

各種設定方法

- GA4 と Claude の連携方法

※GA4 を ChatGPT/Copilot Studio/Gemini CLI と連携する方法は以下 URL から確認ください

<https://www.ga4.guide/related-service/ai/gamcp-compare/>

- Search Console と Claude の連携方法

- BigQuery と Claude の連携方法

GA4 X Claude 連携

作業フロー

全部で 9 つの STEP になります

1. Google クラウドプロジェクトの作成
2. API の有効化
3. サービスアカウントキーの作成
4. 認証情報ファイルのダウンロード
5. GA4 に権限の付与
6. GA4 からプロパティ ID の取得
7. node.js のインストール
8. Claude Desktop のインストール
9. Claude Desktop での設定

1.Google クラウドプロジェクトの作成

Google Cloud にログインし、Google Cloud ロゴの右側にあるボックスをクリックして、「新しいプロジェクト」を選択



1.Google クラウドプロジェクトの作成

「プロジェクト名」を入れて作成を行う

Google Cloud

スラッシュ (/) を使用してリソース、ドキュメント、

新しいプロジェクト

割り当て内の残りのプロジェクト数は 8 projects 件です。プロジェクトの増加をリクエストするか、プロジェクトを削除してください。 [詳細](#)

[Manage Quotas](#)

プロジェクト名 *

Claude GA4 Connection 2

?

プロジェクト ID: claude-ga4-connection-2 後で変更することはできません。 [編集](#)

場所 *

組織なし

[参照](#)

親組織またはフォルダ

作成

キャンセル

1.Google クラウドプロジェクトの作成

作成されたプロジェクト名を選択

スラッシュ (/) を使用してリソース、ドキュメント、プロダクトなどを..

検索

通知

✔ プロジェクト「Claude GA4 Connection 2」
を作成
プロジェクトを選択

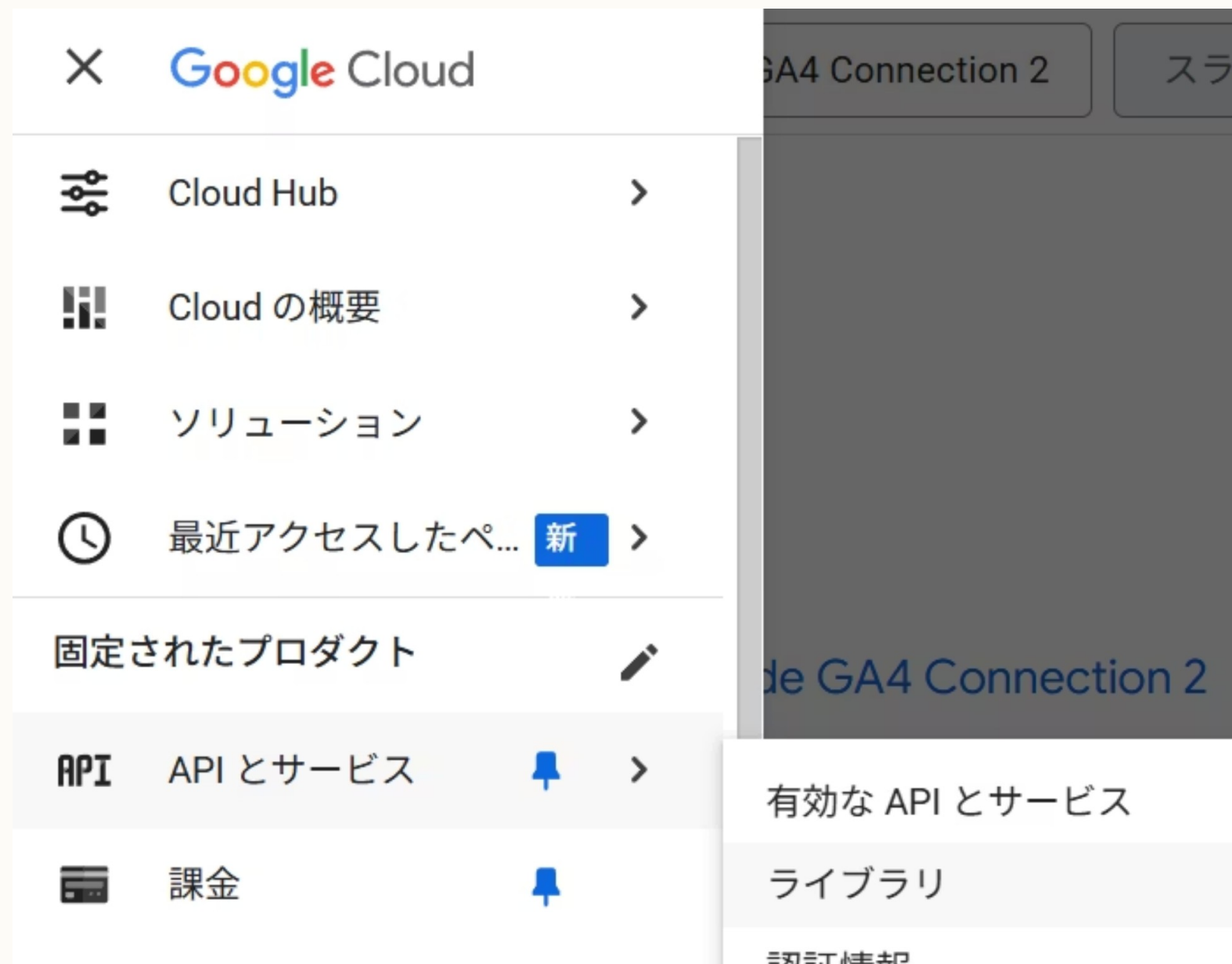
たった今

✔ 次のサービスを有効にします:

12 時間前

2.API の有効化

左側のメニューから「API とサービス」>「ライブラリ」を選択



2.APIの有効化

検索ボックスで「Google Analytics Data API」と入れて検索

Google Cloud

Claude GA4 Connection 2

🔍 ✨ 📄 🔔 ⓘ ⋮ 👤

API API とサービス / API ライブラリ / 参照

API API ライブラリ

🔍 Google Analytics Data API ✕

API ライブラリ > "Google Analytics Data API"

≡ フィルタ

フィルタ テキスト
を入力

19 件の結果

公開設定

公開

カテゴリー

アナリティクス

ビッグデータ

データベース

ML

DevOps

コンピューティング

医療、ライフサイエンス

科学、研究

↑

(19)

↑

(8)

(8)

(3)

(2)

(3)

(1)

(1)

(1)



Analytics Hub API

Google Enterprise API ⓘ

Analytics Hub provides an easy way to publish, discover and subscribe to share BigQuery datasets between users in your organization or other organizations. Once you create an exchange, you can invite others to publish or subscribe to data in the exchange.



Pub/Sub Lite API

Google Enterprise API ⓘ

Pub/Sub Lite is a messaging service with zonal storage. Pub/Sub Lite costs a fraction of Pub/Sub and is meant for high volume streaming pipelines and event-driven system where low cost is the primary consideration. We recommend that most users use Pub/Sub, rather than Pub/Sub Lite, as it offers more flexibility, higher reliability, a broader range of features.



Google Analytics Data API

Google

Accesses report data in Google Analytics. Warning: Creating multiple Customer Applications, Accounts, or Projects to simulate a single Customer Application, Account, or Project (respectively) or to circumvent Service-specific usage limits or quotas is a violation of Google Cloud Platform Terms of Service as well as Google APIs Terms of Service. These actions can result in irreparable damage to your account.

2.APIの有効化

Google Analytics Data API を選択して、「有効にする」を押す



Claude GA4 Connection 2

製品の詳細



Google Analytics Data API

[Google](#)

Accesses report data in Google Analytics.

有効にする

この API を試す 

クリックしてこの API を有効にします

3. サービスアカウントキーの作成

左上ナビゲーションメニューから「IAM と管理」>「サービスアカウント」を選択



3. サービスアカウントキーの作成

「+サービスアカウントを作成」を選択

Google Cloud

Claude GA4 Connection 2

スラッシュ (/) を

IAM と管理 / サービス アカウント

+人

サービス アカウント

+ サービス アカウントを作成

盾

プロジェクト「Claude GA4 Connection 2」のサービス ア

ント

3. サービスアカウントキーの作成

「+サービスアカウントを作成」を入れて「作成して続行」
STEP2/STEP3 はスキップし「完了」を押す

[←](#) サービス アカウントの作成

1 サービス アカウントの作成

サービス アカウント名

Claude Analytics Access 2

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 *

claude-analytics-access-2

✕ ↺

メールアドレス: claude-analytics-access-2@claude-ga4-connection-2.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

2 権限 (省略可)

|

3 アクセス権を持つプリンシパル (省略可)

完了

キャンセル

4. 認証情報ファイルのダウンロード

サービスアカウント名（ claude-analytics-access 以下略） となっている部分を選択

プロジェクト「Claude GA4 Connection 2」のサービス アカウ
ント

サービス アカウントは Google Cloud サービス ID（Compute Engine VM、App Engine アプリ、Google 以外で実行されているシステムなどで実行されているコード）を表しま
す。 [サービス アカウントの詳細をご覧ください。](#)

組織のポリシーを使用してサービス アカウントを保護できます。IAM ロールの自動付与、鍵の作成やアップロード、サービス アカウントの完全な作成など、リスクのあるサー
ビス アカウント機能をブロックすることが可能です。 [サービス アカウントの組織のポリシーの詳細をご覧ください。](#)

≡ フィルタ

プロパティ名または値を入力

?

|||

☐	メール	ステータス	名前 ↑	説明	キー ID	キーの作成日	OAuth 2 クライアント ID ?	操作
☐	📧 claude-analytics-access-2@claude-ga4-connection-2.iam.gserviceaccount.com 📎	✔️ 有効	Claude Analytics Access 2		キーが ありま せん		102557628128235644690	⋮

4. 認証情報ファイルのダウンロード

上部メニューから「鍵」を選択し、ページ下部にある「キーを追加＞新しい鍵を作成」を選択

+人

🛡️

🔒

📊

👤

🔧

📄

📁

🔗

📋

🔑

📅

⚙️

🛡️

🏠

← Claude Analytics Access 2

詳細 権限 **鍵** 指標 ログ アクセス権を持つプリンシパル

鍵

⚠️

サービス アカウント キーは、不正使用されるとセキュリティ上のリスクになる可能性があります。Workload Identity 連携を使用することをおすすめします。Google Cloud でサービス ください。

ℹ️

Google は公開リポジトリで検出されたサービス アカウント キーを自動的に無効にします。この動作をカスタマイズできます。[詳細](#)

新しい鍵ペアを追加するか、既存の鍵ペアから公開鍵証明書をアップロードしてください。

[組織のポリシー](#)を使用して、サービス アカウント キーの作成をブロックします。
[サービス アカウント用の組織のポリシーの設定の詳細](#)

キーを追加 ▾

新しい鍵を作成

既存の鍵をアップロード

キー

作成日

有効期限

4. 認証情報ファイルのダウンロード

JSON 形式を選び「作成」

「Claude Analytics Access 2」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

推奨

☐ P12

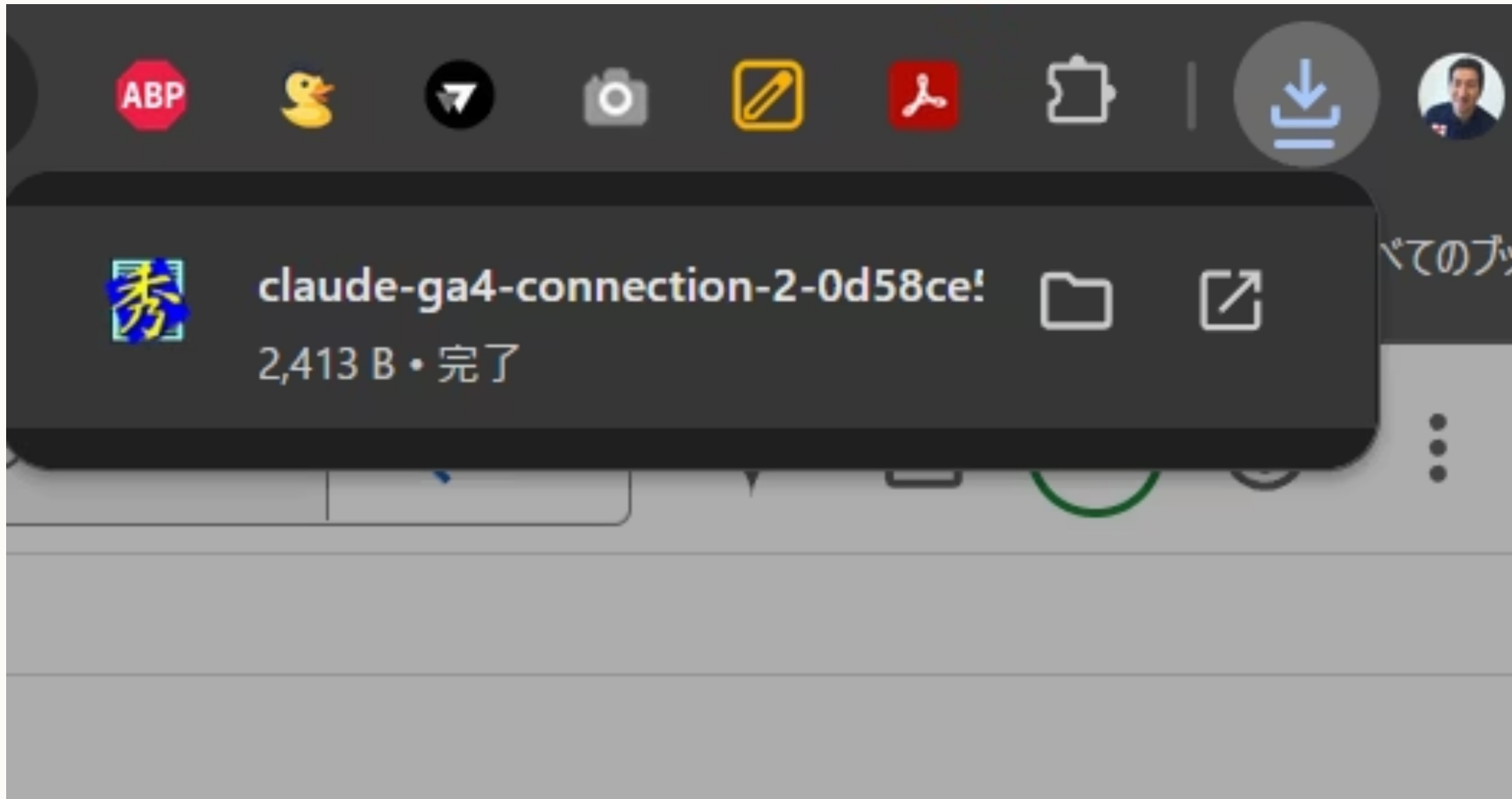
P12 形式を使用したコードとの下位互換性を目的としています

キャンセル

作成

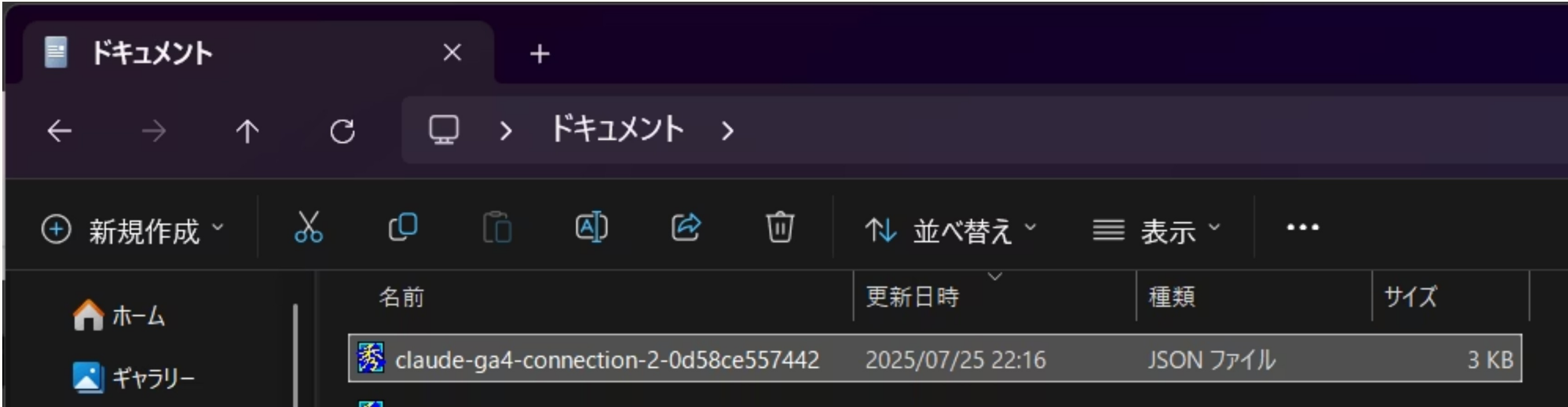
4. 認証情報ファイルのダウンロード

作成されたファイルが自動でダウンロード



4. 認証情報ファイルのダウンロード

今後、このファイルを使うためにわかりやすい場所（例：ドキュメントフォルダ）に移す



5.GA4 に権限付与

1.[OAuth 2.0 Playground](#) を開き、左の入力欄（Input your own scopes）に `https://www.googleapis.com/auth/analytics.manage.users` を入力して「Authorize APIs」をクリックします。

2.GA4 プロパティの管理者権限を持つ Google アカウントで認可し、「Exchange authorization code for tokens」を実行してアクセストークンを取得します。

3.HTTP メソッドを「POST」に変更し、Request URI に以下を入力します（{ プロパティ ID} は GA4 プロパティ ID に置き換え）例：308259408。

`https://analyticsadmin.googleapis.com/v1alpha/properties/{ プロパティ ID}/accessBindings`

4.Enter Request Body を押し 以下のJSON を入力します（メールアドレスは前述のサービスアカウントに置き換え）。

```
{  
  "user": "xxx@xxx.iam.gserviceaccount.com",  
  "roles": ["predefinedRoles/analyst"]  
}
```

5.「Send the request」をクリックし、レスポンスに HTTP/1.1 200 OK が返れば成功です。GA4 管理画面の「プロパティのアクセス管理」にサービスアカウントが表示され、後から役割の変更も可能です。

※ API は必ず `v1alpha` を使用してください（`v1beta` には `accessBindings` リソースが存在せず 404 になります）。利用後は Playground の「サードパーティのアプリとサービス」からリフレッシュトークンのアクセスを削除しておく安全です。

5. GA4 に権限付与



OAuth 2.0 Playground

X

i

∞

⚙

▶ Step 1 Select & authorize APIs

▶ Step 2 Exchange authorization code for tokens

▶ Step 3 Configure request to API

Construct your HTTP request by specifying the URI, HTTP Method, headers, content type and request body. Then click the "Send the request" button to initiate the HTTP Request.

HTTP Method: POST Add headers 0

Request URI: https://analyticsadmin.googleapis.com/v1alpha/properties/3082f

Enter request body 105 Content-Type: application/json

Send the request List possible operations

Note: The OAuth access token in Step 2 will be added to the Authorization header of the request.

