

Google Apps Script を用いたクイズ作成スクリプトの開発

芦生研究林 宮城祐太

1. はじめに

「Google Apps Script は、Google Workspace と統合されるビジネスアプリケーションを迅速かつ簡単に作成できる、アプリケーション開発プラットフォーム」(Google,2024-06-14)である。

Google Apps Script (以下、GAS) は Google アカウントを所有していれば、無料で使用することができるプログラミング言語である。GAS の導入は今回の報告に含めないが、ウェブブラウザで「GAS 導入」などで検索すると導入方法を解説しているウェブサイトが散見されるので、詳細はそちらを参考にしていきたい。

今回は GAS を用いて、Google ドライブ (以下、ドライブ)、Google フォーム (以下、フォーム)、Google スプレッドシート (以下、シート) を連携させ、樹木の画像から樹木名を当てるクイズを、ランダムに 10 問出題するスクリプトを開発した。そこで、スクリプトとクイズの作成方法を報告する。

2. 開発経緯

京都大学フィールド科学教育研究センター (以下、フィールド研) に所属する森林系の教室系技術職員はフィールド研の森林系各施設 (北海道、京都府、和歌山県) に異動をすることがある。職務上樹木に対する知識は必須であるが、異動先で日々の業務に追われつつ樹木を覚えることは容易ではない。そこでフォーム上で樹木画像をランダムに出題し、樹木名を当てる仕組みを作れば、自己学習が捗るのではないかと考え、本スクリプトを開発した。

3. データの準備

3-1 ドライブでのクイズ用画像ファイルの準備

スクリプトでクイズを作成する際、クイズで使用する樹木の画像データを、ドライブ上の任意のフォルダに用意しておく必要がある。図 1 のように「樹木クイズ用」というフォルダを作成し、その中に樹木の画像データを格納した。フォルダ内には画像データのみが格納されている。画像ファイルの拡張子は jpg とした。データ数は試験的に 100 ファイルとした。各ファイル名は樹木名にした。樹木名にした理由は、ファイル名を正答データのためのテキストデータとして使用するためである。また、フォルダ ID を付録 1 のデータ取得スクリプト内で使用するの、画像を格納したフォルダ ID (今回の例では樹木クイズ用フォルダの ID) を事



図 1 ドライブ上に用意した .jpg 画像

前に取得した。フォルダ ID はフォルダ URL(<https://drive.google.com/drive/u/1/folders/xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx>)の xxx の箇所である。

3-2 シートでのスクリプト実行準備とフォームの作成

クイズは、付録 1,2 のスクリプトをシート内で実行することで生成される。付録 1 はクイズに使用する画像や樹木名を集約するための「データ取得スクリプト」で、付録 2 は、付録 1 で取得したデータを元にクイズを作成する「クイズ生成スクリプト」である。スクリプトの実行には、新規のシートを作成する必要がある。新規に作成したシートの拡張機能→Apps Script より付録 1,2 のスクリプトを転記する。付録 1 と 2 のスクリプトは、別のスクリプトファイルにする必要がある。付録スクリプトのコピーが終了した状態は図 2 のようになる。



図 2 付録のスクリプトを転記した状態

付録 2 のクイズ作成スクリプトではクイズ表示のためのフォーム ID が必要になる。事前に新規のフォームを作成し ID を取得した。フォーム ID はフォーム URL (<https://docs.google.com/forms/d/e/xx>) の xxx の箇所である。

