

ChatBot-edu の設定方法

1. Firebase の登録

①下記の URL をクリックして Firebase にアクセス

<https://firebase.google.com/?hl=ja>

②「Go to console」をクリック



③「プロジェクトを追加」をクリック



④プロジェクト名（「例：chatbot-setup」）を入力して「続行」をクリック



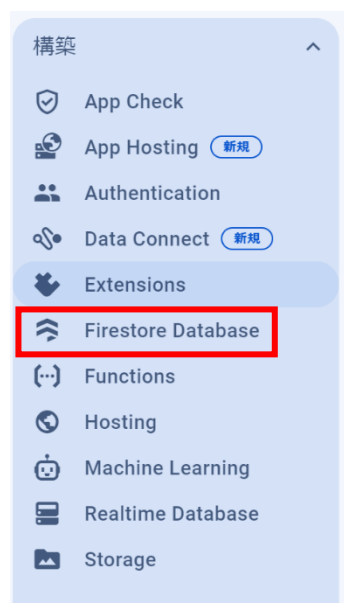
⑤ 「Google アナリティクス」 を無効にするにして「プロジェクトを作成」をクリック



⑥ ④で入力をしたプロジェクト名のプロジェクトが表示されますので、「構築」をクリック



⑦ 「Firestore Database」 をクリック



⑧ 「データベースの作成」をクリック



⑨ロケーションを「asia-northeast1(Tokyo)または(Osaka)」を選択して「次へ」をクリック

データベース ID

(default)

入力なし

ロケーション

nam5 (United States)

リージョン

asia-east1 (Taiwan)

asia-east2 (Hong Kong)

asia-northeast1 (Tokyo)

asia-northeast2 (Osaka)

asia-northeast3 (Seoul)

asia-south1 (Mumbai)

asia-southeast1 (Singapore)

どちらでもよい

⑩「本番モードで開始する」をチェックし「作成」をクリック



本番環境モードで開始する

データはデフォルトで限定公開になります。クライアントの読み取り / 書き込み権限は、セキュリティ ルールで指定されているとおりに付与されます。

⑪「コレクションを開始」をクリック



(default)

+ コレクションを開始

⑫コレクション ID に「school-data」を入力して「次へ」をクリック

コレクション ID ②

school-data|

キャンセル

次へ

⑬下記のように設定を行い、「保存」をクリック

フィールドを「authenticated」
タイプを「array」とする

フィールドを「deployed」
タイプを「map」とする

フィールドを「pass」
タイプを「map」とする

ドキュメントのスキーマ
/ school-data
ドキュメント ID ②
フィールド タイプ 値
authenticated = array
0 string 0
フィールドを追加
deployed = map
フィールド タイプ 値
test01 = string test02
フィールドを追加
pass = map
フィールド タイプ 値
test03 = string test04
フィールドを追加

管理者及び使用生徒のドメイン
(例: gmail.com)

タイプを「string」
値を管理者の「ユーザー名」とする
(例:「iizuna」←iizuna@gmail.com)

フィールドを「test01」
タイプを「string」
値を「test02」とする

フィールドを「test03」
タイプを「string」
値を「test04」とする

⑭「コレクションを開始」をクリック

(default)
+ コレクションを開始
school-data >

⑮コレクション ID に「publish-data」を入力して「次へ」をクリック

コレクション ID ②
publish-data
キャンセル 次へ

⑯ドキュメント ID は「自動 ID」として下記のように設定を行い「保存」をクリック

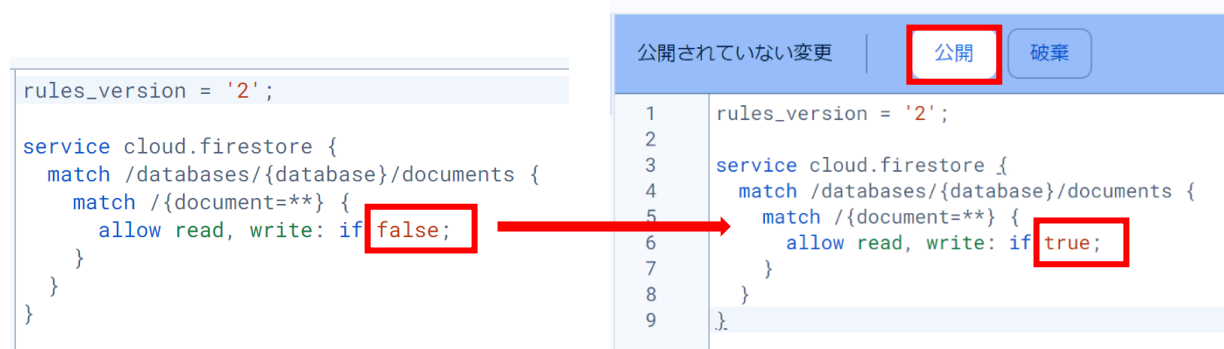
ドキュメント ID ②
自動 ID
必須
フィールド タイプ 値
test05 = string test06
フィールドを追加