Request / Response

```
POST /analyticsadmin.googleapis.com/v1alpha/properties/3082f HTTP/1.1
Host: analyticsadmin.googleapis.com
Content-Type: application/json
Authorization: Bearer ya29.a0AfHjSbp...
Accept-Encoding: gzip;q=1.0,*/*;q=0.5
Accept-Language: en-US
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7; rv:109.0) Gecko/20100101 Firefox/109.0
Accept: */*
Referer: https://developers.google.com/oauthplayground/
```

```
{
  "user": "ga4clienttest@gtmga4nanage.iam.gserviceaccount.com",
  "roles": ["predefinedRoles/analyst"]
}
```

```
HTTP/1.1 200 OK
Content-length: 265
X-xss-protection: 0
X-content-type-options: nosniff
Transfer-encoding: chunked
Vary: Origin, X-Origin, Referer
Server: ESF
-content-encoding: gzip
Date: Fri, 22 May 2026 08:20:40 GMT
X-frame-options: SAMEORIGIN
Alt-svc: h3=":443"; ma=2592000, h3-29=":443"; ma=2592000
Content-type: application/json; charset=UTF-8
```

```
{
  "name": "properties/308259408/accessBindings/AYVX2eynXeKksv2rd0Ivg51-o GC9LpETTNzgutMecG5ra1bkV0IH 9EkZdPkYoX ClfFpVEC kaemD5YWsa-paLjIAEPkJsg==",
  "roles": [
    "predefinedRoles/analyst"
  ],
  "user": "ga4clienttest@gtmga4nanage.iam.gserviceaccount.com"
}
```

6.GA4 からプロパティ ID を取得

管理＞プロパティ＞プロパティの詳細を選択。プロパティ ID をメモしておきます

- ❑ U-XXXXXX 形式の測定 ID ではありません
9桁の数値の番号になります

アナリティクス | すべてのアカウント > ryuka01 | HA(GA4計測メイン) | 「ユーザー数で上位の国」と検索してみてください

プロパティの詳細

プロパティ ID: 227084301

プロパティ ID をコピー

プロパティ名 (必須)

HA(GA4計測メイン)

レポートのタイムゾーン

日本 (GMT+09:00) 日本時間

通貨の表示

日本円 (¥)

作成

管理

設定

設定アシスタント

アカウント設定

アカウント

プロパティ設定

プロパティ

プロパティの詳細

プロパティのアクセス管理

プロパティを移行

ゴミ箱に移動

7.node.js のインストール

nodejs.orgにアクセス。上のメニューから「ダウンロード」を選択

The screenshot shows the Node.js website with a dark theme. The top navigation bar includes links for '学ぶ' (Learn), 'はじめに' (Getting Started), 'ダウンロード' (Download), 'ブログ' (Blog), 'ドキュメント' (Documentation), '貢献' (Contribute), and '認定資格' (Certification). A search bar on the right says '入力を開始...' (Start typing...). The main content area features the heading 'どこでもJavaScriptを使おう' (Use JavaScript anywhere) and a paragraph describing Node.js as a cross-platform JavaScript runtime environment. Below this are two buttons: 'Node.js®を入手' (Get Node.js) and 'セキュリティサポート' (Security Support) for Node.js 18 and below. On the right, a code editor titled 'Create an HTTP Server' shows a JavaScript snippet for creating a simple HTTP server. A 'JavaScript' label and a 'クリップボードにコピー' (Copy to clipboard) button are at the bottom of the code editor. At the very bottom, a footer text reads: '私たちの学習教材でNode.jsでできることをさらに学んでみましょう。' (Let's learn more about what we can do with Node.js using our learning materials.)

nodejs.orgにアクセス。上のメニューから「ダウンロード」を選択

nodejs.org

学ぶ はじめに ダウンロード ブログ ドキュメント 貢献 認定資格

🔍 入力を開始... Ctrl + K

どこでもJavaScript を使おう

Node.js®はクロスプラットフォームに対応した
フリーでオープンソースのJavaScript実行環境で
す。開発者にサーバー、ウェブアプリ、コマンド
ラインツール、スクリプトなどを開発する環境を
提供します。

Node.js®を入手

セキュリティサポート
Node.js 18以下向け

Create an HTTP Server Write Tests Read and Hash a File Streams Pipeline Work with Threads

```
1 // server.mjs
2 import { createServer } from 'node:http';
3
4 const server = createServer((req, res) => {
5   res.writeHead(200, { 'Content-Type': 'text/plain' });
6   res.end('Hello World!\n');
7 });
8
9 // starts a simple http server locally on port 3000
10 server.listen(3000, '127.0.0.1', () => {
11   console.log('Listening on 127.0.0.1:3000');
12 });
13
14 // run with `node server.mjs`
```

JavaScript クリップボードにコピー

私たちの学習教材でNode.jsでできることをさらに学んでみましょう。

7.node.js のインストール

自分が使っている OS を選んでください。Version に関しては「LTS」とついている最新のバージョンを選択し、ページ下部にあるインストーラーをクリック

node

学ぶはじめにダウンロードブログドキュメント貢献認定資格

🔍 入力を開始... Ctrl + K

🌞

Node.js®をダウンロードする

Windows

用のNode.js®

v22.17.1 (LTS)

 と

npm

 を

Docker

を使ってダウンロードする

補足 すぐに最新の機能を試したいですか？最新バージョンのNode.jsを代わりにインストールして最新の機能を試してください！

```
1 # DockerにはOSごとにそれぞれのインストール方法があります。
2 # 詳しくは https://docker.com/get-started/ にある公式ドキュメントを参照してください
3
4 # Node.jsのDockerイメージを取得する：
5 docker pull node:22-alpine
6
7 # Node.jsのコンテナを作成しシェルを起動する：
8 docker run -it --rm --entrypoint sh node:22-alpine
9
10 # Node.jsのバージョンを確認する：
11 node -v # "v22.17.1"が表示される。
12
13 # npmのバージョンを確認する：
14 npm -v # "10.9.2"が表示される。
```

PowerShell

クリップボードにコピー

Dockerはコンテナ化に関するプラットフォームです。もし何らかの問題を見つけた場合はDockerのウェブサイト を参照してください。

x64

アーキテクチャーで動作する

Windows

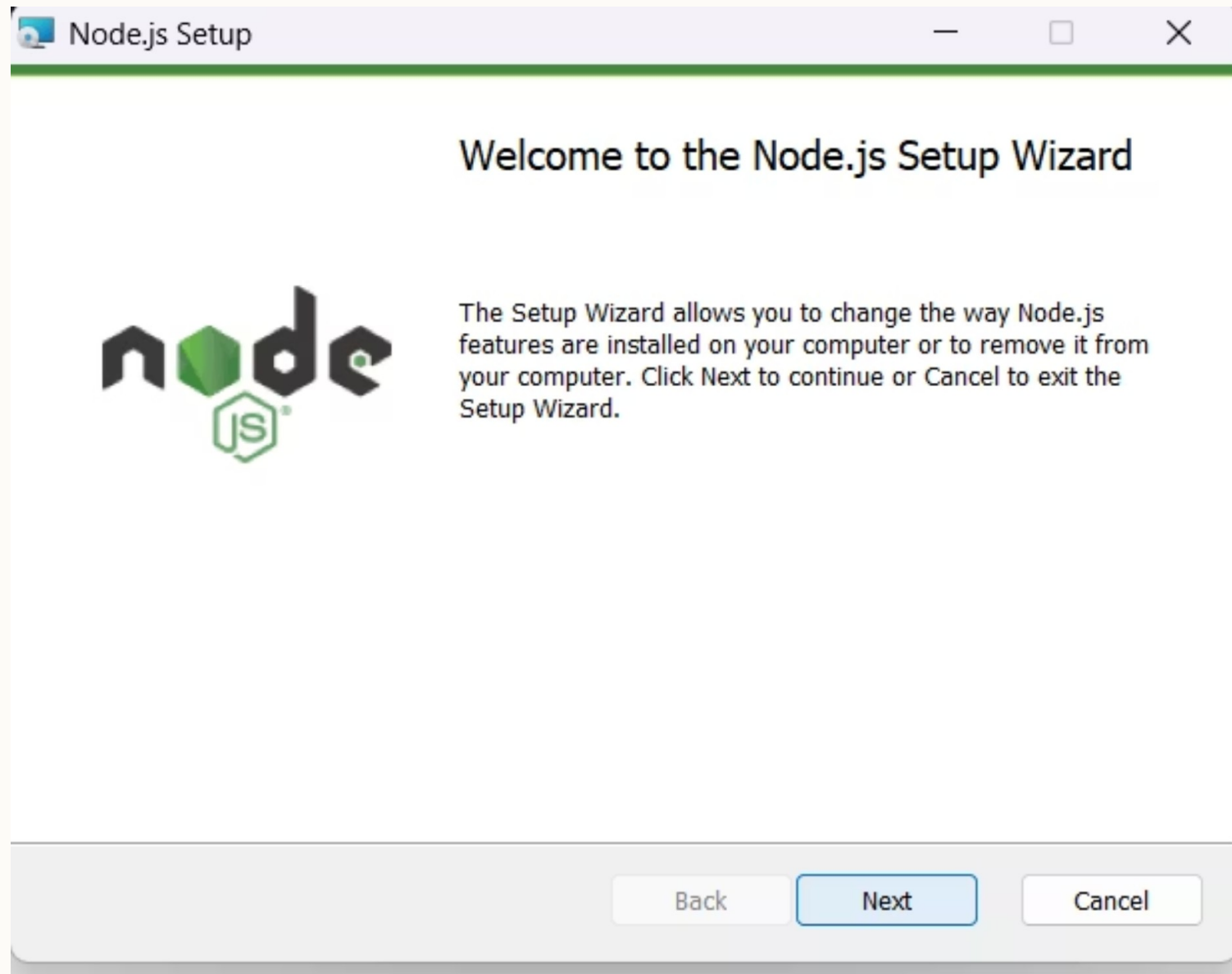
用のビルド済みのNode.js®も利用できます。

📄 Windows インストーラー (.msi)

📄 スタンドアロンのバイナリー (.zip)

7.node.js のインストール

ダウンロードが完了したら、ファイルを実行。何も変更せずインストールを進める



8.Claude Desktop のインストール

Claude の[ダウンロードページ](#)に移動し、ファイルをダウンロードしインストール
アカウントを持っていない場合、[アカウントを先に作成](#)しておきましょう



9. Claude Desktop の設定

Claude Desktop がインストールされたら起動をして、左上のハンバーガーメニューから「ファイル＞設定」を選択



9.Claude Desktop の設定

開発者のメニューを選び、「設定を編集」を選択



9.Claude Desktop の設定

エクスプローラーが表示されますので、「 claude_desktop_config.json 」というファイルをエディタで開く。

	Service Worker	2025/07/25 9:39	ファイル フォルダー	
	Session Storage	2025/07/25 22:29	ファイル フォルダー	
	Shared Dictionary	2025/07/25 9:39	ファイル フォルダー	
	WebStorage	2025/07/25 9:39	ファイル フォルダー	
	claude_desktop_config	2025/07/25 9:43	JSON ファイル	3 KB
	config	2025/07/25 22:33	JSON ファイル	1 KB
	DIPS	2025/07/25 22:29	ファイル	36 KB

9. Claude Desktop の設定

ファイルの中身を以下の通り編集

A) まずは以下内容に差し替え

```
{ "mcpServers":  
  { "google-analytics":  
    { "command": "npx",  
      "args": [ "-y", "mcp-server-google-analytics" ],  
      "env": { "GOOGLE_CLIENT_EMAIL": "your-service-account@project.iam.gserviceaccount.com",  
                "GOOGLE_PRIVATE_KEY": "-----BEGIN PRIVATE KEY-----\nYOUR_PRIVATE_KEY_HERE\n-----END  
PRIVATE KEY-----",  
                "GA_PROPERTY_ID": "123456789" }  
    }  
  }  
}
```

9.Claude Desktop の設定 変更箇所 1

B)**GOOGLE_CLIENT_EMAIL** の部分を **STEP4** でダウンロード **json** ファイルの中身と差し替え（STEP4 でドキュメントフォルダ配下においた json ファイルを開いて確認してください）

```
**BEFORE
```

```
"your-service-account@project.iam.gserviceaccount.com"
```

```
↓
```

```
**AFTER
```

```
"xxxxxxxxxxxx@xx.iam.gserviceaccount.com"
```

9.Claude Desktop の設定 変更箇所 2

C)GOOGLE_PRIVATE_KEY の部分を STEP4 で作成したJSON ファイル内にある private_key と差し替え

****BEFORE**

```
"-----BEGIN PRIVATE KEY-----\nYOUR_PRIVATE_KEY_HERE\n-----END PRIVATE KEY-----"
```

↓

****AFTER**

```
"-----BEGIN PRIVATE KEY-----\nMIIEvQIBADANBgkqhkiG9 (中略) FqyNgfwfoXobOf+hY5DNXthV\nEZNwc1+\n+1mDnMuwlmlJlullw=\n-----END PRIVATE KEY-----\n"
```

9.Claude Desktop の設定 変更箇所 3

D)GA_PROPERTY_ID の部分を STEP6 で取得したプロパティ ID と差し替え

ファイルを保存し、 Claude Desktop を閉じてください。

```
**BEFORE  
"123456789"  
↓  
**AFTER  
"XXXXXXXXX"
```

9. Claude Desktop の設定

最終的に以下のような形になっていれば OK

```
{ "mcpServers":  
  { "google-analytics":  
    { "command": "npx",  
      "args": [ "-y", "mcp-server-google-analytics" ],  
      "env": { "GOOGLE_CLIENT_EMAIL": "xxxxxxxxxxx@xx.iam.gserviceaccount.com",  
                "GOOGLE_PRIVATE_KEY": "—BEGIN PRIVATE KEY—\nMIIEvQIBADANBgkqhkiG9 (中  
略) FqyNgfwfoXobOf+hY5DNXthV\nEZNwc1++1mDnMuwlmlJlullw=\n—END PRIVATE KEY—\n",  
                "GA_PROPERTY_ID": "204983712"      }  
      }  
    }  
  }  
}
```

9.Claude Desktop の設定

タスクマネージャーでも確認し、残っている場合は、「タスクの終了」を選んでください (Ctrl+Shift+esc で起動。 Claude を右クリックで「タスクの終了」を選択)

タスク マネージャー

検索する名前、発行元、PID...

プロセス

新しいタスクを実行する

タスクを終了する

効率モード

名前	状態	6% CPU	56% メモリ	1% ディスク	0% ネットワ...
> Claude (14)		1.3%	447.8 MB	0 MB/秒	0 Mbps
> Google Chrome (26)		1.7%	2,494.6 ...	0.1 MB/秒	0 Mbps

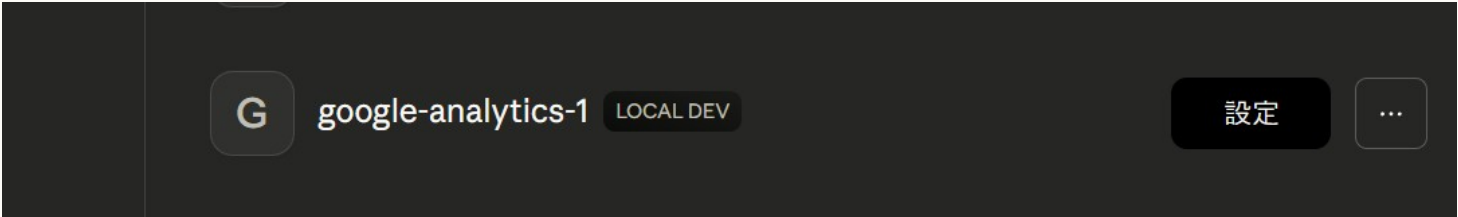
9. Claude Desktop の設定

再度 Claude Desktop を起動。「ファイル>コネクタ」を確認し、google-analytics がローカルで追加されているか確認



9.Claude Desktop の設定

「設定」ボタンを押して「RunReport」以外を OFF にしてください（数値のずれが発生しにくくなります）



一般的には、より柔軟で透明性の高いRunReportの使用をおすすめします。GetPageViewsは内部的にRunReportをラップしているため、何が集計されているか不明瞭な場合があります。具体的にどのような差異が出ているか教えていただければ、より詳しく原因を特定できます!



Claudeは間違えることがあります。回答内容を必ずご確認ください

G google-analytics-1 LOCAL DEV

ツール

RunReport

Run a report to get analytics data

監視なしで許可

GetPageViews

Get page view metrics for a specific date range

監視なしで許可

GetActiveUsers

Get active users metrics for a specific date range

監視なしで許可

GetEvents

Get event metrics for a specific date range

監視なしで許可

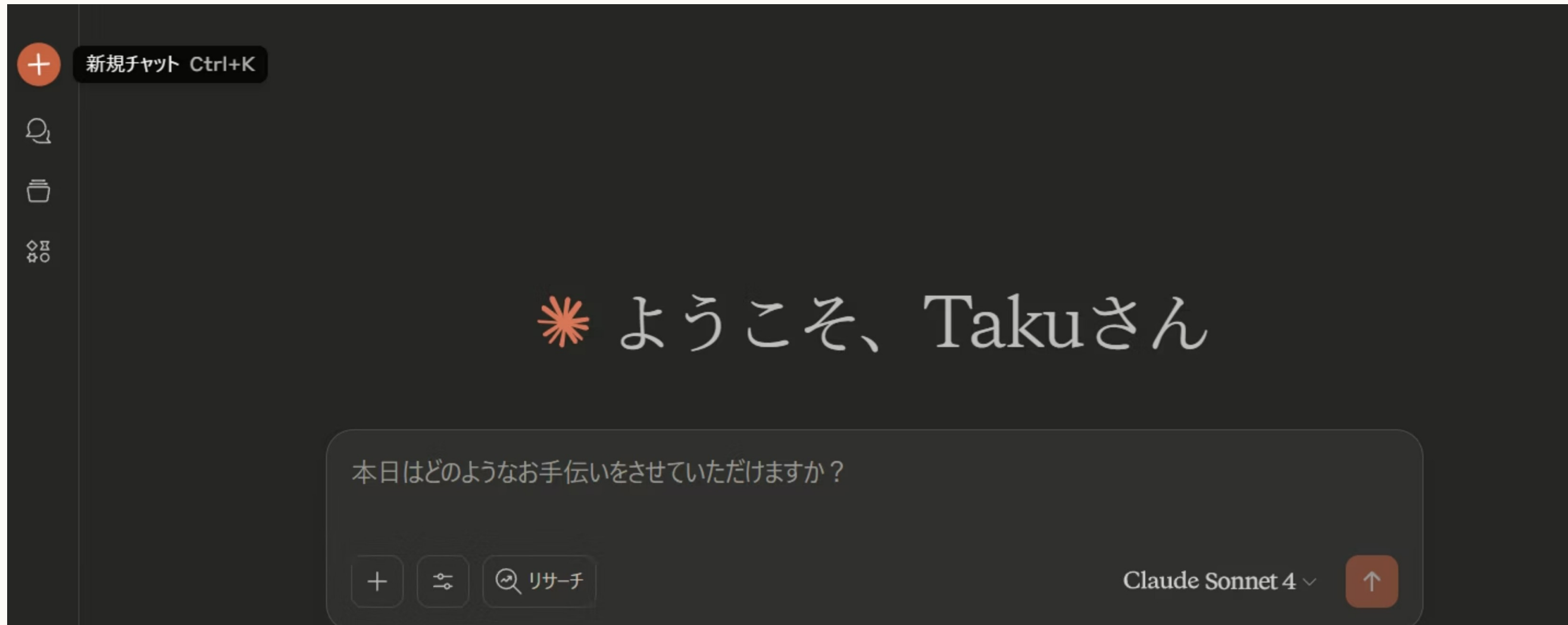
GetUserBehavior

Get user behavior metrics like session duration and bounce rate

監視なしで許可

9. Claude Desktop の設定

新規チャットを押して分析を開始することができます



❏ 入力する際に「接続した GA4 の」と最初につけてください

設定に関する FAQ

よくある質問

エラー 1-1: プロジェクトが作成できない

✖ エラー症状

「新しいプロジェクト」ボタンが押せない
プロジェクト作成後に一覧に表示されない
「権限がありません」とエラー表示

🔍 原因

Google アカウントに組織の管理権限がない
GCP の組織ポリシーで制限されている
既存のプロジェクト数が上限に達している（無料枠は 12 個まで）

✔ 対処方法

【即座に試せる対処】

1. ブラウザのシークレットモードで再ログイン
2. 別の Google アカウントで試す
3. 左上の組織選択で「組織なし」を選択

【管理者に依頼する場合】

- 「GCP Project Creator」ロールの付与を依頼
- または管理者にプロジェクト作成を代行依頼

よくある質問

エラー 1-2: Google Analytics Data API が見つからない

✖ エラー症状

検索ボックスで「Google Analytics Data API」と入力しても出てこない
API一覧に表示されない

🔍 原因

プロジェクトが正しく選択されていない（左上のプロジェクト名を確認）
組織のAPIポリシーでブロックされている
リージョン制限がかかっている

✔ 対処方法

【確認手順】

1. 左上の「Google Cloud ロゴの右」でプロジェクト名を確認
→ 自分が作成したプロジェクト名になっているか？
2. API ライブラリの URL 直接アクセス
<https://console.cloud.google.com/apis/library/analyticsdata.googleapis.com>
3. それでも出ない場合
→ 組織管理者に「Google Analytics Data API の利用許可」を依頼

よくある質問

エラー 1-3: API 有効化ボタンがグレーアウト

✖ エラー症状

「有効にする」ボタンが押せない
クリックしても反応がない

🔍 原因

API 管理の権限が不足している
請求先アカウント（Billing Account）が設定されていない

✔ 対処方法

【権限確認】

1. IAM と管理 → IAM を開く
2. 自分のアカウントに以下の権限があるか確認
 - 「編集者（Editor）」または
 - 「Service Usage Admin」

【Billing 確認】

1. 左メニュー「お支払い」を開く
2. 請求先アカウントがリンクされているか確認
3. 無い場合は無料トライアルを有効化（クレカ登録必要）

よくある質問

エラー 2-1: サービスアカウントが作成できない

✖ エラー症状

「+ サービスアカウントを作成」ボタンが無い / 押せない
「権限がありません」エラー

🔍 原因

サービスアカウント作成権限がない
プロジェクトの IAM 設定で制限されている

✔ 対処方法

【必要な権限】

「Service Account Admin」ロールが必要

【代替手段】

1. 管理者にサービスアカウント作成を依頼
2. または「編集者（Editor）」ロールを付与してもらう

【確認コマンド（Cloud Shell）】

```
gcloud projects get-iam-policy [PROJECT_ID] --flatten="bindings[].members" --  
filter="bindings.role:roles/iam.serviceAccountAdmin"
```

よくある質問

エラー 2-2: JSON ファイルがダウンロードされない

✖ エラー症状

「作成」ボタンを押してもファイルが保存されない
ダウンロードフォルダに何も入っていない

🔍 原因

ブラウザのダウンロードブロック設定
ポップアップブロッカーが作動
ファイル名に特殊文字があり保存失敗

✔ 対処方法

【ブラウザ設定確認】

Chrome の場合：

1. 設定 → プライバシーとセキュリティ → サイトの設定
2. 「追加のコンテンツ設定」 → 「自動ダウンロード」
3. cloud.google.com を許可リストに追加

【代替手段】

1. キーの一覧画面で「：」 → 「キーをダウンロード」から再取得
2. 別ブラウザ（Edge、Firefox）で試す
3. シークレットモードで試す

よくある質問

エラー 2-3: JSON ファイルをどこに保存したか分からない

✖ エラー症状

ダウンロードしたはずだが見つからない
ファイル名が分からない

🔍 原因

ダウンロードフォルダから移動させた
複数回ダウンロードして上書きされた

✔ 対処方法

【Windows の場合】

1. エクスプローラーで検索

検索ワード: *.json 場所: ダウンロード 更新日時: 今日

2. ファイル名のパターン

[プロジェクト名]-[ランダム文字列].json

例: claude-ga4-abc123def456.json

【見つからない場合】

再度ダウンロードして、すぐに「ドキュメント」フォルダに移動
→ 「GA4-MCP-Credentials」などの専用フォルダを作成推奨

よくある質問

エラー 3-1: GA4 で「プロパティのアクセス管理」が開けない

✖ エラー症状

「管理」→「プロパティのアクセス管理」メニューが無い
クリックしても開けない

🔍 原因

GA4 プロパティの管理者権限がない
閲覧者やアナリスト権限しかない

✔ 対処方法

【権限確認】

1. GA4 の「管理」→「プロパティ設定」を開く
2. 開けない場合 → 管理者権限なし

【必要な権限】

「編集者」または「管理者」権限が必要

【対処】

プロパティの管理者に以下を依頼：

「サービスアカウントのメールアドレスに『アナリスト』権限を付与してください」

よくある質問

エラー 3-2: メールアドレスを追加しても「無効なメール」エラー

✖ エラー症状

サービスアカウントのメールを貼り付けても「無効」と表示
追加ボタンが押せない

🔍 原因

余分なスペースや改行が混入
全角文字が含まれている
メールアドレスが不完全（コピー範囲ミス）

✔ 対処方法

【正しいコピー手順】

1. JSON ファイルをメモ帳で開く
2. "client_email" の行を探す
3. ダブルクォート内のみをコピー
 - × "client_email": "xxxxxx@xxx.iam.gserviceaccount.com",
 - xxxxx@xxx.iam.gserviceaccount.com

