

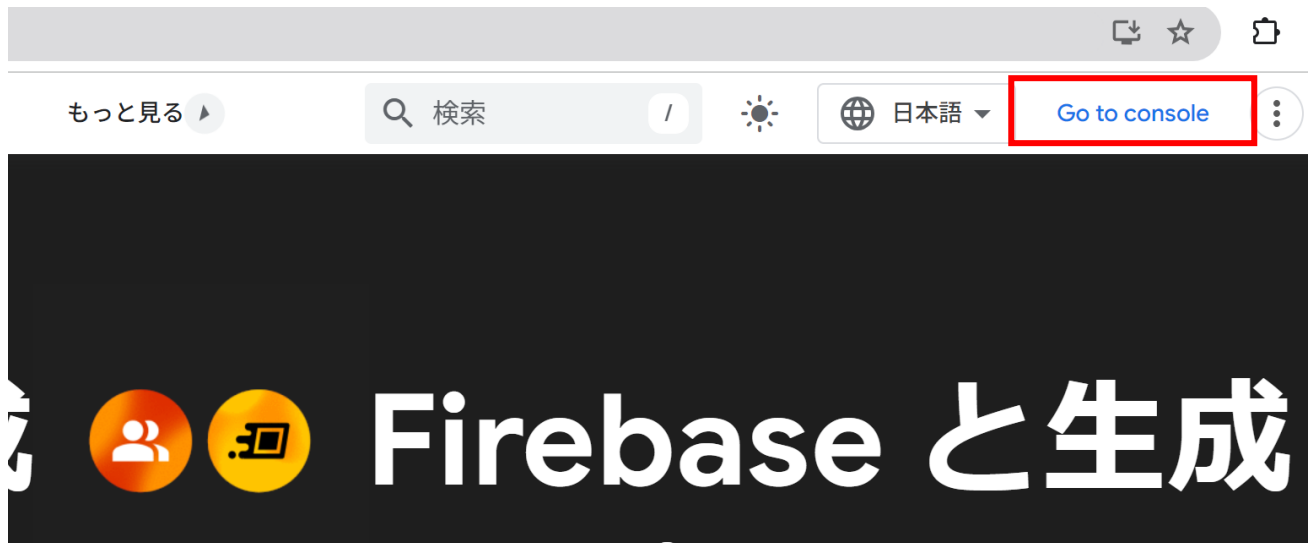
ChatBot-edu の設定方法

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



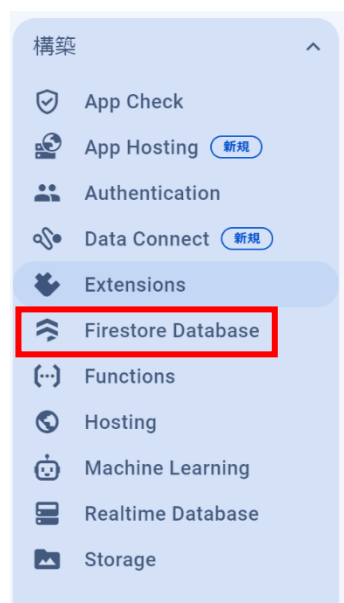
⑤ 「Google アナリティクス」を無効にするにして「プロジェクトを作成」をクリック



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 自動 ID (管理者及び使用生徒のドメイン (例: gmail.com))

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

フィールドを追加

フィールド タイプ 値

deployed = map

フィールドを追加

フィールド タイプ 値

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

フィールドを追加

フィールド タイプ 値

pass = map

フィールドを追加

フィールド タイプ 値

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

フィールドを追加

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

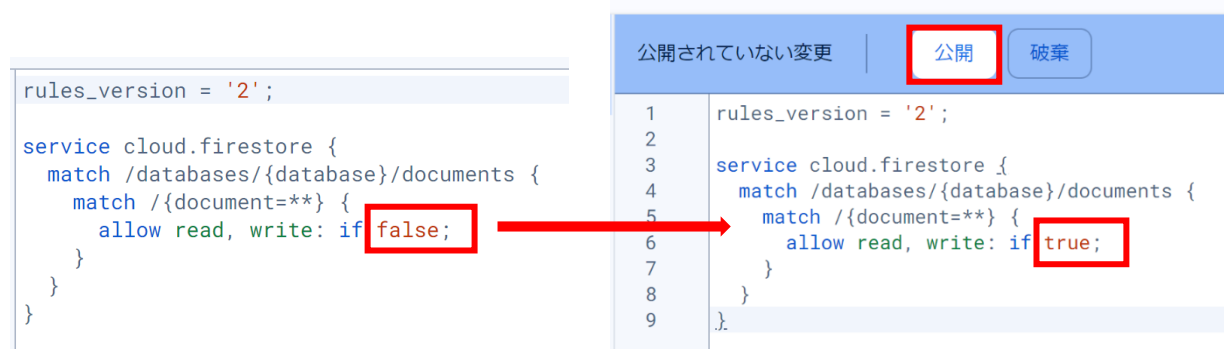
test05 = string test06 フィールドを「test05」
タイプを「string」
値を「test06」とする