フィールドを「test05」
タイプを「string」
値を「test06」とする

⑪ 「ルール」 をクリック



⑫ 「false」 を 「true」 に変更し、「公開」 をクリック

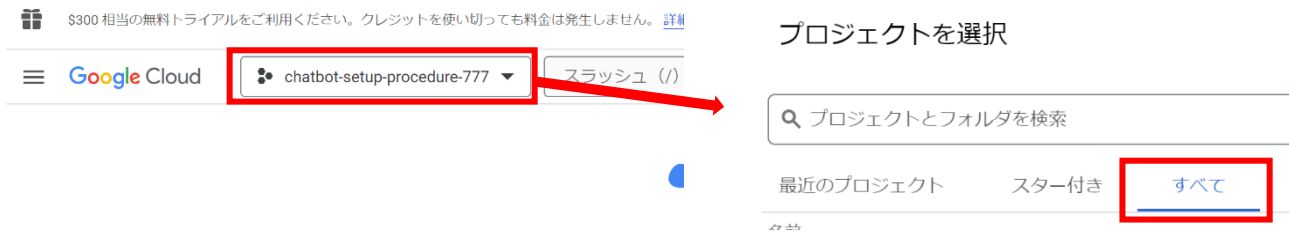


1. GoogleCloud の登録

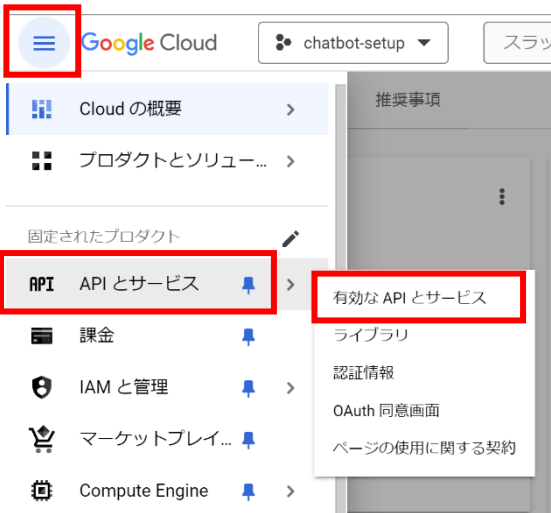
①下記の URL をクリックして Firebase にアクセス

<https://console.cloud.google.com/welcome>

②プロジェクト名を選択（例：chatbot-setup）



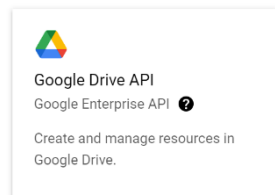
③「ナビゲーションウィンドウ」→「API とサービス」→「有効な API とサービス」をクリック



④「+API とサービスを有効にする」をクリック



⑤「Google Drive API」をクリック

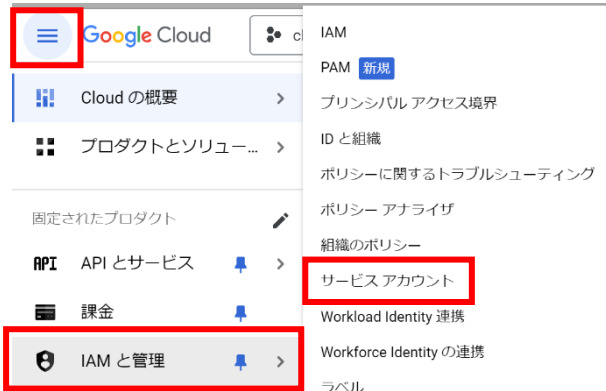


⑥「有効にする」をクリック



＊「Google Docs API」、「Google Forms API」、「Google Sheets API」、「Apps Script API」も同様に有効にする

⑦「ナビゲーションウィンドウ」→「IAM と管理」→「サービスアカウント」をクリック



⑧「+サービスアカウントを作成」をクリック

サービス アカウント [+ サービス アカウントを作成](#)

⑨サービスアカウント名を入力（例：firebase-connector）して「作成して続行」

1 サービス アカウントの詳細

サービス アカウント名
firebase-connector

このサービス アカウントの表示名

サービス アカウント ID *
firebase-connector

メールアドレス: firebase-connector@chatbot-setup.iam.gserviceaccount.com

サービス アカウントの説明
このサービス アカウントで行うことを説明します

[作成して続行](#)

⑩「ロールを選択」→「Cloud Datastore」で検索し→「Cloud Datastore オーナー」→「続行」

2 このサービス アカウントにプロジェクトへのアクセスを許可する (省略可)

このサービス アカウントに chatbot-setup へのアクセス権を付与して、プロジェクト内のリソースに対する特定のアクションを完了する権限を付与します。 [詳細](#)

ロール
Cloud Datastore オーナー

IAM の条件 (省略可) ?
[+ IAM の条件を追加](#)

[+ 別のロールを追加](#)

[続行](#)

⑪ 「完了」をクリック

- ✓ サービス アカウントの詳細
 - ✓ このサービス アカウントにプロジェクトへのアクセスを許可する (省略可)
 - 3 ユーザーにこのサービス アカウントへのアクセスを許可 (省略可)
Grant access to users or groups that need to perform actions as this service account. [Learn more](#)
- サービス アカウント ユーザー ロール

このサービス アカウントを使用してジョブと VM をデプロイする権限をユーザーに付与します
- サービス アカウント 管理者 ロール

このサービス アカウントを管理する権限をユーザーに付与します。

完了 キャンセル

⑫ 「詳細を管理」をクリック

☐	firebase-connector@chatbot-setup.iam.gserviceaccount.com	✓ 有効	firebase-connector	キーがありません	108048804061462031307	⋮ 詳細を管理 権限を管理 鍵を管理
--------------------------	--	------	--------------------	----------	-----------------------	--

⑬ 「キー」をクリック

詳細 権限 **キー** 指標 ログ

⑭ 「鍵を追加」 → 「新しい鍵を作成」をクリック

- 鍵を追加 ▾
- 新しい鍵を作成
- 既存の鍵をアップロード

⑮ キーのタイプを「JSON」に選択し「作成」をクリック

「firebase-connector」の秘密鍵の作成

秘密鍵を含むファイルをダウンロードします。この鍵を紛失すると復元できなくなるため、ファイルは大切に保管してください。

- キーのタイプ

☒ JSON

☐ P12
- 推奨

P12 形式を使用したコードとの下位互換性を目的としています

キャンセル **作成**

* 保存された JSON ファイルは非常に重要

⑩「ナビゲーションウィンドウ」→「API とサービス」→「OAuth 同意画面」をクリック



⑪「External」を宣して「作成」をクリック

User Type

☐ Internal ?

Only available to users within your organization. You will not need to submit your app for verification. [Learn more about user type](#)

☒ External ?

