

# 目次

1. 本システムの目的 3
2. 全体構想 4
3. 処理の概要 5
  - 3-1. オンライン授業の実施 5
  - 3-2. 教室管理者によるオンライン授業の後の作業 8
  - 3-3. 生徒に見える映像(課題) 17
  - 3-4. 講師の作業 18
4. 処理の詳細 21
  - 4-1. ソースコード 21
  - 4-2. 本システムの使用方法 26
5. 運用上の注意点 37
6. 残された課題 39
  - 6-1. ユーザーインターフェイスの首尾一貫性 39
  - 6-2. Youtube への配信 39
7. サンプルプログラムによる事前技術的検討例 40
  - 7-1. Google Apps Script(GAS) で、Google Classroom に投稿する 40
  - 7-2. Google Apps Script : 時間主導のトリガを使用して、CRON のように一定間隔でスクリプトを実行する 41
  - 7-3. Google Apps Script : 動画を添付して Classroom の指定のトピックへ課題として投稿する 47
8. 補足 48
  - 8-1. 着想の解説 48
  - 8-2. 行った機能調査 49



## 1. 本システムの目的

Zoom (SB C&S 株式会社) を使ってオンライン授業を実施し、Zoom の機能を使ってその映像を録画する。録画された映像データは WEB 環境で利用可能なファイル形式に変換後、クラウド上に保存する。その録画された映像データは自習教材として、生徒や教師が使いやすい状態に整理して、ユーザインターフェイスにつなぐ。

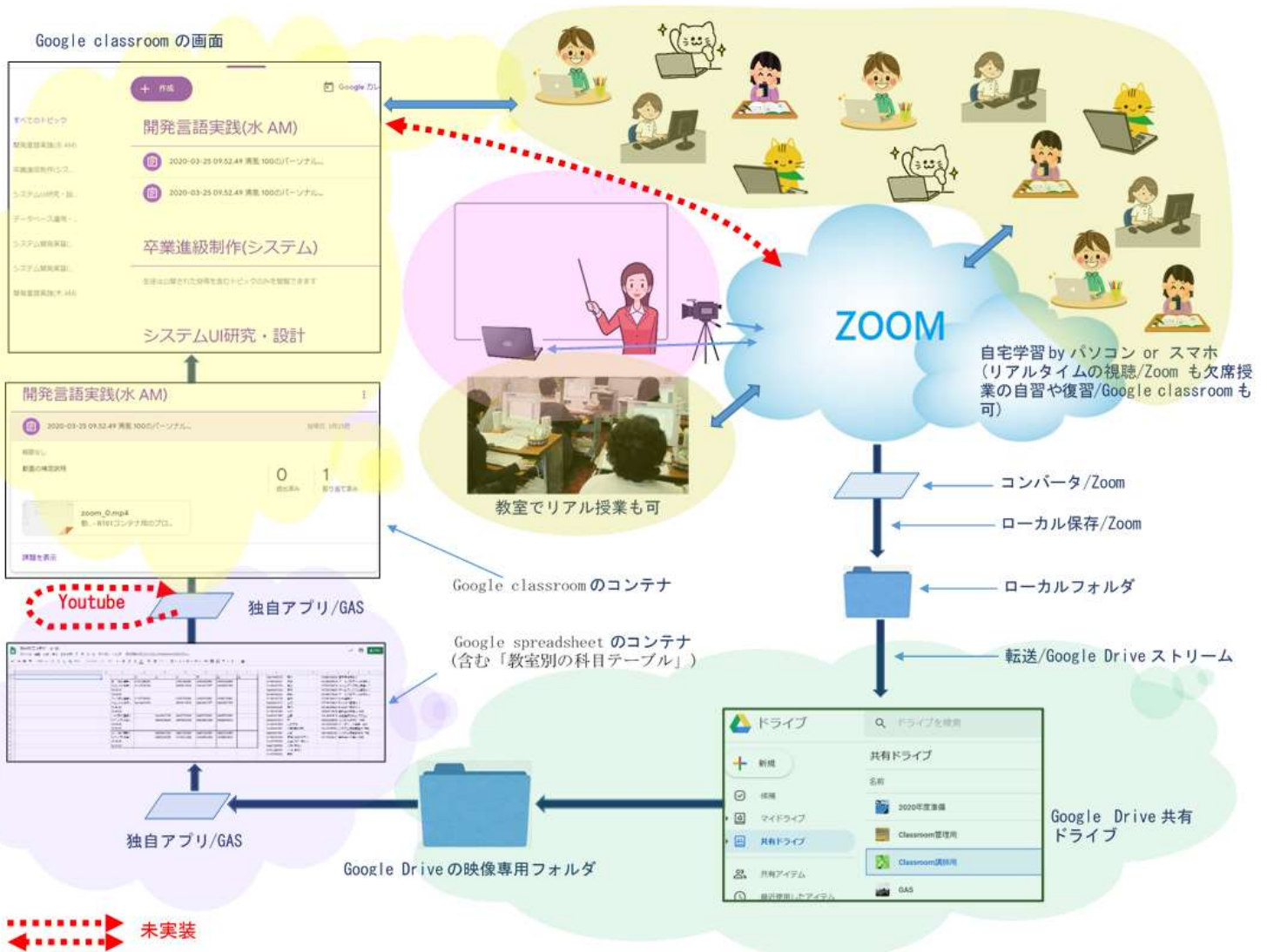
本システムは、これらの処理が、可能な限り人手を介することなくできるよう支援する目的で構築されたものである。

本システムが本格的に稼働すれば、オンライン授業の実施の翌日から生徒は復習のためにこれを活用したり、オンライン授業に参加できなかった生徒は自習するためにこの映像教材を活用することができることが期待されるものである。

## 2. 全体構想

「オンライン授業 to 自習映像教材：シームレス運用支援システム」の全体の構造は数のとおりです。

講師の演ずる教育活動が Zoom を通じて配信されるだけでなく、Google classroom とシームレスに連動することによって、事後も繰り返し利用可能なコンテンツがほぼ自動的に生成し、そのまま翌日には利用が可能となります。



赤の破線矢印で示した「ユーチューブへの配信」と「Zoom を Google classroom の配下に加える」については、現在検討中であるが、実装はしていない。こんごのかだいとします。

### 3. 処理の概要

Zoom(SB C&S 株式会社)を使って撮影した動画を G Suite(Google) のサービスである Google Classroom の『課題』として添付して投稿する。