【検証方法】

コピーしたメールアドレスを一度メモ帳に貼って確認

- 前後にスペース無し
- 改行無し
- すべて半角英数字

よくある質問

エラー 3-3: 権限を付与したのに「アクセスできません」エラー

✖ エラー症状

GA4 で権限付与は完了したが、Claude で「アクセスできません」と出る
データ取得時にエラー

🔍 原因

権限が「閲覧者」になっている（アナリスト以上が必要）
権限反映に時間がかかっている（最大 5 分）
プロパティ ID が間違っている

✔ 対処方法

【権限レベル確認】

GA4 の「プロパティのアクセス管理」で

- 「アナリスト」✔
- または「編集者」✔
- 「閲覧者」✖（データ取得不可）

【待機時間】

権限付与後、5 分待ってから再試行

【プロパティ ID 再確認】

管理 → プロパティの詳細 → 9 桁の数値をコピー

よくある質問

エラー 4-1: 測定 ID (G-XXXXXX) を使ってしまう

✖ エラー症状

設定後、「プロパティが見つかりません」エラー
データ取得できない

🔍 原因

最も多い間違い: 測定 ID とプロパティ ID の混同
測定 ID (G-XXXXXX) は使用不可

✔ 対処方法

【正しいプロパティ ID の形式】

✖ G-ABC123XYZ (測定 ID)

✖ UA-12345678-1 (旧 GA のトラッキング ID)

✔ 123456789 (9桁の数値のみ)

【正しい取得場所】

1. GA4 の「管理」をクリック
2. 「プロパティ」列の「プロパティの詳細」をクリック
3. 画面右上の「プロパティ ID」をコピー
(「測定 ID」ではない)

よくある質問

エラー 5-1: Node.js インストーラーが実行できない

✖ エラー症状

msi ファイルをダブルクリックしても何も起こらない
「管理者権限が必要です」エラー

🔍 原因

Windows の標準ユーザー権限でインストール不可
会社 PC でインストール制限がかかっている

✔ 対処方法

【管理者権限で実行】

1. msi ファイルを右クリック
2. 「管理者として実行」を選択
3. UAC ダイアログで「はい」

【社内 PC の場合】

1. IT 部門にインストール申請
2. または「ポータブル版 Node.js」を検討
(管理者権限不要だが設定が複雑)

【インストール確認】

コマンドプロンプトで

> node --version

v20.x.x のように表示されれば OK

よくある質問

エラー 5-2: Claude Desktop が起動しない

✖ エラー症状

インストール後、アイコンをクリックしても起動しない
「アプリケーションエラー」と表示

🔍 原因

セキュリティソフトがブロック（よくある）
Windows Defender の制限
古いバージョンの Windows で非対応

✔ 対処方法

【Windows Defender 除外設定】

1. 設定 → 更新とセキュリティ → Windows セキュリティ
2. ウイルスと脅威の防止 → 設定の管理
3. 除外 → 除外の追加 → フォルダー
4. C:\Users\[ユーザー名]\AppData\Local\Programs\claude を追加

【ウイルス対策ソフト】

ノートンやウイルスバスターの場合：

- アプリケーション制御から Claude.exe を許可

【動作確認】

タスクマネージャー（Ctrl+Shift+Esc）で
"Claude" プロセスが動いているか確認

よくある質問

エラー 6-2: JSON 編集後に Claude Desktop が起動しない

✖ エラー症状

設定ファイル保存後、Claude Desktop が起動しなくなった
起動しても即座に落ちる

🔍 原因

最頻出：JSON 構文エラー（カンマ、括弧、クォートのミス）
全角文字の混入
改行コードの問題

✔ 対処方法

- 【JSON 構文チェック】
1. <https://jsonlint.com/> を開く
 2. `claude_desktop_config.json` の内容を貼り付け
 3. 「Validate JSON」をクリック
 4. エラー箇所が表示される

よくある質問

エラー 6-2: JSON 編集後に Claude Desktop が起動しない

(続き)

【よくある間違い】

✗ {"mcpServers": {...}},} ← 最後のカンマ

✓ {"mcpServers": {...}}

✗ 'GOOGLE_CLIENT_EMAIL' ← シングルクォート

✓ "GOOGLE_CLIENT_EMAIL" ← ダブルクォート

✗ "env": {
GOOGLE_CLIENT_EMAIL: "..." ← クォート無し

✓ "env": {
"GOOGLE_CLIENT_EMAIL": "..."

【復旧方法】

1. バックアップが無い場合は資料 P39 の内容を再度コピー
2. 一つずつ値を置き換える
3. 置き換える度に JSONLint で検証

よくある質問

エラー 6-3: GOOGLE_PRIVATE_KEY の改行エラー

✖ エラー症状

設定後、「認証エラー」「Invalid private key」と表示
接続できない

🔍 原因

Private Key の \n（改行コード）が正しく扱われていない
ダブルエスケープ（\\n）になっている
改行が実際の改行になってしまっている

✔ 対処方法

よくある質問

エラー 6-3: GOOGLE_PRIVATE_KEY の改行エラー

【正しい PRIVATE_KEY の形式】

JSON ファイルから：

```
"private_key": "-----BEGIN PRIVATE KEY-----\nMIIEvQIBA...( 中略 )...=\n-----END PRIVATE KEY-----\n"
```

claude_desktop_config.json へ：

```
"GOOGLE_PRIVATE_KEY": "-----BEGIN PRIVATE KEY-----\\nMIIEvQIBA...\\n-----END PRIVATE KEY-----\\n"
```

【確認方法】

メモ帳で開いて、\n が以下のようにになっているか確認：

- 「¥n」と表示（日本語フォント）
- 実際に改行されている

【修正手順】

1. JSON ファイルから private_key をコピー
2. テキストエディタで \n を \\n に全置換
(VSCode なら Ctrl+H で置換)
3. claude_desktop_config.json に貼り付け

よくある質問

エラー 6-4: 「google-analytics」がコネクタに表示されない

✖ エラー症状

設定後、「ファイル」→「コネクタ」を開いても空っぽ
google-analytics が表示されない

🔍 原因

JSON ファイルの構文エラー（エラー 6-2 参照）

Node.js がインストールされていない

npx コマンドが実行できない環境

【設定ファイル確認】

```
{  
  "mcpServers": {  
    "google-analytics": { ← この名前が表示される  
      "command": "npx",  
      "args": ["-y", "mcp-server-google-analytics"],  
      ...  
    }  
  }  
}
```

【強制再読み込み】

1. Claude Desktop を完全終了（タスクマネージャーで確認）
2. 設定ファイルを再度開いて保存（Ctrl+S）
3. 再起動

よくある質問

エラー 7-1: 設定変更が反映されない

✖ エラー症状

claude_desktop_config.json を修正して保存したのに変化なし
古い設定のまま動作している

🔍 原因

超頻出：Claude Desktop のプロセスが完全に終了していない
バックグラウンドで動き続けている

✔ 対処方法

【完全終了手順（Windows）】

1. Ctrl + Shift + Esc でタスクマネージャー起動
2. 「プロセス」タブで "Claude" を探す
3. 右クリック → 「タスクの終了」
4. 複数ある場合はすべて終了

【確認】

タスクマネージャーで "Claude" が無いことを確認してから
→ Claude Desktop を再起動

【自動起動を停止】

タスクマネージャー → スタートアップタブ
→ Claude を「無効」に設定（任意）

よくある質問

エラー 8-1: 「接続した GA4 の～」と入力しても反応しない

✖ エラー症状

プロンプトを入力してもデータが取得されない
「申し訳ございません、データにアクセスできません」と返答

🔍 原因

コネクタが正しく設定されていない
GA4 の権限不足

プロパティ ID の間違い

✔ 対処方法

【診断フローチャート】

STEP1: コネクタ確認

「ファイル」→「コネクタ」に "google-analytics" が表示されているか？

→ NO: エラー 6-4 を参照

STEP2: 接続テスト

以下の簡単なプロンプトを試す：

「接続した GA4 のプロパティ名を教えて」

→ エラーが返る：STEP3 へ

→ 正常に返答：設定完了！

STEP3: エラーメッセージ確認

- "Property not found" → プロパティ ID ミス（エラー 4-1）

- "Permission denied" → GA4 権限不足（エラー 3-3）

- "Authentication failed" → Private Key エラー（エラー 6-3）

よくある質問

エラー 8-2: 「データが見つかりません」エラー

✖ エラー症状

接続は成功するが「データがありません」と返る
空の結果が返ってくる

🔍 原因

プロパティ ID は合っているがデータが無い期間を指定
GA4 の計測自体が動いていない
サービスアカウントの権限が「閲覧者」になっている

よくある質問

エラー 8-2: 「データが見つかりません」エラー

✓ 対処方法

【GA4 でデータ確認】

1. GA4 の「レポート」→「リアルタイム」を開く
2. 現在アクティブユーザーがいるか確認
3. いない場合 → 自分でサイトにアクセスして確認

【期間指定を変更】

× 「今日のデータを見せて」（データ反映に遅延あり）

✓ 「昨日のデータを見せて」

✓ 「先月のデータを見せて」

【権限再確認】

GA4 の「プロパティのアクセス管理」で

サービスアカウントの権限が「アナリスト」以上か確認

サーチコンソールと Claude の連携方法

設定のための前提

利用するには以下が必要となります

- GA4 との MCP サーバー連携がすでに終わっている状態（GCP のアカウントがあり、Node.js と Claude Desktop がインストールされていること）
- **サーチコンソールの設定が終わっており、データが取得できていること**

注意点

Windows を前提に説明しますが、Mac でも可能です

作業時間

1 時間程度見ておきましょう

作業フロー

全部で 12 の STEP になります

1. Google クラウドプロジェクトの作成
2. API の有効化
3. 認証情報の作成
4. プロジェクトフォルダの作成
5. 必要なライブラリのインストール
6. プログラムファイルの作成
7. プログラムファイルの実行
8. プロジェクトフォルダの作成
9. MCP サーバーのコード作成
10. 認証用スクリプトの作成
11. 認証を実行
12. Claude Desktop の設定

1. Google Cloud プロジェクトの作成

新規にプロジェクトを作成します

Google Cloud

スラッシュ (/) を1

新しいプロジェクト

割り当て内の残りのプロジェクト数は 2 projects 件です。プロジェクトの増加をリクエストするか、プロジェクトを削除してください。[詳細](#)

[Manage Quotas](#)

プロジェクト名 *

GoogleSearchConsole

?

プロジェクト ID: master-ember-474313-e5 後で変更することはできません。
[編集](#)

場所 *

組織なし

[参照](#)

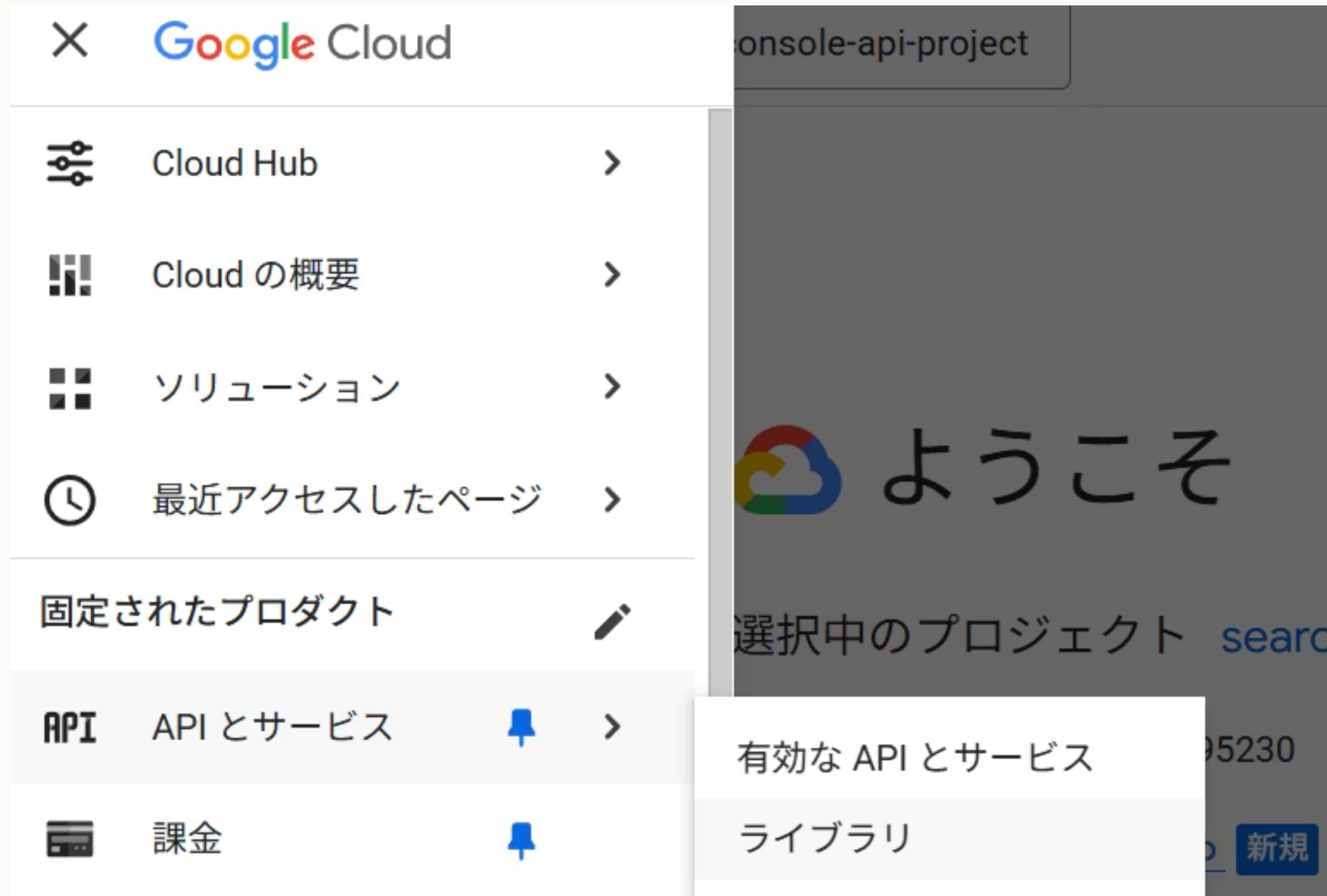
親組織またはフォルダ

作成

キャンセル

2.API の有効化

API とサービス>ライブラリを選択します



2.APIの有効化

Google Search Console で検索して API をインストールしてください

API API とサービス / API ライブラリ / 参照

API API ライブラリ

Google search console

API ライブラリ > “Google search console”

≡ フィルタ

フィルタ テキスト
を入力

1 件の結果

公開設定

公開

カテゴリ

その他

↑

(1)

↑

(1)



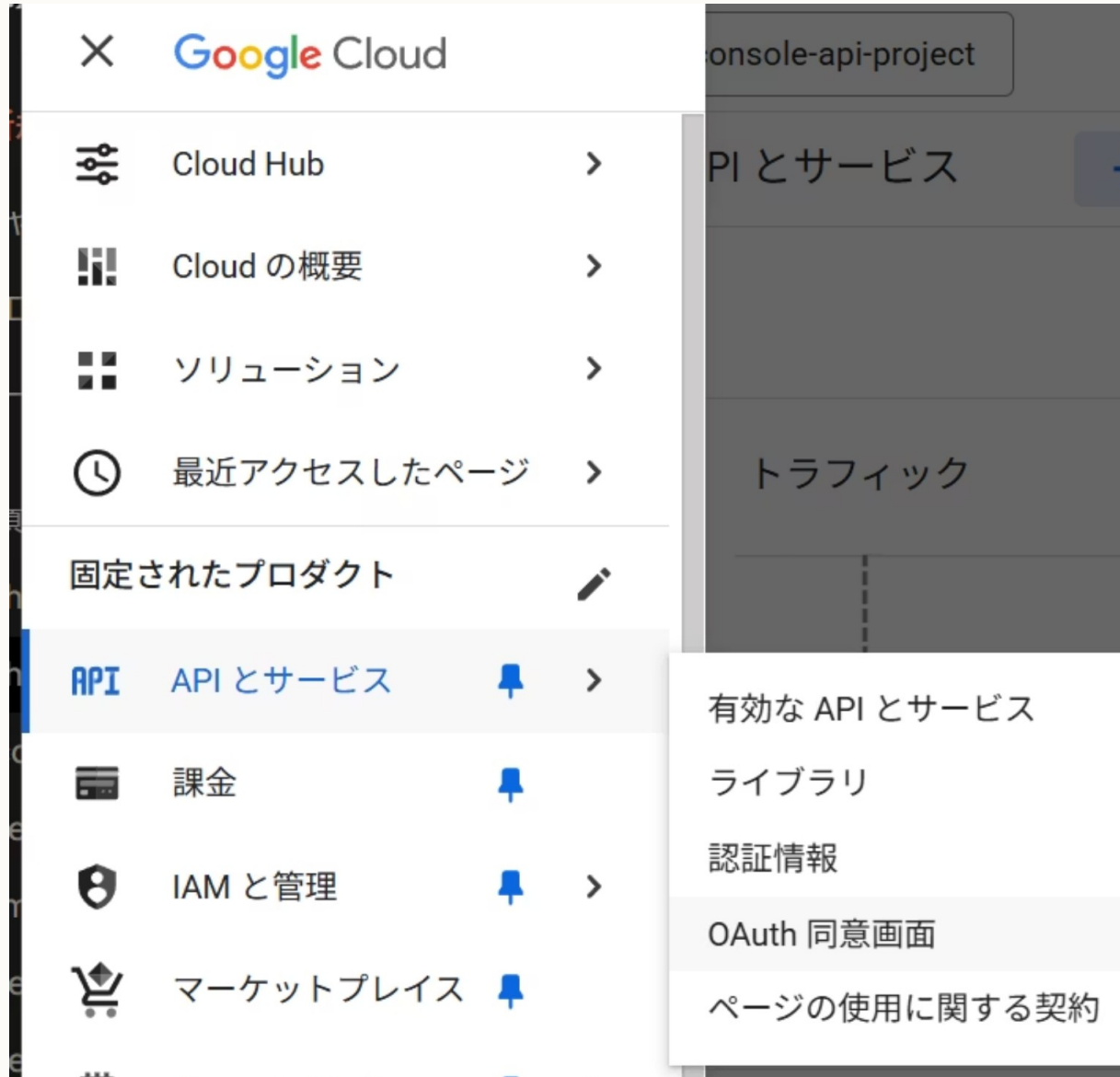
Google Search Console API

Google

The Search Console API provides access to both Search Console data (verified users only) and to public information on an URL basis (anyone)

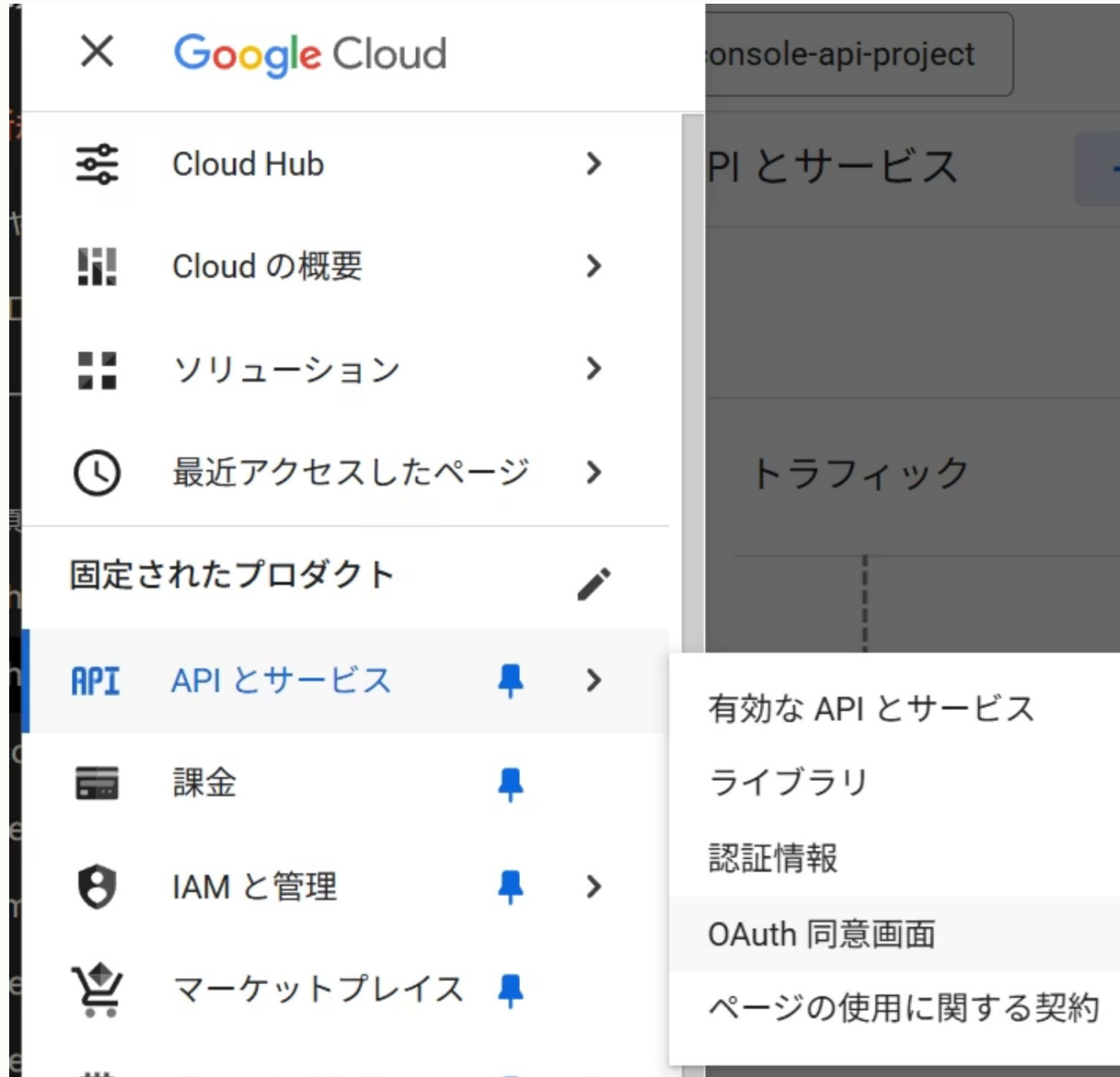
3. 認証情報の作成

API とサービスから OAuth 同意画面を選んでください



3. 認証情報の作成

API とサービスから OAuth 同意画面を選んでください



3. 認証情報の作成

開始を押して下さい

OAuth の概要



Google Auth Platform はまだ構成されていません
アプリケーションの ID を構成すること、および Google API の呼び出しと
Google でログインのための認証情報を管理することを開始します。 [詳細](#)



