

<https://github.com/lsvakis/5-Simple-Google-Apps-Script-Exercises-Using-Gemini>

5 Simple Google Apps Script Exercises Using Gemini (With Full Code + Downloadable Repo)

AI is transforming how we work inside Google Workspace—and with Google Apps Script + Gemini, you can now add AI superpowers directly into Docs, Sheets, and your everyday workflows.

To help you get started quickly, I've created a set of **five simple, beginner-friendly Apps Script exercises**. Each one introduces a different way Gemini can integrate into your Workspace tools—from answering questions to summarizing Docs to generating ideas based on cell values.

These exercises are perfect for:

- Students learning Apps Script
- Developers experimenting with Gemini
- Educators building AI-enhanced tools
- Anyone looking to automate Google Workspace with AI

A **GitHub-ready project + ZIP download** is included at the end.

What You'll Build

Across the exercises, you'll learn how to:

- ✓ Send prompts to Gemini
- ✓ Build a custom menu inside Google Sheets
- ✓ Summarize selected text in Google Docs
- ✓ Generate creative ideas from spreadsheet data
- ✓ Build a custom spreadsheet function: `=GEMINI("your prompt")`
(Just like a built-in formula!)

Each exercise includes:

- ✓ Code

Get more Apps Script Content at <https://basescripts.com/> by Laurence Svekis

- ✓ Test functions
- ✓ Practical use cases

Let's begin.



Exercise 1 — Ask Gemini a Simple Question

The easiest possible starting point: send a prompt to Gemini and log the answer.

Perfect for testing that your API key + permissions are working correctly.

Use Cases

- Ask coding questions
 - Generate writing prompts
 - Create learning summaries
-



Exercise 2 — Add a Custom Menu to Google Sheets

Next, you'll add a new **Gemini** menu inside Sheets.

Clicking *Ask Gemini* triggers an AI request and displays the result in a popup. This pattern is powerful—it mirrors how real add-ons are built.

Use Cases

- Explain formulas
 - Provide writing feedback
 - Generate step-by-step instructions
-

Exercise 3 — Summarize Selected Text in Google Docs

Highlight text → run the function → get an instant, AI-generated summary.

This is one of the most practical Docs automations you can build.

Use Cases

- Summaries for meetings, articles, research
 - Simplify educational content
 - Rewrite content for clarity
-

Exercise 4 — Generate Creative Ideas From a Cell

In Sheets:

- Put a topic inside **A1**
- Run the script
- Gemini writes ideas into **A2**

A simple structure, but extremely powerful.

Use Cases

- Brainstorming project ideas
 - Generating marketing concepts
 - Creating topic lists for blogs or courses
-

Exercise 5 — Create a Custom Function: **=GEMINI()**

This one is a game changer.

You can now use Gemini like a regular spreadsheet function:

```
=GEMINI("Write a haiku about coding.")
```

Your sheet becomes an **AI-powered productivity tool** with infinite possibilities.

Use Cases

- Rewrite text
 - Explain concepts
 - Create summaries automatically
 - Generate variations of content
-



Testing Your Exercises

Every exercise includes a matching **test function** so you can validate it directly from the Script Editor.

Also included is a **master test runner**:

```
function test_allGeminiExercises() {  
  ...  
}
```

Run it once and it will sequentially verify all 5 exercises.



Download the GitHub-Ready Project

To help you get up and running fast, I've packaged everything into a GitHub-ready ZIP:

👉 **Download the full project**

<https://github.com/lsveki/5-Simple-Google-Apps-Script-Exercises-Using-Gemini>

The ZIP includes:

- `Code.gs` with all exercises + helpers + tests
- `appscript.json` configuration
- `README.md` with setup instructions

You can drop this directly into a bound Google Sheet/Doc project or upload it to GitHub as a sample repository.

Why Learn Apps Script + Gemini?

Because together, they let you:

- Automate daily tasks
- Build custom AI tools
- Enhance student learning
- Improve workplace productivity
- Prototype ideas extremely fast

You don't need servers, frameworks, or deployments.

Just open a Google Doc or Sheet → Extensions → Apps Script → start coding.

It's one of the fastest ways to create real, useful AI-powered tools.