#### 3-1. オンライン授業の実施

##### (1) サインイン

教室管理者が教師と生徒を招待するので、招待された教師と生徒は Zoom にサインインする。



##### (2) 新規ミーティング

教師と生徒は教室管理者から招待されて新規のミーティングに参加することになります。



### (3) このコンピュータにレコーディング

オンライン授業を録画(レコーディング)するためには、教室管理者が、教師画面のポップアップメニューから「このコンピュータにレコーディング」を選ぶ。

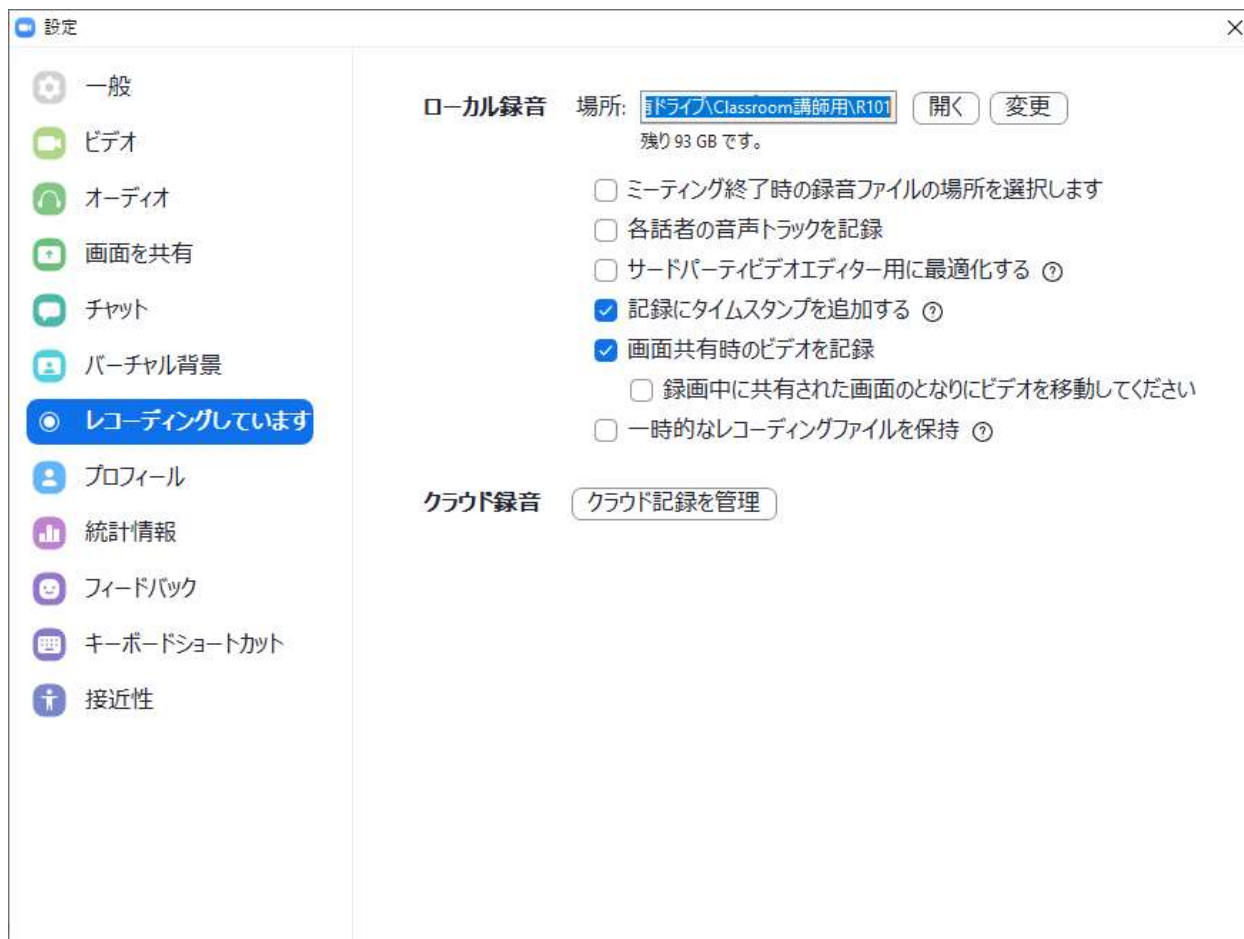


オンライン授業が終了した際に画面右下の「ミーティングの終了」を選ぶと、WEB環境で視聴できる動画ファイル(mp4形式)が作成されます。

### 3-2. 教室管理者によるオンライン授業の後の作業

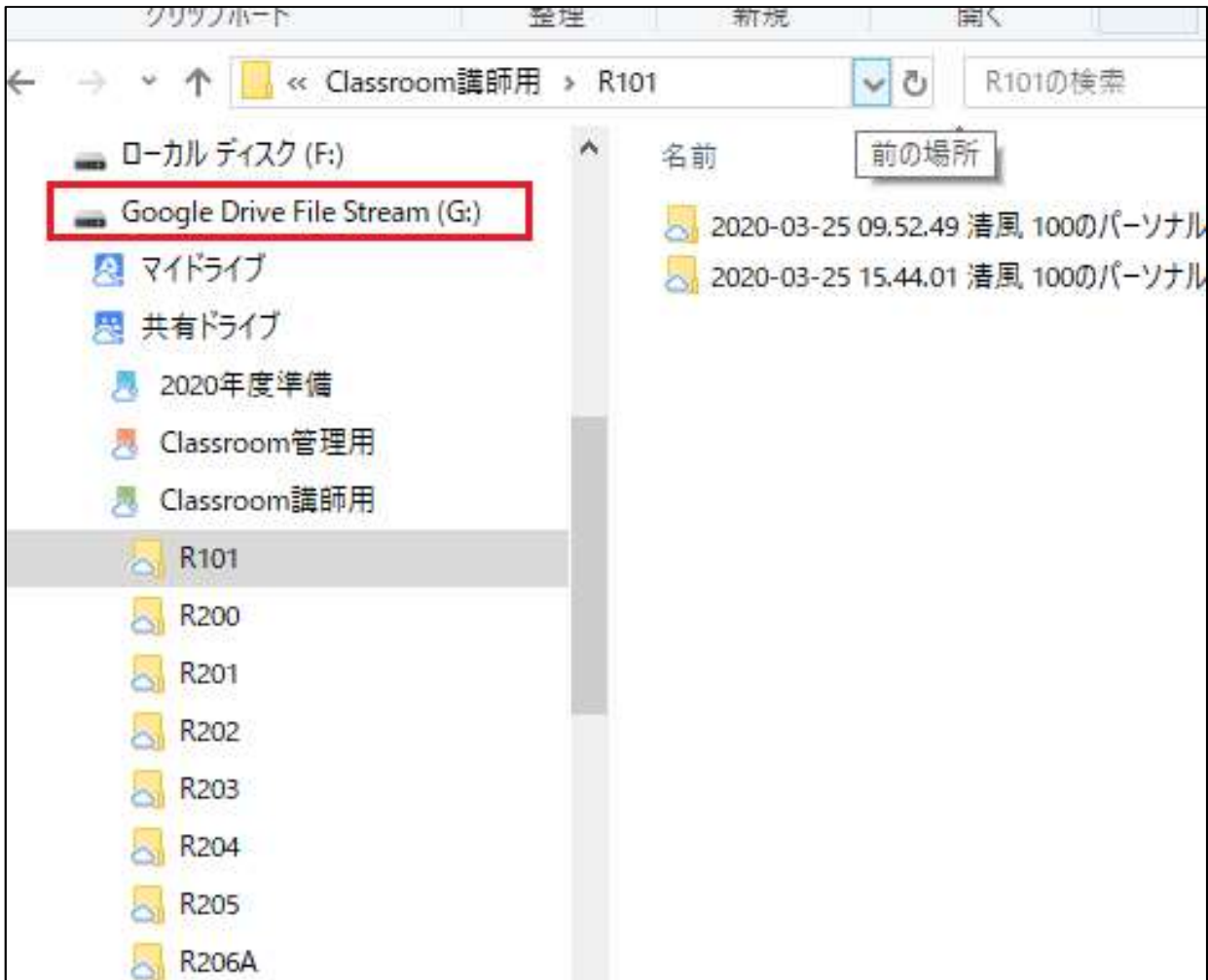
作成される場所は、設定でローカルのフォルダを選択できます。

#### (1) 映像をローカルフォルダに保存



(2) 共有ドライブへの格納

この際指定するフォルダを、Google Drive File Stream をインストールして G Suite での共有ドライブの中のフォルダを指定する事によって、mp4(動画ファイル) が自動的にアップロードされます。





Google Drive File Stream は、以下の URL より Windows 版と Mac 版がダウンロード可能です

<https://support.google.com/drive/answer/7329379>

## ドライブ ファイルストリームのスタートガイド

ドライブ ファイルストリームを使用すると、パソコンで Google ドライブからファイルを検索して開くことができます。具体的には、次の操作を行うことができます。

- 共有ドライブも含め、特定のファイルやフォルダをオフラインで保存
- 容量を使用せずにパソコンのファイル システムでファイルを表示、整理
- パソコンでファイルを開く

---

### ドライブ ファイルストリームをダウンロードしてインストールする ▲

ドライブ ファイルストリームは、利用できない場合やユーザーの代わりに組織がインストールを行う場合があります。ご不明な点がある場合は、[管理者にお問い合わせください](#)。

ドライブ ファイルストリームをダウンロードするには:

[WINDOWS 版をダウンロード](#) [MAC 版をダウンロード](#)

1. お使いのパソコンで、対応するインストーラを実行します。
  - Windows の場合: **DriveFileStream.exe**
  - Mac の場合: **DriveFileStream.dmg**
2. 表示される手順に沿って操作します。

### (3) 共有フォルダ内の教室別のフォルダとアップロードされるファイル

#### • 教室のアカウント

動画を作成する単位は、学校における教室単位になります。

ここがまず最初に重要で、複数の教室で授業が展開されますが、教室に対応する Zoom のアカウントを一対一で準備します。

#### • 教室アカウントの例

ここで、仮に R101 で行われるオンライン授業については、以下のように定義します。

アカウント : R101 専用 ZOOM

共有ドライブ管理者アカウント : classroom@i-seifu.jp

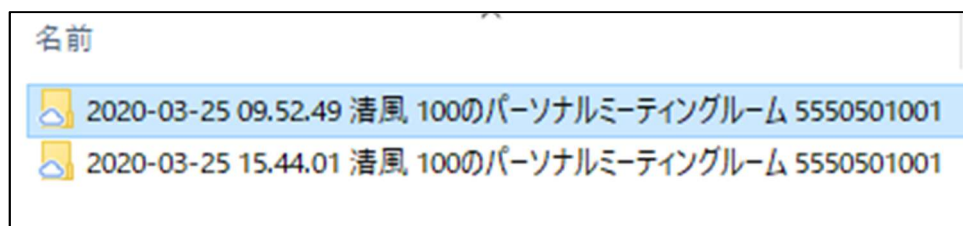
アップロード先 : 上記アカウントが作成した共有フォルダの中に作成した R101 というフォルダ

つまり、R101 教室で記録された動画は必ず共有フォルダ内の R101 というフォルダに

アップロードされます。

・アップロードされるもの

Zoom でアップロードされるファイルは、フォルダであり、フォルダの中に実際の mp4 ファイルが 1 つ以上保存されます。フォルダ名の先頭には、録画を開始した日付と時分秒が記録されています。



この日付と時間を使って、R101 教室で使用される科目の時間割テーブルより Classroom と科目を特定します。

※ Google の内部的には、クラスはコースと呼ばれ、科目はトピックと呼ばれます

(4) 教室別の科目テーブル

例えば、R101 教室で行われる科目について着目すると、以下が R101 の科目テーブルとなります。

	C	D	E	F	G	H	I
		月	火	水	木	金	土
コースID(講師)		54407226065	54408403250	54407226065	54407226065	54407226065	
トピック(科目)		54434493630	56066536269	56098112610	54416452377	54433667289	
09:20:00							
10:59:00							
コースID(講師)		54407226065	54408403250	54407226065	54407226065	54407226065	
トピック(科目)		54434493630	56066536269	56098112610	54416452377	54433667289	
11:00:00							
12:40:00							
コースID(講師)				54407226065	54407226065	54407226065	
トピック(科目)				54416452382	54433667284	54422856612	
13:30:00							
15:09:00							
コースID(講師)				54407226065	54407226065	54407226065	
トピック(科目)				54416452382	54433667284	54422856612	
15:10:00							
16:50:00							





#### (6) Google ドライブ内のファイルの移動について

Google ドライブでは、場所を移動しても、ファイルの ID が変わる事がないのでこのような事が可能です

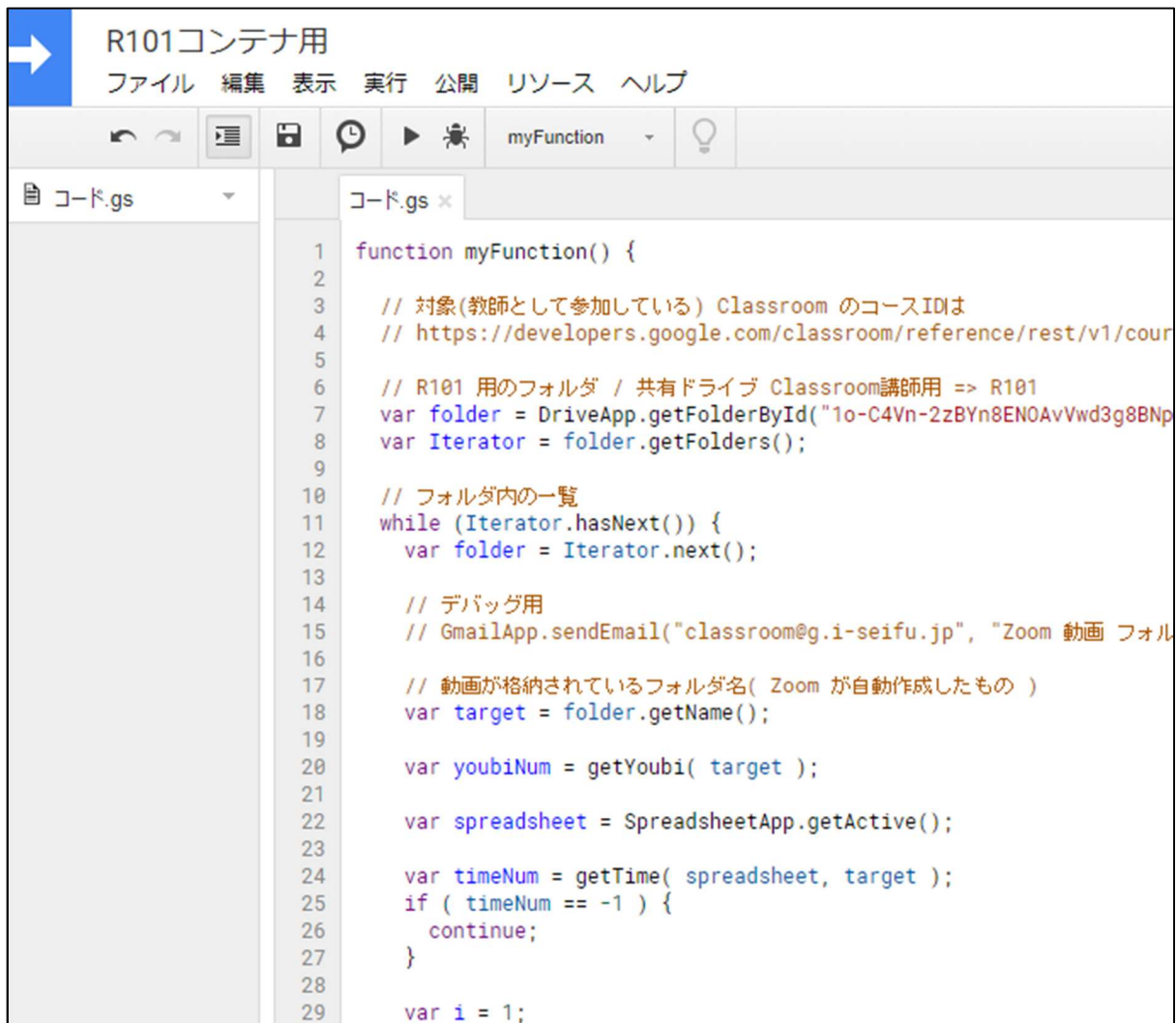
実際のプログラムは、Google Apps Script という JavaScript を使った API を利用します。

