

2020年2月27日
株式会社インプレスR&D
<https://nextpublishing.jp/>

経営管理を Google Apps Script でモダンに自動化！
「現役公認会計士が解説！ GAS で自動化する経営管理」発行
技術の泉シリーズ、2月の新刊

インプレスグループで電子出版事業を手がける株式会社インプレス R&D は、『現役公認会計士が解説！ GAS で自動化する経営管理』（著者：稲垣 大輔）を発行いたします。

最新の知見を発信する『技術の泉シリーズ』は、「技術書典」や「技術書同人誌博覧会」をはじめとした各種即売会や、勉強会・LT 会などで頒布された技術同人誌を底本とした商業書籍を刊行し、技術同人誌の普及と発展に貢献することを目指します。

『現役公認会計士が解説！ GASで自動化する経営管理』
<https://nextpublishing.jp/isbn/9784844378402>



著者：稲垣 大輔
小売希望価格：電子書籍版 1600 円（税別）／印刷書籍版 1800 円（税別）
電子書籍版フォーマット：EPUB3／Kindle Format8
印刷書籍版仕様：B5 判／カラー／本文 118 ページ
ISBN：978-4-8443-7840-2
発行：インプレス R&D

<<発行主旨・内容紹介>>

本書は業務設計に必要な基本的な考え方と、IT ツールの実装に関する解説書です。最新ツールを入れたのに経理業務が効率化されない、そもそも最適な業務フローがわからない。そんな悩みはありませんか？

経営管理のプロである会計士が Google Apps Script や SaaS の API を組み合わせた活用方法をレクチャーします。また、Google Apps Script をモダンに書くための TypeScript 導入方法も掲載しており、プログラミング入門としても活用

できます。

(本書は、次世代出版メソッド「NextPublishing」を使用し、出版されています。)

Google Apps Script を基本から解説

第2章 Google Apps Script 入門

2.1 Google Apps Script(GAS)とは

Google Apps Script とは何なのでしょう。Google Apps という単語から「Google のアプリを操作する何かである」とイメージがつかなくとも構いません。公式サイトには以下の通り書かれています。

Google Apps Script は、G Suite と統合するビジネスアプリケーションを迅速かつ簡単に作成するための迅速なアプリケーション開発プラットフォームです。

要するに、G Suite や Google Apps を容易に稼働させることができる、JavaScript ベースのプログラミング言語です。JavaScript ベースの言語でコードを書くことにより、Gmail や Google ドライブなどの Google Apps の組み込み API を容易に稼働させることができます。

GAS を活用する理由と簡単な実例

数多くのプログラミング言語がある中で、コーポレート業務において Google Apps Script を活用する理由は以下の通りです。

- Google アカウントを保有していればすぐにでも導入可能であること
- 経営管理業務において日常的に使用されている Google ドライブや Google スプレッドシートとの連携機能が充実していること

いくらプログラムによる業務自動化を行いたいと思っても、急に高度な IT ツールを導入するのは心理的にも予算的にもハードルが高いはず。しかし、Google Apps Script は Google アカウントさえ持っていればすぐに導入できる気軽さと、コーポレート業務で日常的に使っている Google スプレッドシートや Google ドライブとの親和性の高さが特徴であるため、コーポレート業務の効率化に役立つものになります。

以下に、GAS を使って作成できるプログラムの実例を挙げておきます。

- Gmail のフォルダの中から添付ファイルだけを自動取得して特定のドライブに保存する
- Google カレンダーの一日の予定を取得して毎日 Slack 等のチャットツールに通知する
- スプレッドシートに記載されている情報をランダムに通知する
- スプレッドシートが更新される都度更新内容を Gmail やチャットツールに通知を行う

このように、GAS は様々な Google Apps と外部アプリを連携することが得意です。アイデア次第ではありますが様々な効率化ツールを作成することができるでしょう。

2.2 GAS の導入方法

Google Apps Script を利用する方法は主にふたつあります。

スタンドアロンスクリプト (直接スクリプトエディタを開く)

- Google ドライブから右クリック
- そこから Google Apps Script を選択

図 2.1:



コンテナバインドスクリプト (Google スプレッドシートからスクリプトエディタを開く)

- Google スプレッドシートを開く
- その後スプレッドシートからプルダウンメニューからスクリプトエディタをクリック

20 | 第2章 Google Apps Script 入門

第2章 Google Apps Script 入門 | 21

free が提供する API を使った実例を紹介

第4章 freeAPI を活用した実例

4.1 はじめに

この章では、実際に freeAPI を活用したサービスを作ってみます。free を使ったサービスを作る流れは通常は以下の通りになります。

- アクセストークンを取得する
- GAS で API を叩く
- API を用いて取得した JSON を指定の様式に加工する
- その結果の情報を指定の場所に格納する

この流れを一度身につけることができれば、他のソフトで API を叩いて利用することは容易にできるようになります。より効果的な学習のために、実際に手を動かしながらこの章を読み進めていきますと幸いです。

4.2 事前準備

freeAPI を叩くために必要なものは、以下の通りになります。もしまだインストールできていない方は、各サービスのアカウントを作成いただけますと幸いです。

- free アカウント
- Google ドライブ

4.3 アクセストークンを取得する

free でアプリ作成

free へのログインができましたら、まずは会計 free の画面の右上にある「free アプリストア」をクリックします。

図 4.1: free アプリストアへのログイン



図 4.2: 開発者ページ



その後 free アプリストアより開発者ページタブをクリックして、中央にある「今すぐアプリを作成」をクリックします。

44 | 第4章 freeAPI を活用した実例

第4章 freeAPI を活用した実例 | 45

第9章 TypeScriptでコードを書いてみよう

今までの章で TypeScript の使い方を理解することができたと思います。そこで、実際に TypeScript と clasp を用いたコーディングを実施してみようと思います。実行される結果は変わらないコードになりますが、「TypeScript を活用する」という機能があるのかと、比較しながら学習をしてみましょう。

9.1 FS を取得する

```

authorizations
const CLIENT_ID = ''; //Client ID
const SECRET = ''; //Client Secret
/**
 * 目的: Authの認証を行う
 */
function showAlertAuth() {
  const service = getService();
  const authorizationUrl = service.getAuthorizationUrl();
  const template = HtmlService.createTemplateFromFile('認証ダイアログ');
  template.authorizationUrl = authorizationUrl;
  const page = template.evaluate();
  SpreadsheetApp.getUi().showModalDialog(page, '認証をしてください');
}
/**
 * 認証用のURLを叩くためのコード
 */
export function getService() {
  return OAuth2.createService('freee')
    .setAuthorizationBaseUrl('https://secure.freee.co.jp/oauth/authorize')
    .setTokenUrl('https://secure.freee.co.jp/oauth/token')
    .setClientId(CLIENT_ID)
    .setClientSecret(SECRET)
    .setCallbackFunction('authCallback')
    .setPropertyStore(PropertiesService.getUserProperties());
}
/**
 * 認証コールバックのためのコード

```

```

*/
function authCallback(request) {
  const service = getService();
  const isAuthorized = service.handleCallback(request);
  const accessToken = service.getAccessToken();
  if (isAuthorized) {
    return HtmlService.createHtmlOutput('認証成功。閉じてOK');
  } else {
    return HtmlService.createHtmlOutput('認証失敗');
  }
}
/**
 * 終了のためのコード
 */
function clearService() {
  OAuth2.createService('freee')
    .setPropertyStore(PropertiesService.getUserProperties())
    .reset();
}

```

```

get-fs
import { getService } from './authorization';
import { settings } from './settings';

function setBs() {
  const bs = getFs('bs');
  const array = createArray(bs);
  setSpreadsheet(array, 'BS');
}

function setPl() {
  const bs = getFs('pl');
  const array = createArray(bs);
  setSpreadsheet(array, 'PL');
}

type fs = 'bs' | 'pl';
function getFs(fs: fs, startMonth: number = 1, endMonth: number = 12) {
  const accessToken = getService().getAccessToken();
  const baseUrl = (fs: fs) => {
    if (fs === 'bs') {