Available to any test user with a Google Account. Your app will start in testing mode and will only be available to users you add to the list of test users. Once your app is ready to push to production, you may need to verify your app. [Learn more about user type](#)

⑫アプリ名（例：チャットボットアプリシート）とユーザーサポートメールを入力する

ユーザーサポートメールは **Firebase** でログインをしているアカウントとする

アプリ情報

この情報は同意画面に表示されるため、デベロッパーのユーザー情報とデベロッパーへの問い合わせ方法をエンドユーザーが把握できます。

アプリ名*
チャットボットアプリシート
同意を求めるアプリの名前

ユーザー サポートメール*
@gmail.com
同意に関して問い合わせる際に使用します。 [詳細](#)

⑬デベロッパーの連絡先情報に連絡のつくアドレスを入力して「保存して次へ」をクリック

デベロッパーの連絡先情報

メールアドレス*
@gmail.com

これらのメールアドレスは、プロジェクトの変更について Google からお知らせするために使用します。

キャンセル

⑩「保存して次へ」をクリック

🔒 制限付きのスコープ

制限付きのスコープとは、機密性の高いユーザー
するスコープです。

API ↑	範囲	ユーザー
表示する行がありません		

保存して次へ

キャンセル

⑪「+ADD USERS」を選択して Firebase のアカウントを入力、「追加」をクリック

テストユーザー

公開ステータスが「テスト中」に設定されている間は、
がアプリにアクセスできます。アプリの確認前の許可
は 100 で、この上限はアプリの全期間でカウントされ

+ ADD USERS

@gmail.com ✕



0 / 100

追加

⑫「保存して次へ」をクリック

≡ フィルタ プロパティ名または値を入力

ユーザー情報

@gmail.com



保存して次へ

キャンセル

①下記の URL より ZIP ファイルをダウンロード

②ダウンロードした ZIP ファイルをデスクトップに解凍する

①下記の URL より Firebase CLI をインストール

```

Welcome to...

#####  #####  #####  ##  #####  #####  ##
##  ##  ##  ##  ##  ##  ##  ##  ##  ##  ##
#####  ##  #####  #####  #####  #####  ##
##  ##  ##  ##  ##  ##  ##  ##  ##  ##
##  #####  #####  ##  ##  #####  #####  ##

~ Let's make sure your Firebase CLI is ready...
~ Looks like your CLI needs to be set up.

~ This may take a moment
+ Alright, your CLI is set up!

~ Looks like you're not authenticated. Let's log in!
i Firebase optionally collects CLI and Emulator Suite usage and error reporting information to help improve our product
s. Data is collected in accordance with Google's privacy policy (https://policies.google.com/privacy) and is not used to
identify you.

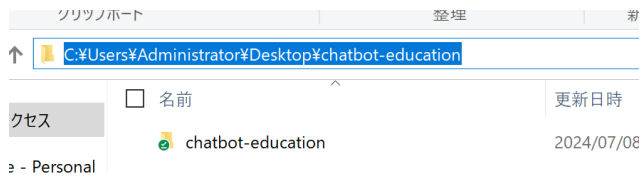
? Allow Firebase to collect CLI and Emulator Suite usage and error reporting information? (Y/n) _

```

information? という問いに対して Enter キーを押す

③ 「firebase login」を入力してEnterキーを押す

④解凍した ZIP ファイル (chatbot-education) を開き、アドレスバーの情報をコピー



⑤Firebase CLI に戻って

「cd コピーした情報をペースト」を入力して Enter キーを押す
(例 : cd C:\Users\Administrator\Desktop\chatbot-education)

④「firebase init」を入力して Enter キーを押す

⑤? Are you ready to proceed? (Y/n) という問いに対して Enter キーを押す

⑥Hosting: Configure files for Firebase Hosting and (optionally) set up GitHub Action

```
? Which Firebase features do you want to set up for this directory? Press Space to select features, then Enter to confirm your choices. (Press <space> to select, <a> to toggle all, <i> to invert selection, and <enter> to proceed) instance
( ) Firestore: Configure security rules and indexes files for Firestore
( ) Functions: Configure a Cloud Functions directory and its files
> (*) Hosting: Configure files for Firebase Hosting and (optionally) set up GitHub Action deploys
( ) Hosting: Set up GitHub Action deploys
( ) Storage: Configure a security rules file for Cloud Storage
( ) Emulators: Set up local emulators for Firebase products
```

deploys に移行し、スペースキーを押すと「*」が付く、その状態で Enter キーを押す

```
? Please select an option: (Use arrow keys)
> Use an existing project
  Create a new project
  Add Firebase to an existing Google Cloud Platform project
  Don't set up a default project
```

⑦Use an existing project を選択し、Enter キーを押す

```
> chatbot-setup (chatbot-setup)
```

⑧Firebase で立ち上げたプロジェクトを選択し、Enter キーを押す

⑨? What do you want to use as your public directory? (public) という問いに対して Enter キーを押す

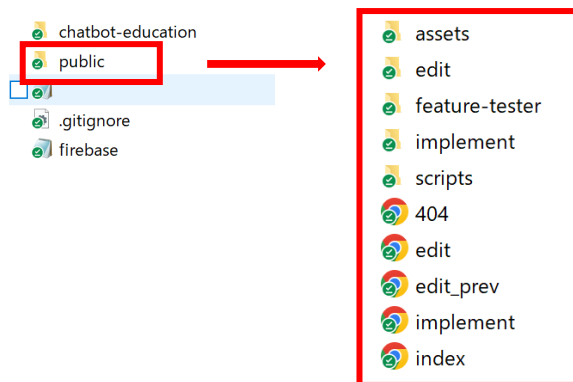
⑩? Configure as a single-page app (rewrite all urls to /index.html)? (y/N) という問いに対して Enter キーを押す

⑪? Set up automatic builds and deploys with GitHub? (y/N) という問いに対して Enter キーを押す

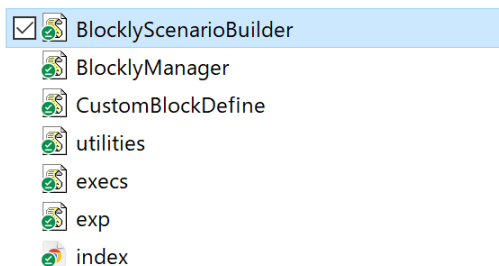
⑫chatbot-education フォルダに移行し、その中の chatbot-education フォルダの中身をすべてコピー

