```
└─ GAS入門_Gmail(スタンドアロンスクリブト)
   GASフォルダ配下に、本章での作業に使う 「Gmail」フォルダを作成してください。次に、コンテナバインドスクリプト
   を作成します。
   2章で学習したダッシュボードを開き、「新しいプロジェクト」をクリックしてください。

参 自分のプロジェクト - Apps Script × +

    ← → C @ script.google.com/ho

	≡ Magnetic Apps Script

     新しいプロジェクト
     ☆ スター付きのプロジェクト
     □ 自分のプロジェクト
     すべてのプロジェクト
   スタンドアロンスクリプトが開きます。名前は自分がわかりやすいものに変更しておきましょう。
   ここでは「GAS入門_Gmail」とします。
    9.4 スレッドやメッセージを取り扱う - GmailAppクラス
   GmailAppクラスとは、Gmail のスレッドやメッセージおよびラベルを取り扱うクラスです。
   ここでは、GmailAppクラスの中のメソッドについて説明します。
    メソッド
                                    説明
                           戻り値の型
                                    メールを送信する
    sendEmail(recipient, subject, body)
                           GmailApp
   引数は、以下の3つです。
    recipient: 宛先(複数の場合はカンマ区切りのリスト)
    ■ subject:件名
    ■ body:本文
          新規メッセージ
                                                          - * ×
                                 ←recipient
          宛先
                                 ←subject
          件名
                                 ↑ body (本文)
                          ⊕ ⊕ ♠ ■ ♣ ፆ ਜ਼ █ :
            送信
                                                             Û
   GmailAppクラスのメソッド一覧については公式サイトに記載されているため、興味のある方は参照してみてください。
    sendEmail(recipient, subject, body)
   sendEmail(recipient, subject, body) とは、メールを送信するメソッドです。 以下のように使用します。
   スクリプトファイル(見本)
  1 GmailApp.sendEmail('samurai@example.com', 'メールの件名', 'メールの本文');
   それでは、実際にスクリプトを記述してみましょう。先ほど開いたスクリプトエディタで作業します。関数名は自由です
   が、ここでは「myGmailApp」とします。
   GAS入門_Gmail
 1 -function myFunction() {
 3 -}
 4 +function myGmailApp() {
 5 + // 宛先・件名・本文
 6 + let recipient = 'ご自身のメールアドレスに変更してください';
 7 + let subject = 'GASによるメール';
 8 + let body = 'これはGASによるメールです。';
 9 +
 10 + // メール送信
 11 + GmailApp.sendEmail(recipient, subject, body);
 12 }
   recipientには、ご自身のメールアドレスを指定してください。保存後「myGmailApp」を実行して、メールの宛先に送信さ
   れているか確認してください。
   マイドライブにスクリプト (GAS入門_Gmail) が保存されています。GASフォルダ配下のGmailフォルダに移動しておい
   てください。
    9.5 スレッドを取り扱う - GmailThreadクラス
   GmailThreadクラスとは、Gmailのスレッドを取り扱うクラスです。「Gmailのスレッド」とは、メールに対する返信や転
   送など、同じ件名のメールを1つのグループにまとめたものです。
   ここでは、GmailThreadクラスの中のメソッドについて説明します。
    メソッド
                      戻り値の型
                                説明
                      String
                                スレッド内最初のメッセージ件名を取得する
    getFirstMessageSubject()
                                スレッド内のメッセージ数を取得する
    getMessageCount()
                      Integer
                                スレッド内に未読メッセージが含まれているか否かを返す
    isUnread()
                      Boolean
                                このスレッドを既読にする
    markRead()
                      GmailThread
                      GmailThread
                                このスレッドを未読にする
    markUnread()
   GmailThreadクラスのメソッド一覧については公式サイトに記載されているため、興味のある方は参照してみてくださ
   110
   ここでは練習として、メッセージの未読 / 既読を切り替えるスクリプトを作成してみます。 あとで一緒にスクリプトを書
   くため、まずは手順を確認してください。
   (1) Gmailの受信トレイを確認する
   本節では、Gmailの受信トレイ内にあるメッセージの未読/既読を切り替えます。
   そのため、Gmailを初めて使いメールが1通もない状況では作業ができません。もし1通もメールが受信トレイにない場
   合、自分宛てに何通かメール送信しておいてください。
    (2) 受信トレイ内のスレッドを取得する
   まず、受信トレイ内のスレッドを取得するには、GmailAppクラスのgetInboxThreadsメソッドを使用します。第1引数は
   「最初に取得するスレッドのインデックス」、第2引数は「取得するスレッドの最大数」を指定します。
   スクリプト(解説用)
 1 let start = 0;
 2 let max = 2;
 4 // Gmail受信トレイ内のスレッドを取得(最大2スレッド)
 5 let threads = GmailApp.getInboxThreads(start, max);
   (3) スレッドの件名とスレッドに含まれるメッセージ数を取得する
   配列でスレッドが取得されるため、for文で繰り返し処理します。getMessageCount()で、繰り返し処理でスレッドの件
   名とスレッドに含まれるメッセージ数を取得します。
   また、isUnread() でスレッドの未読を判断して、markRead() / markUnread() で未読 / 既読を切り替えます。
   スクリプト(解説用)