開始

3. 認証情報の作成

以下の通り選択して作成を行ってください。

アプリ名: 任意 ユーザーサポート: SearchConsole の権限が付与されているメールアドレス

1 アプリ情報

アプリ名 *
SearchConsole

同意を求めるアプリの名前

ユーザー サポートメール *
otofuukei@gmail.com ▼

同意に関して問い合わせる際に使用します。 [詳細](#)

次へ

3. 認証情報の作成

以下の通り選択して作成を行ってください。

「外部」を選択。 Step1 と同じメールアドレスを入力し「作成」

2 対象

☐ 内部 ⓘ

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 ⓘ

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

3 連絡先情報

メールアドレス *
otofuukei@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。

次へ

4 終了

作成

キャンセル

3. 認証情報の作成

API とサービスから認証情報を選択



3. 認証情報の作成

メニューの「+認証情報を作成」から「OAuth クライアント ID」を選択

API API とサービス

有効な API とサービス

ライブラリ

認証情報

OAuth 同意画面

ページの使用に関する契約

認証情報

+ 認証情報を作成

削除

削除した認証情報を復元

有効な API にアクセス

API キー

☐ ☒

名前

表示する API キー

OAuth 2.0 クラ

☐

名前

表示する OAuth

API キー

シンプル API キーを使用してプロジェクトを識別し、割り当てとアクセスを確認します

OAuth クライアント ID

ユーザーのデータにアクセスできるようにユーザーの同意をリクエストします

サービス アカウント

ロボット アカウントによるサーバー間でのアプリレベルの認証を有効にします

ウィザードで選択

使用する認証情報の種類を決定するため、いくつかの質問をします

3. 認証情報の作成

アプリケーションの種類から「デスクトップ アプリ」を選び、任意の名称をつけたら「作成」を押す

☰ 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	← OAuth クライアント ID の作成 クライアント ID は、Google の OAuth サーバーで個々のアプリを識別するために使用します。アプリが複数のプラットフォームで実行される場合、それぞれに独自のクライアント ID が必要になります。詳しくは、OAuth 2.0 の設定をご覧ください。OAuth クライアントの種類の詳細 アプリケーションの種類 * デスクトップ アプリ 名前 * SearchConsole OAuth 2.0 クライアントの名前。この名前はコンソールでクライアントを識別するためにのみ使用され、エンドユーザーには表示されません。 注: 設定が有効になるまで 5 分から数時間かかることがあります 作成 キャンセル
--	--

3. 認証情報の作成

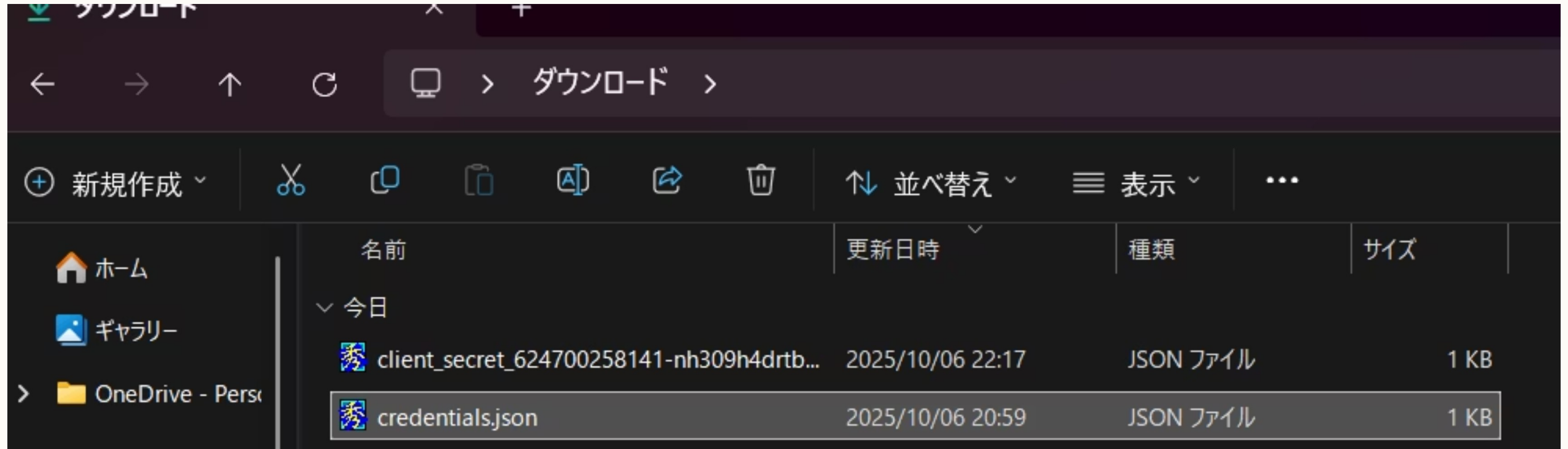
ポップアップが表示されるので「JSON をダウンロード」を選択。

★重要★ OK を押すと、ダウンロードできなくなり再作成となります。



3. 認証情報の作成

ダウンロードしたファイルを credentials.json という名前に変更

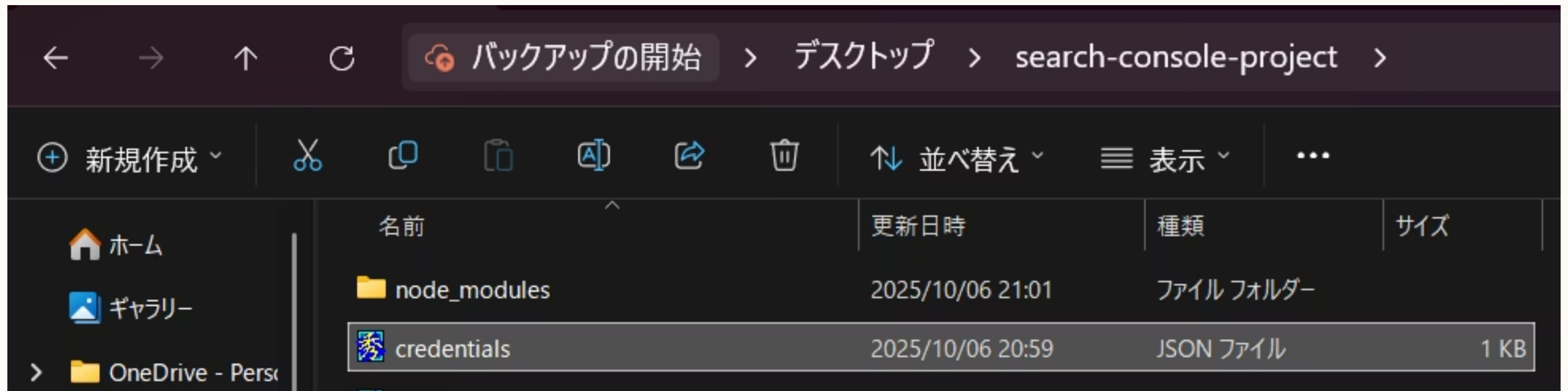


4. プロジェクトフォルダの作成

デスクトップに「search-console-project」という名前のフォルダを作成

先ほどダウンロードした credentials.json をこのフォルダに入れる

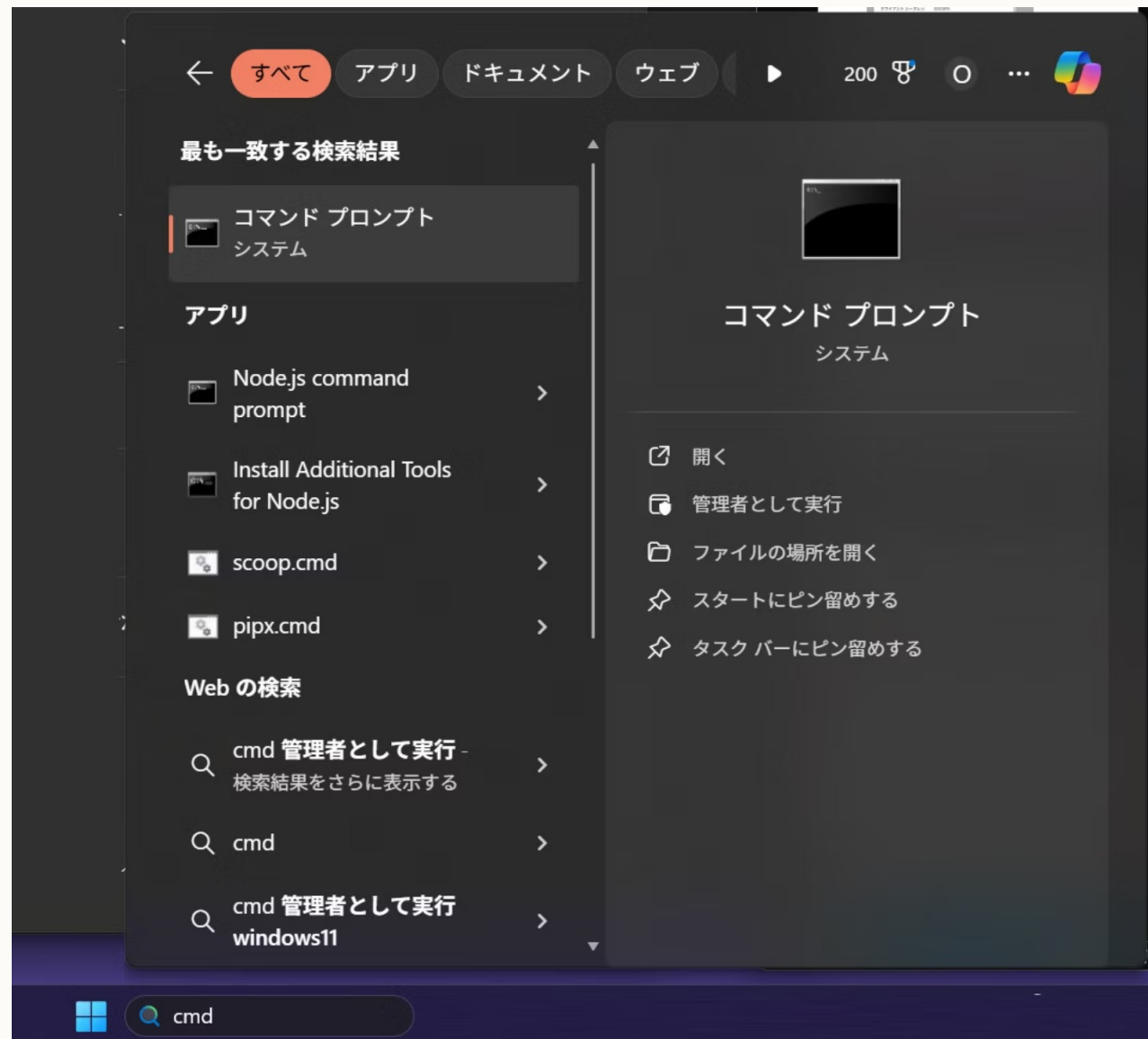
※ デスクトップではなくてもよいですが、その場合はこの後出てくるコマンドプロンプトでのディレクトリ指定を変えてください



4. プロジェクトフォルダの作成

コマンドプロンプト（Win）あるいはターミナル（Mac）を起動

Windows の場合は Windows の検索ボックスで「cmd」と入れて検索



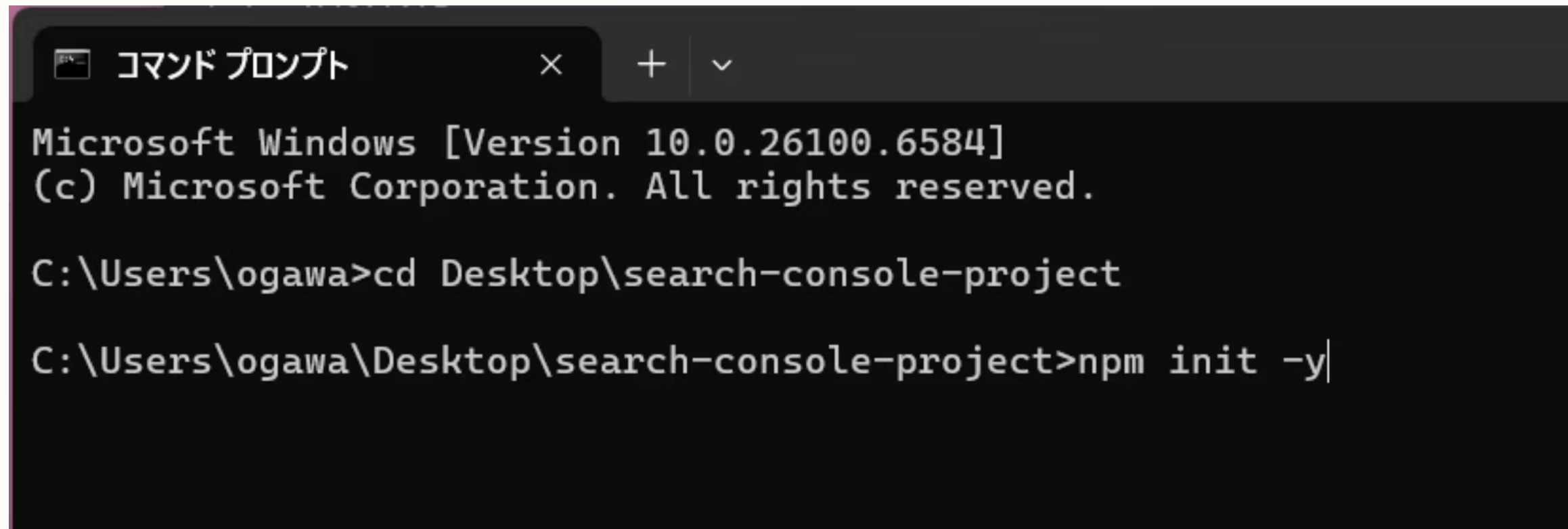
4. プロジェクトフォルダの作成

```
cd Desktop\search-console-project
```

と入力して Enter を押し、ディレクトリを移動する

```
npm init -y
```

と入力して Enter を押すと package.json というファイルが自動生成されます。



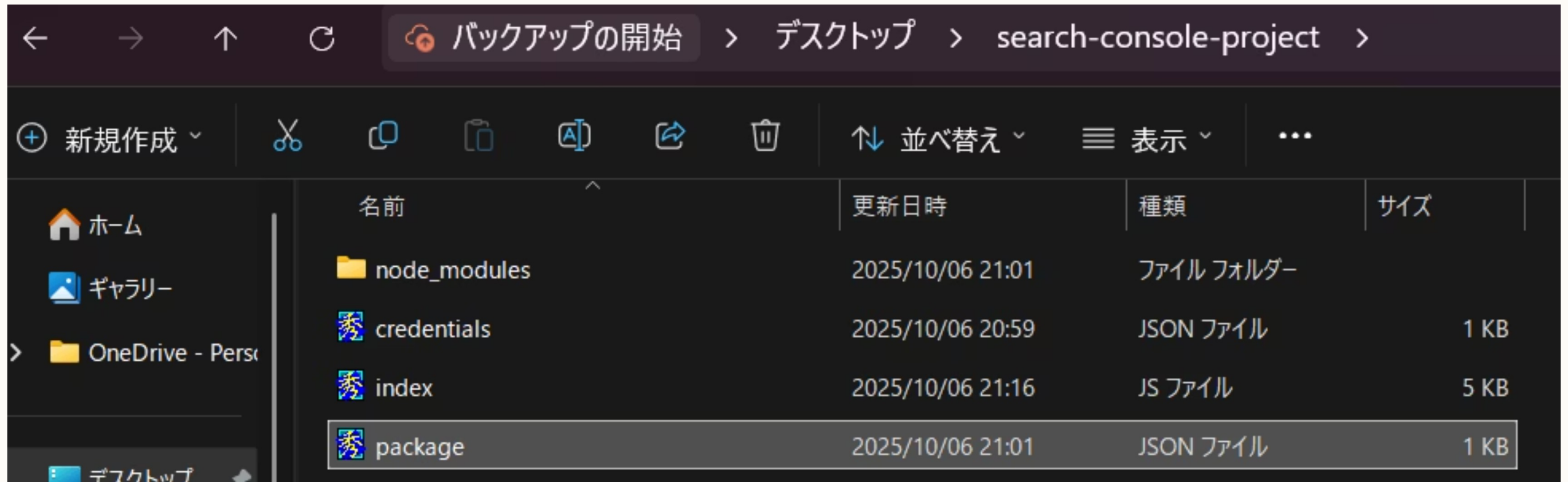
```
コマンド プロンプト
Microsoft Windows [Version 10.0.26100.6584]
(c) Microsoft Corporation. All rights reserved.

C:\Users\ogawa>cd Desktop\search-console-project

C:\Users\ogawa\Desktop\search-console-project>npm init -y|
```

4. プロジェクトフォルダの作成

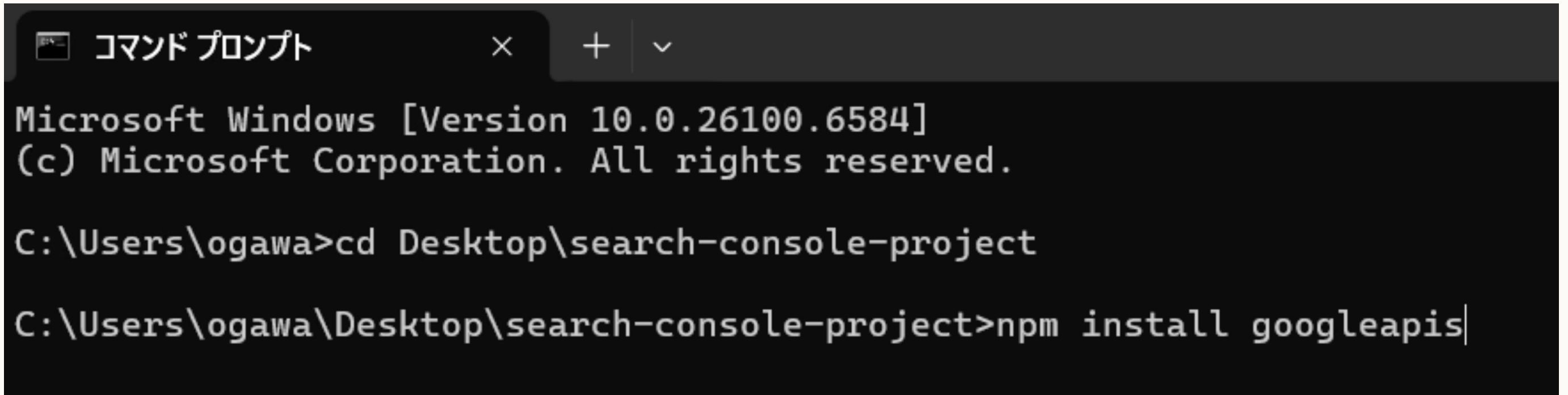
先ほどのフォルダに package.json が入っていれば OK です



5. 必要なライブラリのインストール

`npm install googleapis`

と入力し、Google API を使えるようにします

A screenshot of a Windows Command Prompt window. The title bar shows 'コマンド プロンプト' (Command Prompt) with standard window controls. The text inside the window shows the following sequence of commands and system messages:
1. 'Microsoft Windows [Version 10.0.26100.6584]'
2. '(c) Microsoft Corporation. All rights reserved.'
3. 'C:\Users\ogawa>cd Desktop\search-console-project'
4. 'C:\Users\ogawa\Desktop\search-console-project>npm install googleapis|'
The cursor is at the end of the last command line.

```
Microsoft Windows [Version 10.0.26100.6584]
(c) Microsoft Corporation. All rights reserved.

C:\Users\ogawa>cd Desktop\search-console-project

C:\Users\ogawa\Desktop\search-console-project>npm install googleapis|
```