フィールドを追加

⑪ 「ルール」 をクリック



⑫ 「false」 を 「true」 に変更し、「公開」 をクリック

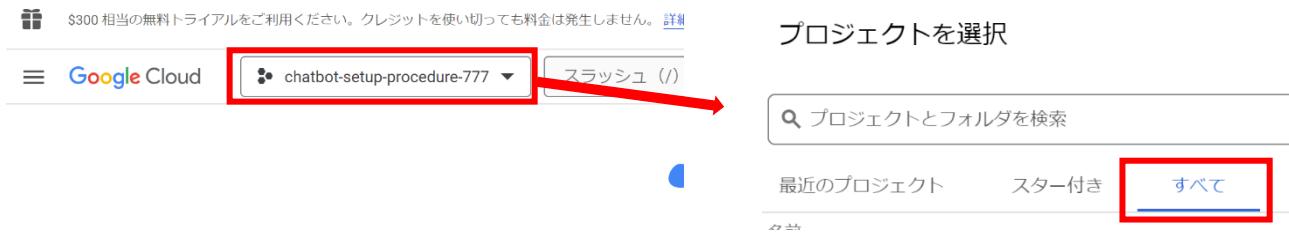


1. GoogleCloud の登録

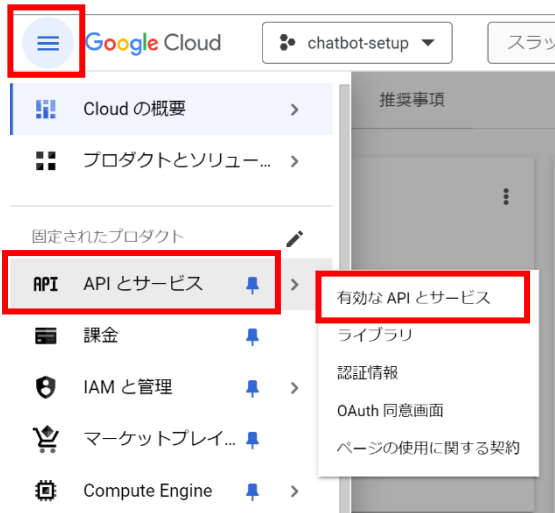
①下記の URL をクリックして Firebase にアクセス

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

②プロジェクト名を選択（例：chatbot-setup）



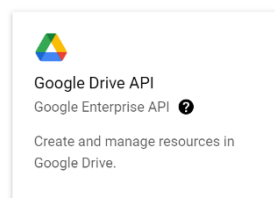
③「ナビゲーションウィンドウ」→「API とサービス」→「有効な API とサービス」をクリック



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

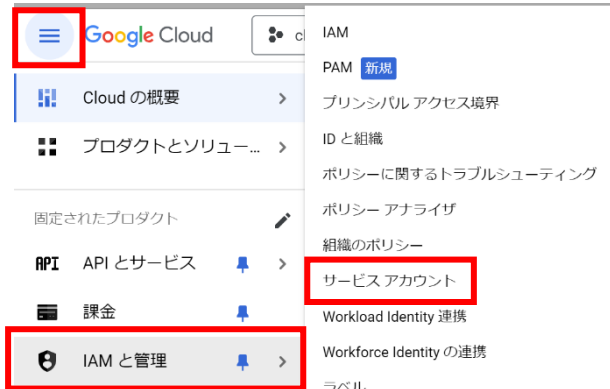


⑥「有効にする」をクリック



＊「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

| | | | | | | |
|--------------------------|--|------|--------------------|----------|-----------------------|-------------------------------------|
| <input type="checkbox"/> | firebase-connector@chatbot-setup.iam.gserviceaccount.com | ✓ 有効 | firebase-connector | キーがありません | 108048804061462031307 | <div>詳細を管理 権限を管理 鍵を管理</div> |
|--------------------------|--|------|--------------------|----------|-----------------------|-------------------------------------|

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

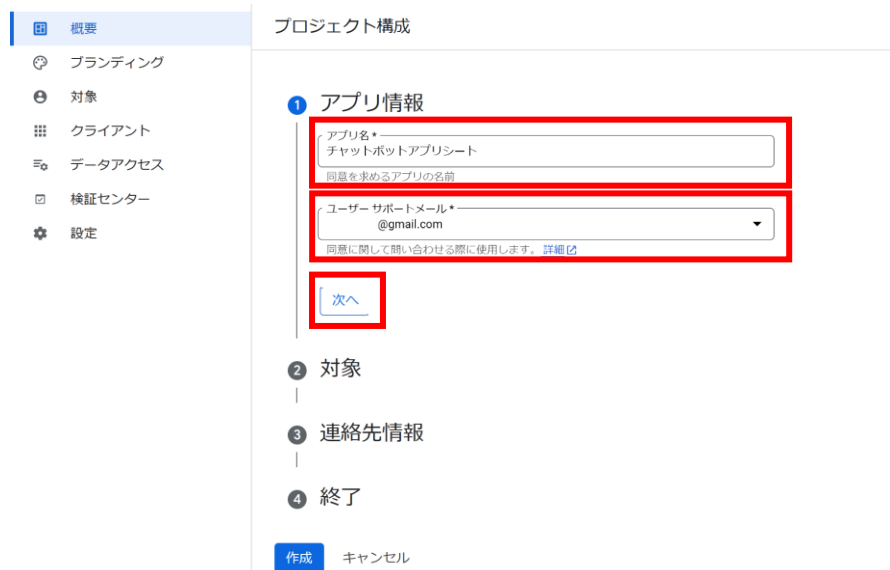
⑩「ナビゲーションウィンドウ」→「API とサービス」→「OAuth 同意画面」をクリック