そのスクリプト用のエディタは、コンテナ( スプレッドシート )のメニューより開く事ができ、『プロジェクト』という位置付けになります



## (7) スクリプトエディタ

スクリプトエディタを選ぶと次のようなウィンドウが開かれます。ここでは「myFunction」が選ばれています。



```
1 function myFunction() {
2
3 // 対象(教師として参加している) Classroom のコースIDは
4 // https://developers.google.com/classroom/reference/rest/v1/cour
5
6 // R101 用のフォルダ / 共有ドライブ Classroom講師用 => R101
7 var folder = DriveApp.getFolderById("1o-C4Vn-2zBYn8EN0AvVwd3g8BNp
8 var Iterator = folder.getFolders();
9
10 // フォルダ内の一覧
11 while (Iterator.hasNext()) {
12     var folder = Iterator.next();
13
14     // デバッグ用
15     // GmailApp.sendEmail("classroom@g.i-seifu.jp", "Zoom 動画 フォル
16
17     // 動画が格納されているフォルダ名( Zoom が自動作成したもの )
18     var target = folder.getName();
19
20     var youbiNum = getYoubi( target );
21
22     var spreadsheet = SpreadsheetApp.getActive();
23
24     var timeNum = getTime( spreadsheet, target );
25     if ( timeNum == -1 ) {
26         continue;
27     }
28
29     var i = 1;
```

## (8) トリガーを使用しての起動

このコードの起動方法は、テスト中はエディタより関数を選択して実行が可能です。実際の運用時は、『トリガー』という時間主導で定期的に起動できるサービスに登録して実行させます。

画面遷移を以下の連続する図群によって示します。



### R101コンテナ用のトリガーを追加

実行する関数を選択	エラー通知設定 +
myFunction ▼	毎日通知を受け取る ▼
実行するデプロイを選択	
Head ▼	
イベントのソースを選択	
時間主導型 ▼	
時間ベースのトリガーのタイプを選択	
日付ベースのタイマー ▼	
時刻を選択	
午前 3 時~4 時 ▼	
(GMT+09:00)	
	キャンセル <span style="background-color: #0070C0; color: white; padding: 2px 5px;">保存</span>

#### (9) コンテナの作成

教室管理者は、このコンテナを教室ごとに作成し、Classroom に教師を招待した後にクラスコードを生徒に配布して参加してもらいます。



### 3-3. 生徒に見える映像(課題)

#### (1)生徒の Classroom の中

この事によって、翌日には自分の受けた授業の動画が、自分の参加する Classroom の中にあるトピックに課題として投稿されます。

次には、トピックに掲載された科目名と映像名の例を示します。





映像名の一つをクリックすると次のようになります。

## (2)映像の一つを選ぶ

下記のような画面に遷移します。ここで、「課題を選ぶ」を選択すると映像が視聴ができるようになる。



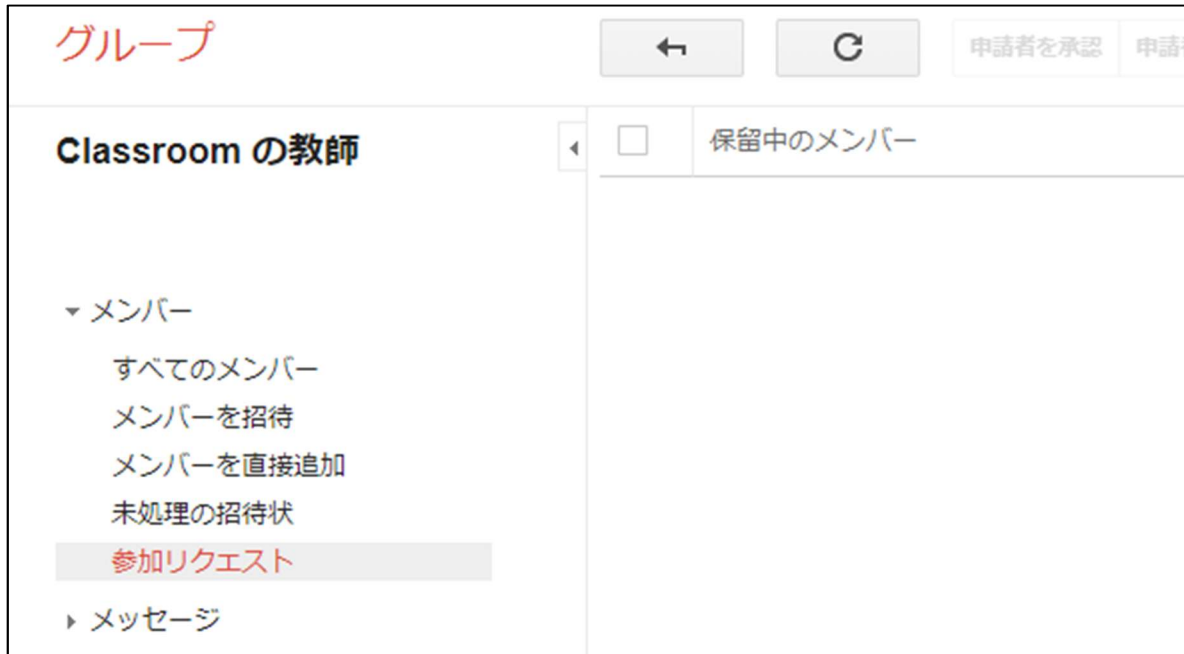
## 3-4. 講師の作業

Class ルームの名前は講師名で作成すると解りやすいですが、副担任のように複数の登録も可能です。

しかし、決められた時間の範囲に行われる科目は一つである必要があります。

そして、各 Classroom は classroom@i-seifu.jp が管理者として全て作成して教師を招待します。

その際、Google 的には一時的な教師登録となり、管理者が自動的に作成させる Google グループ内で『承認』する必要があります。

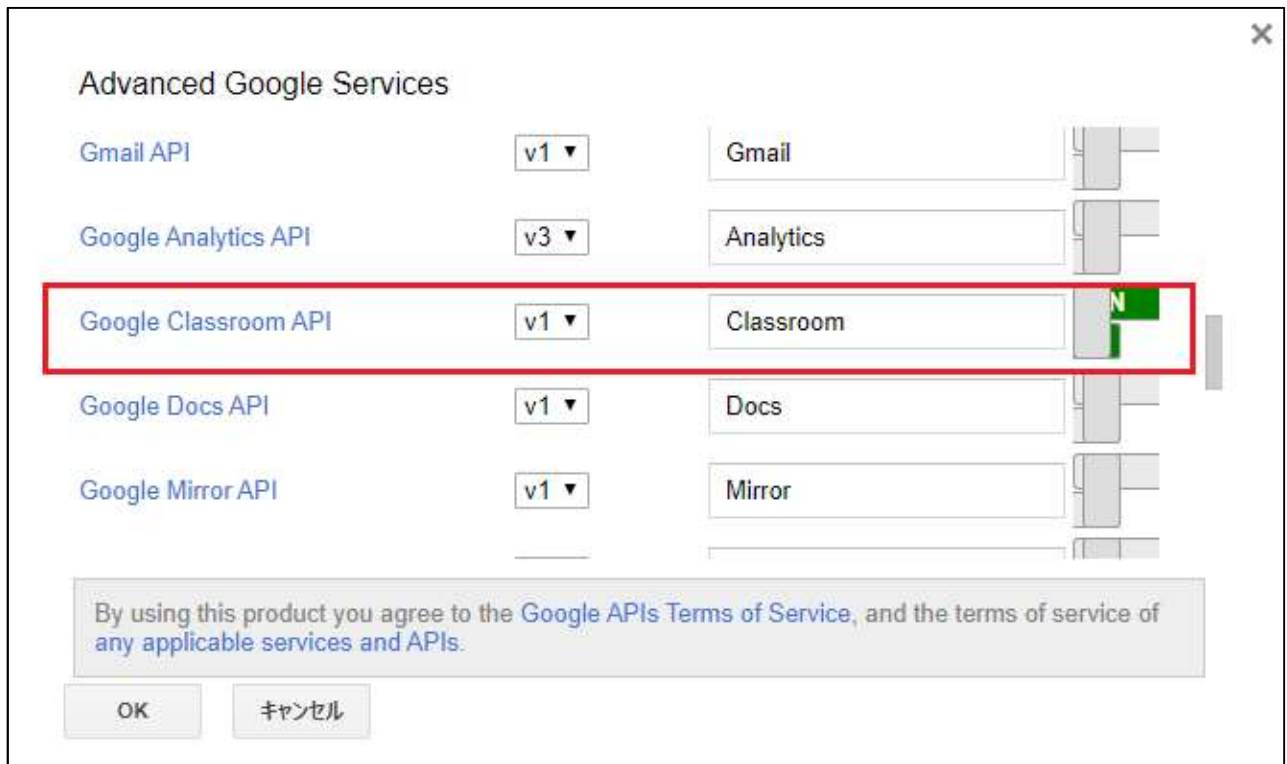


この「参加リクエスト」内に承認すべき教師がリストでならんでいるので、管理者権限のあるアカウントで、承認作業をして下さい。上の図は承認後ですので、教師リストが消えたあとです。

承認すれば、下の画面のように「すべてのメンバー」のところに移動されます。



Google Apps Script で使用できる標準 API では、Classroom の API は使用できないので、一旦 classroom@i-seifu.jp のアカウントで Google Cloud Console で利用の承認を行い、API の利用をチェックする必要があります( Google Cloud Console のプロジェクトを作る必要はありません )



さらに最初の実行時には、アカウントと権限の確認ダイアログで承認する必要があります。

## 4. 処理の詳細

### 4-1. ソースコード

起動はトリガで夜間バッチです。

処理は、主として下記の時間割テーブルを対象に行われる。

#### ▼ 時間割テーブル

C	D	E	F	G	H	I
	月	火	水	木	金	土
コースID(講師)	54407226065		54407226065	54407226065	54407226065	
トピック(科目)	54434493630		56098112610	54416452377	54433667289	
09:20:00						
10:59:00						
コースID(講師)	54407226065		54407226065	54407226065	54407226065	
トピック(科目)	54434493630		56098112610	54416452377	54433667289	
11:00:00						
12:40:00						
コースID(講師)		54408403250	54407226065	54407226065	54407226065	
トピック(科目)		56066536269	54416452382	54433667284	54422856612	
13:30:00						
15:09:00						
コースID(講師)		54408403250	54407226065	54407226065	54407226065	
トピック(科目)		56066536269	54416452382	54433667284	54422856612	
15:10:00						
16:50:00						