6. プログラムファイルの作成

メモ帳などで index.js というファイルを作成して、先ほどのフォルダに追加します。

中身は以下の通りです。あるいは[こちら](#)に置いておきます。

```
const { google } = require('googleapis');
const fs = require('fs');
const path = require('path');

const CREDENTIALS_PATH = path.join(__dirname, 'credentials.json');
const INDEX_PATH = path.join(__dirname, 'index.html');
const SCOPES = ['https://www.googleapis.com/auth/webmasters.readonly'];

async function authorize() {
  if (fs.existsSync(CREDENTIALS_PATH)) {
    const { client_secret, client_id, redirect_uri } =
      JSON.parse(fs.readFileSync(CREDENTIALS_PATH, 'utf8'));
    const oAuth2Client = new google.auth.OAuth2(
      client_id, client_secret, redirect_uri);
    oAuth2Client.setCredentials({});
    return oAuth2Client;
  }
}

const { client_secret, client_id, redirect_uri } =
  JSON.parse(fs.readFileSync(CREDENTIALS_PATH, 'utf8'));
const oAuth2Client = new google.auth.OAuth2(
  client_id, client_secret, redirect_uri);

const authUrl = oAuth2Client.generateAuthUrl({
  access_type: 'offline',
  scope: SCOPES,
});

console.log('Open this URL in your browser to authenticate');
console.log(authUrl);
console.log('Enter the code from the URL:');

const readline = require('readline').createInterface({
  input: process.stdin,
  output: process.stdout,
});

readline.question('Enter the code: ')
  .then((code) => {
    const { tokens } = await oAuth2Client.getAccessToken(code);
    oAuth2Client.setCredentials(tokens);
    console.log('Authentication successful');
    resolve(oAuth2Client);
  })
  .catch(() => {});

async function getSearchConsoleData(auth) {
  const searchConsole = google.searchconsole({ version: 'v2', auth });

  const sites = await searchConsole.sites.list();
  console.log('Your sites:');
  sites.data.forEach(site => {
    console.log(`${site.id} ${site.title}`);
  });

  const readline = require('readline').createInterface({
    input: process.stdin,
    output: process.stdout,
  });

  return new Promise(resolve) => {
    readline.question('Select the number (1-3) of the site you want to search: ')
      .then((choice) => {
        let siteIndex = parseInt(choice) - 1;
        if (siteIndex < 0 || siteIndex >= sites.data.length) {
          console.log('Invalid selection. Using the first site');
          siteIndex = 0;
        }
        const siteId = sites.data[siteIndex].id;
        console.log('Fetching data for: ' + siteId);

        const endDate = new Date();
        const startDate = new Date(
          endDate.getTime() - 7 * 24 * 60 * 60 * 1000);

        const response = await searchConsole.searchanalytics.query({
          siteId: siteId,
          startDate: startDate.toISOString().split('T')[0],
          endDate: endDate.toISOString().split('T')[0],
          dimensions: ['query'],
          metrics: ['*'],
        });

        console.log('Top 10 Search Queries (Last 7 Days)');
        console.log('-----');

        let csvContent = 'Rank,Query,Clicks,Impressions,CTR,Average Position';

        response.data.forEach((row, index) => {
          const query = row.query;
          const clicks = row.clicks;
          const impressions = row.impressions;
          const ctr = (clicks / impressions * 100).toFixed(2);
          const position = row.position.toLowerCase();

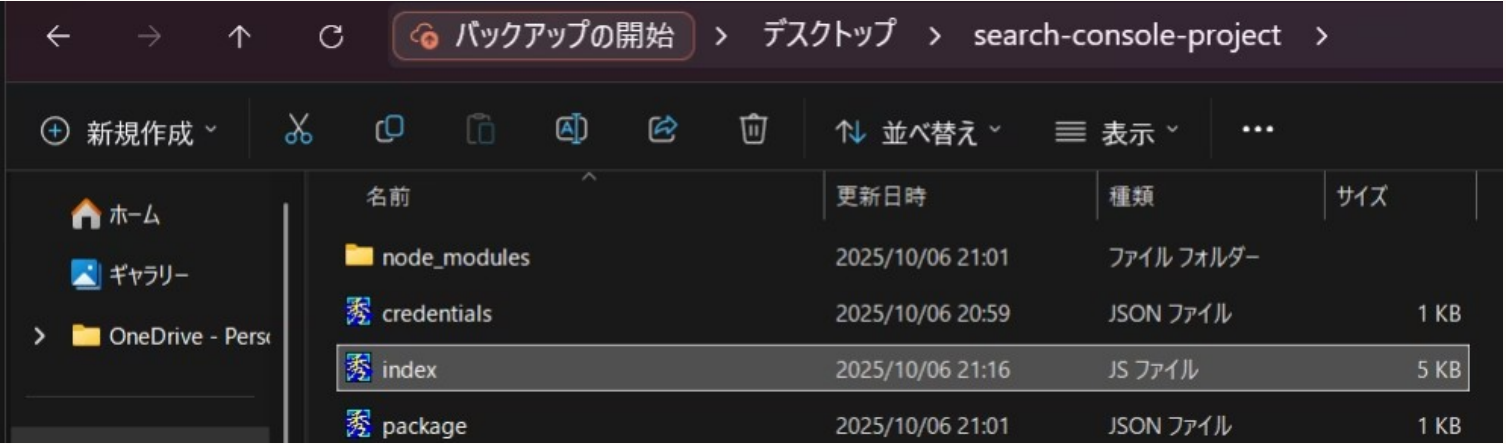
          console.log(`${index + 1} ${query}`);
          console.log(`Clicks: ${clicks}, Impressions: ${impressions}, CTR: ${ctr}, Position: ${position}`);
          console.log('-----');
        });

        csvContent += '\n' + sites.data[siteIndex].id + ',' + query + ',' + clicks + ',' + impressions + ',' + ctr + ',' + position + '\n';

        const fileName = `search_console_data_${siteId}_${index + 1}.csv`;
        fs.writeFileSync(fileName, csvContent);
        console.log('Data saved to: ' + fileName);
        console.log('You can open this CSV file in Excel to view Japanese text properly');
        resolve();
      })
      .catch(() => {});
  };

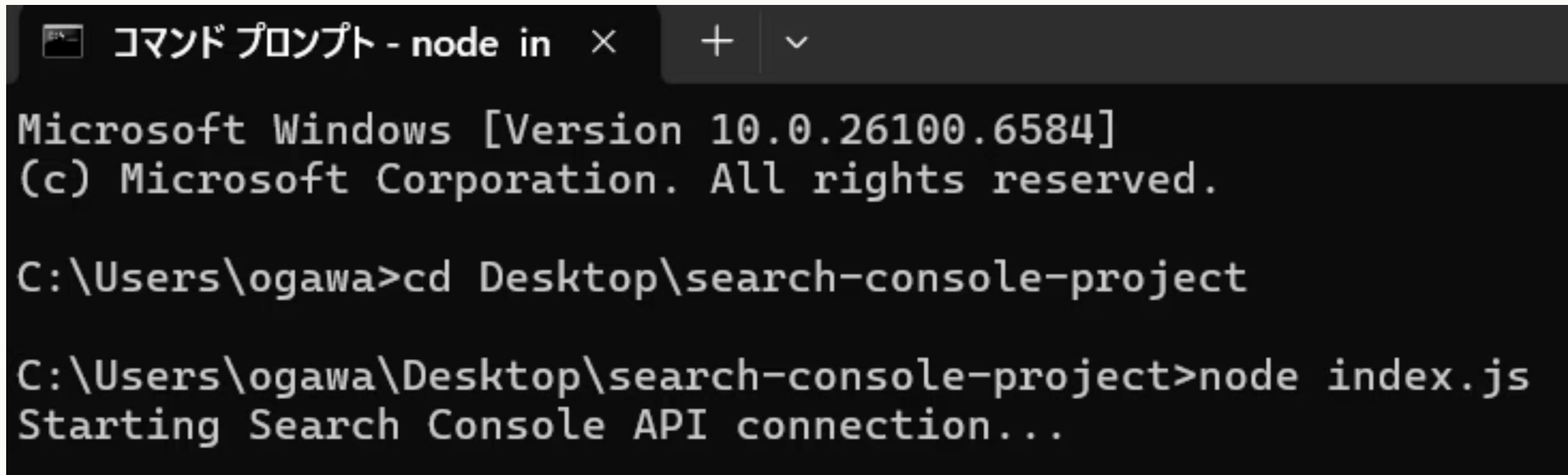
  async function main() {
    try {
      console.log('Starting Search Console API connection...');
      const auth = await authorize();
      await getSearchConsoleData(auth);
      console.log('Completed');
    } catch (error) {
      console.error('Error occurred:', error.message);
    }
  }

  main();
}
```



7. プログラムの実行

コマンドプロンプトで `node index.js` を入力して Enter を押します。



```
コマンド プロンプト - node in × + ∨  
Microsoft Windows [Version 10.0.26100.6584]  
(c) Microsoft Corporation. All rights reserved.  
  
C:\Users\ogawa>cd Desktop\search-console-project  
  
C:\Users\ogawa\Desktop\search-console-project>node index.js  
Starting Search Console API connection...
```

7. プログラムの実行

URL が表示されるので、そちらをコピー & ペーストしてブラウザで開きます。Google アカウントの認証が終わると、localhost を含む URL にアクセスされ、ブラウザでページが表示されないエラーが出ます。

以下のような URL になります。

http://localhost/?code=4/0AVGzR1DbRzoNlmktoYORp8iy9hk_bglG950RVZeU6k-WGKQqgwMtdJ1YdW1fuY4KQZPgoQ&scope=https://www.googleapis.com/auth/webmasters.readonly

この中の code= の後ろの部分をコピーします。

```
4/0AVGzR1DbRzoNlmktoYORp8iy9hk_bglG950RVZeU6k-WGKQqgwMtdJ1YdW1fuY4KQZPgoQ
```

こちらをコマンドプロンプトに戻ると表示される「コード：」の後に張り付けて Enter キーを押してください。

認証情報を保存しました。と表示されデータが表示されれば OK です

ここから後半

データの取得が確認できるようになりました。これで接続は上手くいっています。

しかし、これは直近7日間の検索クエリが表示されるだけで、AIに直接質問をすることができません。

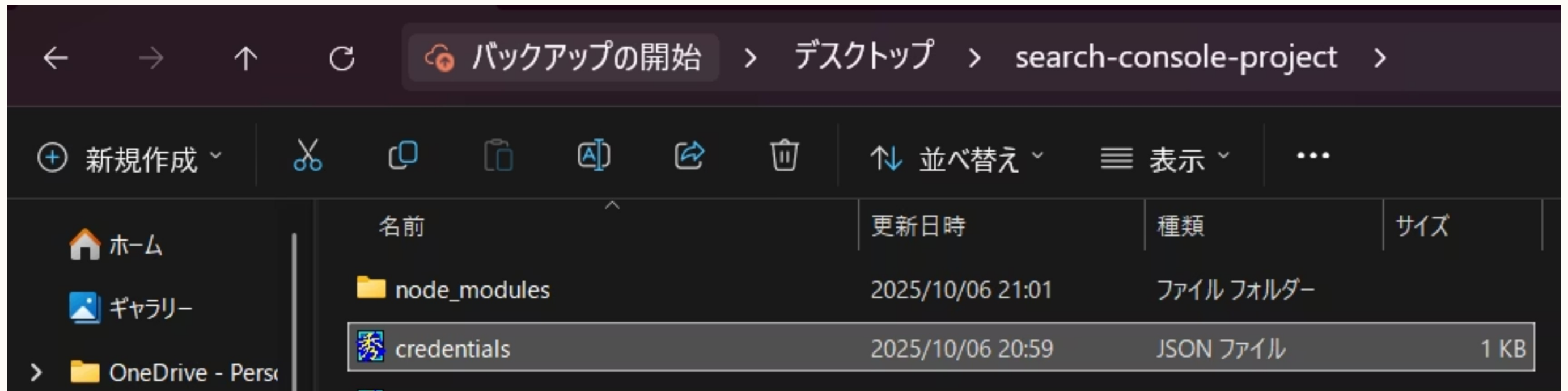
ここから、Claude との連携で MCP サーバーを構築していきます

8. プロジェクトフォルダの作成

デスクトップに「search-console-mcp」という名前のフォルダを作成

先ほど作成した credentials.json をコピー＆ペーストでフォルダに入れる

※ デスクトップではなくてもよいですが、その場合はこの後出てくるコマンドプロンプトでのディレクトリ指定を変えてください



8. プロジェクトフォルダの作成

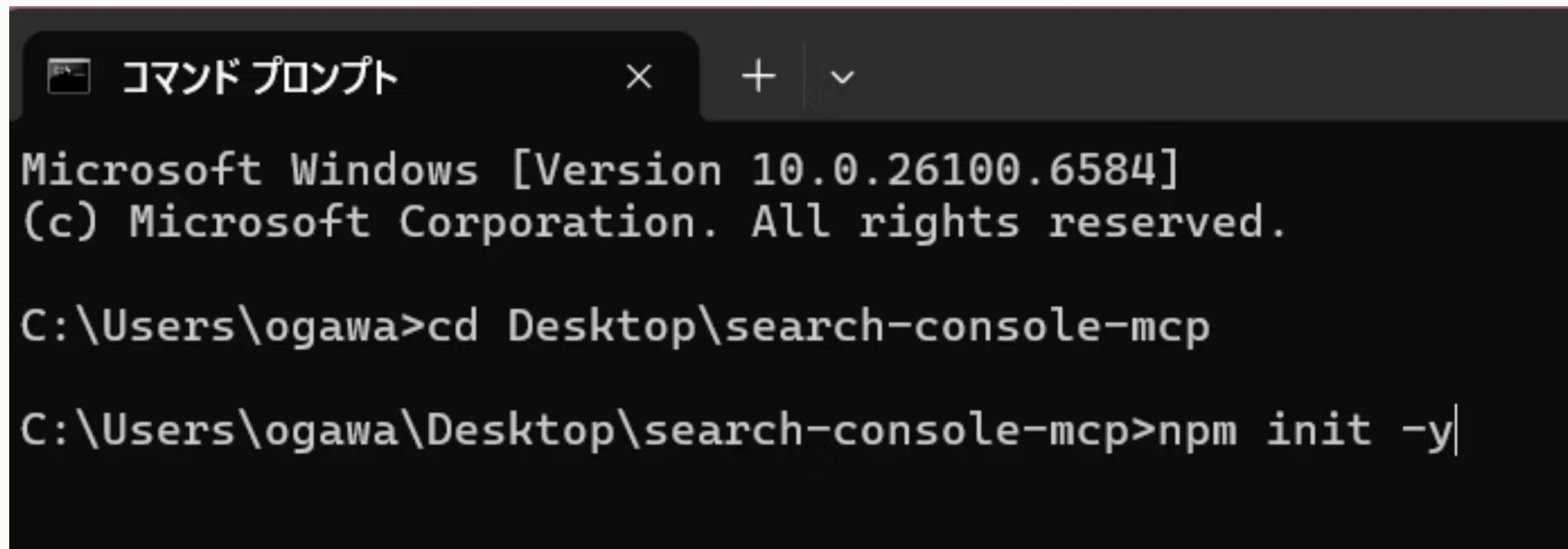
コマンドプロンプトを再起動し

```
cd Desktop\search-console-mcp
```

と入力して Enter を押し、ディレクトリを移動する

```
npm init -y
```

と入力して Enter を押すと package.json というファイルが自動生成されます。

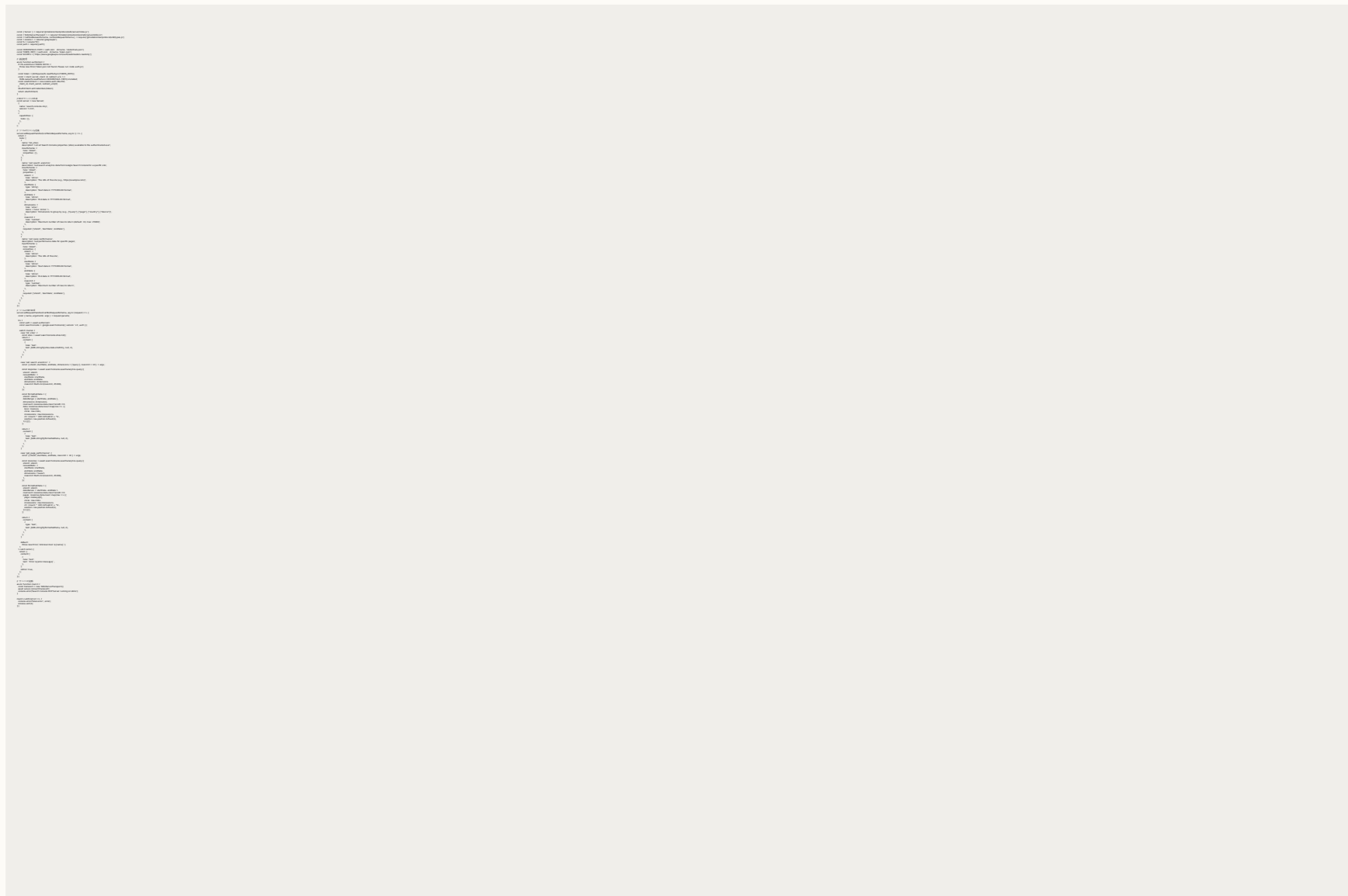


```
Microsoft Windows [Version 10.0.26100.6584]  
(c) Microsoft Corporation. All rights reserved.  
  
C:\Users\ogawa>cd Desktop\search-console-mcp  
  
C:\Users\ogawa\Desktop\search-console-mcp>npm init -y|
```

9.MCP サーバーのコード作成

server.js というファイルを作成し同じディレクトリにおきます。

中身は以下の通りです。あるいは[こちら](#)に置いておきます。



10. 認証用スクリプトの作成

auth.js というファイルを作成し同じディレクトリにおきます。
中身は以下の通りです。あるいは[こちら](#)に置いておきます。

```
const { google } = require('googleapis');
const fs = require('fs');
const path = require('path');
const readline = require('readline');

const CREDENTIALS_PATH = path.join(__dirname, 'credentials.json');
const TOKEN_PATH = path.join(__dirname, 'token.json');
const SCOPES = ['https://www.googleapis.com/auth/webmasters.readonly'];

async function authenticate() {
  if (!fs.existsSync(CREDENTIALS_PATH)) {
    console.error('Error: credentials.json not found!');
    console.error('Please copy your credentials.json file to this directory.');
```

```
    process.exit(1);
  }

  const { client_secret, client_id, redirect_uris } =
    JSON.parse(fs.readFileSync(CREDENTIALS_PATH)).installed;

  const oAuth2Client = new google.auth.OAuth2(
    client_id, client_secret, redirect_uris[0]
  );

  const authUrl = oAuth2Client.generateAuthUrl({
    access_type: 'offline',
    scope: SCOPES,
  });

  console.log('Open this URL in your browser to authenticate:');
  console.log(authUrl);
  console.log('\nEnter the code from the URL:');
```

```
  const rl = readline.createInterface({
    input: process.stdin,
    output: process.stdout,
  });

  rl.question('Code: ', async (code) => {
    rl.close();

    try {
      const { tokens } = await oAuth2Client.getToken(code);
      oAuth2Client.setCredentials(tokens);
      fs.writeFileSync(TOKEN_PATH, JSON.stringify(tokens));
      console.log('\nAuthentication successful!');
      console.log('token.json has been created.');
```

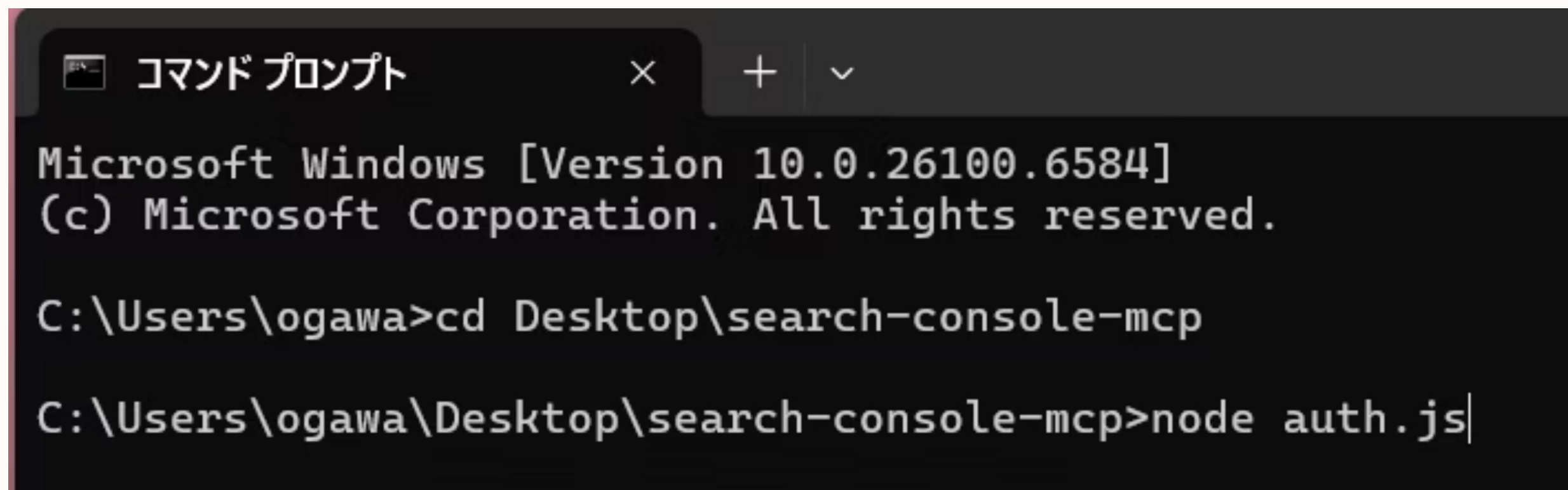
```
      console.log('\nYou can now use the MCP server with Claude Desktop.');
```

```
    } catch (error) {
      console.error('Authentication failed:', error.message);
      process.exit(1);
    }
  });
}
```

```
authenticate();
```

11. 認証を実行

コマンドプロンプトで `node auth.js` を入力して Enter を押します。



```
Microsoft Windows [Version 10.0.26100.6584]  
(c) Microsoft Corporation. All rights reserved.  
  
C:\Users\ogawa>cd Desktop\search-console-mcp  
  
C:\Users\ogawa\Desktop\search-console-mcp>node auth.js|
```

11. 認証を実行

URL が表示されるので、そちらをコピー & ペーストしてブラウザで開きます。Google アカウントの認証が終わると、localhost を含む URL にアクセスされ、ブラウザでページが表示されないエラーが出ます。

以下のような URL になります。

http://localhost/?code=4/0AVGzR1DbRzoNlmktoYORp8iy9hk_bglG950RVZeU6k-WGKQqgwMtdJ1YdW1fuY4KQZPgoQ&scope=https://www.googleapis.com/auth/webmasters.readonly

この中の code= の後ろの部分のコピーします。

4/0AVGzR1DbRzoNlmktoYORp8iy9hk_bglG950RVZeU6k-WGKQqgwMtdJ1YdW1fuY4KQZPgoQ

こちらをコマンドプロンプトに戻ると表示される「コード：」の後に張り付けて Enter キーを押してください。

Authentication successful!token.json has been created.