 1 // スレッドのルーブ
 2 for(let thread of threads){
 3 // 件名
 4 console.log('件名:' + thread.getFirstMessageSubject());
 5 // メッセージ数
 6 console.log('スレッド内のメッセージ数:' + thread.getMessageCount());
 8 // 未読なら既読に、既読なら未読にする
    if(thread.isUnread()){
     thread.markRead();
 11 } else {
       thread.markUnread();
 13 }
 14 }
   実際に記述しよう
   それでは、実際にスクリプトを記述してみましょう。ここでは、「myGmailThread」という関数を新たに作成します。
   GAS入門_Gmail
   1 + function myGmailThread() {
   2 + let start = 0;
       let max = 2;
      // Gmail受信トレイ内のスレッドを取得(最大2スレッド)
   6+
       let threads = GmailApp.getInboxThreads(start, max);
   9 + for(let thread of threads){
   10 +
         console.log('件名:' + thread.getFirstMessageSubject());
   11+
         console.log('スレッド内のメッセージ数:' + thread.getMessageCount());
   14+
         // 未読なら既読に、既読なら未読にする
        if(thread.isUnread()){
         thread.markRead();
   18 +
         } else {
   19+
          thread.markUnread();
   21 + }
   22 + }
   スクリプトを保存して「myGmailThread」を実行すると、コンソールログに以下のように表示されます。※受信トレイの
   内容で結果は変わります。
 1 件名:サンブルメール
 2 スレッド内のメッセージ数:1
 3 件名:テストメール
 4 スレッド内のメッセージ数:3
   また、実行のたびに先頭2スレッドの既読・未読が切り替わります。なお、受信トレイにメールがないと本スクリプトは動
   きません。

■ M Gmail

                   Q メールを検索
                                                             ② ♦ ፡
     0 作成
                   □ メイン
     □ 受信トレイ
                   🗆 🕁 🐌 自分
                                    サンプルメール - サンプルメールです。
     ☆ スター付き
     ○ スヌーズ中
                                    テストメール - テストメールです。
                   □ ☆ > 自分3
                                    未読(上図)と既読(下図)が切り替わる

■ M Gmail

                   Q メールを検索
                                                             ② ♦ ##
     0 作成
                   □ メイン
     □ 受信トレイ
                                   サンブルメール - サンブルメールです。
                   □☆●自分
     ☆ スター付き
       スヌーズ中
                   □ ☆ ▶ 自分3
        前半はここまで!
        休憩してリフレッシュしましょう
        後半もがんばりましょう!
    9.6 メッセージを取り扱う - GmailMessageクラス
   GmailMessageクラスは、Gmailのメッセージを取り扱うクラスです。
   ここでは、GmailMessageクラスの中のメソッドについて説明します。
    メソッド
               戻り値の型
                        説明
                        メッセージの送信者を取得する
    getFrom()
               String
    getTo()
               String
                        メッセージの宛先を取得する
    getSubject()
               String
                        メッセージの件名を取得する
               Date
                        メッセージの日時を取得する
    getDate()
                        HTML形式を使用せずにメッセージの本文を取得する
    getPlainBody()
               String
   GmailMessageクラスのメソッド一覧については公式サイトに記載されているため、興味のある方は参照してみてくださ
   110
   ここでは練習として、受信トレイ内のメールを取得して件名や本文などの情報を取得するスクリプトを作成してみます。
   あとで一緒にスクリプトを書きますので、まずは手順を確認してください。
   (1) スレッドに含まれるメッセージを配列で取得
   まず、GmailThreadクラスの getMessages() メソッドで、スレッドに含まれるメッセージを配列で取得します。
   スクリプト(解説用)
 1 // スレッドからメッセージ群(返信・転送したメールも含める)の取得
 2 let messages = thread.getMessages();
     (2) メッセージの情報を取得
   次に、メッセージの繰り返し処理でメッセージの情報を取得します。先ほど学習した、以下の5つのメソッドを利用しま
   す。
    getFrom()
    getTo()
    getSubject()
    getDate()

    getPlainBody()

   スクリプト(解説用)
 1 // スレッド内のメッセージループ
 2 for(let message of messages){
 4 // メッセージの情報を取得
 5 console.log('From:' + message.getFrom());
 6 console.log('To:' + message.getTo());
 7 console.log('件名:' + message.getSubject());
 8 console.log('目時' + message.getDate());
    console.log('本文:' + message.getPlainBody());
 10 }
   実際に記述しよう
   それでは、実際にスクリプトを記述してみましょう。ここでは、「myGmailMessage」という関数を新たに作成します。
   GAS入門_Gmail
   1 + function myGmailMessage() {
      let start = 0;
       let max = 2;
   4+
       // Gmail受信トレイ内のスレッドを取得(最大2スレッド)
