



Entrenamiento: App Script

Sesión 1



¡Hola! Soy Jimmy Corzon

APP SCRIPT SPECIALIST

- Arquitecto de Soluciones de Software y especialista en la automatización de procesos de negocio.
- Apasionado por transformar herramientas cotidianas como Google Sheets y Gmail en potentes aplicaciones empresariales.



ARQUITECTO DE SOFTWARE

APP SCRIPT TRAINER



1. Objetivos

2. ¿Qué es App Script?

3. Aplicaciones

CONTENIDO

¿Qué es App Script?



- *Google Apps Script es una plataforma de desarrollo rápido.*
- *Permite crear soluciones empresariales personalizadas.*
- *Integra y automatiza los servicios de Google.*
- *Ahorra tiempo en tareas repetitivas.*
- *Amplía la funcionalidad de las aplicaciones de Google.*



Google Apps Script



Serie de instrucciones escritas en un lenguaje de programación o lenguaje de scripting para realizar una tarea particular.

Las instrucciones son escritas y salvadas como un script que es ejecutado bajo las circunstancias que el "programador" defina.

¿Qué hace Apps Script?

Integra las herramientas de Google.

Personaliza tareas de herramientas de Google.

Automatiza tareas.

Maximiza valor de la información.

Envía información de un documento a otro

Envía información por correos.

Herramientas compatibles con Apps Script



Google Spreadsheet



Gmail



Groups



Google Docs



Google Drive



Google Maps



Google form



Google Calendar



Google Translate



Google Site



Google Contacts



1. Una Cuenta de Google Workspace.
2. Un navegador (Preferiblemente Google Chrome).
3. Conocimientos en básicos de programación.
4. Un editor de código (Opcional).

¿Qué necesito?

TRES EN UNO



Ejecución de JavaScript

APIs ya construidas para conectar y automatizar servicios de Google.

Extiende funcionalidades y construye aplicaciones personalizadas.



Ejecución de JavaScript

Basado en **Cloud Scripting Language** (JavaScript).

Herramientas de cumplimiento, depuración y rendimiento integradas.



Editor en el Browser

Cero instalación o configuración.

Archivos, compartir y versiones gestionadas en la nube.

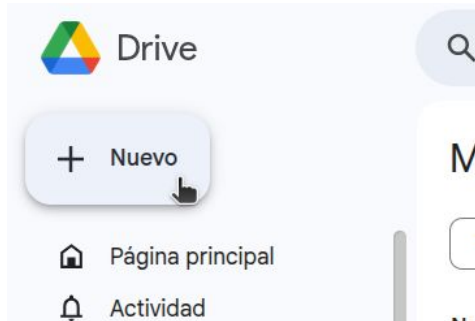


Ingresar al entorno de Apps Script

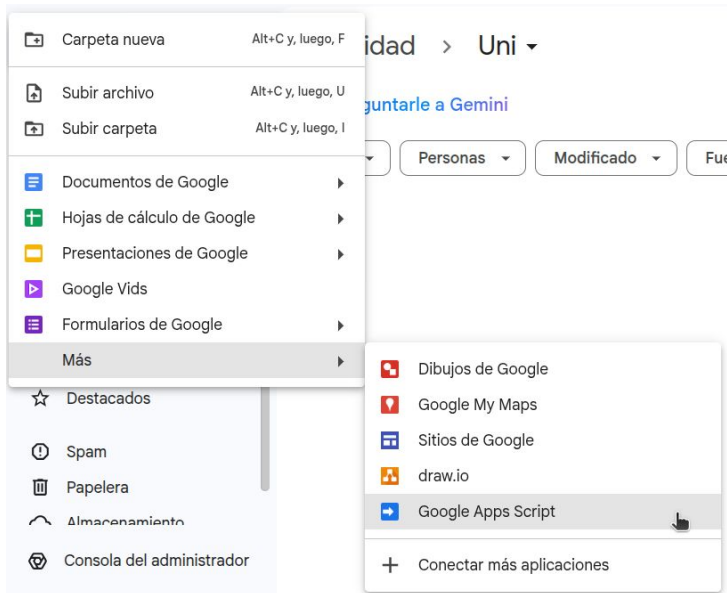
1. Ingresar a Google Drive.