You can now use the MCP server with Claude Desktop.

と出れば OK です。コマンドプロンプトを閉じて大丈夫です。

12.Claude Desktop の設定

Claude Desktop の設定ファイルを編集します。

Claude Desktop の設定＞開発者から 「設定を編集」を選んでください



12.Claude Desktop の設定

表示された claude_desktop_config を開きます

Shared Dictionary	2025/07/25 9:39	ファイル フォルダー	
WebStorage	2025/07/25 9:39	ファイル フォルダー	
ant-did	2025/09/26 15:32	ファイル	1 KB
claude_desktop_config	2025/10/06 21:30	JSON ファイル	3 KB
claude_desktop_config2	2025/10/06 21:30	JSON ファイル	3 KB
config	2025/10/06 22:53	JSON ファイル	2 KB
DIPS	2025/10/06 21:31	ファイル	36 KB

12.Claude Desktop の設定

開くと GA4 の記述がすでに入っているかと思います。

```
1 {
2   "mcpServers": {
3     "google-analytics": {
4       "command": "npx",
5       "args": ["-y", "mcp-server-google-analytics"],
6       "env": {
7         "GOOGLE_CLIENT_EMAIL": "eimyistoire@eimyistoire-474311.iam.gserviceaccou
8 nt.com",
9         "GOOGLE_PRIVATE_KEY": "-----BEGIN PRIVATE KEY-----\nMIIEvAIBADANBgkqhkiG
10 9w0BAQEFAASCBKYwggSiAgEAAoIBAQDWRghBg9DtYA5ZYN90cNFeJEi32MzRH8DDiQ0iV/p+0aHN1/Ib
11 mRX5IiwQFTjaIx4GAPXtCRG++dYmeAYnt5kz1tI71JoG0rdjsfZ+OVxW1yQv1Z1/OLHDQWEfSJXKNRbj
12 JyDHQ8kKZoJER/OIYNcuT5WtM6FCGWbgoHgMDfJJSSYFhd9h98Bnfb4rkk7kUkE3DR0DKT/qISFXoUw0
13 MzYNyI8twjrIdNS1lI8Db0gJD8QLanBrf4nqrfiFDHlPInvW4p9/dtRWftVboKHHK7RqYNwVnf8VJ8HL
14 nRkB4vt+p6RW14s70KUSVEDerQ34efaryNPtAELUXyk2piDnKmp5R2YN7RD3xAXPAgMBAAECggEAEQ4C
15 Impol6EBhp8JDGoVZRZJmAp+GxBcwJU+nqJrjuk4YNQG7oDPTNHJYKUHYTKe9T35u3QK2EDQQj/KZ5+7
16 QzOvVJJmNHZ7IZzm50ic08I7Z9YNIS00J15NtpM5u1GUo+AQtFUmlDByv7BbmmZ5WnMwTH3M58VJ+BT
17 UOKsNQ0uscyIYNhW+Z4+X6tFYLRz9B2uXhu0acC6F6JPuV+AZfPugCTvAKlw2pU4Bvqm7ACJiQCSvLYn
18 SrKOKIALCTEHABPCyNmSRksXHSpfOEjwM/6kpkfKwFmP7CmhBGKV9ugqQ+IMbYtYNsIDseFMDiPXnoD
19 6gMNynVzU+s3hx+5n+YIFxdkRy5QKBgQD/ctTgtPzIhCevQZjiYNsSRRbTpBqXna2o+g72I9vN75I01R
20 Ej72zJgNoT0gw6JKnS7BR+1G/Wh6BxtamnVbYNnLB2pw4VCITjr05EJ1tVhmPfiyDApjI3R9HrkSUE7e
21 69xMwUdGxYJdKQLE+gdTAYNJkQIHpiL8q+c+dEN9opNN18EiwKBgQDWvHJOLvVIAPvSuIaV3J/8wR8x
22 QijwsnB0YNHN7fDU7VFstMQ8qA2QWvm5Laq9hP40KXrpDUzz9sDIJXwB8rwIPN8hQtViRWGesMYnUyBg
23 5JlpgvkGcEcX8Fqu+/wpu20uTEvTwp7FnNGyyS50AsYzweU2NnadrWG8xrAeYNVcul33v4TQKBgDtIAd
24 sPW26JRNBMl72stjz7VT6xvuasPBLEHL4iW7iDDK/4xQXvYNWwYmuhHlDL90FPCj6YvE+T8AcTeoir20
25 8jrG7/yvmm9gWpB0dby2UGOrU3GB4Dt1YN/mBlSBco1PPTP00uS4p1VJVC2etApAfnmX7nx80Kh/is/P
26 f5lATxaaqbAoGAPIL2YNIXoQfK3qSMM/CdQWMX+qbpMaPlnKU/jGxUxWvq2htGgWFKETfRz2ETEDz99W
27 OK+cYNmJueX7BWuquIHQuX2gMxtGTl8G11Uw/JvAQWf59E1Y+mQGhe5ZiUwp3PxToy1AWwYNADfkz+Dp
28 ej+d4s/s644FdwF43V2QG7xJmwCTSrECgYAakxONGYpnH9fRVMWtupV0YNmgaYuHsc64j0den1AIwN28
29 VNGf6MmfJHnbhU2z/xi7CEigRYqpumfhD2MC2WU9shYNpvIdltPz+DgC5QnJc6Mf4Kzra+yT1Q2fgE2s
30 krLb561KJtbf7Kmk9UkiyNifc7ahYNhXlnAQEPu2wQurDNTJW40Q==\n-----END PRIVATE KEY-----
31 -\n",
32   "GA_PROPERTY_ID": "350644389"
33 }
34 }
35 }
36 } [EOF]
```

12.Claude Desktop の設定

最後の3行を置き換えます。

```
31 -¥n", ↓  
32 "GA_PROPERTY_ID": "350644389",  
33 } ↓  
34 } ↓  
35 } ↓  
36 } [EOF]
```



```
32 "GA_PROPERTY_ID": "350644389" ↓  
33 } ↓  
34 }, ↓  
35 "search-console": { ↓  
36 "command": "node", ↓  
37 "args": [ ↓  
38 "C:¥¥Users¥¥ogawa¥¥Desktop¥¥search-console-mcp¥¥server.js" ↓  
39 ] ↓  
40 } ↓  
41 } ↓  
42 } [EOF]
```

12.Claude Desktop の設定

置き換える内容は以下の通りです

C:\\Users\\ogawa\\

の部分は自分のディレクトリに置き換えてください。

```
{,
  "search-console": {
    "command": "node",
    "args": [
      "C:\\Users\\ogawa\\Desktop\\search-console-mcp\\server.js"
    ]
  }
}
```

12.Claude Desktop の設定

タスクマネージャーでも確認し、残っている場合は、「タスクの終了」を選んでください (Ctrl+Shift+esc で起動。 Claude を右クリックで「タスクの終了」を選択)

タスク マネージャー

検索する名前、発行元、PID...

プロセス

新しいタスクを実行する

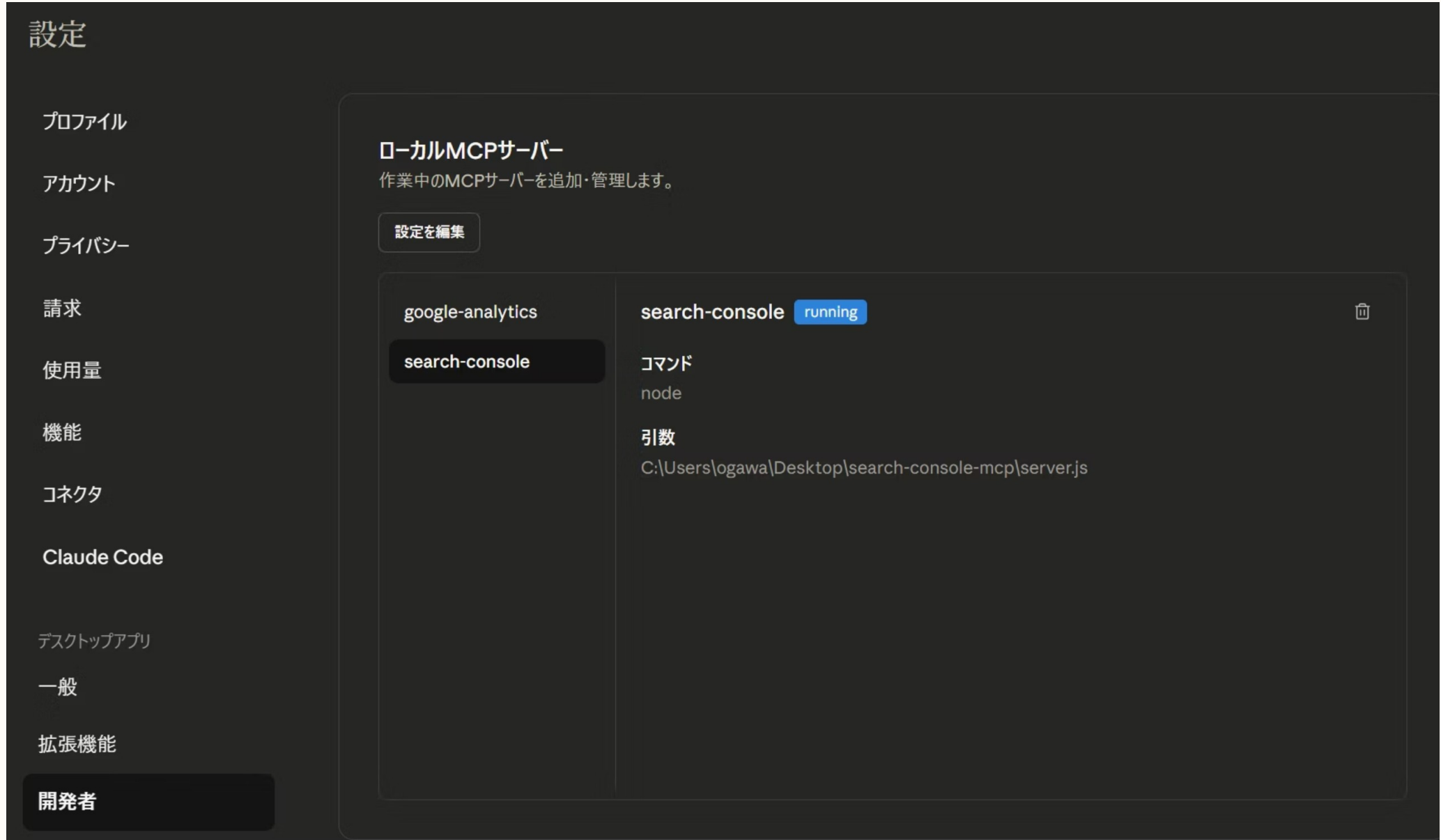
タスクを終了する

効率モード

名前	状態	6% CPU	56% メモリ	1% ディスク	0% ネットワ...
> Claude (14)		1.3%	447.8 MB	0 MB/秒	0 Mbps
> Google Chrome (26)		1.7%	2,494.6 ...	0.1 MB/秒	0 Mbps

12.Claude Desktop の設定

再度 Claude を起動し、設定＞開発者と確認し、search-console が追加されていれば完了です。



完了

後は Chat で Search Console に登録されているサイトを確認してみましょう。

TO Search Consoleに登録されているサイトを教えて

Search Consoleに登録されているサイトを確認しますね。

s List sites

Search Consoleに登録されているサイトは以下の通りです：

オーナー権限のサイト：

- ✦ [redacted]
- ✦ [redacted]

フルユーザー権限のサイト：

- ✦ [redacted]
- ✦ [redacted]
- ✦ [redacted]
- ✦ [redacted]

完了

その上でプロンプトを入れれば自由に分析できます

TO

happyanalytics.co.jpでの検索流入が多い10キーワードを直近1か月で出してください

happyanalytics.co.jpの直近1ヶ月間で検索流入が多いキーワードを取得します。

S

Get search analytics

▼

happyanalytics.co.jpの直近1ヶ月（2025年9月6日～10月6日）で検索流入が多いトップ10キーワードは以下の通りです：

順位	キーワード	クリック数	表示回数	CTR	平均掲載順位
1	ハッピーアナリティクス	31	39	79.49%	1.0位
2	小川卓	26	216	12.04%	3.7位
3	happy analytics	22	26	84.62%	1.5位
4	小川 卓	7	45	15.56%	3.4位

設定に関する FAQ

よくある質問

エラー 1-1: Search Console API が見つからない

✖ エラー症状

検索しても「Google Search Console API」が出てこない
「結果なし」と表示される

🔍 原因

GA4 の時と同じくプロジェクト選択ミス
検索ワードの誤り

✔ 対処方法

【正しい検索ワード】

× "サーチコンソール API"

× "Search Console Data API"

✔ "Google Search Console API"

【直接アクセス】

<https://console.cloud.google.com/apis/library/searchconsole.googleapis.com>

【確認】

左上のプロジェクト名が自分の作成したものか確認

よくある質問

エラー 1-2: OAuth 同意画面の設定で「メールアドレスが必要です」エラー

✖ エラー症状

OAuth 同意画面の作成時に進めない
「ユーザーサポートメール」が選択できない

🔍 原因

複数の Google アカウントでログインしている
GCP プロジェクトのアカウントと違うアカウントでログイン中

✔ 対処方法

【アカウント確認】

1. 右上のアイコンをクリック
2. 現在ログイン中のアカウントを確認
3. GCP プロジェクトを作成したアカウントと一致しているか確認

【切り替え方法】

1. 別のアカウントでログインし直す
2. または、シークレットモードで単一アカウントのみでログイン

【メールアドレス選択】

自分の Gmail アドレスをドロップダウンから選択
(プルダウンに出ない場合は上記のアカウント確認)

よくある質問

エラー 1-3: OAuth 同意画面で「外部」が選択できない

✖ エラー症状

「外部」のラジオボタンがグレースアウト
「内部」しか選択できない

🔍 原因

Google Workspace 組織に所属しているアカウント
組織ポリシーで外部アプリが制限されている

✔ 対処方法

【個人アカウントの場合】

「外部」を選択して OK（推奨）

【組織アカウントの場合】

- 「内部」を選択
- ただし、組織外のサーチコンソールデータにはアクセス不可
- 個人用 Google アカウントで別途プロジェクト作成を推奨

【確認方法】

自分のメールアドレスが @gmail.com なら個人アカウント
@company.com など企業ドメインなら組織アカウント

よくある質問

エラー 2-1: 「認証情報を作成」が押せない

✖ エラー症状

「認証情報を作成」ボタンがグレースアウト
クリックしても反応がない

🔍 原因

OAuth 同意画面の設定が未完了
権限不足

✔ 対処方法

【 OAuth 同意画面の完了確認】

1. 「API とサービス」→「 OAuth 同意画面」を開く
2. ステータスが「公開」または「テスト中」になっているか確認
3. なっていない場合 → STEP 3 をやり直し

【権限確認】

IAM で「 OAuth Config Editor 」ロールがあるか確認

よくある質問

エラー 2-2: `credentials.json` がダウンロードできない

✖ エラー症状

「ダウンロード」アイコンをクリックしても何も起こらない
ファイルが見つからない

🔍 原因

ブラウザのダウンロードブロック
ポップアップブロッカー
ファイル名が分からない

✔ 対処方法

【ダウンロード確認】

1. Chrome: 右上の「⋮」→「ダウンロード」 (Ctrl+J)
2. 最新のファイルに "client_secret_" で始まる JSON ファイルがあるか確認

【再ダウンロード】

1. 「API とサービス」→「認証情報」を開く
2. 「OAuth クライアント ID」の一覧から該当の ID をクリック
3. 右上の「JSON をダウンロード」アイコン

【ファイル名の確認】

client_secret_[長い文字列].apps.googleusercontent.com.json
→ これを「credentials.json」にリネーム

よくある質問

エラー 3-1: npm init が実行できない

✖ エラー症状

コマンドプロンプトで `npm init -y` を実行しても
「npm は内部コマンドまたは外部コマンドとして認識されていません」エラー

🔍 原因

Node.js がインストールされていない
環境変数の PATH が通っていない

✔ 対処方法

【Node.js 確認】
> `node --version`
> `npm --version`

両方でバージョンが表示されるか確認

【表示されない場合】

1. Node.js を再インストール (P.32-34 参照)
2. インストール時に「Add to PATH」にチェックが入っているか確認
3. インストール後、コマンドプロンプトを再起動

【環境変数の確認 (上級者向け)】

1. システムのプロパティ → 環境変数
2. Path に以下が含まれているか確認:
C:\Program Files\nodejs\

よくある質問

エラー 3-2: npm install googleapis でエラー

✖ エラー症状

npm install googleapis 実行時にエラー

「EACCES: permission denied」

「npm ERR! code ENOENT」

🔍 原因

管理者権限不足

プロジェクトフォルダのパスが間違っている

ネットワーク制限

✔ 対処方法

【権限エラー（EACCES）の場合】

1. コマンドプロンプトを管理者として実行

- スタートメニューで「コマンドプロンプト」を右クリック
- 「管理者として実行」

2. 再度フォルダに移動して実行

> cd C:\search-console-mcp

> npm install googleapis

【ENOENT エラーの場合】

フォルダが存在しないか、パスが間違っている

> cd C:\search-console-mcp

> dir ← フォルダの中身を確認

package.json があるか確認

【ネットワークエラーの場合】

社内プロキシ設定が必要な場合：

> npm config set proxy http://proxy.company.com:8080

> npm config set https-proxy http://proxy.company.com:8080

よくある質問

エラー 4-1: node auth.js 実行時に「Cannot find module」エラー

✖ エラー症状

Error: Cannot find module 'googleapis'

Error: Cannot find module './credentials.json'

🔍 原因

npm install が正しく完了していない

credentials.json が正しい場所がない

ファイル名が間違っている

よくある質問

エラー 4-1: node auth.js 実行時に「Cannot find module」エラー

✓ 対処方法

【googleapis が見つからない】

1. node_modules フォルダがあるか確認
> dir
2. 無い場合は再インストール
> npm install googleapis

【credentials.json が見つからない】

1. フォルダ構成を確認
C:\search-console-mcp\
├── credentials.json ← ここに配置
├── auth.js
├── package.json
└── node_modules\
2. ファイル名を確認
× client_secret_xxx.json ← ダウンロード時の名前
✓ credentials.json ← 正しいリネーム後
3. パス確認
auth.js 内のパスが正しいか確認
const CREDENTIALS_PATH = './credentials.json';

よくある質問

エラー 4-2: 認証 URL が表示されない

✖ エラー症状

node auth.js を実行しても何も表示されない
または途中でエラーが出る

🔍 原因

credentials.json の内容が不正
OAuth 同意画面の設定が未完了
スクリプトのコードに誤りがある

よくある質問

エラー 4-2: 認証 URL が表示されない

✓ 対処方法

【credentials.json の検証】

1. ファイルをメモ帳で開く
2. 以下の項目があるか確認:

```
{  
  "installed": {  
    "client_id": "...",  
    "client_secret": "...",  
    "redirect_uris": ["http://localhost"]  
  }  
}
```

3. JSON 構文が正しいか確認

<https://jsonlint.com/> でチェック

【OAuth 設定確認】

GCP → API とサービス → OAuth 同意画面

→ ステータスが「テスト中」以上になっているか

【スクリプト確認】

資料の auth.js のコードと完全一致しているか確認

特に以下の行:

```
const SCOPES = ['https://www.googleapis.com/auth/webmasters.readonly'];
```