```

<<目次>>

第1章 はじめに

- 1.1 経営管理の概要
- 1.2 経理業務とは
- 1.3 経理の現状の問題
- 1.4 解決手段
- 1.5 経営管理のあるべき姿
- 1.6 現状を理解する
- 1.7 業務改善を行う
- 1.8 さいごに

第2章 Google Apps Script 入門

- 2.1 Google Apps Script(GAS)とは
- 2.2 GAS の導入方法
- 2.3 文法編
- 2.4 Google Apps を操作してみよう
- 2.5 よく使用する関数
- 2.6 さいごに

第3章 API 入門

- 3.1 Web API とは何か
- 3.2 経営管理とAPI
- 3.3 API の使い方
- 3.4 API と認可

- 3.5 さいごに
- 第4章 freeAPI を活用した実例
 - 4.1 はじめに
 - 4.2 事前準備
 - 4.3 アクセストークンを取得する
 - 4.4 実践例
 - 4.5 さいごに
- 第5章 ケーススタディ
 - 5.1 プロジェクト概要
 - 5.2 さいごに
- 第6章 Google Apps Script をローカルで書くための環境をインストールしよう
 - 6.1 はじめに
 - 6.2 Node.js npm
 - 6.3 clasp とは
 - 6.4 @types とは
 - 6.5 Prettier とは
 - 6.6 ESLint とは
 - 6.7 さいごに
- 第7章 ES6(ECMA2015)
 - 7.1 ES6 とは
 - 7.2 ES6 で追加された文法一覧
 - 7.3 さいごに
- 第8章 TypeScript 入門
 - 8.1 TypeScript とは
 - 8.2 TypeScript のメインの型
 - 8.3 関数の型定義
 - 8.4 さいごに
- 第9章 TypeScript でコードを書いてみよう
 - 9.1 FSを取得する
 - 9.2 変更した部分
 - 9.3 TypeScript をトランスパイルした結果
 - 9.4 さいごに

<<著者紹介>>

稲垣 大輔

公認会計士。現在はスタートアップや上場準備企業の経理支援やシステム導入や、監査法人にて IT 支援業務を行う。また、会計×IT をテーマにした勉強会である「PyCPA」運営として講師や企画を担当している。

<<販売ストア>>

電子書籍:

Amazon Kindle ストア、楽天 kobo イーブックストア、Apple Books、紀伊國屋書店 Kinoppy、Google Play Store、honto 電子書籍ストア、Sony Reader Store、BookLive!、BOOK☆WALKER

印刷書籍:

Amazon.co.jp、三省堂書店オンデマンド、honto ネットストア、楽天ブックス

※ 各ストアでの販売は準備が整いしたい開始されます。

※ 全国の一般書店からもご注文いただけます。

【インプレス R&D】 <https://nextpublishing.jp/>

株式会社インプレス R&D(本社:東京都千代田区、代表取締役社長:井芹昌信)は、デジタルファーストの次世代型電子出版プラットフォーム「NextPublishing」を運営する企業です。また自らも、NextPublishing を使った「インターネット白書」の出版など IT 関連メディア事業を展開しています。

※NextPublishing は、インプレス R&D が開発した電子出版プラットフォーム(またはメソッド)の名称です。電子書籍と印刷書籍の同時制作、プリント・オンデマンド(POD)による品切れ解消などの伝統的出版の課題を解決しています。これにより、伝統的出版では経済的に困難な多品種少部数の出版を可能にし、優秀な個人や組織が持つ多様な知の流通を目指しています。

【インプレスグループ】 <https://www.impressholdings.com/>

株式会社インプレスホールディングス(本社:東京都千代田区、代表取締役:唐島夏生、証券コード:東証1部9479)を持株会社とするメディアグループ。「IT」「音楽」「デザイン」「山岳・自然」「旅・鉄道」「学術・理工学」を主要テーマに専門性の高いメディア&サービスおよびソリューション事業を展開しています。さらに、コンテンツビジネスのプラットフォーム開発・運営も手がけています。

【お問い合わせ先】

株式会社インプレス R&D NextPublishing センター

TEL 03-6837-4820

電子メール: np-info@impress.co.jp