- assets
- edit
- feature-tester
- implement
- scripts
- 404
- edit
- edit_prev
- implement

⑬public フォルダの中で貼り付けを行う



⑭public¥scripts¥blockly 中の「BlocklyScenarioBuilder.js」を開く



⑮ GeminiAPI をペーストして上書き保存 * GeminiAPI 取得方法 P 2 8 ~

```
/**
 * 生成AI (gemini-pro:generateContent)
 * @param {BlocklyJsonBlock} block
 */
async GenerateAI(block) {
  console.log("GenerateAI => block:", block);
  let prompt = block.fields["PROMPT"] || "";
  let userText = this.inputString || "";
  let combinedPrompt = prompt + "\nユーザー入力: " + userText;

  try {
    let response = await fetch(
      "https://generativelanguage.googleapis.com/v1beta/models/gemini-pro:generateContent?key="
      {
        method: "POST",
        headers: { "Content-Type": "application/json" },
        body: JSON.stringify({
          contents: [
            {
              text: combinedPrompt
            }
          ]
        })
      }
    );
    let data = await response.json();
    let generatedText = data.candidates[0].content.parts[0].text;
    console.log("Generated text:", generatedText);
    this.outputText.setText(generatedText);
  } catch (error) {
    console.error("Error generating content:", error);
  }
}
```

API キーをペースト

* モデル変更可能

⑯ Firebase CLI に戻り、「firebase deploy」を入力して Enter キーを押す

⑰ 下図が完了画面 Hosting URL はメモしておく

```
> firebase deploy
=== Deploying to 'chatbot-setup'...

i deploying hosting
i hosting[chatbot-setup]: beginning deploy...
i hosting[chatbot-setup]: found 178 files in public
+ hosting[chatbot-setup]: file upload complete
i hosting[chatbot-setup]: finalizing version...
+ hosting[chatbot-setup]: version finalized
i hosting[chatbot-setup]: releasing new version...
+ hosting[chatbot-setup]: release complete

+ Deploy complete!

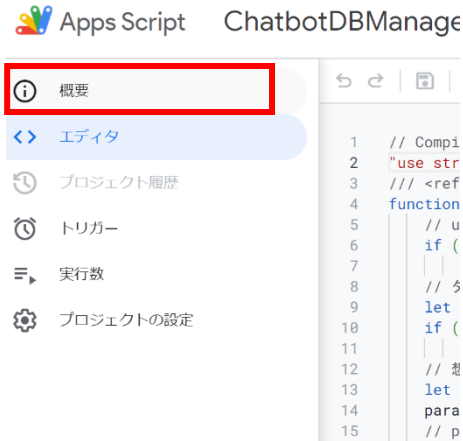
Project Console: https://console.firebase.google.com/project/chatbot-setup/overview
Hosting URL: https://chatbot-setup.web.app
```

4. ChatbotDBManager の設定

①下記の URL をクリックして ChatbotDBManager にアクセス

https://script.google.com/home/projects/1Gb5-LG_fRUdW1jFKQJde8F01nzBJLPOBTITeDR7GDQM2zm09gANLaRJN/edit

②「概要」をクリック



③「コピーを作成」をクリック

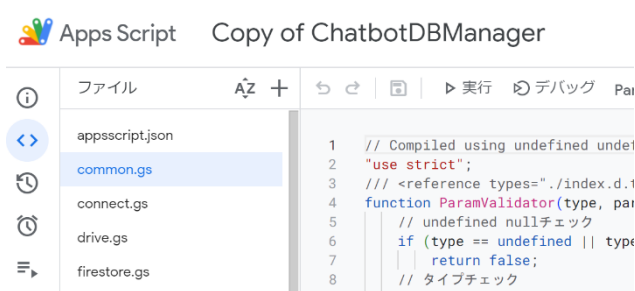
プロジェクトの詳細



Apps Script

オーナー	作成日時	最終更新
チャットボット	2023/09/27	2024/07/06
		変更者: チャットボット さん
ロール	ステータス	最終実行日時
閲覧者	導入済み	2024/07/06

自分の Google Drive に「Copy of ChatbotDBManager」が作成される



④「Copy of ChatbotDBManager」の「Copy of」部分を削除し「ChatbotDBManager」とする
プロジェクトの名前を変更

プロジェクトタイトル*

ChatbotDBManager

キャンセル 名前を変更

⑤「プロジェクトの設定」をクリック

- 概要
- エディタ
- プロジェクト履歴
- トリガー
- 実行数
- プロジェクトの設定**

⑥「プロジェクトを変更」→GCP のプロジェクト番号を入力し「プロジェクトを設定」をクリック

Google Cloud Platform

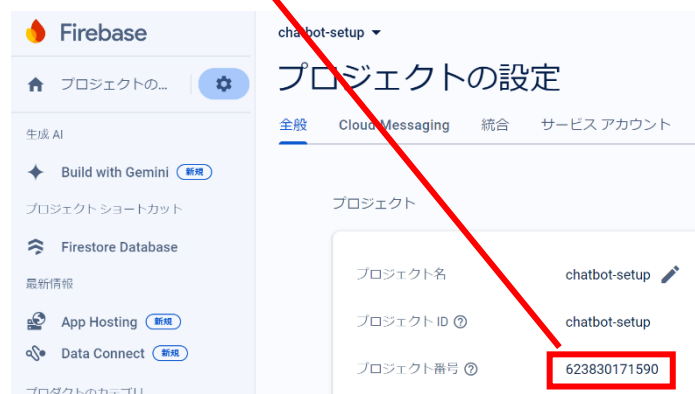
Apps Script プロジェクトで
詳細が管理されます。詳しくは
ヘルプをご覧ください。