3-3 データ取得スクリプトの実行によるクイズデータの集約

次に、これまでの情報を含めて、クイズに用いる情報を集約したシートを作成する。付録 1 のスクリプトを実行することにより、任意のシートに以下の情報が自動的に挿入される。1 行目：各列の表題、A 列(question)：フォーム上で出題した樹木画像にマウスオーバーした際に表示される文章、B 列(answer)：画像に対応する樹木名（正答データ、画像ファイル名から自動取得）、H 列(file_url)：画像ファイル URL を API (Application Programming Interface) 形式に変更したもの。シートの C~F,G 列は自動で挿入されないため、入力する必要がある。C~F 列(fake)は、クイズの誤答となる選択肢である。B 列の答えを貼り付けた後、シートの標準機能である「範囲のランダム化」を用いると簡易的に列を埋めることができる。ただし同一問題内に同一選択肢が存在すると、付録のスクリプト 2 を実行した際にエラーが発生するので、同一行内の B~F 列の値(答えと選択肢)は全て異なるようにする必要がある。また「範囲のランダム化」では、クイズ上でスギの画像が表示されている時に、選択肢が「トチノキ」といったような、あからさまに異なる選択肢になる場合もあるので、問題のレベルに応じて調整が必要である。G 列にはクイズに対するフィードバックを記入する。例えばスギであれば、「針葉樹」などの樹木に関する情報を記載する。最終的に必要な情報が記載されたシートは図 3,4 のようになる。このシートの 2 行目以降の各行が 1 問に対応する。

	A	B	C	D	E	F
1	question	answer	fake_1	fake_2	fake_3	fake_4
2	この画像の樹種名は？	アセビ	ソヨゴ	ヤマグルマ	クサギ	サワフタギ
3	この画像の樹種名は？	トチノキ	コシアブラ	ホオノキ	フカノキ	アケビ

図3 シート上に必要な情報を記載した状態(A-F列)

G	H
feedback	file_url
互生で僅かに鋸歯がある葉。常緑落葉樹。毒があるので、馬が食べたときにフラフラした様子から馬が酔っ木と書いてアセビと呼ばれるようになった。	https://drive.google.com/uc?id=[redacted]
対生で鋸歯がある葉。掌状複葉。落葉広葉樹。トチは「十」「千」と書き、多くの実をつけることからこう呼ばれた。トチの実は、山間部の冬場の貴重な蛋白質源(トチ餅等)にな	https://drive.google.com/uc?id=[redacted]

図4 シート上に必要な情報を記載した状態(G-H列)

4. クイズの作成

上記データの準備を行い、クイズデータが集約されたシートを用意した状態で、付録2のスク립トを実行すると、IDを指定したフォーム上にクイズが生成される。シートからランダムに10行が選択されることで、図5のような形式で10問連続してクイズが表示される。上部に画像が表示され、その下に正解と誤答の選択肢がランダムに配置される。任意の選択肢を選んで、最後に回答を行うことで、採点を行い、また各問題のフィードバックが表示される。

また付録2のスク립トを実行した後、シートのメニューバーには「スク립ト実行」というメニューが追加される。「スク립ト実行」からクイズ作成をクリックすることで、クイズの更新を行えるようになる。このスク립トはIDで指定したフォーム上のクイズを一度すべて消去し、再度クイズを生成するので、同じフォームURLに異なる問題を表示することができる。

さらに、GASの機能である「トリガー」を活用することで、スク립トを自動的に実行し、クイズを自動更新することが可能である。例えば、ホームページ上にフォームURLを公開し、トリガー機能を活用すれば、毎日自動更新されるオリジナルのクイズコンテンツを提供することが可能になる。

5. まとめ

当初は樹木のみを想定して画像クイズ生成スク립トの開発を始めたが、あらゆる生物や景色の画像等にも対応可能な汎用性の高いスク립トを開発することができた。さらに、事前に



図5 クイズ生成後のフォーム画面

動画投稿サイトに鳥の鳴き声等をアップロードしておけば、画像のみならず音声や動画のクイズにも拡張可能であり、使用の幅も大きく広がる。

今回のスクリプトは私が独学で作成した。スクリプトの作成には、インターネットを通じて無償公開されている情報が大変役に立った。例えば、問題の選択肢をランダムにする方法や、ドライブ上の画像をスクリプトにて読み込む方法は、個人のブログで公開されているスクリプト例が参考になった。ただし、今回付録として公開したスクリプトは変数の宣言や記述方法にはまだ改善の余地があると考えられる。