⑪「開始」をクリック



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し、次へをクリック



⑱外部を選択して次ぎへをクリック

2 対象

内部 ②

組織内のユーザーのみが使用できます。確認を受けるためにアプリを送信する必要はありません。[ユーザーの種類の詳細](#)

[🔗](#)

外部 ②

Google アカウントを持つすべてのテストユーザーが使用できます。アプリはテストモードで起動し、アプリを使用できるのは、テストユーザーのリストに追加されたユーザーに限られます。アプリを本番環境に移す準備ができれば、アプリの確認が必要となる場合があります。[ユーザーの種類の詳細](#) [🔗](#)

次へ

⑲連絡できるメールを入力して次ぎへをクリック

3 連絡先情報

メールアドレス *

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。

次へ

⑳作成をクリック

㉑対象から「+ADD USERS」を選択して Firebase のアカウントを入力、「追加」をクリック

🏠 概要

🕒 ブランディング

👤 対象

👥 クライアント

🔑 データアクセス

🔍 検証センター

⚙️ 設定

対象

OAuth ユーザー数の上限 ②

公開ステータスが「テスト中」に設定されている間は、テストユーザーのみがアプリにアクセスできます。アプリの確認前の許可済みユーザー数の上限は 100 で、この上限はアプリの全期間でカウントされます。
[詳細](#) [🔗](#)

ユーザー 0 人 (0 人がテストユーザー、残り 0 人) / ユーザーの上限数は 100 人

②

テストユーザー

+ Add users

≡ フィルタ プロパティ名または値を入力 ②

ユーザー情報

表示する行がありません


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

スタンドアロン バイナリ npm

Firebase CLI のバイナリをダウンロードして実行する方法は次のとおりです。

1. [Windows 用の Firebase CLI バイナリ](#)  をダウンロードします。
2. バイナリにアクセスして、`firebase` コマンドを実行できるシェルを開きます。
3. 続けて[ログインして CLI をテスト](#)します。

```

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.

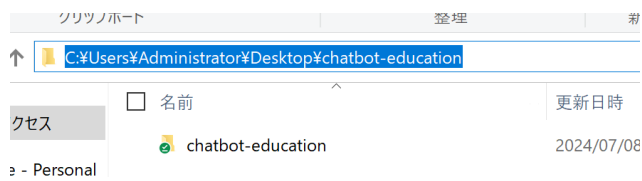
? Allow Firebase to collect CLI and Emulator Suite usage and error reporting information? (Y/n) _

```

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: Set up deployments for static web apps 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

```
Are you ready to proceed? Yes
? Which Firebase features do you want to set up for this directory? Press Space to select features, then Enter to confirm your choices.
  ( ) Genkit: Setup a new Genkit project with Firebase
  ( ) Functions: Configure a Cloud Functions directory and its files
  ( ) App Hosting: Set up deployments for full-stack web apps (supports server-side rendering)
  > (*) Hosting: Set up deployments for static web apps
  ( ) Storage: Configure a security rules file for Cloud Storage
  ( ) Emulators: Set up local emulators for Firebase products
  ( ) Remote Config: Configure a template file for Remote Config
↑ ↓ navigate · space select · a all · i invert · ↵ submit
```

⑦Use an existing project を選択し、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
```

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

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

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

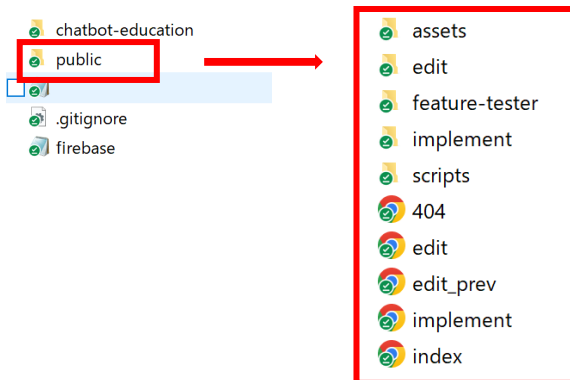
⑩? Configure as a single-page app (rewrite all urls to /index.html)? (Y/n) という問いに対して Enter キーを押す

⑪? Set up automatic builds and deploys with GitHub? (Y/n) という問いに対して「n」を入力し、Enter キーを押す

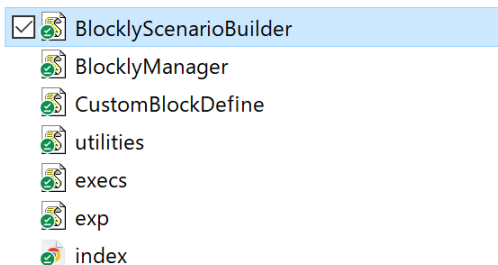
⑫chatbot-education フォルダに移行し、その中の chatbot-education フォルダの中身をすべてコピー

- 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「ここに入力」に必要事項をペーストして上書き保存 *AZURE_API 取得方法 P 2 8 ~