GCP デフォルト

プロジェクトを変更

GCP のプロジェクト番号

プロジェクトを設定



⑦「スクリプトプロパティを追加」をクリック

スクリプト プロパティ

スクリプト プロパティを使用すると、
スクリプト プロパティを定義および公開できます。
詳しくは、ヘルプをご覧ください。

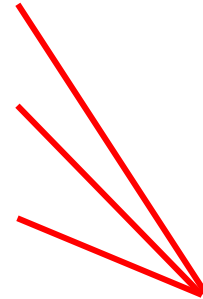
スクリプト プロパティを追加

⑧プロパティに「client_email」「private_key」「project_id」を追加する

スクリプトプロパティ

スクリプト プロパティを使用すると、特定のオブジェクト インスタンスに対して簡単にカスタム プロパティを定義および公開できます。詳しくは、[スクリプトプロパティに関するドキュメント](#)をご覧ください。

プロパティ client_email	値 firebase-connector@chatbot-setup-procedur
プロパティ private_key	値 -----BEGIN PRIVATE KEY-----\nMIIEvQIBADANI
プロパティ project_id	値 chatbot-setup

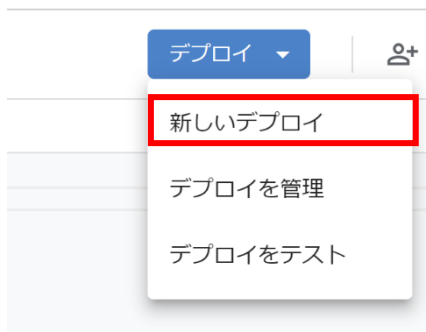


* 1. GoogleCloud の登録^⑤で保存した JSON ファイルを開き、値を入力する

```
chatbot-setup-cb0353544b25 - メモ帳
ファイル(F) 編集(E) 書式(O) 表示(V) ヘルプ(H)
[{"type": "service_account", "project_id": "chatbot-setup", "private_key_id": "cb0353544b257429d604c7a0a96358050712f8b6", "private_key": "-----BEGIN PRIVATE KEY-----\nz9QC4KeYnOM6VihFUF6+gkLOWts/yzRnTLHivF/8d8JcifyciNwHq2CjbtKe8h7fyTMD+DsJyYndY4ujbxYPUC2HZOM03taxyJhZQK8gQDJ8B48gMJE6zsqJZo1IBx6VWdazJjKsi+B4nXgk0u60trHmkHMbd6JvP52sL1L9h2/auth", "token_uri": "https://oauth2.googleapis.com/token", "auth_provider_x509_cert_url": "https://www.googleapis.com/oauth2/v1/certs", "client_x509_cert_url": "h
```

*private_key は「-----BEGIN PRIVATE KEY-----¥n」から「-----END PRIVATE KEY-----¥n」まで
すべてを貼り付けること

⑨「デプロイ」→「新しいデプロイ」クリック



⑩「種類の選択」→「ライブラリ」→「デプロイ」をクリック

新しいデプロイ

種類の選択  設定 

ウェブアプリ

- ✓ ウェブアプリ
- 実行可能 API
- アドオン
- ライブラリ

自分 (k07b220a@gmail.com)

このウェブ アプリケーションを実行するために、あなたのアカウント データを使用することを許可します。

アクセスできるユーザー

全員

ライブラリとしても利用できます。詳細

キャンセル デプロイ

⑪「アクセスを承認」をクリック

新しいデプロイ

このウェブ アプリケーションを

アクセスを承認

⑫アカウントを選択後、「詳細を隠す」→「ChatbotDBManager のコピーに移動」をクリック



Googleはこのアプリを検証していません

アプリは Google アカウントの機密情報へのアクセスを要求しています。開発者 (k07b220a@gmail.com) が Google でこのアプリを確認するまで、このアプリを使用しないでください。

詳細を隠す

安全に戻る

リスクを理解し、開発者 (k07b220a@gmail.com) を信頼している場合のみ続行してください。

[ChatbotDBManagerのコピーに移動 \(安全ではありません\)](#)

⑬「続く」→「許可する」をクリックすることでデプロイが完了する

ライブラリ

URL

https://script.google.com/macros/library/d/1FC6WjpLoAtJCMVK5_IkONEPNkI2QLibByDKEDQby-n6HVAwiq1PVIgR/1

 コピー

スクリプト ID

5. Chatbot ベースシート_origin からの設定

①下記の URL にアクセス

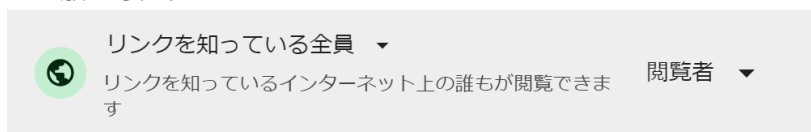
<https://drive.google.com/drive/folders/15CIU1sgn8SW6LScb0WEUY7ZOeQGTuygd?usp=sharing>

②3つのファイルをすべて自分の Google Drive にコピーする



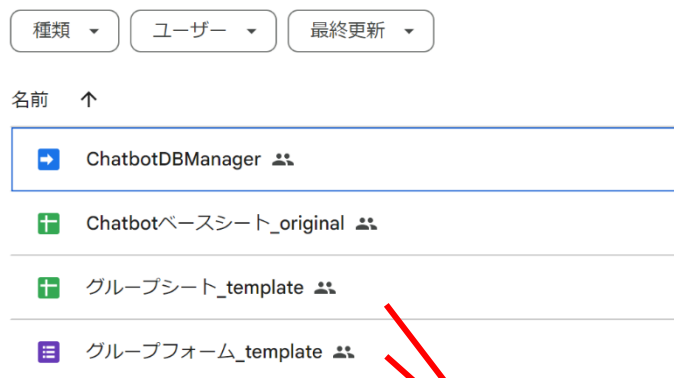
③自分の Google Drive を開き、マイドライブに「Chatbot-template」フォルダを作成する