Full code

```
{  
  
  "timeZone": "America/Toronto",  
  
  "dependencies": {},  
  
  "exceptionLogging": "STACKDRIVER",  
  
  "runtimeVersion": "V8",  
  
  "oauthScopes": [  
  
    "https://www.googleapis.com/auth/script.properties",  
  
    "https://www.googleapis.com/auth/script.external_request",  
  
  ]  
}
```

Get more Apps Script Content at <https://basescripts.com/> by Laurence Svekis

```
"https://www.googleapis.com/auth/spreadsheets.currentonly",  
"https://www.googleapis.com/auth/documents.currentonly"  
]  
}
```

```
/**
```

```
* Gemini Apps Script Exercises
```

```
*
```

```
* Requirements:
```

```
* 1. Create a Gemini API key in your Google account.
```

```
* 2. In the Apps Script editor, go to:
```

```
* - Project Settings → Script properties
```

```
* - Add key: GEMINI_API_KEY
```

```
* - Paste your API key as the value.
```

```
*
```

```
* This file contains:
```

```
* - 5 exercises using Gemini
```

```
* - Test functions for each exercise
```

```
* - A master test runner
```

```
*/
```

```
const GEMINI_MODEL_ID = 'gemini-2.5-flash';
```

```
/**
```

```

* Get Gemini API key from Script Properties.

*/

function getGeminiApiKey_() {

  const scriptProps = PropertiesService.getScriptProperties();

  const key = scriptProps.getProperty('GEMINI_API_KEY');

  if (!key) {

    throw new Error('GEMINI_API_KEY is not set in Script Properties.');
```

```

  }

  return key;

}

/**

* Core Gemini helper: send a text prompt, get a text response.

*

* @param {string} prompt

* @returns {string}

*/
```

```

function callGemini(prompt) {

  const apiKey = getGeminiApiKey_();

  const url =

    'https://generativelanguage.googleapis.com/v1beta/models/' +

    GEMINI_MODEL_ID +

    ':generateContent?key=' +
```

```
apiKey;
```

```
const payload = {  
  contents: [  
    {  
      parts: [{ text: prompt }]  
    }  
  ]  
};
```

```
const res = UrlFetchApp.fetch(url, {  
  method: 'post',  
  contentType: 'application/json',  
  payload: JSON.stringify(payload),  
  muteHttpExceptions: true  
});
```

```
const result = JSON.parse(res.getContentText());
```

```
if (!result.candidates ||  
    !result.candidates[0] ||  
    !result.candidates[0].content ||  
    !result.candidates[0].content.parts ||  
    !result.candidates[0].content.parts[0] ||
```

```

!result.candidates[0].content.parts[0].text) {

  throw new Error('Unexpected Gemini response: ' + res.getContentText());

}

return result.candidates[0].content.parts[0].text;

}

/**
 * =====
 * EXERCISE 1 – Simple Prompt
 * =====
 * Logs a one-sentence explanation of JavaScript from Gemini.
 */

function exercise1_simplePrompt() {

  const answer = callGemini('Explain JavaScript in one sentence. ');

  Logger.log(answer);

}

/**
 * Test for Exercise 1.
 */

function test_exercise1_simplePrompt() {

  Logger.log('Running Exercise 1 test...');

  exercise1_simplePrompt();

```

```

    Logger.log('Check Logs for Gemini response.');
```

```

}
```

```

/**
 * =====
 * EXERCISE 2 – Custom Menu in Google Sheets
 * =====
 * Adds a "Gemini" menu with "Ask Gemini".
 * When clicked, it shows tips in an alert.
 */
function onOpen(e) {
    // Only runs in Sheets UI
    SpreadsheetApp.getUi()
        .createMenu('Gemini')
        .addItem('Ask Gemini', 'runGeminiMenu')
        .addToUi();
}

/**
 * Triggered from the custom menu in Sheets.
 */
function runGeminiMenu() {
    const prompt = 'Give me 3 tips for learning JavaScript faster.';
    const answer = callGemini(prompt);