2. Dar clic en el botón “Nuevo” o “New”.



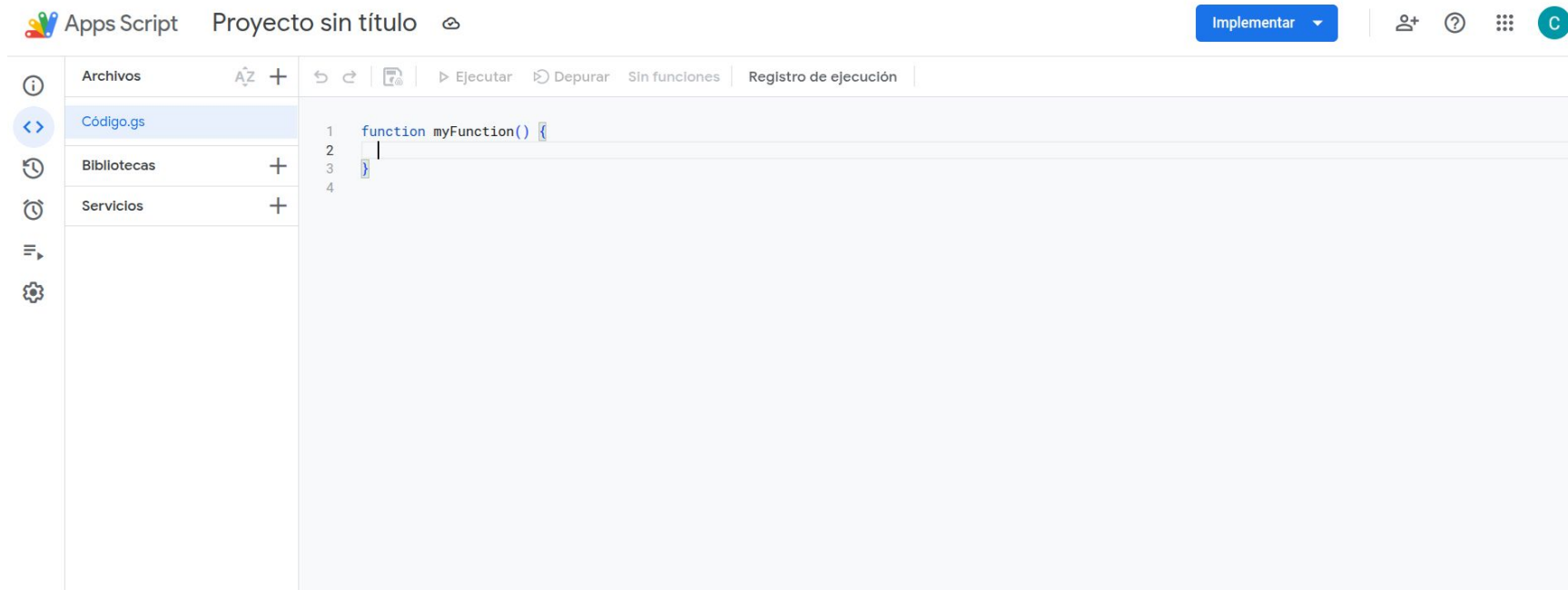
3. Dar clic en “Mas”.



4. Si encontramos **Google Apps Script** entonces Dar Clic.

Ingresar al entorno de Apps Script

5. Abrir el entorno de trabajo.



Coffee Break



TIPOS DE SCRIPTS

Google Apps Script

Tipos


Container Bound

Asociados a un documento de Google (Hojas de cálculo, Documentos, Sites).

Standalone


Independientes, creados desde script.google.com y guardados en Google Drive.


Modos de Ejecución

 **Manual** – El usuario ejecuta la función desde el editor.

 **Por tiempo** – Usando temporizadores programados.

Tipos

 **Basados en tiempo** – Ejecutan funciones según una programación (minutos, horas, días, semanas, meses).

 **Basados en eventos** – Responden a acciones como abrir, editar o enviar un formulario.

Ejemplos de Contenedor

Google Sheets

Google Docs







Google Slides

ESTRUCTURA DE ARCHIVOS

Proyecto típico en Google Apps Script



Proyecto

-  **appsscript.json** – es un archivo de manifiesto que contiene información esencial del proyecto
-  **code.gs** – Lógica principal del script
-  **main.gs** – Funciones auxiliares o controladores
-  **index.html** – Interfaz de usuario (HTML + CSS + JS embebido)
-  **sidebar.html** – Vistas adicionales (barra lateral)
-  **modal.html** – Componentes reutilizables



DISPARADORES

Google Apps Script



Basados en Tiempo

Ejecutan funciones de forma programada.