④Chatbot-template フォルダを右クリックし、「共有」→「リンクを知っている全員」にする
一般的なアクセス



⑤先ほどコピーした4つのファイルを検索し、Chatbot-template フォルダ内に移動させる

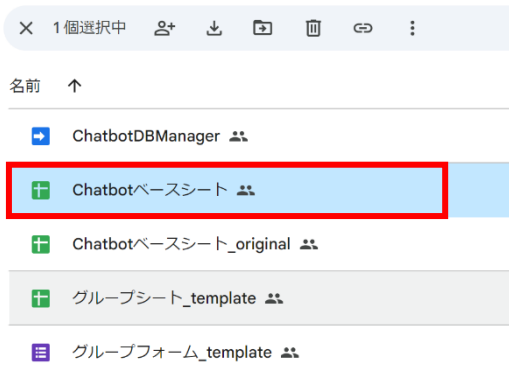
マイドライブ > chatbot-template



この2つのファイルが各生徒に反映される
各学校やテーマに合わせて編集可能

⑥「Chatbot ベースシート_original」をコピーし、名前を「Chatbot ベースシート」とする

マイドライブ > chatbot-template ▾



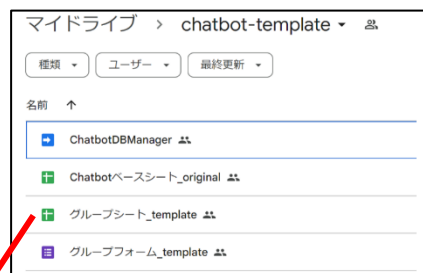
⑦マイドライブに戻り、新しく「Chatbot」フォルダを作成する

Chatbot-template フォルダと同様に一般的なアクセスを「リンクを知っている全員」にしておく

⑧Chatbot ベースシートを Chatbot-template フォルダから Chatbot フォルダに移動させる

⑨Chatbot ベースシートを開く

⑩説明に従い、値を入力していく



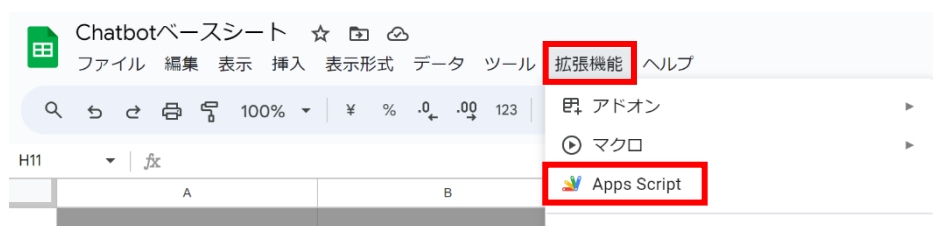
Google フォームを
一度ファイルを開いてコピー

docs.google.com/forms/d/1GvmOC9hVX3ZY2y464gsehmB8b6fTq5g2HaQ2_khGw/edit

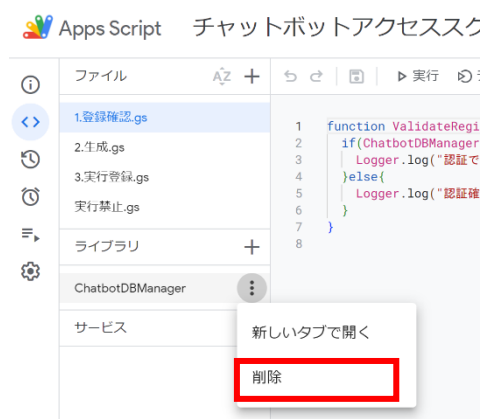
	値	説明
PassPhrase	password	このシートのGASへアクセスするために必要な合言葉です 半角英数6~12文字程度で設定してください
SheetTemplateURL	https://docs.google.com/spreadsheets/d/1GvmOC9hVX3ZY2y464gsehmB8b6fTq5g2HaQ2_khGw/edit	キーワード等を格納しておくためのテンプレートシートのURLです シートを開いた時のURLをそのまま貼り付けてください(tab等は押さないでください)
FormTemplateURL	https://docs.google.com/forms/d/1GvmOC9hVX3ZY2y464gsehmB8b6fTq5g2HaQ2_khGw/edit	チャットボット終了後に表示するフォームのテンプレートです フォームを開いた時のURLを貼り付けてください(回答URLではありません)
TargetYear	2024	授業を実施する年度を西暦で入力してください
TargetGrade	2	授業を実施する学年を数字で入力してください
ClassSheetSuffix	組	クラスシートであることを示す末尾の文字列です 例:「{任意の組名}組」というシートをクラスシートにする場合は「組」を指定
TargetSemester	1	授業が実施される学期を数字で入力してくださいex:1学期である場合は1を入力
SchoolDomain	gmail.com	学校で利用しているメールアドレスのドメイン
ChatbotOriginURL	https://chatbot-setup.web.app	授業で利用するチャットボットのオリジンURL

```
hosting[chatbot-setup]: release complete
+ Deploy complete!
Project Console: https://console.firebase.google.com/project/chatbot-setup/overview
Hosting URL: https://chatbot-setup.web.app
```

⑪「拡張機能」→「App Script」をクリック



⑫「ChatbotDBManager」を削除する



⑬ライブラリの追加「+」→スクリプト ID を入力し、「検索」をクリック



⑭バージョンを「1」とし、「追加」をクリック

ライブラリの追加

利用可能なライブラリを ID で検索できます。[詳細](#)

スクリプト ID *
1FC6WjpLoAtJCMVK5_IlkONEPNkI2QLIbByDKEDQby-n6HVAwi

ライブラリのプロジェクト設定で確認できるライブラリのスクリプト ID。

検索

ライブラリ ChatbotDBManager を検索しました。

バージョン
1

利用可能なバージョン。

ID *
ChatbotDBManager

このプロジェクト内でこのライブラリを参照する際に使用します。

キャンセル

追加

⑮「1. 登録確認.gs」→「実行」をクリック

ファイル	実行	デバッグ
1.登録確認.gs	▶ 実行	🐞 デバッグ
2.生成.gs		
3.実行登録.gs		

```
1 function ValidateRegister(){
2   if(ChatbotDBManager.Applicati
3     Logger.log("認証できました。生
4   }else{
```