```

```

    SpreadsheetApp.getUi().alert(answer);
  }

  /**
   * Test for Exercise 2 (without using UI menu).
   */
  function test_runGeminiMenu() {
    Logger.log('Running Exercise 2 test (no UI)...');
    const prompt = 'Give me 3 tips for learning JavaScript faster.';
    const answer = callGemini(prompt);
    Logger.log('Gemini answer:\n' + answer);
  }

  /**
   * =====
   * EXERCISE 3 – Summarize Selected Text in a Doc
   * =====
   * In Google Docs:
   * 1. Select some text.
   * 2. Run summarizeSelection() from the editor.
   * 3. A popup shows Gemini's summary.
   */
  function summarizeSelection() {
    const doc = DocumentApp.getActiveDocument();

```

```

const selection = doc.getSelection();

if (!selection) {
  DocumentApp.getUi().alert('Select some text first!');
  return;
}

let text = "";

selection.getRangeElements().forEach(function (el) {
  const element = el.getElement();
  if (element.editAsText) {
    text += element.asText().getText() + '\n';
  }
});

if (!text.trim()) {
  DocumentApp.getUi().alert('Selected text is empty.');
```

return;

```

}

const prompt = 'Summarize this text in a short paragraph:\n' + text;
const summary = callGemini(prompt);
DocumentApp.getUi().alert(summary);

```

```

}

/**
 * Test for Exercise 3 using sample text (no Doc selection needed).
 */
function test_summarizeSelection_withSampleText() {
  const sampleText =
    'JavaScript is a programming language used to add interactivity to web pages. ' +
    'It runs in the browser and allows developers to create dynamic user interfaces.';

  const prompt =
    'Summarize this in one short paragraph:\n' + sampleText;

  Logger.log('Running Exercise 3 test with sample text...');
  const summary = callGemini(prompt);
  Logger.log('Summary:\n' + summary);
}

/**
 * =====
 * EXERCISE 4 – Idea Generator (Sheet cell A1 → A2)
 * =====
 * In Sheets:
 * - Put a topic in A1.

```

```

* - Run ideaGenerator().

* - Gemini's ideas are written to A2.

*/

function ideaGenerator() {

  const sheet = SpreadsheetApp.getActiveSheet();

  const topic = sheet.getRange('A1').getValue();

  if (!topic) {

    sheet.getRange('A2').setValue('Please enter a topic in A1.');
```

return;

```
  }

  const prompt = 'Give me 5 creative ideas about: ' + topic;

  const answer = callGemini(prompt);

  sheet.getRange('A2').setValue(answer);

}

/**

* Test for Exercise 4.

*/

function test_ideaGenerator() {

  const sheet = SpreadsheetApp.getActiveSheet();
```

```

if (!sheet.getRange('A1').getValue()) {
    sheet.getRange('A1').setValue('beginner JavaScript projects');
}

Logger.log('Running Exercise 4 test...');

ideaGenerator();

Logger.log('Check cell A2 for Gemini's generated ideas.');
```

```

}

/**
 * =====
 * EXERCISE 5 – Custom Function =GEMINI("prompt")
 * =====
 * Usage in Sheets:
 * =GEMINI("Write a haiku about coding.")
 */

function GEMINI(prompt) {
    if (typeof prompt !== 'string') {
        return 'Prompt must be a string.';
    }
    return callGemini(prompt);
}

/**
```

```

* Test for Exercise 5.

*/

function test_GEMINI() {

  Logger.log('Running Exercise 5 test for GEMINI()...');

  const result = GEMINI(

    'Write a short motivational message for new coders.'

  );

  Logger.log('GEMINI() result:\n' + result);

}

```

```

/**

* =====

* Master Test Runner

* =====

* Runs all test functions in sequence.

*/

function test_allGeminiExercises() {

  Logger.log('=== Testing Exercise 1 ===');

  test_exercise1_simplePrompt();


  Logger.log('=== Testing Exercise 2 ===');

  test_runGeminiMenu();


  Logger.log('=== Testing Exercise 3 (sample text) ===');

```

```
test_summarizeSelection_withSampleText();
```

```
Logger.log('=== Testing Exercise 4 ===');
```

```
test_ideaGenerator();
```

```
Logger.log('=== Testing Exercise 5 ===');
```

```
test_GEMINI();
```

```
Logger.log(
```

```
    'All test functions executed. Check Logs and Docs/Sheets for results.'
```

```
);
```

```
}
```