本スクリプトは選択式のクイズを生成するものである。今後記述式に対応したメソッドが公開されれば、上記の改善を行うとともに、より学習を深化させることができると考えられる記述式クイズに対応したスクリプトも開発したい。

6. 謝辞

芦生研究林のデータフォルダに保存されていた画像を拝借し、本スクリプトを開発しました。芦生研究林に在籍されていた教職員ならびに関係者の皆様にお礼申し上げます。また、本報告を執筆するにあたり、文章表現等の助言をいただきました松岡講師にお礼申し上げます。

7. 参考文献

Google.”Google Apps Script の概要”.GoogleWorkspace.2024-03-06.<https://developers.google.com/apps-script/overview?hl=ja>,(参照 2024-06-14)

Google.”Class DriveApp”.GoogleWorkspace.2023-12-01.<https://developers.google.com/apps-script/reference/drive/drive-app?hl=ja>,(参照 2024-06-14)

Google.”Class FormApp”.GoogleWorkspace.2023-12-01.<https://developers.google.com/apps-script/reference/forms/form-app?hl=ja>,(参照 2024-06-14)

Google.”Class SpreadsheetApp”.GoogleWorkspace.2023-12-01. <https://developers.google.com/apps-script/reference/spreadsheet/spreadsheet-app?hl=ja>,(参照 2024-06-14)

8. 付録

```
function getdata() {  
  
    //現在開いているシートを取得 (sheet という定数を設定)  
    const sheet = SpreadsheetApp.getActiveSheet();  
  
    //フォルダ ID を定数として設定 フォルダ ID はフォルダにより異なる  
    const folderId = "xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx";  
  
    //folder という定数を設定  
    const folder = DriveApp.getFolderById(folderId);  
  
    //contents という定数を設定  
    const contents = folder.getFiles();  
}
```

```

//スプレッドシート先頭行の配列
const title =
['question', 'answer', 'fake_1', 'fake_2', 'fake_3', 'fake_4', 'feedback', 'file_url']

//先頭行にタイトルを挿入
for(let i = 0; i < title.length; i++){
  sheet.getRange(1, 1 + i).setValue(title[i]);
}

//file という変数を準備
var file;

//ドライブから取得したファイル名とファイル URL を二行目から挿入するために、
row=2 を設定
var row = 2

//文字列置換用に B 列にファイル名を挿入し、H 列にファイル URL を挿入する
while(contents.hasNext()) {
  file = contents.next();
  sheet.getRange(row, 2).setValue(file.getName());
  sheet.getRange(row, 8).setValue(file.getUrl());
  row++;
}

//最終行を取得する
const lastRow = sheet.getLastRow();

//範囲を取得
const nameRange = sheet.getRange('B2:B' + lastRow)
const urlRange = sheet.getRange('H2:H' + lastRow);

//値を取得
let nameValues = nameRange.getValues();
let urlValues = urlRange.getValues();

//空配列を用意
let deleteArray = [];
let replaceArray = [];

//配列に格納
for(let i = 0; i < urlValues.length; i++){

```

```

deleteArray.push(nameValues[i][0]);
replaceArray.push(urlValues[i][0]);
}

//処理後の空配列を準備
let deletedArray = [];
let replacedArray = [];

//ファイル名には.JPG や.jpg といった拡張子が含まれ、正答のテキストに使用できない
//ので整形
for(let i = 0; i < deleteArray.length; i++){
    deleteArray[i]= deleteArray[i].replace(/.jpg/i ,");
    deletedArray.push(deleteArray[i]);
}

//ファイル URL はフォームで直接読み込めないなので、URL の一部を置換
!個人ブログを参考にしています。!

for(let urlValues of replaceArray){
    let urlIndex = replaceArray.indexOf(urlValues);
    replacedArray[urlIndex] =
urlValues.replace('file/d/','uc?id=').replace('/view?usp=drivesdk','&.jpg');
    //replacedArray[urlIndex] = urlValues.replace('/view?usp=drivesdk','&.jpg');
}

//A 列にはマウスオーバーした際に表示される文字列を挿入 B 列に整形済みのファイル
//名を挿入 H 列に整形済みの URL を挿入
for(let i = 0; i < lastRow; i++){
    sheet.getRange(i + 2, 1).setValue('この画像の樹木名は?');
    sheet.getRange(i + 2, 2).setValue(deletedArray[i]);
    sheet.getRange(i + 2, 8).setValue(replacedArray[i]);
}
}