⑯「2. 生成.gs」→コメントを確認して→「実行」をクリック

ファイル	実行	デバッグ	GenerateSheet	実行ログ
1.登録確認.gs	▶ 実行	🐞 デバッグ	GenerateSheet ▼	実行ログ
2.生成.gs				
3.実行登録.gs				
実行禁止.gs				
ライブラリ	+			
ChatbotDBManager				
サービス	+			

```
1 // クラスシートを生成する際は、組と組の班編成数を変更し実行してください
2 // 各クラスのフォーム、シート、画像フォルダが自動で生成されます
3
4 // ひと組づつの生成となるので、一学年三組の場合は、targetClassとunitCountを変更しながら三回実行してください
5 // targetClass:生成するクラスの番号です
6 // unitCount:生成する組み分けの量です。例えば一組30人で5人で組み分けする場合は6を入力してください
7 // 二人一組なら15、一人ずつに割り振るなら30となります
8 // ただし、30に近い数字を入力すると中断される可能性が高いため、何度か試行してください
9
10
11 function GenerateSheet(){
12
13   // 生成する組(半角数字)
14   let targetClass=1;
15
16   // 組の班編成(半角数字)(初期は6班編成)
17   let unitCount=6;
18
19   Generate(targetClass,unitCount);
20
21   Logger.log(`${targetClass}組のファイル、フォルダ生成を完了しました`);
22 }
23
24
25 // すべての組を生成したら、登録ステップに進んでください
```

⑰ 「3. 実行登録.gs」 → 「デプロイ」 をクリック

Apps Script チャットボットアクセススクリプト

デプロイ ▼

ファイル ▲ ▼ + ↺ ↻ 📁 ▶ 実行 ⚙️ デバッグ Register ▼ 実行ログ

1.登録確認.gs	⋮
2.生成.gs	
3.実行登録.gs	
実行禁止.gs	
ライブラリ	+
ChatbotDBManager	

```
1  // 右上のボタンからデプロイを実行後、表示されるウェブアプリURLをコピーして、置き換えを行ってください
2  // デプロイURLは、https://script.google.com/macros/ ではじまり、/execで終わるアドレスのことです
3  function Register(){
4
5      // 下記部分をデプロイURLで置き換えてください
6      // let deployURL="この部分をデプロイURLで置き換えてください";
7      let deployURL="";
8
9      ChatbotDBManager.DeployURLRegister(deployURL)
10
11 }
```

⑱ 「新しいデプロイ」 → 「デプロイ」 から URL が作成させる → 「コピー」

ウェブアプリ

URL

<https://script.google.com/macros/s/AKfycbzkvbeWuF95I2K4HPKOKj2zy34eXhXthVwiBqZLqUjpE-FyHnpVwNhS0feo9sQ...>

📋 コピー

⑲ コピーした URL を 「let deployURL=" ";」 の 「" "」 内に貼り付ける

```
1  // 右上のボタンからデプロイを実行後、表示されるウェブアプリURLをコピーして、置き換えを行ってください
2  // デプロイURLは、https://script.google.com/macros/ ではじまり、/execで終わるアドレスのことです
3  function Register(){
4
5      // 下記部分をデプロイURLで置き換えてください
6      // let deployURL="この部分をデプロイURLで置き換えてください";
7      let deployURL="";
8
9      ChatbotDBManager.DeployURLRegister(deployURL)
10
11 }
```



```
1  // 右上のボタンからデプロイを実行後、表示されるウェブアプリURLをコピーして、置き換えを行ってください
2  // デプロイURLは、https://script.google.com/macros/ ではじまり、/execで終わるアドレスのことです
3  function Register(){
4
5      // 下記部分をデプロイURLで置き換えてください
6      // let deployURL="この部分をデプロイURLで置き換えてください";
7      let deployURL="https://script.google.com/macros/s/AKfycbzkvbeWuF95I2K4HPKOKj2zy34eXhXthVwiBqZLqUjpE-FyHnpVwNhS0feo9sQk9guV/exec";
8
9      ChatbotDBManager.DeployURLRegister(deployURL)
10
11 }
```

②⑩「上書き保存」したら「実行」をクリック

ファイル ⌕ + ↶ ↷ 📄 ▶ 実行 🔍 デバッグ Register ▼ | 実行ログ |

```

1 // 右上のボタンからデプロイを実行後、表示されるウェブアプリURLをコピーし
2 // デプロイURLは、https://script.google.com/macros/ ではじまり、
3 function Register(){
4
5     // 下記部分をデプロイURLで置き換えてください
6     // let deployURL="この部分をデプロイURLで置き換えてください";
7     let deployURL="https://script.google.com/macros/s/AKfycbzkv
8     exec";
9
10    ChatbotDBManager.DeployURLRegister(deployURL)
11 }
  
```

②⑪Chatbot ベースシートに戻るとシートが追加されていることを確認

A	B	C	D	E	F	G	H	I	J
unit	sheet	form	folder	answer	deploy	save	code	editUrl	implUrl
101	https://docs.google.com/spreadsheets/d/1...	https://docs.google.com/forms/d/1...	https://drive.google.com/drive/folders/1...	https://docs.google.com/forms/d/1...	code101	storage101	7366	https://chatbot-setup-99bd7.web.app/	https://chatbot-setup-99bd7.web.app/
102	https://docs.google.com/spreadsheets/d/1...	https://docs.google.com/forms/d/1...	https://drive.google.com/drive/folders/1...	https://docs.google.com/forms/d/1...	code102	storage102	5720	https://chatbot-setup-99bd7.web.app/	https://chatbot-setup-99bd7.web.app/
生徒番号	データベース	フォーム回答	画像フォルダ	フォームの編集		編集ページへのログインパスワード		編集ページ	実装ページ

+ ≡ env ▼ 1組 ▼

②⑫【編集ページの確認】code（例：7366）をコピーして editUrl にアクセスする

A	B	C	D	E	F	G	H	I	J
unit	sheet	form	folder	answer	deploy	save	code	editUrl	implUrl
101	https://docs.google.com/spreadsheets/d/1...	https://docs.google.com/forms/d/1...	https://drive.google.com/drive/folders/1...	https://docs.google.com/forms/d/1...	code101	storage101	7366	https://chatbot-setup-99bd7.web.app/	https://chatbot-setup-99bd7.web.app/
102	https://docs.google.com/spreadsheets/d/1...	https://docs.google.com/forms/d/1...	https://drive.google.com/drive/folders/1...	https://docs.google.com/forms/d/1...	code102	storage102	5720	https://chatbot-setup-99bd7.web.app/	https://chatbot-setup-99bd7.web.app/