```
/**
 * 生成AI (Azure OpenAI)
 * @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;
```

```
// Azure OpenAI設定 (環境変数または設定から取得)
const AZURE_OPENAI_API_KEY = "ここに入力";
const AZURE_OPENAI_API_VERSION = "ここに入力";
const AZURE_OPENAI_DEPLOYMENT = "ここに入力";
const AZURE_OPENAI_ENDPOINT = "ここに入力";
```

AZURE_OPENAI_API_VERSION は
「2025-01-01-preview」でよい

⑩Firebase CLIに戻り、「firebase deploy」を入力してEnterキーを押す

⑪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!

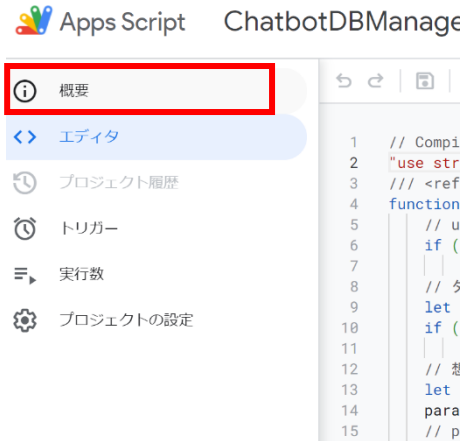
Project Console: https://console.firebase.google.com/project/chatbot-setup/overview
Hosting URL: https://chatbot-setup.web.app
```

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/d/1EEIWB1MfJEISVqva_qMtHv4aO71GxfMwBZzRdEYq1vfBuDH4MaUoe1e8/edit?usp=sharing

②「概要」をクリック



③「コピーを作成」をクリック

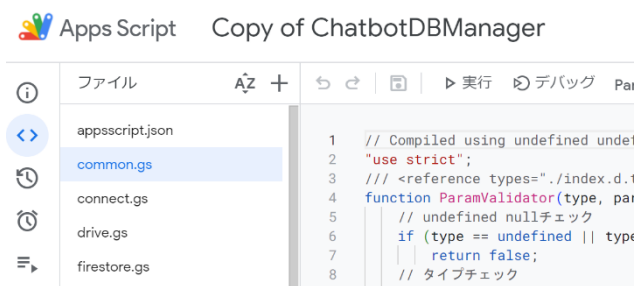
プロジェクトの詳細



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



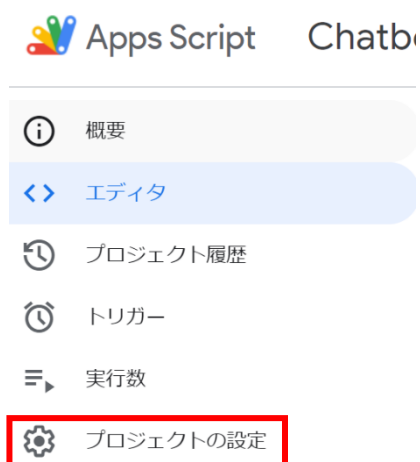
④「Copy of ChatbotDBManager」の「Copy of」部分を削除し「ChatbotDBManager」とする
プロジェクトの名前を変更

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



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

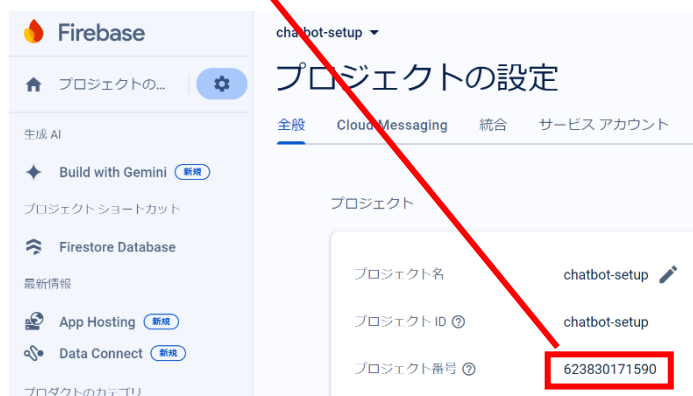
Apps Script プロジェクトで
詳細が管理されます。詳しくは

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
プロパティを定義および公開できます
ご覧ください。

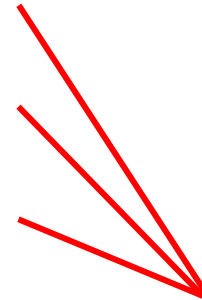
スクリプト プロパティを追加

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 | 値 |
| client_email | firebase-connector@chatbot-setup-procedur |
| private_key | -----BEGIN PRIVATE KEY-----\nMIIIEvQIBADANI |
| project_id | chatbot-setup |