ソースコードは以下の通りです。

```
// *****  
// 教室と一対一のフォルダより新しく登録された  
// フォルダの中にあるZoom動画ファイルを該当する  
// Classroom のコース内の該当するトピック  
// に登録する  
// *****  
function myFunction() {  
    // 教室用のフォルダ / 共有ドライブ に作成済  
    // ( id は、表示されている URL より取得 )  
    var folder = DriveApp.getFolderById("1o-C4Vn-2zBYn8ENOAvVwd3g8BNp9xS-E");  
    var Iterator = folder.getFolders();  
  
    // フォルダ内の一覧  
    while (Iterator.hasNext()) {  
        var folder = Iterator.next();  
  
        // デバッグ用  
        // GmailApp.sendEmail("メールアドレス", "Zoom 動画 フォルダ名", folder.getName());  
  
        // 動画が格納されているフォルダ名( Zoom が自動作成したもの )  
        var target = folder.getName();  
    }  
}
```

```

// Zoom フォルダ名より曜日番号を取得
var youbiNum = getYoubi(target);

var spreadsheet = SpreadsheetApp.getActive();

var timeNum = getTime(spreadsheet, target);
if (timeNum == -1) {
    continue;
}

var i = 1;
var exist = false;

while (true) {
    // 登録済のフォルダを排除する為に順に比較していく
    var targetRange = spreadsheet.getRange('A' + i);
    var cellWork = targetRange.getValue().toString();
    if (cellWork == "") {
        // 新規なので、スプレッドシートの A 列の最後に追加する
        targetRange.setValue(target);

        // MIME で絞ってファイル一覧を取得
        var files = folder.GetFilesByType("video/mp4");
        // フォルダ内の動画ファイルをチェック
        while (files.hasNext()) {

            var file = files.next();

            // 拡張子 .mp4 のみを取得
            var work = file.getName();
            var exts = work.split(".");
            if (exts[1] != "mp4") {
                continue;
            }

            // Google ドライブ上の 動画の ID
            var fileId = file.getId();
            var courseId = getCourseId(spreadsheet, youbiNum, timeNum);
            // 本来は条件を元に決定する
            var topicId = getTopicId(spreadsheet, youbiNum, timeNum);

            // 課題作成用の JSON
            var json = {
                "materials": [
                    {
                        "driveFile": {
                            "driveFile": {
                                "id": fileId
                            }
                        },
                        "shareMode": "VIEW"
                    }
                ],
                "description": "動画の補足説明",
                "assigneeMode": "ALL_STUDENTS",
                "state": "PUBLISHED",
                "title": target,
                "topicId": topicId,
                "workType": "ASSIGNMENT"
            };

            // デバッグ用
            // GmailApp.sendEmail("メールアドレス", "JSON", JSON.stringify(json));

```

```

        // 課題を投稿
        try {
            Classroom.Courses.CourseWork.create(json, courseId);
        }
        catch(e) {
            targetRange.setValue("");
            GmailApp.sendEmail("メールアドレス", "Classroom 投稿エラー", target);
        }
    }
}
break;
}
else {
    if (cellWork == target) {
        exist = true;
        break;
    }
    else {
        i++;
    }
}
}
}
}

// *****
// Zoom フォルダ名より曜日番号を取得
// *****
function getYoubi(folderName) {
    var ymdString = folderName.substr(0, 10);
    var dateData = new Date(ymdString);
    var youbiNum = dateData.getDay();

    return youbiNum;
}

// *****
// Zoom フォルダ名より時間割テーブル上の
// 時限番号を取得
// *****
function getTime(spreadsheet, folderName) {
    var timeNum = -1;

    var rangeData1 = [];
    var rangeData2 = [];

    // C 列にある時限の時間範囲を配列で取得
    for (var i = 1; i < 10; i++) {
        rangeData1.push(spreadsheet.getRange("C" + (4 + 4 * (i - 1))).getValue().toString());
        rangeData2.push(spreadsheet.getRange("C" + (5 + 4 * (i - 1))).getValue().toString());
    }
}

```

```

// var rangeData1 = ["09:20:00", "11:00:00", "13:30:00", "15:10:00"];
// var rangeData2 = ["10:59:00", "12:40:00", "15:09:00", "16:50:00"];

// 対象の時刻部分を取得
var ymdString = folderName.substr(0, 19);
ymdString = ymdString.replace(/¥./g, ":");
var dateData = new Date(ymdString);

// 範囲を持つ配列り時限を決定
for (var i = 0; i < 9; i++) {
    if (rangeData1[i] != "") {
        var rangeWork1 = new Date(folderName.substr(0, 10) + " " + rangeData1[i]);
        var rangeWork2 = new Date(folderName.substr(0, 10) + " " + rangeData2[i]);
        if (rangeWork1 <= dateData && dateData <= rangeWork2) {
            timeNum = i + 1;
        }
    }
}

return timeNum;
}

// *****
// 曜日と時限よりコースを取得
// *****
function getCourseId(spreadsheet, youbiNum, timeNum) {
    var youbiRange = ["", "D", "E", "F", "G", "H", "I"];

    var targetRange = spreadsheet.getRange(youbiRange[youbiNum] + (2 + 4 * (timeNum - 1)));
    var courseId = targetRange.getValue().toString();

    return courseId;
}

// *****
// 曜日と時限よりトピックを取得
// *****
function getTopicId(spreadsheet, youbiNum, timeNum) {
    var youbiRange = ["", "D", "E", "F", "G", "H", "I"];

    var targetRange = spreadsheet.getRange(youbiRange[youbiNum] + (3 + 4 * (timeNum - 1)));
    var topicId = targetRange.getValue().toString();

    return topicId;
}

// *****
// 作業用のコースID 一覧(日本語名も)を K・L
// 列に作成
// トピック一覧も N・O 列に作成
// *****
function setCourseId() {
    var spreadsheet = SpreadsheetApp.getActive();
    var json = Classroom.Courses.list();
    var cnt = json.courses.length;

```

```

}
// *****
// 作業用のコースID 一覧( 日本語名も ) を K・L
// 列に作成
// トピック一覧も N・O 列に作成
// *****
function setCourseId() {
    var spreadsheet = SpreadsheetApp.getActive();
    var json = Classroom.Courses.list();
    var cnt = json.courses.length;
    for (var i = 0; i < cnt; i++) {
        var targetRange = spreadsheet.getRange('K' + (i + 1));
        targetRange.setValue(json.courses[i].id);
        targetRange = spreadsheet.getRange('L' + (i + 1));
        targetRange.setValue(json.courses[i].name);
    }
    var jsonTopic = null;
    var l = 0;
    for (var i = 0; i < cnt; i++) {
        jsonTopic = Classroom.Courses.Topics.list(json.courses[i].id);
        try {
            for (var j = 0; j < jsonTopic.topic.length; j++) {
                var targetRange = spreadsheet.getRange('N' + (l + 1));
                targetRange.setValue(jsonTopic.topic[j].topicId);
                targetRange = spreadsheet.getRange('O' + (l + 1));
                targetRange.setValue(jsonTopic.topic[j].name);
                l++;
            }
        } catch(e) {
        }
    }
}
}
}

```

上記ソースコードによって、教室と一対一のフォルダより新しく登録されたフォルダの中にある Zoom 動画ファイルを該当する Classroom の コース内の該当するトピックに登録することができます。



## 4-2. 本システムの生成とその確認

コンテナの機能を中心に画面の様子を確認しながら「オンライン授業 to 自習映像教材 シームレス運用支援システム」を用いて実際に運用するシステムを生成する手順とその結果得られる動作の確認を示します。

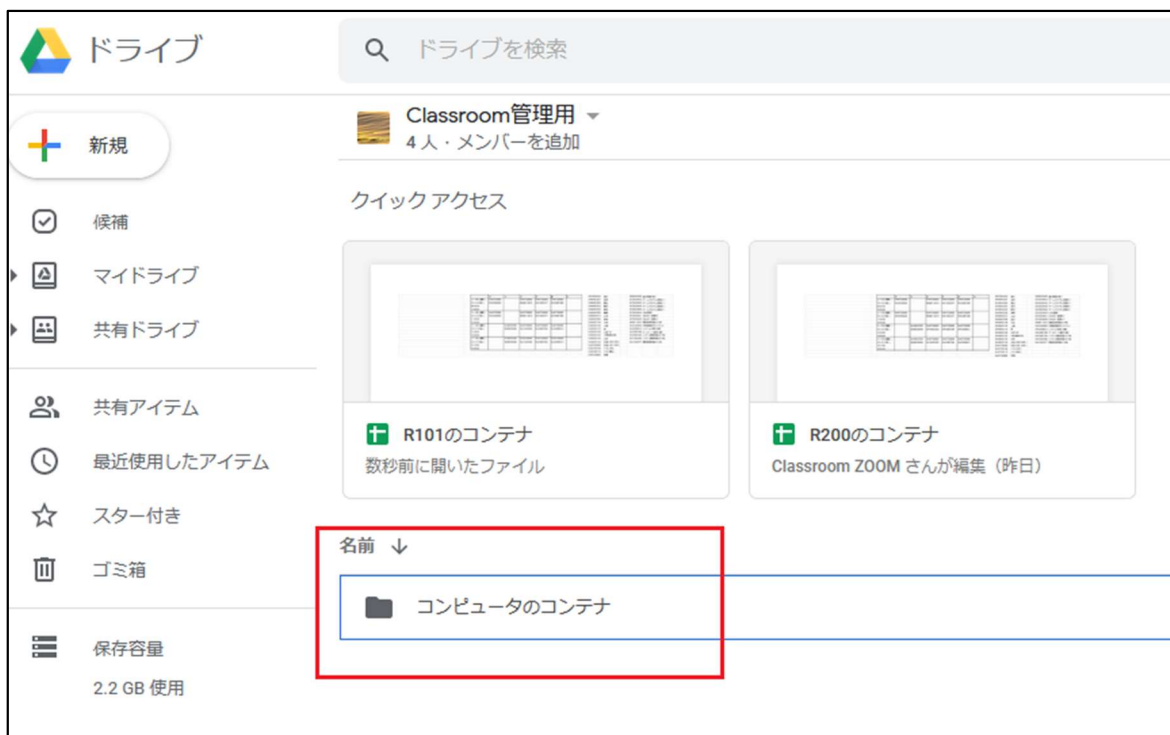
### (1) 教室管理機能の生成

教室管理者は、グーグルドライブを用いて、各教室ごとのコンテナを生成しておきます。その様子は、次のようにすることによってみることができます。

まず、グーグルドライブ-共有ドライブ-「classroom 管理用」を選びます。



次に「コンピュータのコンテナ」を選択します。



ここで、「コンピュータのコンテナ」を選択します。

ここでは、サンプルとして「R101」と名付けられた教室のデータを見てゆくことにします。



当該コンテナの画面が開かれます。

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1				月	火	水	木	金	土		56476830030	堀川		56066536269	基本情報対策 I	
2			コースID(講師)	54407226065		54407226065	54407226065	54407226065			54408403257	老田		55728238433	ゲームプログラム実習 D I	
3			トピック(科目)	54434493630		56098112610	54416452377	54433667289			54408403250	尾立		55728238430	ゲームプログラム実習 C I	
4			09:20:00								54408403243	長井		55728238429	ゲームプログラム実習 B I	
5			10:59:00								54408403235	勝谷		55706979929	ゲームプログラム実習 A I	
6			コースID(講師)	54407226065		54407226065	54407226065	54407226065			54408403228	美野		55728238424	Unity基礎 I	
7			トピック(科目)	54434493630		56098112610	54416452377	54433667289			54408403213	土井		55728238421	DirectX11基礎 B I	
8			11:00:00								54408403206	若本		55728238420	DirectX11基礎 A I	
9			12:40:00								54408403199	中田		56098112610	開発言語実践(水 AM)	
10			コースID(講師)		54408403250	54407226065	54407226065	54407226065			54408403190	上野		54434493630	卒業進級制作(システム)	
11			トピック(科目)		56066536269	54416452382	54433667284	54422856612			54408403179	林		54422856612	システムUI研究・設計	
12			13:30:00								54408403169	山田マホ		54433667289	データベース運用・設計	
13			15:09:00								54408403162	中野(専攻2年)		54433667284	システム開発実習(水 PM)	
14			コースID(講師)		54408403250	54407226065	54407226065	54407226065			54408403155	山田		54416452377	システム開発実習(水 PM)	
15			トピック(科目)		56066536269	54416452382	54433667284	54422856612			54408403148	友会(2022 年卒)		54416452377	開発言語実践(水 AM)	
16			15:10:00								54407226256	友会(2021 年卒)				
17			16:50:00								54407226186	八木(本科)				
18											54407226116	八木(専攻)				
19											54407226065	百寿				
20																