②⑬シートアクセス用の合い言葉を入力する（例：password）

* 合い言葉は Chatbot ベースシートで設定した値になる

chatbot-setup-99bd7.web.app の内容

シートアクセス用の合い言葉を入力

password

OK キャンセル

	値	説明
PassPhrase	password	このシートのGASへアクセスするために必要な合言葉です 半角英数6~12文字程度で設定してください
SheetTemplateURL	https://docs.google.com	キーワード等を格納しておくためのテンプレートシートのURLです シートを開いた時のURLをそのまま貼り付けてください(tab等は押さないでください)

chatbot-setup-99bd7.web.app の内容

個人用合言葉(4桁)を入力

OK

キャンセル

②⑤設定完了 *このページまで移行できない場合にはキャッシュをクリアし、ページ更新

生徒への配付

*ユーザーへの質問を変更する場合には「answer」を編集する。その場合、対応した sheet 内のシート名「レビュー」の内容を変更する必要あり

[illegible]

【アクセスできない場合】



このよう画面が表示され、アクセスできない場合があります。

そのような場合は、システムを登録したドメインと生徒のドメインが違う場合に起こります。

キー	値	説明
PassPhrase	password	このシートのGASへアクセスするために必要な合言葉です 半角英数6~12文字程度で設定してください
SheetTemplateURL	https://docs.google.com	キーワード等を格納しておくためのテンプレートシートのURLです シートを開いた時のURLをそのまま貼り付けてください(tab等は押さないでください)
FormTemplateURL	https://docs.google.com	チャットボット終了後に表示するフォームのテンプレートです フォームを開いた時のURLを貼り付けてください(回答URLではありません)
TargetYear	2024	授業を実施する年度を西暦で入力してください
TargetGrade	2	授業を実施する学年を数字で入力してください
ClassSheetSuffix	組	クラスシートであることを示す末尾の文字列です 例: 「{任意の組名}組」というシートをクラスシートにする場合は「組」を指定
TargetSemester	1	授業が実施される学期を数字で入力してくださいex: 1学期である場合は1を入力
SchoolDomain	iizuna.ed.jp	学校で利用しているメールアドレスのドメイン
ChatbotOriginURL	https://chatbot-setup-99	授業で利用するチャットボットのオリジンURL

生徒のドメイン

システムを登録したドメイン

生徒のドメイン

この状態だとアクセスできない

解決策

システムを登録したドメイン

この情報をコピー

生徒のドメイン

ペースト

(default)	school-data	gmail.com
+ コレクションを開始	+ ドキュメントの追加	+ コレクションを開始
publish-data	gmail.com	+ フィールドを追加
school-data	iizuna.ed.jp	authenticated
		0 "k0001"
		deployed
		20240201: "https://script.google.com/macros/s/AKfycbzkvtFyHnpVwNhSOfeo9sQk9guV/exec"
		test01: "test02"

(default)	school-data	iizuna.ed.jp
+ コレクションを開始	+ ドキュメントの追加	+ コレクションを開始
publish-data	gmail.com	+ フィールドを追加
school-data	iizuna.ed.jp	authenticated
		0 "t0001"
		deployed
		20240201: "https://script.google.com/macros/s/AKfycbzkvtFyHnpVwNhSOfeo9sQk9guV/exec"
		test01: "test02"
		pass
		test03: "test04"

「GEMINI_API_KEY」

Gemini API の取得方法

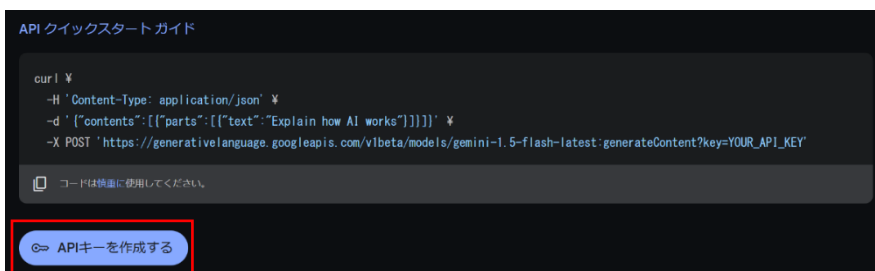
(1) 下記の URL にアクセスする

<https://ai.google.dev/gemini-api?hl=ja>

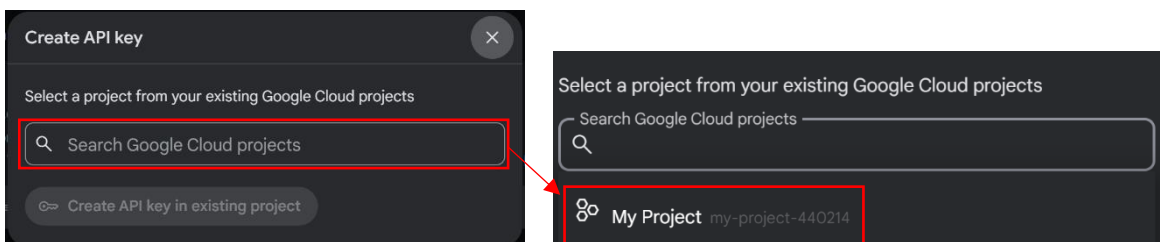
(2) 「Google AI Studio で API キーを取得する」をクリックする



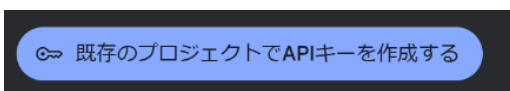
(3) 「API キーを作成する」をクリックする



(4) 「Search Google Cloud Projects」→ Google Cloud で作成したプロジェクトを選択する



(5) 「既存のプロジェクトで API キーを作成する」をクリックする



(6) 「コピー」をクリックする