* 1. GoogleCloud の登録⑮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+gk1OWtS/yzRnTLHivF/8d8JcifyciNwHq2CjbtKe8h7fyTMD+DsJy4ndY4ujbxYPUC2HZOMQ3taxyJhZQKBgQDJ8B48gMJE6zsQJZol1Bx6VWdazJjKsi+BYnxgk0u60trHmkHMbd6Jvp52sL1L9h2/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



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/15ClU1sgn8SW6LScb0WEUY7ZOeQGTuygd?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

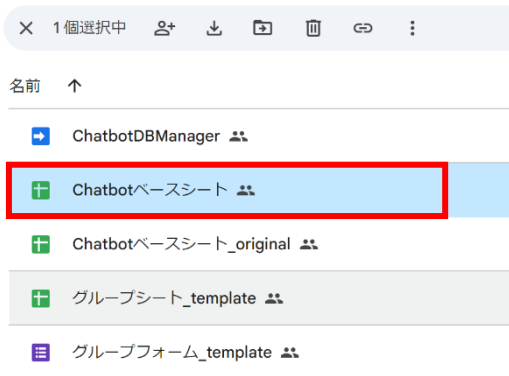
マイドライブ > chatbot-template



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 ▾



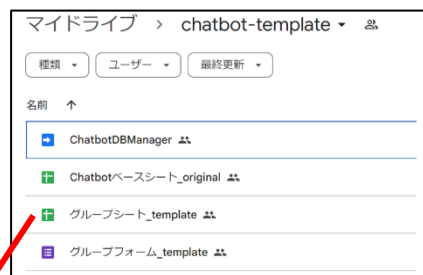
⑦マイドライブに戻り、新しく「Chatbot」フォルダを作成する

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



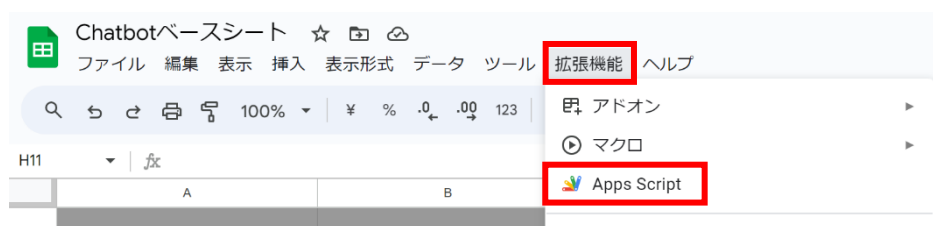
Google フォームを
一度ファイルを開いてコピー

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

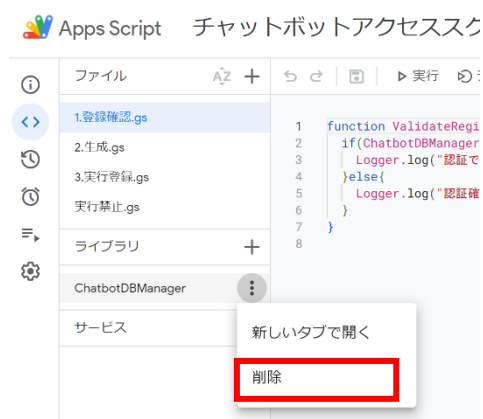
| | 値 | 説明 |
|------------------|---|---|
| PassPhrase | password | このシートのGASへアクセスするために必要な合言葉です 半角英数6~12文字程度で設定してください |
| SheetTemplateURL | https://docs.google.com/spreadsheets/d/1GvmOC9hVX3ZY2y464gsehmB8b6fTq5g2HaQ2_khGw/edit | キーワード等を格納しておくためのテンプレートシートのURLです シートを開いた時のURLをそのまま貼り付けてください(tab等は押さないでください) |
| 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 | https://docs.google.com | https://drive.google.com | https://docs.google.com | code101 | storage101 | 7366 | https://chatbot-setup-99bd7.web.app/ | https://chatbot-setup-99bd7.web.app/ |
| 102 | https://docs.google.com | https://docs.google.com | https://drive.google.com | https://docs.google.com | 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 | https://docs.google.com | https://drive.google.com | https://docs.google.com | code101 | storage101 | 7366 | https://chatbot-setup-99bd7.web.app/ | https://chatbot-setup-99bd7.web.app/ |
| 102 | https://docs.google.com | https://docs.google.com | https://drive.google.com | https://docs.google.com | 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) |

