



<https://basescripts.com/>

Google Workspace





Introduction to Google Apps Script

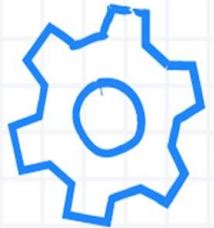
Scripting language based on JavaScript - Automate various tasks and workflows within Google Workspace applications. Extending the functionality of Google Workspace applications.

Automation: data entry, report generation, and email notifications

Customization: create custom menus, add-ons, and even create your own web applications

Integration: integrate different Google services and even non-Google services using APIs

Easy to Access: Cloud based access it from anywhere, share and collaborate on projects. **Free to Use!**

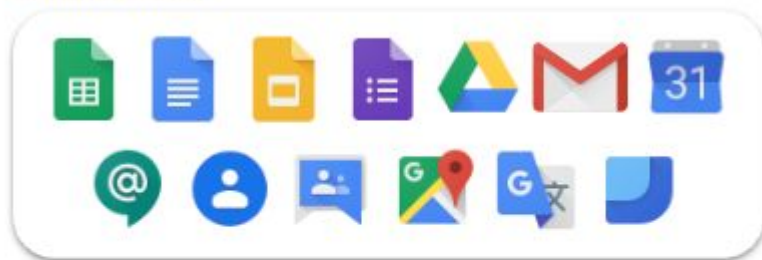


Apps Script can bring it all together

Apps Script can be used as a powerful tool to bring Workspace Services and Advanced services together, quickly and seamlessly. With just a few lines of code.

It just works!!! Making developers smile every time.

G Suite Services



Advanced Services





Experts
workspace

Laurence Svekis ✓

Best Selling Course Author - 1,000,000+
Students. Web Technology Expert - Googl...



INSTRUCTOR

INSTRUCTOR

Laurence Svekis

Instructor, GDE, Application Developer

Udemy Instructor Partner

Total students

Reviews

1,009,396

64,883

About me

I'm here to help you learn, achieve your dreams, come join me on this amazing
adventure today

Getting Started with Google Apps Script: How to create amazing time saving applications within your Google Workspace with Apps Script Kindle Edition

by Laurence Lars Svekis (Author) Format: Kindle Edition

4.4 ★★★★★ 29 ratings

[See all formats and editions](#)

Kindle

\$0.00

Read with Kindle Unlimited
to also enjoy access to over 3
million more titles
\$9.99 to buy

Google Apps Scr
Script, within jus



**Google Apps Script Complete Course
Beginner to Advanced**

Laurence Svekis

4.5 ★★★★★ (2,995)

26.5 total hours · 331 lectures · All Levels

CA\$34.99 ~~CA\$229.99~~

Bestseller

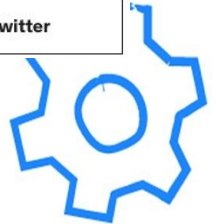
cloud. You can do really amazing things with Apps
le Workspace apps, to save time, and do a lot more.



[↔ Website](#)

[🐦 Twitter](#)

<https://basescripts.com/>



Benefits of Apps Script - Users of Apps Script

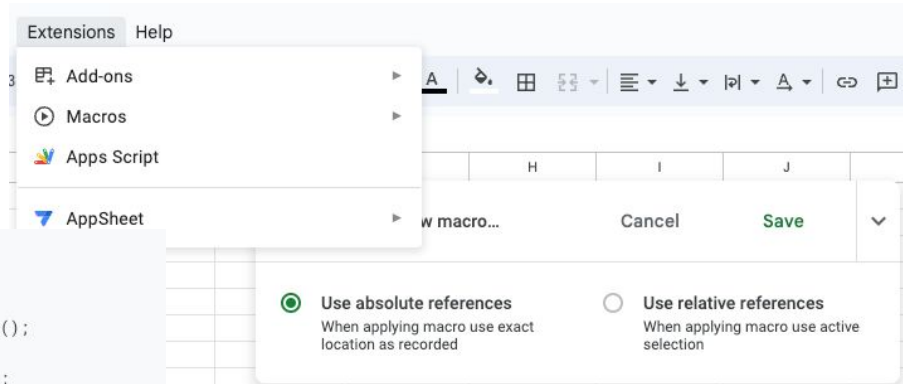
Work collaboratively - Most students I have that come to my courses have a task they are looking to automate with apps script.

Marcos and Apps Script -

Create record the macro then review

the script!

```
1  /** @OnlyCurrentDoc */
2
3  function bolder() {
4    var spreadsheet = SpreadsheetApp.getActive();
5    spreadsheet.getRange('A1').activate();
6    spreadsheet.getCurrentCell().setValue('1');
7    spreadsheet.getRange('B1').activate();
8    spreadsheet.getCurrentCell().setValue('2');
9    spreadsheet.getRange('A1:B1').activate();
10   spreadsheet.getActiveRangeList().setFontWeight('bold');
11 };
```



Common uses for Google apps script

Apps Script is used by individuals, small businesses, educational institutions, nonprofits, and IT professionals to automate tasks, manage data, and streamline workflows.