ここで、「スクリプトエディタ」を選びます。

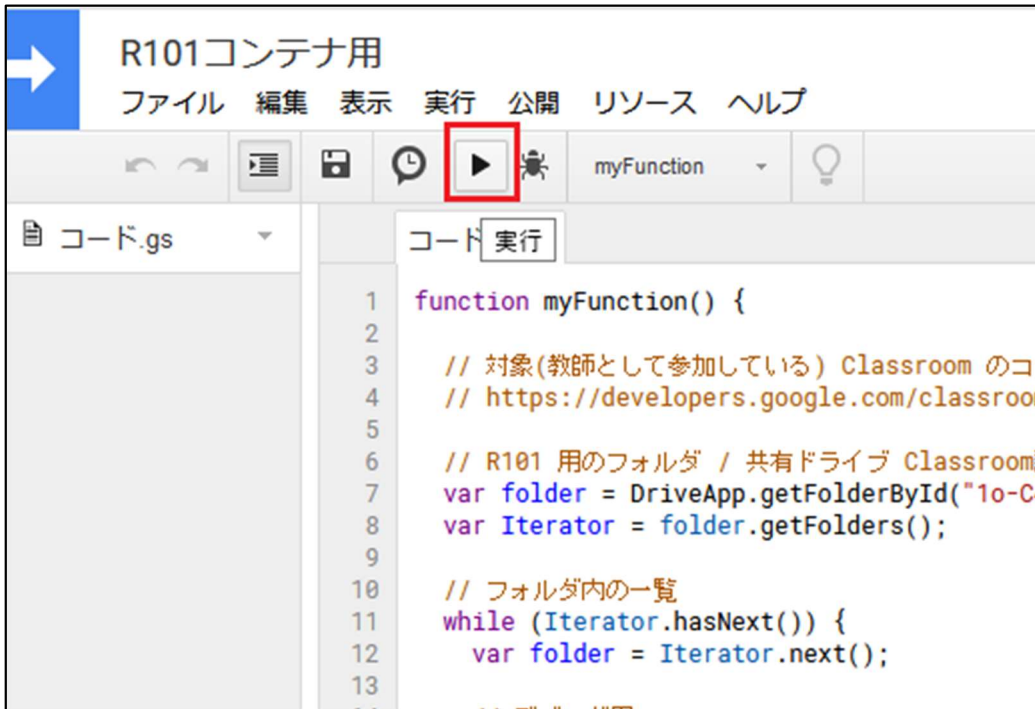
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1				火	水	木	金	土			56476830030	堀川		56066536269	基本情報対策 I	
2			コースID(講師)	54407226065		54407226065	54407226065	54407226065			54408403257	老田		55728238433	ゲームプログラム実習 D I	
3			トピック(科目)	54434493630		56098112610	54416452377	54433667289			54408403250	尾立		55728238430	ゲームプログラム実習 C I	
4			09:20:00								54408403243	長井		55728238429	ゲームプログラム実習 B I	
5			10:59:00								54408403235	勝谷		55706979929	ゲームプログラム実習 A I	
6			コースID(講師)	54407226065		54407226065	54407226065	54407226065			54408403228	美野		55728238424	Unity基礎 I	
7			トピック(科目)	54434493630		56098112610	54416452377	54433667289			54408403213	土井		55728238421	DirectX11基礎 B I	
8			11:00:00								54408403206	若本		55728238420	DirectX11基礎 A I	
9			12:40:00								54408403199	中田		56098112610	開発言語実践(水 AM)	
10			コースID(講師)		54408403250	54407226065	54407226065	54407226065			54408403190	上野		54434493630	卒業進級制作(システム)	
11			トピック(科目)		56066536269	54416452382	54433667284	54422856612			54408403179	林		54422856612	システムUI研究・設計	
12			13:30:00								54408403169	山田マホ		54433667289	データベース運用・設計	
13			15:09:00								54408403162	中野(専攻2年)		54433667284	システム開発実習(水 PM)	
14			コースID(講師)		54408403250	54407226065	54407226065	54407226065			54408403155	山田		54416452377	システム開発実習(水 PM)	
15			トピック(科目)		56066536269	54416452382	54433667284	54422856612			54408403148	友会(2022 年卒)		54416452377	開発言語実践(水 AM)	
16			15:10:00								54407226256	友会(2021 年卒)				
17			16:50:00								54407226186	八木(本科)				
18											54407226116	八木(専攻)				
19											54407226065	百寿				
20																

続いて、オンライン授業で作成された映像データを共有フォルダの所定のフォルダに格納し、同時にコンテナの内容を最新状態に書き換えます。そのため、まず「関数を選択」を選んで「myFunction」を選びます。

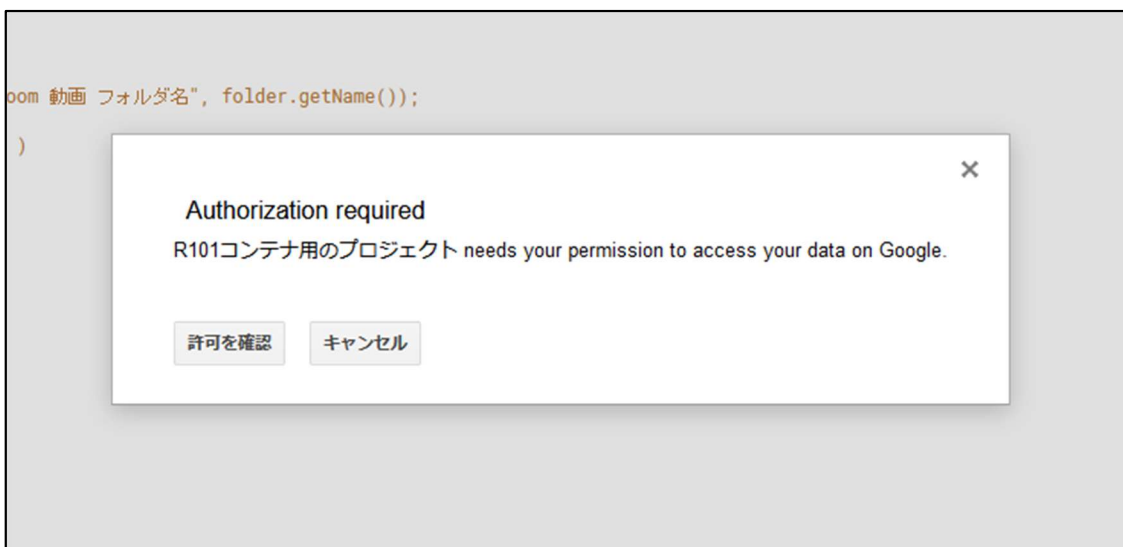
```
1 function my getYoubi
2
3 // 対象(教 getTime Classroom のコースIDは
4 // https: getCourseId e.com/classroom/reference/rest/v1/courses/list より手動で取得
5
6 // R101 用 getTopicId タイプ Classroom講師用 => R101
7 var folde lderById("1o-C4Vn-2zBYn8ENOAvVwd3g8BNp9xS-E");
8 var Itera setCourseId lders());
9
10 // フォルダ内の一覧
11 while (Iterator.hasNext()) {
12   var folder = Iterator.next();
13
14   // デバッグ用
15   // GmailApp.sendEmail("classroom@g.i-seifu.jp", "Zoom 動画 フォルダ名", folder.getName());
16
17   // 動画が格納されているフォルダ名( Zoom が自動作成したもの )
18   var target = folder.getName();
19
20   var youbiNum = getYoubi( target );
21
22   var spreadsheet = SpreadsheetApp.getActive();
23
24   var timeNum = getTime( spreadsheet, target );
25   if ( timeNum == -1 ) {
26     continue;
27   }
28
29   var i = 1;
30   var exist = false;
31
```

「myFunction」を実行すれば、オンライン授業で作成された映像データを共有フォルダの所定のフォルダに格納されます。

「▶」ボタンをクリックすると、今選ばれている関数「myFunction」が実行され、オンライン授業で作成された映像データを共有フォルダの所定のフォルダに格納し、同時にコンテンツの内容を最新状態に書き換えます。