   5+
   6+
       let threads = GmailApp.getInboxThreads(start, max);
       for(let thread of threads){
   10+
   11+
         let messages = thread.getMessages();
   14+
         for(let message of messages){
   15 +
          // メッセージの情報を取得
          console.log('From:' + message.getFrom());
   18 +
   19 +
          console.log('To:' + message.getTo());
          console.log('件名:' + message.getSubject());
          console.log('日付' + message.getDate());
          console.log('本文:' + message.getPlainBody());
   22 +
   25 +
   26 +
        console.log('----');
   27 + }
   28 + }
   実行すると、最新の2スレッド内すべてのメッセージに対するFrom、To、件名、日時、本文が表示されます。
              A2 十 ち ♂ 団 ▶ 実行 むデバッグ myGmailMessage ▼ 実行ログ
      ファイル
                      // スレッド内のメッセージループ
for(let message of messages){
    ⟨> □-F.gs
      ライブラリ
               サービス
                   実行ログ
    (3)
                   21:03:42 お知らせ 実行開始
                   21:83:43 1848 To: - Ogmail.com>
                   21:83:43 16年 日代Mon Nov 14 2822 89:39:49 GMT+8988 (Japan Standard Time)
                   21:83:43 168 From: - Ognail.com>
                   21:83:43 1666 To: - gmail.com-
                   21:03:43 情報 件名:デストメール
                   21:83:43 情報 日付Mon Nov 14 2822 89:35:88 GMT+8988 (Japan Standard Time)
                   21:83:43 情報 本文:テストメールです。
                   21:03:43 1648 From: - Ognail.com
                   21:03:43 149 To: 0gmail.com
   本章の学習は以上です。お疲れさまでした。
    まとめ
   本章では以下の内容を学びました。
    ■ GmailAppクラスとは、Gmail のスレッドやメッセージおよびラベルを取り扱うクラスで、代表的なメソッドとして以
     下のメソッドがある
    メソッド
                           戻り値の型
                                    説明
                                    メールを送信する
                           GmailApp
    sendEmail(recipient, subject, body)
    ■ GmailThreadクラスとは、Gmailのスレッドを取り扱うクラスで、代表的なメソッドとして以下のメソッドがある
    メソッド
                      戻り値の型
                                説明
                                スレッド内の最初のメッセージの件名を取得する
    getFirstMessageSubject()
                      String
                                スレッド内のメッセージ数を取得する
    getMessageCount()
                      Integer
                      Boolean
                                スレッド内に未読メッセージが含まれているか否かを返す
    isUnread()
    markRead()
                      GmailThread
                                このスレッドを既読にする
    markUnread()
                      GmailThread
                                このスレッドを未読にする
    ■ GmailMessageクラスはGmailのメッセージを取り扱うクラス、代表的なメソッドとして以下のメソッドがある
    メソッド
               戻り値の型
                        説明
                        メッセージの送信者を取得する
    getFrom()
               String
    getTo()
               String
                        メッセージの宛先を取得する
    getSubject()
               String
                        メッセージの件名を取得する
               Date
    getDate()
                        メッセージの日時を取得する
    getPlainBody()
               String
                        HTML形式を使用せずにメッセージの本文を取得する
   Gmailの操作のために、スプレッドシートでスクリプトを書いたので驚いたかもしれません。Gmailは単体ではなくスプレ
   ッドシートやフォームと組み合わせて使うことが多いのです。今から慣れておきましょう。
   次章では、スプレッドシートに記載の顧客リストに、Gmailでメールを自動送信する方法について解説します。
                     理解度を選択して次に進みましょう
                    ボタンを押していただくと次の章に進むことができます
                                                    80~100%
   前に戻る
                                  11 / 13 ページ
                                                                      次に進む
く一覧に戻る
                                       ■ 改善点のご指摘、誤字脱字、その他ご要望はこちらからご連絡ください。
                                       © SAMURAI Inc. 利用規約 プライバシーポリシー 運営会社
```