④個人用合言葉を入力する（例：7366）

chatbot-setup-99bd7.web.app の内容

個人用合言葉(4桁)を入力

7366

OK

キャンセル

| 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/1Qm8j... | https://docs.google.com/forms/d/1Qm8j... | https://drive.google.com/folders/1Qm8j... | https://docs.google.com/forms/d/1Qm8j... | code101 | storage101 | 7366 | https://chatbot-setup-99bd7.web.app/ | https://chatbot-setup-99bd7.web.app/ |
| 102 | https://docs.google.com/spreadsheets/d/1Qm8j... | https://docs.google.com/forms/d/1Qm8j... | https://drive.google.com/folders/1Qm8j... | https://docs.google.com/forms/d/1Qm8j... | code102 | storage102 | 5720 | https://chatbot-setup-99bd7.web.app/ | https://chatbot-setup-99bd7.web.app/ |

⑤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

Chatbot編集ページ

2024 年度 02 学年 01 学期 1 組 1 班

シナリオ番号 0

ユーザー入力

ボット出力

分岐番号 1

データベース参照 0

終了

シナリオジャンプ 0

▶ 実行

▶ ステップ実行

▶▶ ステップ

■ 停止

📄 実装ページへ

保存操作 ▶

テキストを入力

▶

生徒への配付

別のスプレッドシートを作成し、「unit」、「sheet」、「folder」、「code」、「editUrl」を伝える

*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

| unit | sheet | form | folder | answer | deploy | save | code | editUrl | implUrl |
|------|---|---|---|---|---------|------------|------|---|---|
| 101 | https://docs.google.com/spreadsheets/d/1Qm8j... | https://docs.google.com/forms/d/1Qm8j... | https://drive.google.com/folders/1Qm8j... | https://docs.google.com/forms/d/1Qm8j... | code101 | storage101 | 7366 | https://chatbot-setup-99bd7.web.app/ | https://chatbot-setup-99bd7.web.app/ |
| 102 | https://docs.google.com/spreadsheets/d/1Qm8j... | https://docs.google.com/forms/d/1Qm8j... | https://drive.google.com/folders/1Qm8j... | https://docs.google.com/forms/d/1Qm8j... | code102 | storage102 | 5720 | https://chatbot-setup-99bd7.web.app/ | https://chatbot-setup-99bd7.web.app/ |

【アクセスできない場合】



このようが画面が表示され、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" |

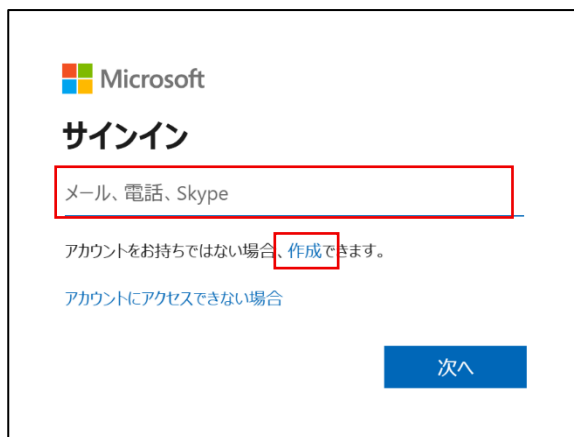
「AZURE_API」に関する情報の取得方法

- ① 下記の URL に移行し、「Azure を無料でお試ください」をクリックする

<https://azure.microsoft.com/ja-jp/pricing/purchase-options/azure-account?icid=azurefreeaccount>



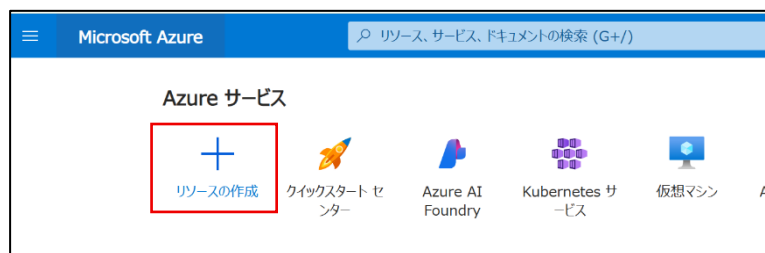
- ② サインインまたはアカウントがない場合には「作成」をクリックする



- ③ サインインが完了したら、下記の URL にアクセスする

<https://portal.azure.com/?quickstart=True#home>

- ④ 「リソースの作成」をクリック



⑤ 「Azure OpenAI」をクリックする



⑥ リソースグループに新規作成で名前を入力（例：Assistant-AI-Chat）

名前*にも名前を入力（例：Assistant-AI-Chat）

価格レベル*を「Standard S0」にする

詳細情報

プロジェクトの詳細

サブスクリプション * ⓘ Azure subscription 1

リソース グループ * ⓘ (新規) Assistant-AI-Chat
新規作成

インスタンスの詳細

リージョン ⓘ East US

名前 * ⓘ Assistant-AI-Chat ✓

価格レベル * ⓘ Standard S0

価格の詳細を表示

< 前へ 次へ

⑦ 「次へ」をクリックする

Microsoft Azure

リソース、サービス、ドキュメントの検索 (G+)

ホーム > リソースの作成 >

Azure OpenAI の作成 ...