よくある質問

エラー 4-3: ブラウザで認証画面が開かない

✖ エラー症状

認証 URL が表示されたが、ブラウザで開いてもエラー
「このサイトにアクセスできません」
「リダイレクト URI が一致しません」エラー

🔍 原因

リダイレクト URI の設定ミス
ブラウザがローカルホストをブロック

✔ 対処方法

【リダイレクト URI 確認】

1. GCP → API とサービス → 認証情報
2. 作成した OAuth クライアント ID をクリック
3. 「承認済みのリダイレクト URI」を確認

正しい設定：

✔ http://localhost

または

✔ http://localhost:3000

【エラーメッセージの場合】

「redirect_uri_mismatch」エラーが出たら：

1. エラーメッセージ内に表示される「redirect_uri」をコピー
2. GCP の OAuth 設定に追加
3. 認証をやり直し

【別ブラウザで試す】

Chrome → Edge または Firefox で試す

よくある質問

エラー 5-1: 認証コードを入力しても「Invalid code」エラー

✖ エラー症状

Enter the code from that page here: [コード入力]

Error: invalid_grant

🔍 原因

認証コードのコピーミス（余分なスペースや改行）

認証コードの有効期限切れ（数分で失効）

複数回入力してコードが無効化

✔ 対処方法

【正しいコピー方法】

1. ブラウザの認証画面でコードが表示される
例：4/0AX4XfWh...（長い文字列）

2. コード全体を選択してコピー

- マウสดラッグで選択
- Ctrl+C でコピー

3. コマンドプロンプトに貼り付け

- 右クリック → 貼り付け
- または Ctrl+V

【検証】

貼り付けたコードを確認：

- 前後にスペース無し
- 改行無し
- 全て半角英数字

【有効期限切れの場合】

1. 認証 URL を再取得

> node auth.js

2. 新しい URL をブラウザで開く

3. 新しいコードをすぐに入力

よくある質問

エラー 5-2: token.json が生成されない

✖ エラー症状

認証コード入力後、エラーは出ないが token.json ができない
「Authentication successful!」などのメッセージが出ない

🔍 原因

スクリプトが途中で止まっている
ファイル書き込み権限がない
スクリプトのコードミス

✔ 対処方法

【権限確認】

1. フォルダを右クリック → プロパティ → セキュリティ
2. 自分のユーザーに「書き込み」権限があるか確認

【フォルダ確認】

> dir
token.json があるか確認

【スクリプト確認】

auth.js の最後に以下があるか確認：
fs.writeFileSync('token.json', JSON.stringify(tokens));
console.log('Token stored to token.json');

【デバッグ実行】

スクリプトに console.log を追加して動作確認

よくある質問

エラー 6-1: server.js 実行時に「SyntaxError」

✖ エラー症状

SyntaxError: Unexpected token 'export'

SyntaxError: Cannot use import statement outside a module

🔍 原因

Node.js のバージョンが古い (v14 以下)

package.json に "type": "module" がない

import/export の書き方が間違っている

✔ 対処方法

【 Node.js バージョン確認】

> node --version

v18.0.0 以上が推奨

古い場合：最新の LTS 版を再インストール

【 package.json 修正】

ファイルを開いて以下を追加：

```
{  
  "type": "module", ← これを追加  
  "dependencies": {  
    "googleapis": "^..."  
  }  
}
```

保存後、再実行：

> node server.js

【代替方法】

CommonJS 形式に書き換え：

× import { google } from 'googleapis';

✔ const { google } = require('googleapis');

よくある質問

エラー 6-2: server.js 実行時に「token.json が見つからない」

✖ エラー症状

Error: ENOENT: no such file or directory, open 'token.json'

🔍 原因

token.json が生成されていない (エラー 5-2 参照)

ファイルパスが間違っている

別のフォルダで実行している

✅ 対処方法

【ファイル確認】

```
> dir
```

以下のファイルがすべて同じフォルダにあるか確認:

- credentials.json
- token.json
- server.js
- auth.js
- package.json

【パス確認】

server.js 内のパスを確認:

```
const TOKEN_PATH = './token.json';
```

```
const CREDENTIALS_PATH = './credentials.json';
```

相対パス (./) になっているか確認

【実行フォルダ確認】

```
> cd C:\search-console-mcp
```

```
> node server.js
```

必ず正しいフォルダで実行

よくある質問

エラー 6-3: MCP サーバー起動後、すぐに終了する

✖ エラー症状

node server.js を実行すると一瞬で終わる
エラーメッセージが出ない

🔍 原因

サーバーのリスニング処理がない
スクリプトが途中で終了している

✅ 対処方法

【正しい server.js 】

資料のコードには以下のような構造があるはず：

```
// MCP サーバーの起動
const server = createServer({
  name: 'search-console',
  version: '1.0.0'
});
```

```
// リクエスト処理
server.setRequestHandler(...);
```

```
// 標準入出力で待機
server.connect(...);
```

【デバッグ】

console.log('MCP Server started'); を追加して
起動確認

よくある質問

エラー 7-1: `claude_desktop_config.json` に `search-console` 設定を追加できない

✖ エラー症状

既存の `ga4` 設定に追加しようとするときエラー
JSON 構文エラーになる

🔍 原因

カンマの位置が間違っている
括弧の対応がずれている

よくある質問

エラー 7-1: `claude_desktop_config.json` に `search-console` 設定を追加できない

✅ 対処方法

【正しい構文（GA4 + サーチコンソール）】

```
{
  "mcpServers": {
    "google-analytics": {
      "command": "npx",
      "args": ["-y", "mcp-server-google-analytics"],
      "env": { ... }
    }, ← カンマ必要
    "search-console": {
      "command": "node",
      "args": ["C:\\search-console-mcp\\server.js"],
      "env": {}
    } ← 最後はカンマ不要
  }
}
```

【パスの注意】

Windows のパスは \\ （バックスラッシュ 2 つ）でエスケープ

× "C:\\search-console-mcp\\server.js"

✅ "C:\\search-console-mcp\\server.js"

【検証】

<https://jsonlint.com/> で構文チェック

よくある質問

エラー 7-2: Claude Desktop 起動後、search-console が表示されない

✖ エラー症状

「ファイル」→「コネクタ」を開いても
google-analytics しか表示されない
search-console が無い

🔍 原因

server.js のパスが間違っている
server.js が正しく動作していない
プロセスが残っている（GA4の時と同じ）

よくある質問

エラー 7-2: Claude Desktop 起動後、search-console が表示されない

✓ 対処方法

【パス確認】

1. エクスプローラーで server.js の場所を開く
2. アドレスバーをクリックしてパスをコピー
例 : C:\search-console-mcp
3. server.js を含めた完全パス :
C:\search-console-mcp\server.js
4. claude_desktop_config.json に記載
"args": ["C:\\search-console-mcp\\server.js"]
※ バックスラッシュ 2 つ

【server.js 動作確認】

コマンドプロンプトで単独実行 :

```
> cd C:\search-console-mcp  
> node server.js
```

エラーが出ないか確認

【プロセス終了】

1. タスクマネージャー (Ctrl+Shift+Esc)
2. "Claude" と "node" の両方を終了
3. Claude Desktop を再起動

よくある質問

エラー 8-1: 「サイトが見つかりません」エラー

✖ エラー症状

プロンプト入力後、「 Site not found 」エラー
データが取得できない

🔍 原因

サイト URL の形式が間違っている
サーチコンソールに登録されていない
権限がない

✔ 対処方法

【正しい URL 形式】

サーチコンソールに登録されている形式と完全一致が必要:

ドメインプロパティの場合:

✔ sc-domain:example.com

URL プレフィックスの場合:

✔ https://www.example.com/

✔ https://example.com/

【確認方法】

1. Search Console を開く
<https://search.google.com/search-console>
2. プロパティ一覧を確認
3. 表示されている通りの形式で Claude に指示

【プロンプト例】

「 <https://www.example.com/> の過去 7 日間の検索パフォーマンスを教えて」

※ 最後のスラッシュも含めて正確に

よくある質問

エラー 8-2: 「Permission denied」エラー

✖ エラー症状

Error: User does not have sufficient permissions for this site

🔍 原因

認証した Google アカウントにサーチコンソールの権限がない
token.json が別のアカウントのもの

✔ 対処方法

【権限確認】

1. Search Console を開く
2. 設定（歯車アイコン）→ ユーザーと権限
3. 認証に使った Google アカウントがあるか確認

権限レベル：

✔ オーナー

✔ フルユーザー

✖ 制限付きユーザー（データ取得不可）

【再認証】

1. token.json を削除
> del token.json

2. 再度認証実行
> node auth.js

3. 正しい権限のあるアカウントでログイン

よくある質問

エラー 8-3: データが空で返ってくる

✖ エラー症状

エラーは出ないが、データが 0 件
「データがありません」と返る

🔍 原因

指定期間にデータがない
サイトに検索トラフィックがない
インデックス未登録

✔ 対処方法

【Search Console で確認】

1. 該当サイトのパフォーマンスレポートを開く
2. データがあるか確認
3. ある場合 → 期間指定を変更

【プロンプト調整】

× 「今日のデータ」（反映に 2-3 日かかる）

✔ 「過去 28 日間のデータ」

✔ 「先月のデータ」

【新規サイトの場合】

- インデックス登録されるまで数日～数週間
- GSC にデータが表示されるまで待つ必要あり

よくある質問

エラー 8-3: データが空で返ってくる

✖ エラー症状

エラーは出ないが、データが 0 件
「データがありません」と返る

🔍 原因

指定期間にデータがない
サイトに検索トラフィックがない
インデックス未登録

✔ 対処方法

【Search Console で確認】

1. 該当サイトのパフォーマンスレポートを開く
2. データがあるか確認
3. ある場合 → 期間指定を変更

【プロンプト調整】

× 「今日のデータ」（反映に 2-3 日かかる）

✔ 「過去 28 日間のデータ」

✔ 「先月のデータ」

【新規サイトの場合】

- インデックス登録されるまで数日～数週間
- GSC にデータが表示されるまで待つ必要あり

BigQuery と Claude 連携方法

設定のための前提

利用するには以下が必要となります

- GA4 との連携がすでに終わっている状態（GCP のアカウントがあり、Node.js と Claude Desktop がインストールされていること）
- GCP の BigQuery 上で GA4 のデータを貯めていること

注意点

Windows を前提に説明しますが、Mac でも可能です

作業時間

30 分程度見ておきましょう

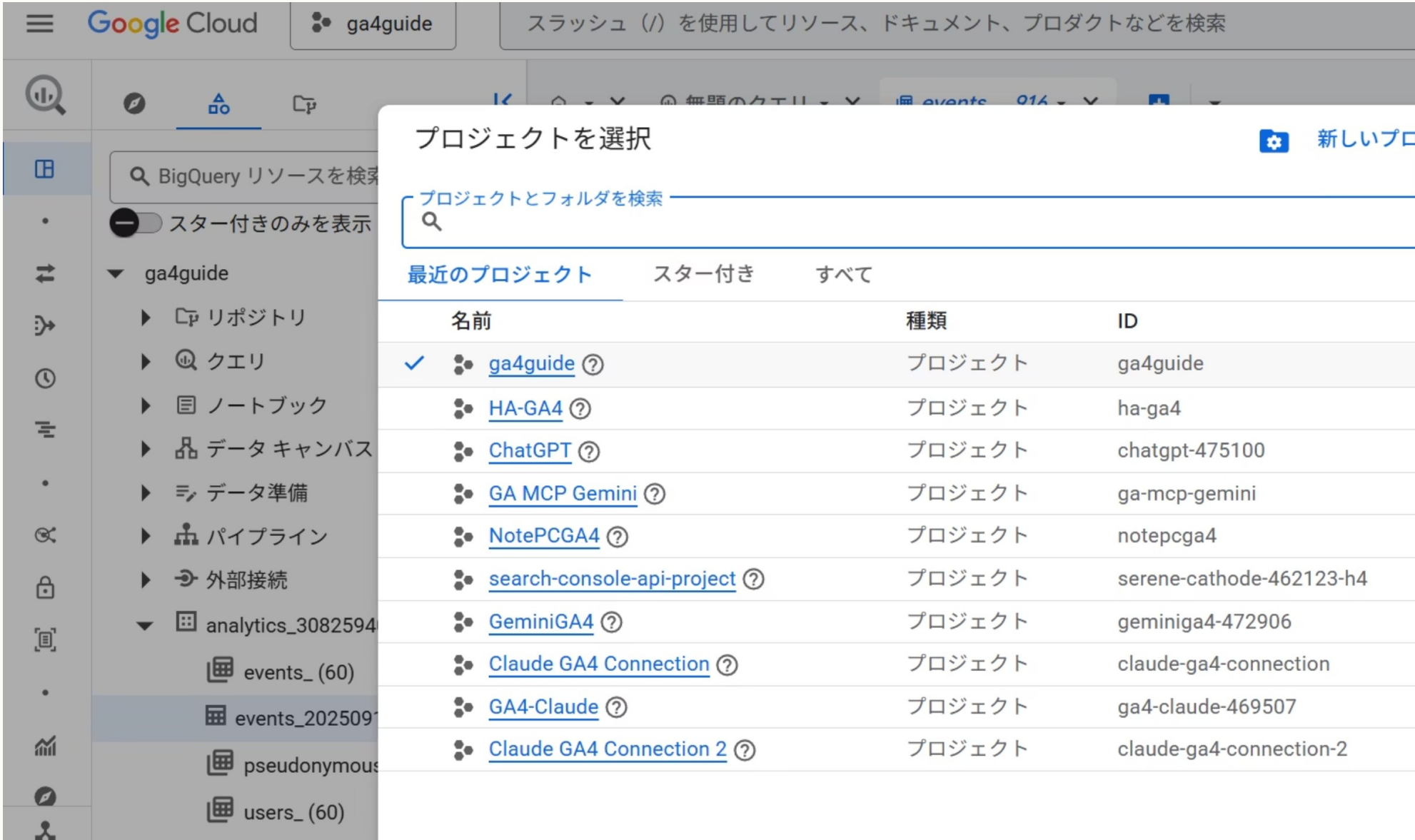
作業フロー

全部で 6 の STEP になります

1. 権限の付与
2. 鍵の取得
3. GA4 のデータセットとテーブルを確認
4. MCP サーバプロジェクトの作成
5. サーバコードの作成
6. Claude Desktop の設定

1. 権限の付与

Google Cloud Console にアクセスし、 BigQuery のデータが入っているプロジェクトを選択する



1. 権限の付与

左上のメニューを開き、IAM と管理から「サービス アカウント」を選ぶ



1. 権限の付与

「+ サービスアカウントの作成」を押し任意のサービスアカウント名をつける

1 サービス アカウントの作成

サービス アカウント名

bigquery-mcp-server

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 *

bigquery-mcp-server



メールアドレス: bigquery-mcp-server@ga4guide.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

1. 権限の付与

権限で「BigQuery データ閲覧者」と「BigQuery ジョブユーザー」のロールを付与

2 権限 (省略可)

このサービス アカウントに ga4guide 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
BigQuery データ閲覧者 ▼

データセットとそのすべてのコンテンツを表示するためのアクセス権

IAM の条件 (省略可) ?

[+ IAM の条件を追加](#)



ロール
BigQuery ジョブユーザー ▼

ジョブを実行するためのアクセス権

IAM の条件 (省略可) ?

[+ IAM の条件を追加](#)



[+ 別のロールを追加](#)

続行

2. 鍵の取得

完了を押して権限付与を完了後、メール部分をクリック

サービス アカウント

+ サービス アカウントを作成

削除

+ アクセスを管理

更新

プロジェクト「ga4guide」のサービス アカウント

サービス アカウントは Google Cloud サービス ID（Compute Engine VM、App Engine アプリ、Google 以外で実行されているサービスの詳細をご覧ください。

組織のポリシーを使用してサービス アカウントを保護できます。IAM ロールの自動付与、鍵の作成やアップロード、サービス機能をブロックすることが可能です。サービス アカウントの組織のポリシーの詳細をご覧ください。

フィルタ

プロパティ名または値を入力

	メール	ステータス	名前 ↑	説明	キー ID
☐	bigquery-mcp-server@ga4guide.iam.gserviceaccount.com	有効	bigquery-mcp-server		キーがありません

2. 鍵の取得

上のメニューから「鍵」を選び、「キーを追加＞新しい鍵を作成」する

+

?

🔒

📅

👤

🔧

📄

☰

🔗

📺

📺

☰

🏷️

➡️

⚙️

⚙️

← bigquery-mcp-server

詳細権限鍵指標ログアクセス権を持つプリンシパル

鍵

⚠️

サービス アカウント キーは、不正使用されるとセキュリティ上のリスクになる可能性があります。サービス Identity 連携を使用することをおすすめします。Google Cloud でサービス アカウントを認証する最適な方法

ℹ️

Google は公開リポジトリで検出されたサービス アカウント キーを自動的に無効にします。組織のポリシーをカスタマイズできます。[詳細](#)

新しい鍵ペアを追加するか、既存の鍵ペアから公開鍵証明書をアップロードしてください。

[組織のポリシー](#)を使用して、サービス アカウント キーの作成をブロックします。
[サービス アカウント用の組織のポリシーの設定の詳細](#)

キーを追加

新しい鍵を作成

既存の鍵をアップロード

キー	作成日	有効期限
----	-----	------

2. 鍵の取得

JSON 形式を選択して、ファイルをダウンロードする。作成したキーは後ほど使用

「bigquery-mcp-server」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
推奨

☐ P12
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

3.GA4 のデータセットとテーブルを確認

左上のメニューから「BigQuery Console」にアクセス



3.GA4 のデータセットとテーブルを確認

GA4 データが含まれているデータセットを確認。

「analytics_XXXXXXXX」をメモしておいてください。

Google Cloud

ga4guide

スラッシュ (/) を使用してリソース、ドキュメント、プロダクトなどを検索

検索

サンドボックス

お支払い情報を設定すると、完全な BigQuery にアップグレードされます。[詳細](#)

BigQuery リソースを検索

スター付きのみを表示

ga4guide

リポジトリ

クエリ

ノートブック

データ キャンバス

データ準備

パイプライン

外部接続

analytics_308259408

events_ (60)

pseudonymous_u

users_ (60)

events_... 014

ga4guide / ... / Tables / events_20251014

2025-10-14

クエリ

次で開く

共有

スキーマ

詳細

プレビュー

テーブル エクスプローラ

プレビュー

分析情報

フィルタ

プロパティ名または値を入力

フィールド名	種類	モード	説明
event_date	STRING	NULLABLE	-
event_timestamp	INTEGER	NULLABLE	-
event_name	STRING	NULLABLE	-
params	RECORD	REPEATED	-
previous_timestamp	INTEGER	NULLABLE	-
event_value_in_usd	FLOAT	NULLABLE	-
event_bundle_sequence_id	INTEGER	NULLABLE	-
event_server_timestamp_offset	INTEGER	NULLABLE	-
user_id	STRING	NULLABLE	-

analytics_308259408

タップすると現在のタブで開きます

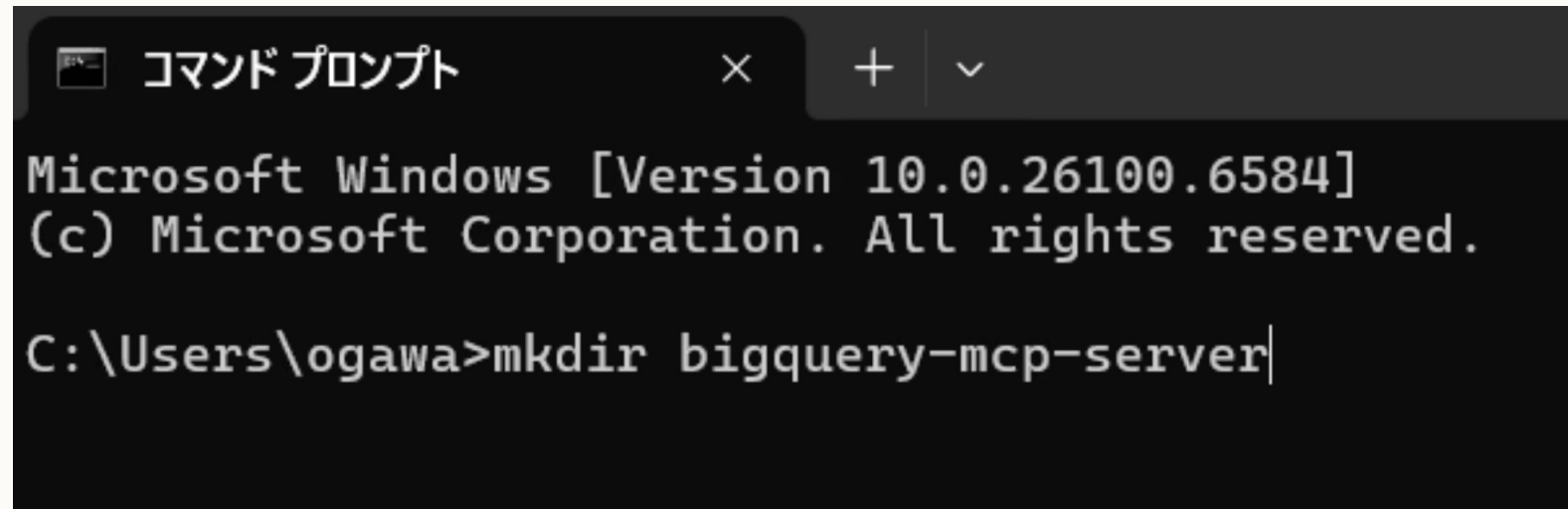
(Ctrl + タップ - 新しいタブで開きます

Shift+Ctrl + タップ - 分割タブで開きます)