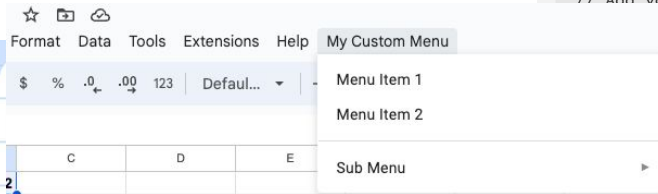
Building custom add-ons: build custom add-ons for Google services, such as Google Sheets and Google Docs. These **add-ons** can add new functionality to Google services and make them more useful for specific use cases.

Creating web applications: Google Apps Script can be used to create custom web applications that interact with Google services. This can be useful for building internal tools, dashboards, and other custom applications.



Custom Menu UI to execute Apps Script Functions

In this example, the `onOpen` function is called automatically when the Google Sheet is opened. It creates a new menu with the name "My Custom Menu" and adds two menu items (Menu Item 1 and Menu Item 2) and a sub-menu (Sub Menu) with two sub-menu items (Sub Menu Item 1 and Sub Menu Item 2). The `menuItem1`, `menuItem2`, `subMenuItem1`, and `subMenuItem2` functions are called when the corresponding menu items are clicked, and you can add your own code in these functions to perform specific actions.



```
function onOpen() {
  var ui = SpreadsheetApp.getUi();
  var menu = ui.createMenu('My Custom Menu');
  menu.addItem('Menu Item 1', 'menuItem1');
  menu.addItem('Menu Item 2', 'menuItem2');
  menu.addSeparator();
  menu.addSubMenu(ui.createMenu('Sub Menu')
    .addItem('Sub Menu Item 1', 'subMenuItem1')
    .addItem('Sub Menu Item 2', 'subMenuItem2'));
  menu.addToUi();
}

function menuItem1() {
  // Add your code here for the action of menu item 1
}

function menuItem2() {
  // Add your code here for the action of menu item 2
}

function subMenuItem1() {
  // Add your code here for the action of sub menu item 1
}

function subMenuItem2() {
  // Add your code here for the action of sub menu item 2
}
```

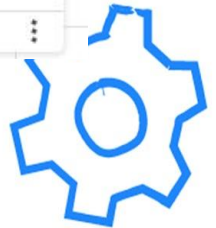
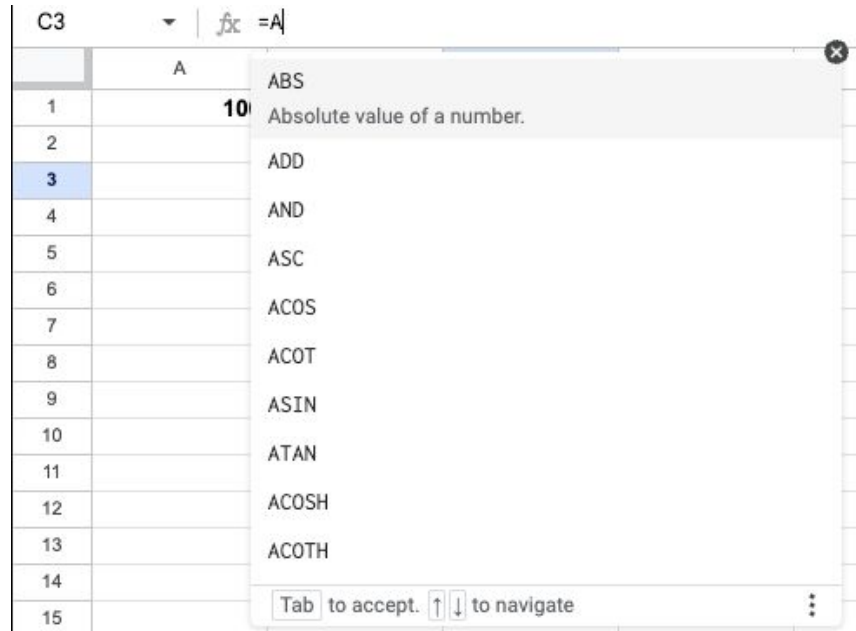
Sheets

Leverage 400+ built in Functions in Sheets

Create your own custom functions with Apps Script

Automate solutions with Macros; No coding experience necessary

Build complete applications with custom UI



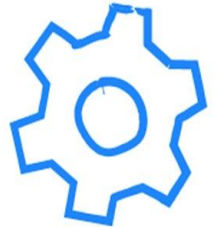
Custom Function in Sheets

This custom function takes two numbers as input (an old value and a new value) and returns the percentage change between the two.