基本

ネットワーク

Tags

レビューおよび送信

1

Configure network security for your Azure AI services resource.

?

種類 *

☒ インターネットを含むすべてのネットワークがこのリソースにアクセスできます。

☐ Selected networks, configure network security for your Azure AI services resource.

☐ 無効になっています。どのネットワークもこのリソースにアクセスできません。このリソースにアクセスする唯一の方法となるプライベート エンドポイント接続を構成できます。

< 前へ

次へ

⑧ 「次へ」をクリックする

Microsoft Azure

リソース、サービス、ドキュメントの検索 (G+)

ホーム > リソースの作成 >

Azure OpenAI の作成 ...

基本

ネットワーク

Tags

レビューおよび送信

タグは名前と値のペアで、同じタグを複数のリソースやリソース グループに適用することでリソースを分類したり、統合した請求を表示したりできるようにします。 [タグに関する詳細情報](#)

タグを作成してから別のタブでリソースの設定を変更すると、タグは自動的に更新されることにご注意ください。

名前 ①

値 ①

リソース

:

2 項目が選択されました ▼

< 前へ

次へ

⑨ 「作成」をクリックする

Microsoft Azure

リソース、サービス、ドキュメントの検索 (G+)

ホーム > リソースの作成 >

Azure OpenAI の作成

基本

ネットワーク

Tags

レビューおよび送信

Automation のテンプレートをダウンロードする

使用条件

"作成" をクリックすることで、お客様は (a) 上記の Marketplace のオファリングに関連する法律条項とプライバシーに関する声明に同意し、(b) Microsoft より、そのオファリングに関連する料金が、現在の支払い方法に対して Azure サブスクリプションと同じ請求頻度で請求されることを認め、かつ、(c) Microsoft がお客様の連絡先情報、使用量情報、取引に関する情報を、レポート、請求、その他の取引上のアクティビティを目的として、オファリングのプロバイダーと共有する可能性があることに同意するものとします。Microsoft は、サードパーティのオファリングに対する権利は提供しません。その他の詳細については、[Azure Marketplace 使用条件](#)を参照してください。

基本

| | |
|-----------|----------------------|
| サブスクリプション | Azure subscription 1 |
| リソース グループ | Assistant-AI-Chat |
| リージョン | East US |
| 名前 | Assistant-AI-Chat |
| 価格レベル | Standard S0 |

ネットワーク

種類

インターネットを含むすべてのネットワーク状のリソースにアクセスできます。

< 前へ

次へ

作成

⑩ デプロイが完了したら「リソースに移動」をクリックする

Microsoft Azure

リソース、サービス、ドキュメントの検索 (G+)

ホーム >

Microsoft.CognitiveServicesOpenAI-20250530170135 | 概要

デプロイ

検索

削除

キャンセル

再デプロイ

ダウンロード

最新の

概要

入力

出力

テンプレート

デプロイが完了しました

デプロイ名

サブスクリプション

リソース グループ

: Microsoft.CognitiveServicesOpenAI-...

: Azure subscription 1

: Assistant-AI-Chat

開始日時

関連付け

> デプロイの詳細

< 次の手順

リソースに移動

フィードバックの送信

デプロイに関するエクスペリエンスについてお聞かせください

⑪ 「Explore Azure AI Foundry portal」をクリックする

The screenshot shows the Microsoft Azure portal interface. At the top, there's a navigation bar with the Microsoft Azure logo and a search bar. Below the navigation bar, the breadcrumb trail reads: ホーム > Microsoft.CognitiveServicesOpenAI-20250530170135 | 概要 >. The main heading is 'Assistant-AI-Chat' with 'Azure OpenAI' underneath. A search bar is present, and a link 'Go to Azure AI Foundry portal' is visible. On the left, there's a sidebar with a '概要' (Overview) section and a list of links: アクティビティ ログ, アクセス制御 (IAM), タグ, 問題の診断と解決, リソース ビジュアライザー, リソース管理, Security, 監視, オートメーション, ヘルプ. The main content area has a '基本' (Basics) tab selected, showing details for the resource group 'Assistant-AI-Chat', status 'アクティブ', location 'East US', subscription 'Azure subscription 1', and subscription ID '5ff62e1e-b237-'. There are also links for 'タグ (編集)' and 'タグの追加'. Below this, there's a section titled 'Build your own secure copilot and generative AI applicati...' with a description: 'Deploy an Azure OpenAI model and start making API calls to generate text, data, call functions, and improve workflow with Azure OpenAI image and speech models. You can access the service through the Azure AI Foundry portal, our Python SDK, or our web-based interface in the Azure AI Foundry portal.' A 'Learn More' link is provided. At the bottom, there's a section titled 'Explore and deploy' with a description: 'Explore and deploy the generative AI models in the Azure AI Foundry portal. Craft unique prompts for your use cases, and fine-tune select models.' A red box highlights the 'Explore Azure AI Foundry portal' button.