ここで、初回のみ以下のプロンプトがポップアップするので、「許可を確認」を選択します。



次の画面では「(管理者)@i-seifu.jp」を選択します。



次の画面に遷移するので、「許可」ボタンをクリックする。



(2) 確認

ここからは、生成された結果を確認していきます。

「myFunction」の実行後、グーグルドライブ-共有ドライブ-「classroom 管理用」を選びます。

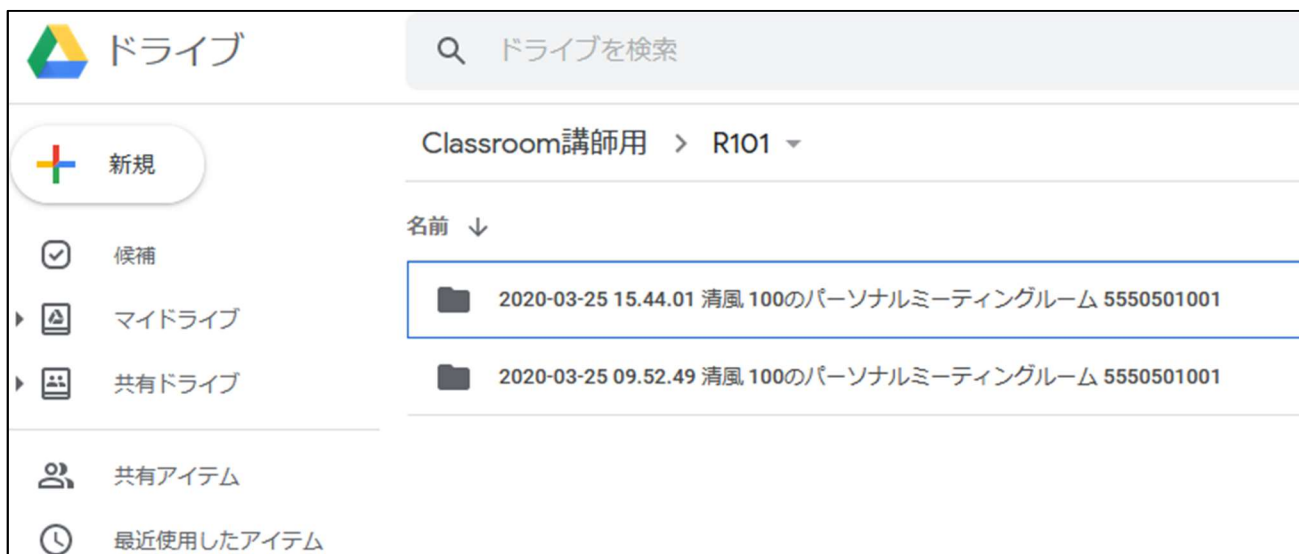


次に表示された画面の「フォルダメニュー」からオンライン授業を行った教室番号(ここでは「R101」)が付いたフォルダを選択します。

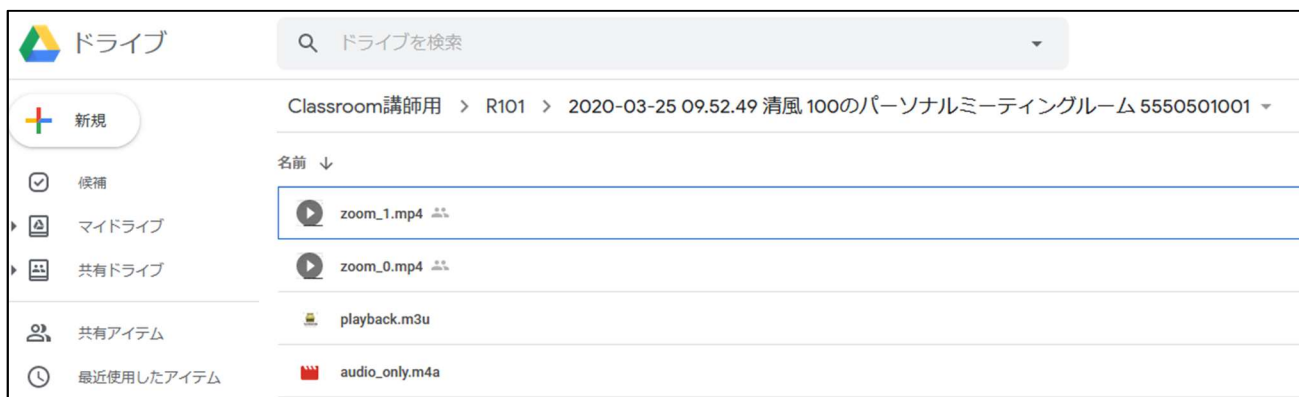
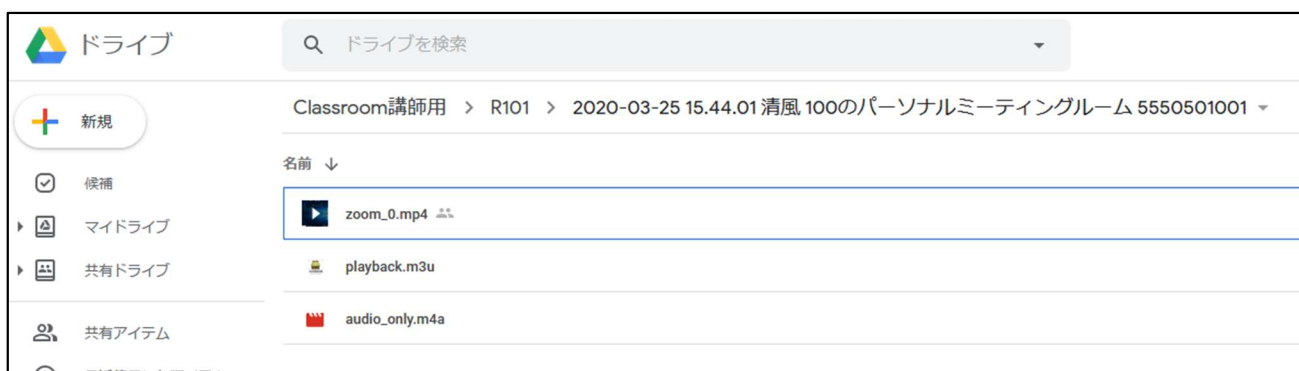
The screenshot shows the Google Drive interface for a Classroom instructor. At the top, there is a search bar with the text "ドライブを検索". Below the search bar, the user is identified as "Classroom講師用" with a dropdown arrow and "20人・メンバーを追加" below it. A "クイックアクセス" section displays two video thumbnails: "講師と学生.mp4" (shared within the last week) and "zoom\_1.mp4" (shared within the last week in a Classroom Zoom session). Below this is a "名前" (Name) dropdown menu. The main area contains a list of folders: "資料用", "アーカイブ", "R301A", "R300", "R206A", "R205", "R204", "R203", "R202", "R201", "R200", and "R101". The "R101" folder is highlighted with a red rectangular box. On the left sidebar, there are navigation options: "新規" (New), "候補" (Candidates), "マイドライブ" (My Drive), "共有ドライブ" (Shared Drives), "共有アイテム" (Shared Items), "最近使用したアイテム" (Recently Used Items), "スター付き" (Starred), "ゴミ箱" (Trash), and "保存容量" (Storage Capacity) showing "2.2 GB 使用". At the bottom left, there is a "管理コンソール" (Management Console) link.



「R101」のフォルダを開けると次のような画面になりますので、該当する授業の日付と時間が確認できます。



該当する授業のフォルダを開けると次のように、必要なファイル(.mp4 ファイルほか)が入っていることが判ります。



確認作業開始前と後のコンテナの様子は次の通りです。

<前>

月	火	水	木	金	土	日	月	火	水	木	金	土	日
コースID(講師)	5440722606	5440722606	5440722606	5440722606	5440722606		5647683000	堀川	5606653629	基本情報対策 I			
トピック(科目)	5443449363	5609811261	5441645237	5443366728			54408403257	老田	5572823843	ゲームプログラム実習 C I			
09:20:00							54408403259	尾立	5572823849	ゲームプログラム実習 C II			
10:59:00							54408403243	野谷	55706979929	ゲームプログラム実習 B I			
コースID(講師)	5440722606	5440722606	5440722606	5440722606			54408403228	野野	55728238424	Unity基礎 I			
トピック(科目)	5443449363	5609811261	5441645237	5443366728			54408403213	土井	55728238421	DirectX11基礎 B I			
11:00:00							54408403206	岩本	55728238420	DirectX11基礎 A I			
12:40:00							54408403199	中田	56098112610	開発言語実務(水 AM)			
コースID(講師)	54408403250	5440722606	5440722606	5440722606			54408403190	上野	54434493630	卒業論文制作(システム)			
トピック(科目)	5606653629	5441645238	54433667284	54422856612			54408403179	林	54422856612	システムUI研究・設計			
13:30:00							54408403169	山田マホ	54433667289	データベース運用・設計			
15:09:00							54408403162	中野(専攻2年)	54433667284	システム開発実習(水 PM)			
コースID(講師)	54408403250	5440722606	5440722606	5440722606			54408403155	山田	54416452382	システム開発実習(水 PM)			
トピック(科目)	5606653629	5441645238	54433667284	54422856612			54408403148	友金(2022年卒)	54416452377	開発言語実務(水 AM)			
15:10:00							54407226186	八木(主幹)					
16:50:00							54407226116	八木(専攻)					
							54407226065	百海					

<後>

月	火	水	木	金	土	日	月	火	水	木	金	土	日
2020-03-25 15:44:01 演習100のパーソナルミーティングルーム 5550501001							5647683000	堀川	5606653629	基本情報対策 I			
2020-03-25 09:52:49 演習100のパーソナルミーティングルーム 5550501001	コースID(講師)	5440722606		5440722606	5440722606	5440722606	54408403257	老田	5572823843	ゲームプログラム実習 C I			
	トピック(科目)	5443449363		5609811261	5441645237	5443366728	54408403259	尾立	5572823849	ゲームプログラム実習 C II			
							54408403243	野谷	55706979929	ゲームプログラム実習 B I			
							54408403235	野谷	55706979929	ゲームプログラム実習 B I			
							54408403228	野野	55728238424	Unity基礎 I			
							54408403213	土井	55728238421	DirectX11基礎 B I			
							54408403206	岩本	55728238420	DirectX11基礎 A I			
							54408403199	中田	56098112610	開発言語実務(水 AM)			
							54408403190	上野	54434493630	卒業論文制作(システム)			
							54408403179	林	54422856612	システムUI研究・設計			
							54408403169	山田マホ	54433667289	データベース運用・設計			
							54408403162	中野(専攻2年)	54433667284	システム開発実習(水 PM)			
							54408403155	山田	54416452382	システム開発実習(水 PM)			
							54408403148	友金(2022年卒)	54416452377	開発言語実務(水 AM)			
							54407226256	友金(2021年卒)					
							54407226186	八木(主幹)					
							54407226116	八木(専攻)					
							54407226065	百海					

(3) Google classroom での確認

Google classroom の画面では、次のように表示されますので日時を確認して必要な授業の映像を選択して「課題を表示」を選ぶと、教室管理者が格納し、講師が許可を与えた映像を見ることができます。

ストリーム 授業 メンバー 採点

+ 作成 📅 Google カレンダー 📁 クラスのドライブ フォルダ

すべてのトピック

- 開発言語実践(水 AM)
- システム開発実習(...)
- 卒業進級制作(シス...
- システムUI研究・設...
- データベース運用・...
- システム開発実習(...)
- 開発言語実践(木 AM)

### 開発言語実践(水 AM) ⋮

📅 2020-03-25 09:52.49 清風 100のパーソナル...
投稿日: 18:37

期限なし

動画の補足説明

0  
提出済み

1  
割り当て済み

**zoom\_0.mp4**

動画 - R101コンテナ用のプロ...

課題を表示

📅 2020-03-25 09:52.49 清風 100のパーソナル...
投稿日: 18:37

### システム開発実習(水 PM) ⋮

📅 2020-03-25 15:44.01 清風 100のパーソナル...
投稿日: 18:36

## 5. 運用上の注意点

### (1) 教室移動の可能性

注意しなければならないのは、同一のクラスの同一の科目であっても場合によって教室を変更しなければならない場合があります。

新型コロナウイルスなど感染症の対策として、本システムを活用する場合、生徒は、通常運用時は「2. 全体構想」の図のように、家庭で受講する者が大多数で、教室で参加する者は通常よりかなり少なくなると推測されます。

この場合は、教室内でもまばらに着席することになるので、密集、密着になりにくいと考えられます。換気に十分気をつけることによって、感染の危険を減少させることが可能です。したがって、このときの講師と出校した生徒は通常通りの教室に入って、対面で学習活動を行いながら、在宅学生たちとはネットを介して学習活動を行うこととなります。

一方、生徒らが長期間出校しないままにするのは学習への緊張感の維持が困難と考えられます。そのため、一定の頻度(例えば週に1回など)では、クラス別に、全員出席日を設けることにします。

全員出席日には、生徒全員が通常の教室にそのまま入室すると密集、密着状態となり、日本語の学習などでは、間断なく大きな発声発話が行われるので、感染の原因となる飛沫が相当程度まき散らされる危険性があります。

感染の危険を避けるため、全員出席日には、普段より大きな教室に変更して、生徒らに間隔をあけて着席してもらう必要があります。

## (2) 教卓パソコンが問題

さて、教室を変えると困ったことが起きます。教室を変えると教卓のパソコンも変更先の教室の教卓パソコンに変わってしまいます。A教室を本拠地としている甲先生がB教室に代わると、B教室のパソコンはいつもは別の乙先生が使っているパソコンです。乙先生が使っているパソコンは甲先生が使っているとは知らずに「乙先生の教科を始める」と勝手に判断してしまいます。教科名が違ってしまったり、映像ファイルが保存されるフォルダが違ってしまふ恐れがあります。