Cada minuto, hora o día.



Frecuencias como semanal o mensual.



Temporizadores personalizados.



Basados en Eventos

Responden a acciones del usuario o del sistema.



Al abrir o cerrar un documento.



Al editar una hoja de cálculo.



Al enviar un formulario.

⚡ **Tip:** Los disparadores se configuran desde el editor en *Editor > Disparadores > Añadir disparador*.

¿Dónde buscar información?

<https://developers.google.com/apps-script>



GRACIAS_

Xertica.ai



Entrenamiento: App Script

Sesión 2





1. Objetivos

2. Herramienta

3. Laboratorio Práctico

CONTENIDO



Patrones en Apps Script

Buenas prácticas y estructuras para código eficiente, reutilizable y mantenible en Google Workspace

Definición (en este curso): Patrones = enfoques reutilizables para automatizar, personalizar e integrar Google Workspace (Gmail, Sheets, Docs, Drive), usando JavaScript moderno, google.script.run, scriptlets en HTML y buenas prácticas de depuración y versionado.

Cómo funcionan

⚙️ **Automatización & Personalización:** menús, diálogos, barras laterales en editores de Google.

🔗 **Integración con Servicios:** Gmail, Calendar, Drive; publicar como Web App o complemento.

♻️ **Código Reutilizable:** funciones modulares y separación de responsabilidades.

Técnicas comunes

🧩 **Scriptlets en HTML** para contenido inicial.

```
<? var user = Session.getActiveUser().getEmail();
<p>Hola, <?= user ?></p>
```

📄 **google.script.run** para interacción dinámica cliente ↔ servidor.

```
// En el HTML (cliente)
google.script.run.withSuccessHandler(render)
  .obtenerDatos();

// En .gs (servidor)
function obtenerDatos() { return SpreadsheetApp.getActiveSheet().getDataRange().getValues(); }
```

Más técnicas

📁 **Registro & Depuración:** Logger.log(), Depurador, registros de ejecución.

🔧 **Funciones personalizadas & Macros** en Sheets.

🔖 **Implementaciones versionadas:** crea versiones y promuévelas a producción con control.

💡 **Beneficios clave:** Reutilización (menos duplicación), Eficiencia (automatiza tareas), y Claridad (código mantenible para equipos).



El Patrón “Generador de Documentos”

Separación de Datos (Sheets) y Presentación (Docs)



Preparando Nuestros Activos

Para construir nuestro **Generador de Documentos**, necesitamos dos elementos claves: una **fuente de datos** y una **plantilla de presentación**.

Google Sheet

Datos de ejemplo con columnas:

- ✚ Cliente
- ✚ Proyecto
- ✚ Monto
- ✚ Fecha

Google Doc Plantilla

Contiene **placeholders**:

```
{{cliente}}  
{{proyecto}}  
{{monto}}  
{{fecha}}
```

Estos marcadores {{...}} serán reemplazados dinámicamente por nuestro script.



Coffee Break



¿Dónde buscar información?

Ejemplos de Apps script



<https://developers.google.com/apps-script>

Repositorio de GitHub



<https://github.com/jimmy-corzon/curso-apps-script-301>

GRACIAS_

Xertica.ai



Entrenamiento: App Script

Sesión 3





1. Objetivos



2. Tipos de proyectos

3. La CLI de Apps Script

4. Requisitos

5. Entorno de desarrollo

CONTENIDO

El Mapa de Proyectos en Apps Script



Web App

Aplicaciones con URL pública que combinan frontend y backend.



Add-on Editor

Extiende editores como Docs, Sheets o Slides con menús y barras.



Add-on Workspace

Extiende todo Google Workspace a nivel organizacional.



Biblioteca

Reutiliza código en múltiples proyectos (DRY: Don't Repeat Yourself).



API Executable

Permite que aplicaciones externas llamen funciones de Apps Script.



Script Vinculado

Vive dentro de un archivo (Docs, Sheets, Forms) y lo automatiza.



Google Apps Script + Clasp CLI


Desarrollo profesional para Apps Script




¿Qué es Clasp?

Command Line Apps Script Projects

✂ Herramienta oficial de Google

 Conecta tu editor local con Apps Script

 Instalar con: `npm install -g @google/clasp`



¿Por qué usarlo?