Applying the formula =PERCENT_CHANGE(A2, B2) would return the result 20.00%, indicating that the new value of 120 is 20% higher than the old value of 100.

```
function P_CHANGE(oldValue, newValue) {  
  var percentChange = (newValue - oldValue) / oldValue * 100;  
  return percentChange.toFixed(2) + "%";  
}
```

C1	A	B	C
	100	120	20.00%

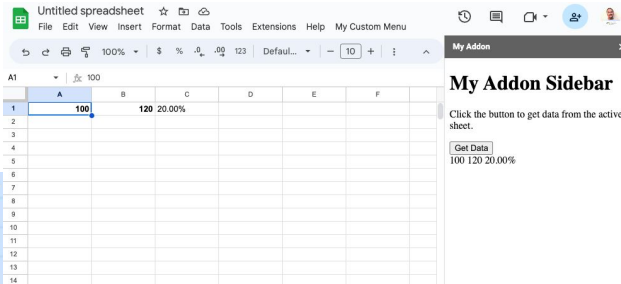


Apps Script

Serverless scripting environment based on JavaScript

Build as simple solutions or deploy as add-ons

Create a custom sidebar that can be launched, and gets data from the sheet and displays it within the sidebar.



```
function onOpen() {  
  var ui = SpreadsheetApp.getUi();  
  var menu = ui.createMenu('My Custom Menu');  
  menu.addItem('Launch', 'showSidebar')  
    .addToUi();  
}  
  
function showSidebar() {  
  var html = HtmlService.createHtmlOutputFromFile('sidebar')  
    .setTitle('My Addon')  
    .setWidth(300);  
  SpreadsheetApp.getUi()  
    .showSidebar(html);  
}  
  
function getData() {  
  var sheet = SpreadsheetApp.getActiveSheet();  
  var data = sheet.getDataRange().getValues();  
  return data;  
}
```



Setting triggers to Automate

Create Triggers to run automatically without user intervention. Run scripts automatically based on specific events or schedules, such as when a user opens a document, when a form is submitted, or on a set schedule

Create Triggers with code or with Trigger Adders

Add Trigger for Recorded Macros (Untitled spreadsheet)

Choose which function to run

P_CHANGE

Choose which deployment should run

Head

Select event source

From spreadsheet

Select event type

On open

Failure notification settings

Notify me daily

	onOpen	onInstall	onEdit	onChange	onSubmit	Time driven	Event Updated	doGet	doPost
	✓	✓	✓	✓	✓	✓			
	✓	✓			✓	✓			
	✓	✓				✓			
	✓	✓				✓			
							✓		
						✓		✓	✓



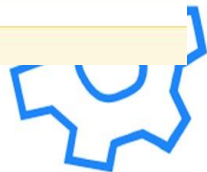
Do a Lot with just one line of Code!

This code defines a function called `translateText()` that takes the text you want to translate and the target language as inputs. In this example, the text is "Hello world" and the target language is Spanish.

```
1 function translateText() {
2   var text = 'Hello world'; // Replace with the text you want to translate
3   var targetLanguage = 'es'; // Replace with the target language code (e.g. 'es'
  for Spanish)
4
5   var translatedText = LanguageApp.translate(text, '', targetLanguage); // Call
  the LanguageApp.translate() method to translate the text
6
7   Logger.log(translatedText); // Log the translated text to the console
8 }
9
```

Execution log

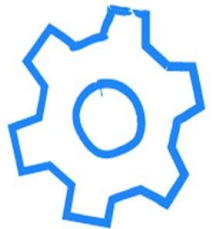
9:39:03 PM	Notice	Execution started
9:39:04 PM	Info	Hola Mundo
9:39:03 PM	Notice	Execution completed



Build with AI Gemini

- Go to <https://aistudio.google.com/app/apikey> to get your API key

```
function sendPromptToGeminiAI(q) {  
  const apiKey = 'YOUR_API_KEY'; // Replace with your actual API key from Google Cloud  
  const url = `https://generativelanguage.googleapis.com/v1beta/models/gemini-pro:generateContent?key=${apiKey}`;  
  const payload = { contents: [{ parts: [{ text: q }] }] };  
  const options = { method: 'POST',  
    contentType: 'application/json',  
    payload: JSON.stringify(payload) };  
  const response = UrlFetchApp.fetch(url, options);  
  const jsonResponse = JSON.parse(response.getContentText());  
  if (jsonResponse.candidates && jsonResponse.candidates.length > 0) { return jsonResponse.candidates[0].content.parts.map(part =>  
    part.text).join(""); }  
  return "No response received."  
}
```



Students of Apps Script

Best responses from students

Many in Academia - working on small projects to automate

New to code - Passionate and ready to explore

Google Apps Script works simply and easily.



Jerico Erin Polintan Udemy Business

Updated 1 month ago



so exciting to learn these new stuffs that I didn't know it would be possible ...



Emy Foscablo

Updated 3 months ago



I have missed coding in JS because it's not my current line of work. But now that I see it in action, it makes me excited. Thank you! :)



Pedro Gehrt

Updated 10 months ago



Complete course with a great professor. Great projects as examples to show what you can do with GAS! This course helped me a lot on my job and i even was congratulated! Thanks!