実は、各教室にあるパソコンには、

- ・教室番号と対応付けられている「Zoom ID」が設定されています。
- ・教室番号がついている保存場所に Zoom のファイルを書き込むように設定されています。

A教室のパソコンがノートパソコンで、甲先生がそのノートパソコンをB教室に運ん使用すれば、問題が起こりません。A教室のパソコンがデスクトップパソコンで、持ち運びできない場合は、B教室のパソコンの設定をA教室と同じにして使用して、授業の終わりにはもとに戻す必要があります。

### a) もともとの教室のパソコンがノートパソコンだったら

そのノートパソコンを移動先の教室に運んで使用します。

ノートパソコンを持ち運ぶと、置忘れやとり違いがしばしば発生することが予想されるので、ノートパソコンの蓋の上には、次のようなゼッケンを張り付けてまぎれのないようにしておくことが望まれます。



b) もともとの教室のパソコンがデスクトップパソコンだったら

Zoom ID をもとの教室(A 教室)の ID に変更する。

Zoom の映像ファイル保存場所(フォルダ)を A 教室の映像ファイル保管場所(フォルダ)に設定しなおす。

Zoom を立ち上げる。

〈通常のオンライン授業を行う〉

撮影済みファイルのコンバートを済ませて、ローカルフォルダに保存する。

B 教室の教卓パソコンの Zoom ID を元の(B 教室固有の) ID に戻す。

B 教室の教卓パソコンの Zoom のファイル保存場所(フォルダ)を B 教室の映像ファイル保管場所(フォルダ)に設定を戻す。

を行います。

## 6. 残された課題

### 6-1. ユーザーインターフェースの首尾一貫性

現在のシステムでは、生徒たちはリアル・オンライン授業では Zoom へのサインインから始めて、復習するときや欠席した授業を改めて受講する場合は Google classroom へのサインインから始めなければなりません。

将来は、どちらの場合でも Google classroom へのサインインから始められるように統一をはかるべきと思われます。つまり、リアル・オンライン授業であっても、Google classroom のメニューから選択できるように機能を改めるようにしたいと考えます。

### 6-2. Youtube への配信

Youtube はダウンロードやコピーができない使用のため、コンテンツの著作権の保護などの観点から、学外にも広く授業映像を公開する場合には、Youtube への配信もほぼ自動的にできるようにしたいと考えています。

今後は、引き続き、クリアすべき技術的課題などを調査し、実装を目指すものとします。

## 7. サンプルプログラムによる事前技術的検討例

### 7-1. Google Apps Script(GAS) で、Google Classroom に投稿する

ドキュメントでは、REST API なので、UrlFetchApp クラスを使うのか? とか思っていますが、そもそも それだと auth が必要なのでまだ試していません。

で、そんな事をしなくてもそれぞれの REST の クラスのメソッドに引数を渡す形で実行できます。

ただし、Classroom Service が拡張なので、スクリプトエディタのリソースメニュー (Google の拡張サービス) より API を実行可能にしておく必要があります。



以下のサンプルでは、Classroom は一つしか作成しておらず、ループである必要はありませんが、複数の場合に名前で特定する時に使えると思います。

```
function createAnnouncements(){
// *****
// 現在は1つしか無いので、対象が course.id となります
// *****
var response = Classroom.Courses.list({});
var courses = response.courses;
var course = "";
for (i = 0; i < courses.length; i++) {
  course = courses[i];
  Logger.log("%s (%s)", course.name, course.id);
}

// *****
// POST する JSON
// *****
var data = {
  "courseId": course.id,
  "materials": [
    {
      "link": {
        "url": "https://news.google.com/?hl=ja&tab=rm1&gl=JP&ceid=JP:ja",
        "title": "Google ニュース",
        "thumbnailUrl": ""
      }
    }
  ],
  "text": "Google Apps Script による Classroom への投稿",
  "assigneeMode": "ALL_STUDENTS",
  "state": "PUBLISHED"
};

Classroom.Courses.Announcements.create(data, course.id);
}
```

Logger.log は、何かと結果が表示されない事が多いので、Gmail で自分のアドレスに送信したほうがいいと思います。

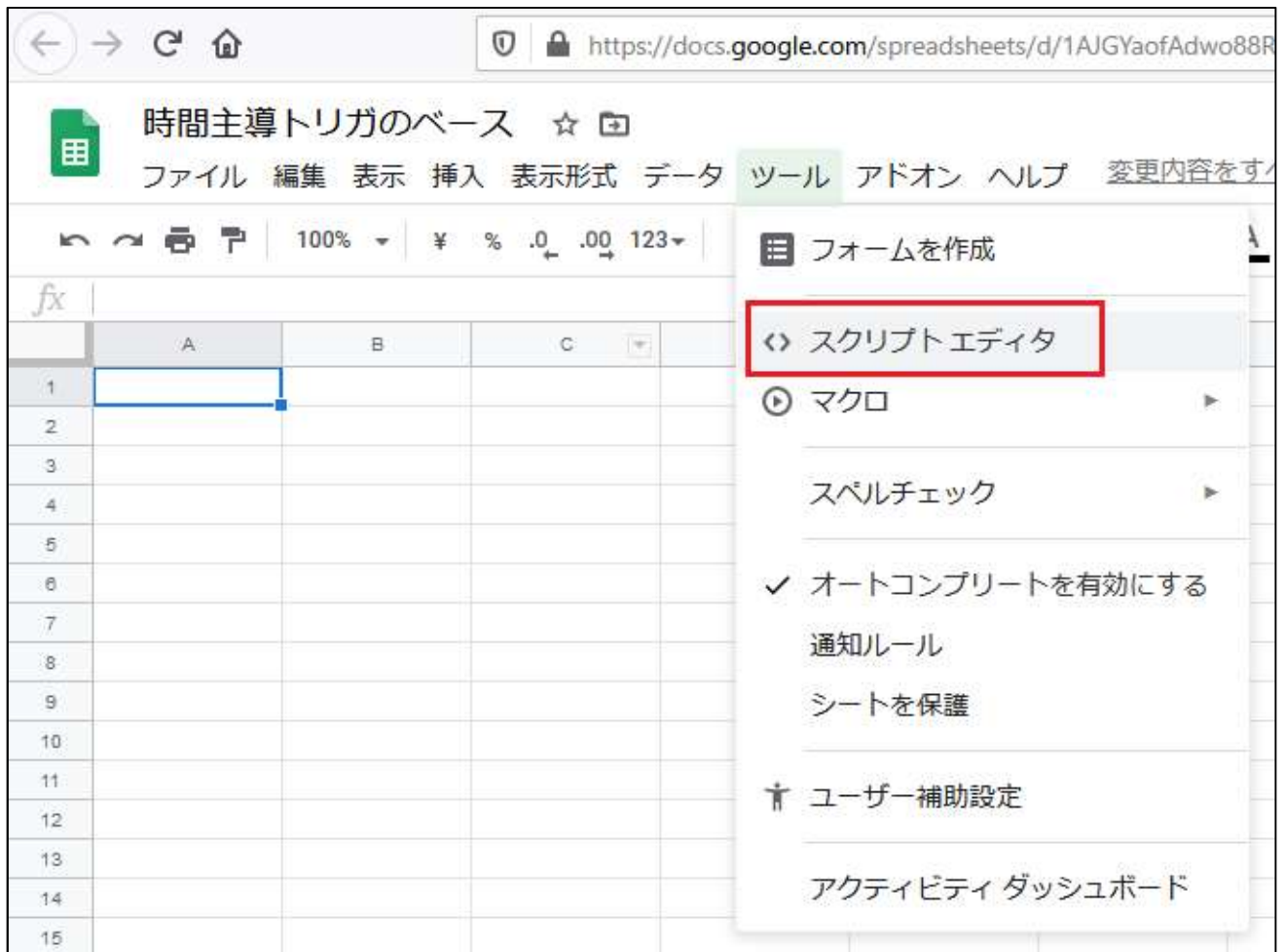
```
// The code below will send an email with the current date and time.
var now = new Date();
GmailApp.sendEmail("mike@example.com", "current time", "The time is: " + now.toString());
```

## 7-2. Google Apps Script : 時間主導のトリガを使用して、CRON のように一定間隔でスクリプトを実行する

Google Apps Script(GAS) で、Google Classroom に投稿することを目指します。一定間隔で、Classroom になんらかの授業用データを作成する為の準備用作業です。



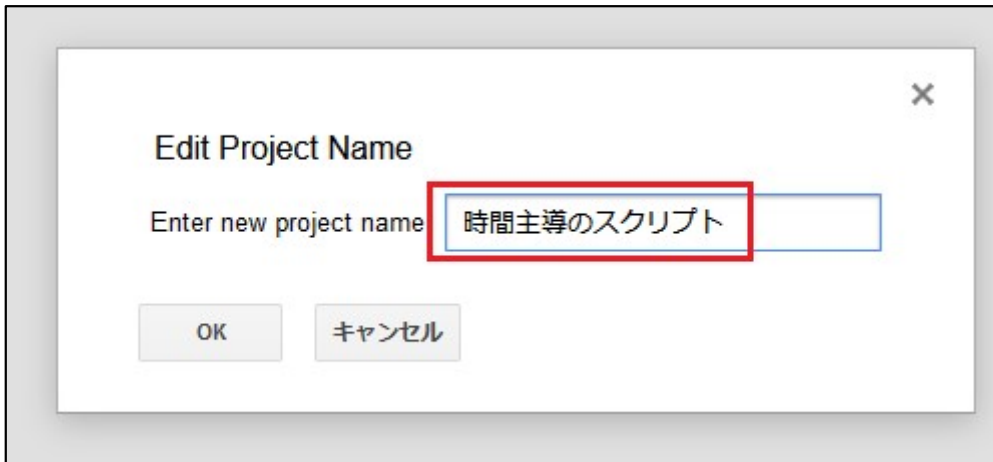
▼ まず Google スプレッドシートを作成してスクリプトエディタを起動します