⑫ 「新しいデプロイの作成」をクリックする

チャットプレイグラウンド

The screenshot shows the Azure AI Foundry portal interface. At the top, there's a navigation bar with 'コードの表示' (Show code) and 'デプロイ' (Deploy) buttons. Below the navigation bar, the main heading is 'セットアップ' (Setup) with a '非表示' (Hide) button. The main content area has a 'デプロイ' (Deploy) section with a red box highlighting the '+ 新しいデプロイの作成' (Create new deployment) button. Below this, there are two options: '基本モデルから' (From base model) and '微調整されたモデルから' (From fine-tuned model). A large blue plus button is also visible at the bottom.

- ⑬ 「gpt-35-turbo」を選択し、「確認」をクリックする(例：gpt-35-turbo)

「AZURE_OPENAI_DEPLOYMENT」 = 「gpt-35-turbo」 P13 の「ここに入力」に貼り付ける

モデル 18 推論タスク: チャットの完了

説明の表示

検索

o1-preview
チャットの完了

○

gpt-4
チャットの完了

○

gpt-35-turbo-16k
チャットの完了

○

gpt-35-turbo-instruct
チャットの完了

○

gpt-35-turbo
チャットの完了

●

gpt-4-32k
チャットの完了

○

< 前へ

次へ >

gpt-35-turbo

タスク: チャットの完了

The gpt-35-turbo (also known as ChatGPT) is the most capable and cost-effective model in the gpt-3.5 family which has been optimized for chat using the Chat Completions API. It is a language model designed for conversational interfaces and the model behaves differently than previous gpt-3 models. Previous models were text-in and text-out, meaning they accepted a prompt string and returned a completion to append to the prompt. However, the ChatGPT model is conversation-in and message-out. The model expects a prompt string formatted in a specific chat-like transcript format and returns a completion that represents a model-written message in the chat.

Learn more at <https://learn.microsoft.com/azure/cognitive-services/openai/concepts/models>

確認

キャンセル

- ⑭ 「デプロイ」をクリックする

gpt-35-turbo の展開

デプロイ名 *

gpt-35-turbo

デプロイの種類

Standard

Standard: API 呼び出しごとに支払い、より低いレート制限が適用されます。Azure データ所在地の確約に従います。量が小規模から中規模の、間欠的に発生するワークロードに最適です。[Standard 展開](#) の詳細をご覧ください。

▼ デプロイの詳細

カスタマイズ

モデル バージョン
0125

AI リソース
Assistant-AI-Chat

容量
100K 1 分あたりのトークン数 (TPM)

リソースの場所
East US

コンテンツの安全性
DefaultV2

バージョン アップグレード ポリシー
新しい既定のバージョンが使用可能になったら

デプロイ

取り消し

- ⑮ ⑪のページまで戻り、「リソース管理」→「キーとエンドポイント」をクリックする

Microsoft Azure

リソース、サービス、ドキュメントの検索 (G+/)

ホーム > Microsoft.CognitiveServicesOpenAI-20250530170135 | 概要 >

Assistant-AI-Chat ☆ ☆ ...
Azure OpenAI

検索

Go to Azure AI Foundry portal 削除

概要

- アクティビティ ログ
- アクセス制御 (IAM)
- タグ
- 問題の診断と解決
- リソース ビジューアライザー
- リソース管理**
 - キーとエンドポイント**
 - 暗号化
 - 価格レベル

基本

リソース グループ (移動) : [Assistant-AI-Chat](#)

状態 : アクティブ

場所 : East US

サブスクリプション (移動) : [Azure subscription 1](#)

サブスクリプション ID : 5ff62e1e-b237-456c-i

タグ (編集) : [タグの追加](#)

開始する Develop Monitor

Build your own secure copilot and

- ⑯ 「キー 1」 = 「AZURE_OPENAI_API_KEY」
「エンドポイント」 = 「AZURE_OPENAI_ENDPOINT」を P13 の「ここに入力」に貼り付ける

Microsoft Azure

リソース、サービス、ドキュメントの検索 (G+/)

Copilot

ホーム > Microsoft.CognitiveServicesOpenAI-20250530170135 | 概要 > Assistant-AI-Chat

Assistant-AI-Chat | キーとエンドポイント ☆ ...
Azure OpenAI

検索

キー 1 の再生成 キー 2 の再生成

概要

- アクティビティ ログ
- アクセス制御 (IAM)
- タグ
- 問題の診断と解決
- リソース ビジューアライザー
- リソース管理
- キーとエンドポイント**
- 暗号化
- 価格レベル
- ネットワーク
- ID
- コスト分析

These keys are used to access your Azure AI Foundry API. Do not share your keys. Store them securely~ for example, using Azure Key Vault. We also recommend regenerating these keys regularly. Only one key is necessary to make an API call. When regenerating the first key, you can use the second key for continued access to the service.

キーの表示

キー 1

キー 2

場所/地域 ⓘ
eastus

エンドポイント
https://assistant-ai-chat.ope...