```

付録 1 データ取得スクリプト

```

function CreateQuizForm() {
// スプレッドシートから質問リストを取得
const sheet = SpreadsheetApp.getActiveSheet();

//最終列を取得
const lastColumn = sheet.getLastColumn();

```

```

//最終行を取得
const lastRow = sheet.getLastRow();

//質問、解答、解答の解説、画像 ID を含む範囲を指定
const range = sheet.getRange(2,1,lastRow,lastColumn);

//範囲の値を取得
var data = range.getValues();

//data に各項目を挿入
for (let i = 0; i < data.length - 1; i++) {
  var question = data[i][0]; // 質問
  var answer = data[i][1]; // 解答
  var choice1 = data[i][2]; // 選択 1
  var choice2 = data[i][3]; // 選択 2
  var choice3 = data[i][4]; // 選択 3
  var choice4 = data[i][5]; // 選択 4
  var explanation = data[i][6]; // 各樹木のフィードバック
  var fileUrl = data[i][7]; // 画像化した URL
}

//空の配列を準備
var randInt = [];

//乱数を発生させて、問題をランダム化
for(let i=0;i<10;i++){
  //2 行目から最終行数-1 までの乱数を生成
  let rand = Math.floor( Math.random() * (lastRow - 1)) + 2;

  //同じ問題が出題されないように調整
  if ( i == 0){
    i = i;
  }else{
    var check = randInt.includes(rand);
  }
  if (check == false || i == 0){
    randInt.push(rand);
  } else{
    i = i - 1;
  }
}
}

```

```

//フォームを設定 xxx にフォーム ID を入力する。ID の前後には' ' を入れる
const form = FormApp.openById('xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx');

//既設の質問を削除
var items = form.getItems();
while (items.length > 0) {
  form.deleteItem(items.pop());
}

//フォームをクイズ形式に設定
form.setIsQuiz(true);

// 質問を追加
for (let x = 0; x <= randInt.length- 1; x++) {

  // 画像の取得とフォームへの追加
  var img = UrlFetchApp.fetch(data[randInt[x]][7]);
  form.addItem()
    .setHelpText([data[randInt[x]][0]]) // 画像にマウスオーバーした際に表示する
文章
    .setImage(img);
  var choices;

  //選択肢をランダムにするための前処理  選択肢をランダムにしないと、全ての問題で
最初の選択肢が正解になってしまう
  var array=[
    [data[randInt[x]][1],true],
    [data[randInt[x]][2],false],
    [data[randInt[x]][3],false],
    [data[randInt[x]][4],false],
    [data[randInt[x]][5],false]
  ]

  //選択肢をランダムにする
  for(let i = array.length - 1; i > 0; i--){
    var r = Math.floor(Math.random() * (i + 1));
    var tmp = array[i];
    array[i] = array[r];
    array[r] = tmp;
  }
}

```

```

//ラジオボタン設置
var item = form.addMultipleChoiceItem();
    item
        .setTitle(data[randInt[x]][0])

        //選択肢の設定
        .setChoices([
            item.createChoice(...array[0]),
            item.createChoice(...array[1]),
            item.createChoice(...array[2]),
            item.createChoice(...array[3]),
            item.createChoice(...array[4]),
        ]);

//1問1点に設定
item.setPoints(1);

//選択強制
item.setRequired(true);

//フィードバックの設定（正解不正解に関わらず、同一のフィードバックを返す）
item.setFeedbackForCorrect(FormApp.createFeedback().setText(data[randInt[x]][6]).build());

item.setFeedbackForIncorrect(FormApp.createFeedback().setText(data[randInt[x]][6]).build());

    }
}

```

付録2 クイズ作成スクリプト