✓ **Editor profesional:** VS Code con autocompletado

✓ **Control de versiones:** Git nativo


✓ **Sin limitaciones:** Desarrollo local completo


✓ **Extensiones:** Todo el ecosistema VS Code



TypeScript Support

 **Tipado estático:** Menos errores

 **IntelliSense:** Autocompletado inteligente

 **Transpilación:** Automática a JavaScript

 **APIs tipadas:** Documentación integrada



Editor Web vs Clasp CLI

Editor Web (script.google.com)

- Acceso directo desde el navegador
- Ideal para scripts simples
- Sin configuración necesaria
- Limitaciones del editor web

✂ **Clasp CLI + VS Code**

- Desarrollo profesional completo
- Control de versiones con Git
- Extensiones y plugins
- TypeScript y linting



Requisitos



El Motor: Node.js + nvm

El entorno de ejecución de nuestro código. Usamos `nvm` (Node Version Manager) para gestionar versiones y evitar conflictos.



El Gestor de Paquetes: npm

Gestor de paquetes. Nuestra tienda de herramientas. Usado para instalar y gestionar librerías de terceros (y `clasp`).



Control de Versiones: Git

Sistema de control de versiones. Esencial para la colaboración y las buenas prácticas.



Flujo de Trabajo

VS Code + Clasp CLI



Comandos Esenciales

`clasp create`

Nuevo proyecto

`clasp clone`

Proyecto existente

`clasp push`

Subir cambios

`clasp pull`

Descargar cambios



Configuración Inicial (Solo una vez)

1. Instalar: `npm install -g @google/clasp`

2. Autenticar: `clasp login`

Coffee Break








Entrenamiento: App Script

Sesión 4



- 
- 
- 
1. Objetivo: Construir una Web App.
 2. doGet(e): Sirviendo la interfaz de usuario.
 3. doPost(e): Recibiendo datos del usuario.
 4. El Flujo de Autenticación de Google
 5. Laboratorio

CONTENIDO

Objetivo: Construir una Web App

Publicar un script como **aplicación web** accesible por URL.



doGet(e)

Sirve la interfaz de usuario

Genera la vista inicial de la Web App.

```
function doGet(e) {  
  return HtmlService  
    .createHtmlOutput("¡Hola Mundo!");  
}
```

doPost(e)

Recibe y procesa datos enviados por el usuario

Normalmente desde formularios HTML o solicitudes HTTP.

```
function doPost(e) {  
  const datos = e.parameter;  
  return ContentService  
    .createTextOutput("Datos recibidos: " + JSON.stringify(datos));  
}
```

El Flujo de Autenticación de Google



El usuario concede permisos → Google genera un token → Apps Script lo usa para acceder a los servicios autorizados.



Bienvenidos

Daremos unos minutos adicionales,
para nuestros participantes faltantes



Entrenamiento: App Script

Sesión 5



GRACIAS_

Xertica.ai

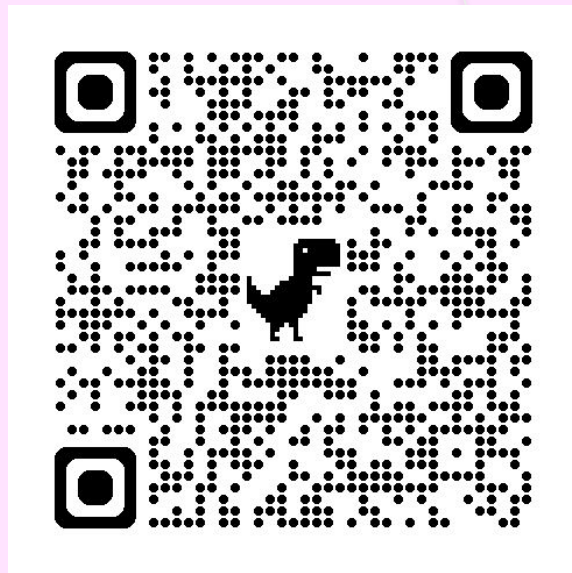
Encuesta de entrenamiento

Denominación del curso:
App Script

Entrenador:
Jimmy Corzon

ID:
K0810

**Contesta por favor la encuesta para
mejorar nuestros servicios.**



GRACIAS_

Xertica.ai

GRACIAS_

Xertica.ai