SAMURAI TERAKOYA

① タイムライン

☆ ホーム

田 教材

→ 課題

Q Q&A

⊘ 学習ログ

の レッスン

よくある質問

リリースノート

教材

Q 検索

ホーム > 教材 > Google App Scriptで業務自動化プログラムを作ろう > Gmailを操作しよう

GASを利用すると、Gmailに対するさまざまな操作を自動化できます。本章では、Gmailに対する基本的な操作方法を解説

GASを利用してメールを操作(作成、送信、検索、削除、など)するには、Gmailサービスを利用します。Gmailサービ

「Gmailのスレッド」とは、メールに対する返信や転送など、同じ件名のメールを1つのグループにまとめたものです。

テストメール - 返信です。 2022年11月14日(月) 9:35

サンプルメール・サンプルメールです。

9章 Gmailを操作しよう

GASでメール送信する方法を学びます。

○50分 ~ □ 未読

9.1 本章の目的

します。

クラス

GmailApp

GmailThread

GmailMessage

■ M Gmail

□ 受信トレイ

☆ スター付き③ スヌーズ中

1 <フォルダ構成>

3 └─ Gmail (フォルダ)

2 GAS

本章では、以下を目標にして学習します。

■ GASでメール送信する方法を理解する

9.2 Gmailサービス

スの代表的なクラスは以下のとおりです。

概要

Gmail のスレッドやメッセージを取り扱う

Gmailのスレッドを取り扱う

Gmailのメッセージを取り扱う

- c :

□ メイン

□ ☆ ▶ 自分

□ ☆ > 自分3

9.3 作業フォルダとファイルを準備しよう

以下のようなフォルダ構成で作業します。

メールのやり取りが1つのスレッドになっている。 1つのスレッドに3つのメッセージが含まれている。

解説に進む前に、まずは作業フォルダと本章で操作するファイルを用意しましょう。

1 GAS(Google App Script)の概

2 スクリプトエディタとダッシュボ

ードの使い方を理解しよう

3 GASの制限事項を理解しよう

Spreadsheetクラス

4 スプレッドシートを操作しよう

5 スプレッドシートを操作しよう

#2 Sheetクラス・Rangeクラス

6 店舗別の売上金額をスプレッドシ

ートで自動集計しよう

#1 SpreadsheetAppクラス・

要を理解しよう

本文 目次 質問一覧 0件