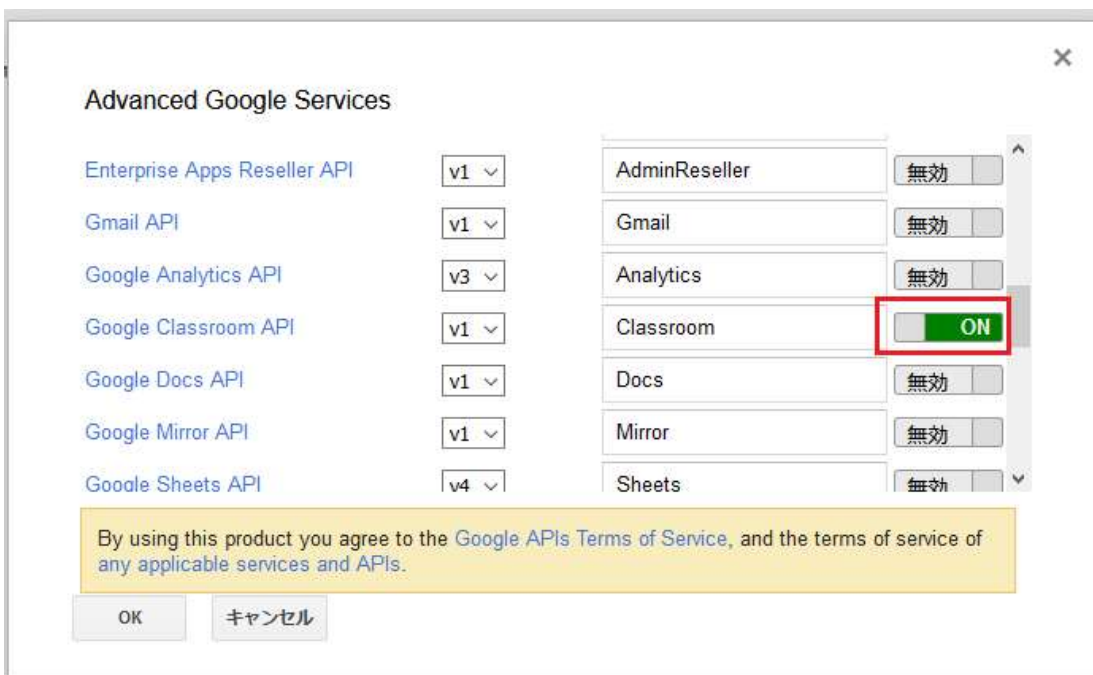
▼ コース id ( 整数文字列 ) と コース名をメールに送信するコードを保存



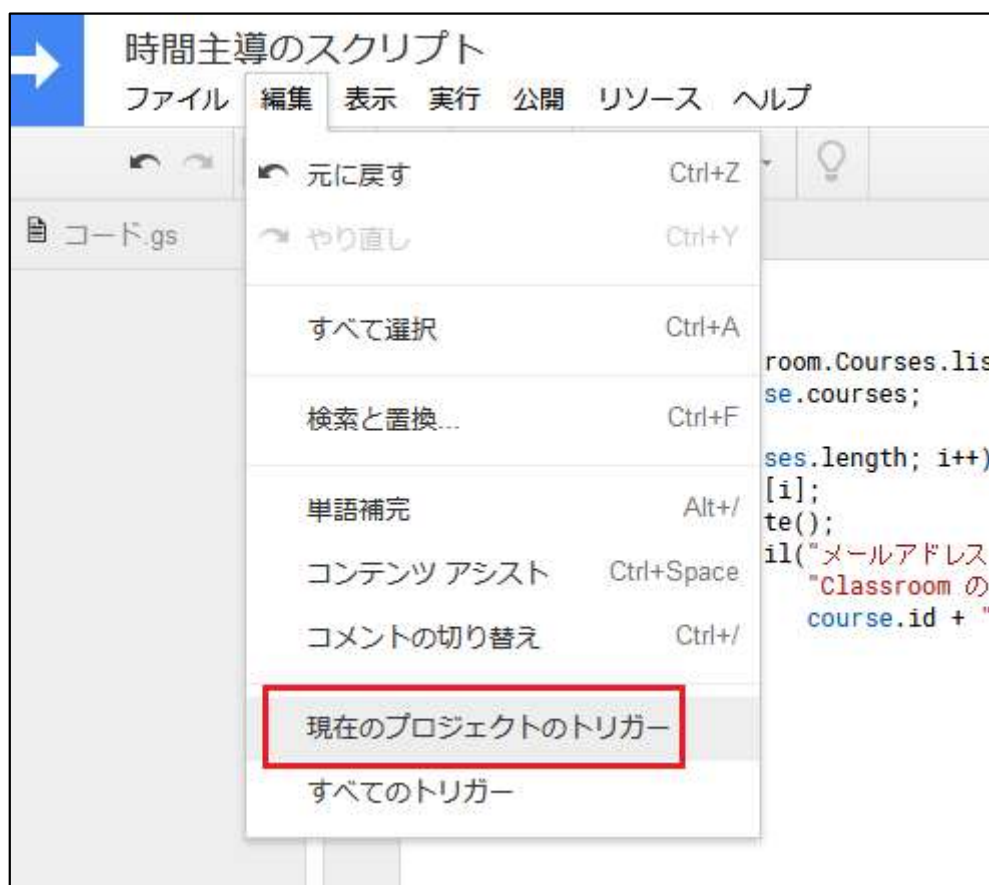
▼ スクリプト名を決定して保存



▼ Google の拡張サービスで、Classroom の API を使用可能にする



▼ トリガ登録画面を表示する



## ▼ トリガ登録

← 時間主導のスク립ト > トリガー

オーナー: 自分

オーナー	前回の実行	導入	イベント
 <b>結果がありません</b> フィルタ条件を調整するか、または新しいトリガーを作成します。			

## ▼ トリガの時間間隔を決定

時間主導のスク립トのトリガーを追加

実行する関数を選択	エラー通知設定 +
myFunction	毎日通知を受け取る
実行するデプロイを選択	
Head	
イベントのソースを選択	
時間主導型	
時間ベースのトリガーのタイプを選択	
分ベースのタイマー	
時間の間隔を選択 (分)	
30 分おき	

キャンセル 保存

このスクリプトは、指定のメールアドレスに現在存在する コース id( 自分はクラス番号と呼んでます ) と名前を送っています。

```
function myFunction() {  
    var response = Classroom.Courses.list({});  
    var courses = response.courses;  
    var course = "";  
    for (i = 0; i < courses.length; i++) {  
        course = courses[i];  
        var now = new Date();  
        GmailApp.sendEmail("メールアドレス",  
            "Classroom の id と名前",  
            course.id + " : " + course.name + " : " + now.toString()  
        );  
    }  
}
```

### 7-3. Google Apps Script : 動画を添付して Classroom の指定のトピックへ課題として投稿する

Google Apps Script : 時間主導のトリガを使用して、CRON のように一定間隔でスクリプトを実行することを目指しました。

このコードは本来、時間主導のトリガを使用して、投稿していない動画があった場合に実行するものです。

( ✓注意 ) いろいろ調べましたが、『資料の作成』は、API からは実行できるようになっていませんでした。

```
function myFunction() {  
    // https://any-  
    ↗ api.com/googleapis_com/classroom/docs/courses/classroom_courses_courseWork_create  
  
    // コースID は最初に取得しておいて、コース毎のトリガを作成する  
    var courseId = "35126354603" // クラス番号  
  
    // スプレッドシートより、動画が保存されているフォルダIDを取得  
    var spreadsheet = SpreadsheetApp.getActive();  
    spreadsheet.getRange('A1').activate();  
    var target = spreadsheet.getCurrentCell().getValue();  
  
    var folder_id = target; // 動画のあるフォルダID  
    var target_folder = DriveApp.getFolderById(folder_id);  
  
    var files = target_folder.getFilesByType("video/mp4"); // mp4 の一覧  
  
    // フォルダ内の動画ファイルをチェック  
    while( files.hasNext() ) {  
  
        var file = files.next();  
        // 本来はこのid が既に存在している場合は continue ( チェックは スプレッドシートを利用する )  
        var fileId = file.getId();  
        // 本来は条件を元に決定する  
        var topicId = "53042801088"; // 課題A( スプレッドシートを利用する )  
  
        // 課題作成用の JSON  
        var json = {  
            "courseId": courseId,  
            "materials": [  
                {  
                    "driveFile": {  
                        "driveFile": {  
                            "id": fileId  
                        }  
                    },  
                    "shareMode": "VIEW"  
                }  
            ],  
            "description": "動画の補足説明",  
            "assigneeMode": "ALL_STUDENTS",  
            "state": "PUBLISHED",  
            "title": "動画のタイトル",  
            "topicId": topicId,  
            "workType": "ASSIGNMENT"  
        };  
  
        // 課題を投稿  
        Classroom.Courses.CourseWork.create(json, courseId);  
    }  
}
```

## 8. 補足

### 8-1. 着想の解説

Excel のマクロの言語は、vba ですが、今回は、javascript となり、vba の関数の及ぶ範囲は、Excel 内のセル等と、pc のハードディスクです。

今回は google のサーバ上にある、Excel もどきと、google ドライブとなり、関数レベルですと、その範囲に加えて gmail でメールが送れる程度です。

★ <https://developers.google.com/apps-script/reference/drive/drive-app>

★ <https://developers.google.com/apps-script/reference/spreadsheet/spreadsheet-app>

★ <https://developers.google.com/apps-script/reference/gmail/gmail-app>

そこに、追加プラグインとして拡張できるのが、Classroom の処理です。

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

今回使った拡張部分の処理は、課題作成と言う処理で、動画が添付出来るということになります。

★ <https://developers.google.com/classroom/reference/rest/v1/courses.courseWork/create>

さらに、通常プログラムを起動するのは、pc の操作員ですが、google のサーバが定期的に行ってくれるということになります。

★ <https://developers.google.com/apps-script/guides/triggers> (トリガー)

システム用語では、これは夜間バッチと呼んでおり、リソースや処理の混雑のない時間帯に行うもので、それら以外は、すべて google の web サービスの ui から作業員が行うものです。

ここからは、運用設計になります。

以前ユニットと表現したものを、教室備え付けのロボットと考え、その、ロボットを遠隔で操作するのが教師です。

ロボットは、その教室で行われる科目をすべて担当して、時間割の最初に録画を始めて次の時間割の直前に録画を終了します。

それらの動画が、ロボットに対応する classroom の、対応する科目に登録できるように準備するのが作業員の役目です。

その際、作るテーブルに必要な情報が教師に対応する classroom のコース id と科目のトピック id です。

作業員は、まず教師の classroom を全て作成(コース)して教師が担当する科目(トピック)を classroom の中に登録します。

そして、ロボットの個体数ぶん(教室)のプロジェクトを作成して、それぞれのコンテナとなる Excel もどきのセルを使って時間割を作成します(次ページ画像)。

それぞれのプロジェクトの、起動はトリガーを追加して、google にまかせます。時間割は、ロボットを操作する教師と、科目をセットで登録します(次ページ画像参照)。

C	D	E	F	G	H	I
	月	火	水	木	金	土
コースID(講師)	54407226065		54407226065	54407226065	54407226065	
トピック(科目)	54434493630		56098112610	54416452377	54433667289	
09:20:00						
10:59:00						
コースID(講師)	54407226065		54407226065	54407226065	54407226065	
トピック(科目)	54434493630		56098112610	54416452377	54433667289	
11:00:00						
12:40:00						
コースID(講師)		54408403250	54407226065	54407226065	54407226065	
トピック(科目)		56066536269	54416452382	54433667284	54422856612	
13:30:00						
15:09:00						
コースID(講師)		54408403250	54407226065	54407226065	54407226065	
トピック(科目)		56066536269	54416452382	54433667284	54422856612	
15:10:00						
16:50:00						

## 8-2. 行った機能調査

- Zoom の機能

使って解ったことは、動画が指定場所に保存できるということだった。

- 動画のアップロードは google drive stram をインストールしたらできた。

ローカルフォルダに共有ドライブが現れるので、そこを Zoom の保存先として指定すればよい。

- 動画を classroom へ投稿

google のドキュメントを読んで、検索して調べて試したらできた。

- <https://logicalerror.seesaa.net/article/473808758.html>

この時点では試しただけで、課題では無かった。

- <https://logicalerror.seesaa.net/article/473924654.html>

これは、ただの CRON のテスト

- <https://logicalerror.seesaa.net/article/473936195.html>



固定値で動画投稿(この時点で、課題でしか投稿できないと解る)

以上