4.MCP サーバープロジェクトの作成

コマンドプロンプトを開き、

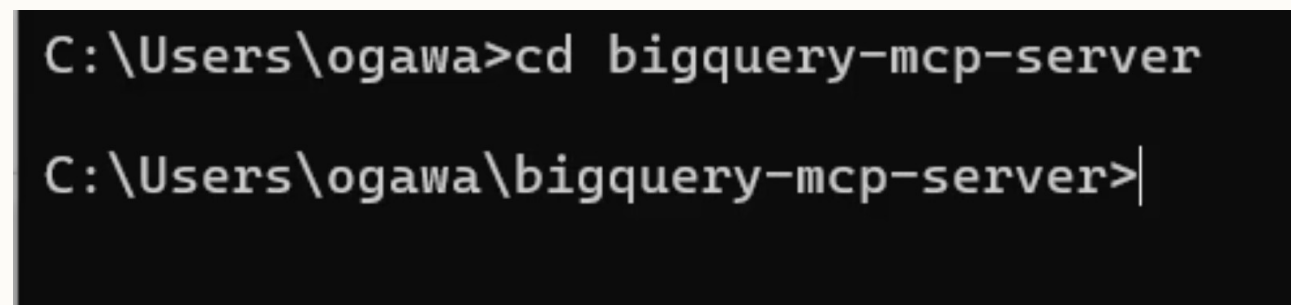
`mkdir bigquery-mcp-server` でディレクトリを作成後



```
コマンド プロンプト
Microsoft Windows [Version 10.0.26100.6584]
(c) Microsoft Corporation. All rights reserved.
C:\Users\ogawa>mkdir bigquery-mcp-server|
```

`cd bigquery-mcp-server`

でフォルダに移動してください

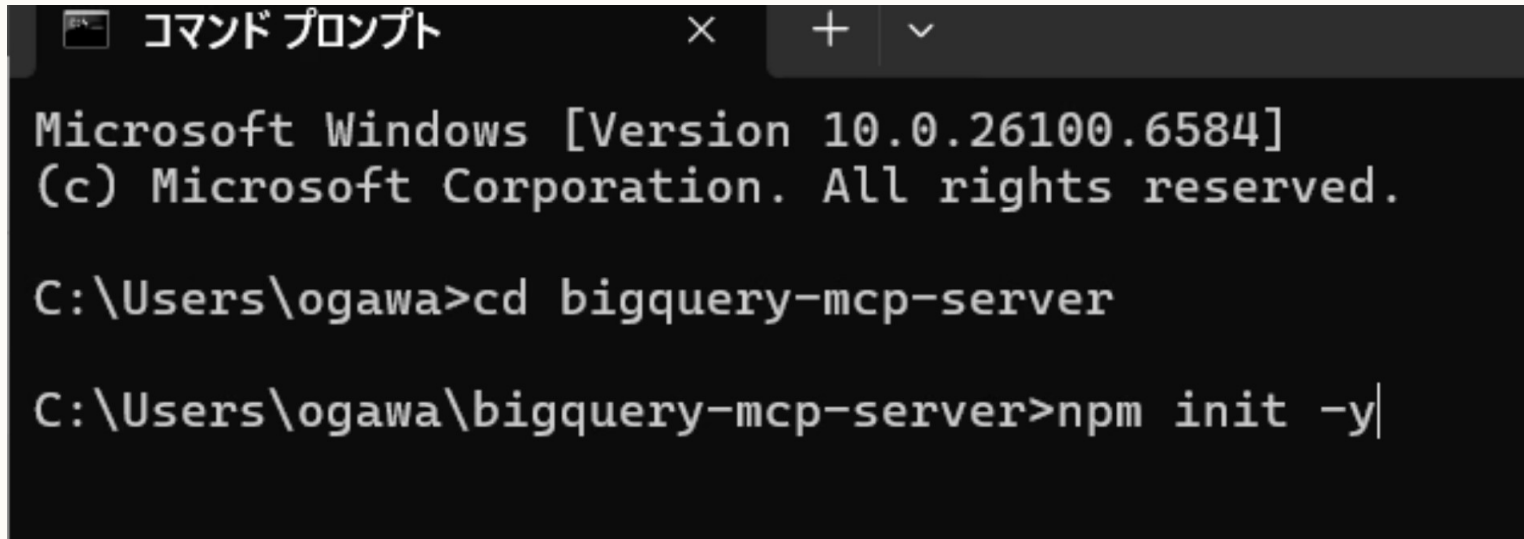


```
C:\Users\ogawa>cd bigquery-mcp-server
C:\Users\ogawa\bigquery-mcp-server>|
```

4.MCP サーバープロジェクトの作成

Node.js プロジェクトの初期化を行うため

`npm init -y` を入力して実行し

A screenshot of a Windows Command Prompt window. The title bar reads 'コマンド プロンプト' (Command Prompt). The window content shows the following text: 'Microsoft Windows [Version 10.0.26100.6584] (c) Microsoft Corporation. All rights reserved. C:\Users\ogawa>cd bigquery-mcp-server C:\Users\ogawa\bigquery-mcp-server>npm init -y|'. The text is displayed in a monospaced font on a dark background.

```
Microsoft Windows [Version 10.0.26100.6584]
(c) Microsoft Corporation. All rights reserved.

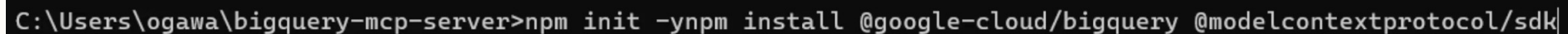
C:\Users\ogawa>cd bigquery-mcp-server

C:\Users\ogawa\bigquery-mcp-server>npm init -y|
```

その後に bigquery 用のパッケージをインストールするため

`npm install @google-cloud/bigquery @modelcontextprotocol/sdk`

を実行します

A screenshot of a Windows Command Prompt window showing the command to install the BigQuery and MCP SDK packages. The text is: 'C:\Users\ogawa\bigquery-mcp-server>npm init -ynpm install @google-cloud/bigquery @modelcontextprotocol/sdk|'. The text is displayed in a monospaced font on a dark background.

```
C:\Users\ogawa\bigquery-mcp-server>npm init -ynpm install @google-cloud/bigquery @modelcontextprotocol/sdk|
```

5. サーバーコードの作成

作成したフォルダにエクスプローラー等で移動してください。

先ほどダウンロードしたJSON ファイルを「bigquery-key」に名称変更します



5. サーバーコードの作成

新規に index.js ファイルを作成します。

中身は以下あるいは[こちら](#)においてあります。

```
const express = require('express');
const app = express();
const port = 3000;

app.get('/', (req, res) => {
  res.send('Hello World!');
});

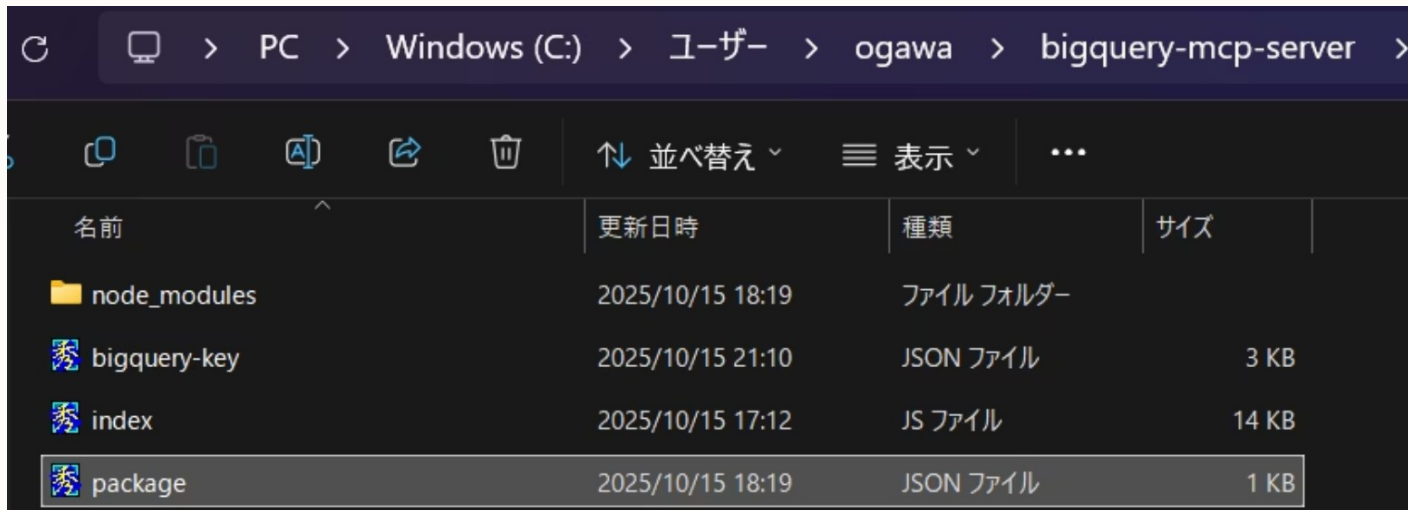
app.listen(port, () => {
  console.log(`Example app listening on port ${port}`);
});
```

5. サーバーコードの作成

package.json ファイルを開いて

"type": "module",

を追加します。



```
1 {  
2   "name": "bigquery-mcp-server",  
3   "version": "1.0.0",  
4   "type": "module",  
5   "main": "index.js",  
6   "bin": {  
7     "bigquery-mcp-server": "index.js"  
8   },  
9   "dependencies": {  
10    "@google-cloud/bigquery": "^7.9.4",  
11    "@modelcontextprotocol/sdk": "^0.5.0"  
12  },  
13  "scripts": {  
14    "test": "echo ¥\"Error: no test specified¥\" && exit 1"  
15  },  
16  "keywords": [],  
17  "author": "",  
18  "license": "ISC",  
19  "description": ""  
20 }
```

6.Claude Desktop の設定

Claude Desktop の設定ファイルを編集します。

Claude Desktop の設定＞開発者から 「設定を編集」 を選んでください



6.Claude Desktop の設定

表示された claude_desktop_config を開きます

Shared Dictionary	2025/07/25 9:39	ファイル フォルダー	
WebStorage	2025/07/25 9:39	ファイル フォルダー	
ant-did	2025/09/26 15:32	ファイル	1 KB
claude_desktop_config	2025/10/06 21:30	JSON ファイル	3 KB
claude_desktop_config2	2025/10/06 21:30	JSON ファイル	3 KB
config	2025/10/06 22:53	JSON ファイル	2 KB
DIPS	2025/10/06 21:31	ファイル	36 KB

6.Claude Desktop の設定

開くと GA4 の記述等がすでに入っているかと思います。

```
1 {
2   "mcpServers": {
3     "google-analytics": {
4       "command": "npx",
5       "args": ["-y", "mcp-server-google-analytics"],
6       "env": {
7         "GOOGLE_CLIENT_EMAIL": "eimyistoire@eimyistoire-474311.iam.gserviceaccou
8 nt.com",
9         "GOOGLE_PRIVATE_KEY": "-----BEGIN PRIVATE KEY-----¥nMIEvAIBADANBgkqhkiG
10 9wOBAQEFAASCBywggSiAgEAAoIBAQDWRghBg9DtYA5Z¥n90cNFeJEi32MzRH8DDiQ0iV/p+OaHN1/Ib
11 mRX5IiwQFTjaIx4GAPXtCRG++dYmeA¥nt5kz1tI71JoGOrdjsfZ+OVxW1yQv1Zl/OLHDQWEfSJXKNRbj
12 JyDHQ8kKZoJER/OI¥ncuT5WtM6FCGWBgoHgMDfJJSSYFhd9h98Bnfb4rkk7kUkE3DRODKT/qISFXoUw0
13 Mz¥nYl8twjrIdNS1lI8Db0gJD8QLanBrf4nqr fi fDHIPInvW4p9/dtRWftVboKHHK7Rq¥nWVnf8VJ8HL
14 nRkB4vt+p6RW14s70KUSVEDerQ34efaryNPtAELUXyk2piDnKmp5R2¥n7RD3xAXPAgMBAAECggEAEQ4C
15 Impol6EBhp8JDGoVZRZJmAp+GxBcwJU+nqJrjuk4¥nQG7oDPTNHJYKUHYTKe9T35u3QK2EDQQj/KZ5+7
16 Qz0vVJJmNHZ7IZzm50ic08l7Z9¥nIS00J15NtpM5u1GUo+AQtFUmlDByv7BbmmmZ5WnMwTH3M58VJ+BT
17 UOKsNQOuscyI¥nhW+Z4+X6tFYLRz9B2uXhu0ac6F6JPuV+AZfPugCTvAKlw2pU4Bvqm7ACJiQCSvL¥n
18 SrKOKIALCTEHABPCyNmSRksXHSpfoEjwM/6kpkfKwFmP7CmhBGKV9ugqQ+IIMbYt¥nsIDseFMDiPXnoD
19 6gMNynVzU+s3hx+5n+YIFxdkRy5QKBgQD/ctTGtPzIhCevQZji¥nSsRRbTpBqXna2o+g72l9vN75I01R
20 Ej72zJgNoT0gw6JKnS7BR+1G/Wh6BxtamnVb¥nnLB2pw4VCITjr05EJ1tVhmPfiyDApjI3R9HrkSUe7e
21 69xMwUdGxYJdKQLLE+gdTA¥nJkQIHpiL8q+c+dEN9opNN18EiwKBgQDWvHJOLvVIAPvSuIaV3J/8wR8x
22 QijwsnB0¥nHN7fDU7VFstMQ8qA2QWvm5Laq9hP40KXrpDUzz9sDIJXwB8rwIPN8hQtViRWGesM¥nUyBg
23 5J1pgvkGcEcX8Fqu+/wpu20uTEvTwp7FnNGyyS50AsYzweU2NnadrWG8xrAe¥nVcu133v4TQKBgDtIAd
24 sPW26JRNBMl72sTjz7VT6xvuasPBLEHL4iW7iDDK/4xQXv¥nWwYmuhHlDl90FPCj6YvE+T8AcTeoir20
25 8jrG7/yvmm9gWpB0dby2UG0rU3GB4Dt1¥n/mBlsbCo1PPTP00uS4p1VJVC2etApAfmX7nx80Kh/is/P
26 f5IATxaaqbAoGAPIL2¥nIxoQfK3qSMM/CdQWMX+qbpMaPInKU/jGxUxWvq2htGgWFKETfRz2ETEDz99W
27 OK+c¥nmJueX7BWuquIHQuX2gMxtGTl8G11Uw/JvAQwZ59E1Y+mQGhe5ZiUwp3PxToy1AWw¥nADfkz+Dp
28 ej+d4s/s644FdwF43V2QG7xJmwCTSrECgYAakxONGYpnH9fRVMWtupVO¥nmgaYuHsc64j0den1AIwN28
29 VNGf6MmfJHnbhU2z/xi7CEigRYqpumfhD2MC2WU9sh¥npvidltPz+DgC5QnJc6Mf4Kzra+yT1Q2fgE2s
30 krLb561KJTbf7Kmk9UkiyNifc7ah¥nhXlnaQEPu2wQurDNTJW40Q==¥n-----END PRIVATE KEY-----
31 -¥n",
32   "GA_PROPERTY_ID": "350644389"
33 }
34 }
35 }
36 }
```

6. Claude Desktop の設定

最後の3行を置き換えます

```
31 -¥n", ↓  
32 "GA_PROPERTY_ID": "350644389",  
33 } ↓  
34 } ↓  
35 } ↓  
36 } [EOF]
```



```
32 "GA_PROPERTY_ID": "350644389" ↓  
33 } ↓  
34 }, ↓  
35 "search-console": { ↓  
36 "command": "node", ↓  
37 "args": [ ↓  
38 "C:¥¥Users¥¥ogawa¥¥Desktop¥¥search-console-mcp¥¥server.js" ↓  
39 ] ↓  
40 } ↓  
41 } ↓  
42 } [EOF]
```

6.Claude Desktop の設定

置き換える内容は以下の通りです

C:\\Users\\logawa\\

の部分は自分のディレクトリに置き換えてください。

```
{,
  "bigquery-ga4": {
    "command": "node",
    "args": [
      "C:\\Users\\logawa\\bigquery-mcp-server\\index.js"
    ],
    "env": {
      "GOOGLE_APPLICATION_CREDENTIALS": "C:\\Users\\logawa\\bigquery-mcp-server\\bigquery-key.json",
      "GCP_PROJECT_ID": "プロジェクト ID",
      "GA4_DATASET_ID": "データセット ID"
    }
  }
}
```

Index.js の置き場

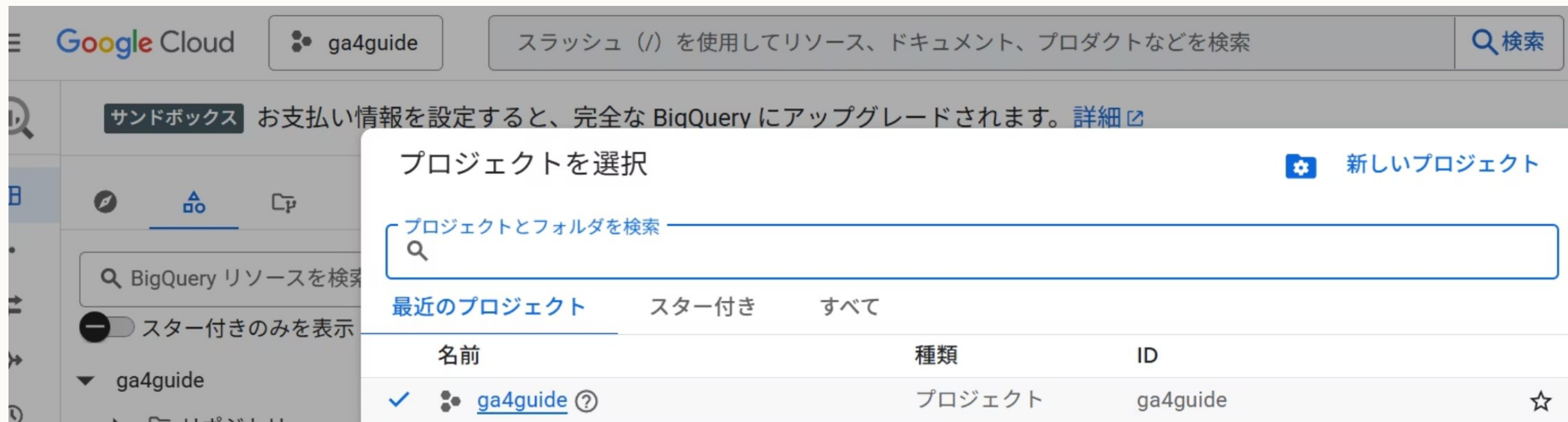
bigquery-key.json
の置き場

次ページ参照

6.Claude Desktop の設定

プロジェクト ID

Google Cloud のプロジェクトボタンを押した時に出てくる一覧の中で今回 GA4 のデータが入っているプロジェクトの ID。
ここでは「ga4guide」

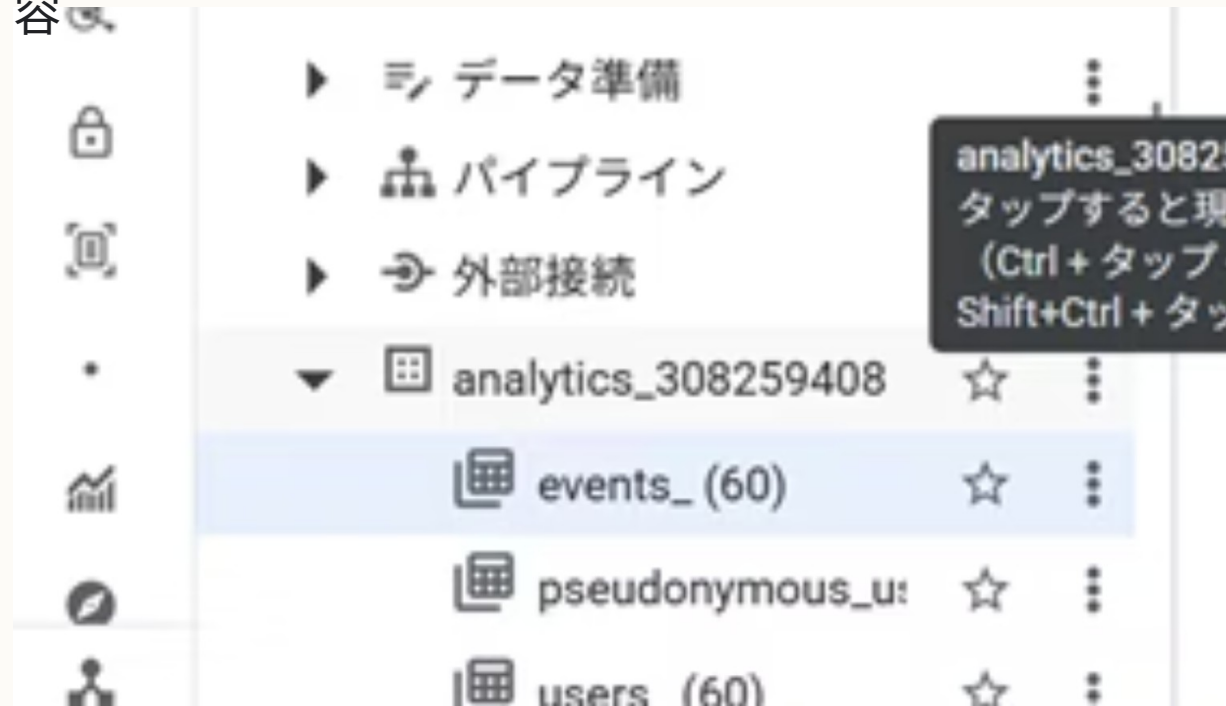


これを前ページのプロジェクト ID と置き換える

6.Claude Desktop の設定

データセット ID

GA4 が含まれているデータセット。手順 3 の最後で確認した内容



これを前ページのデータセット ID と置き換える

6.Claude Desktop の設定

タスクマネージャーでも確認し、残っている場合は、「タスクの終了」を選んでください (Ctrl+Shift+esc で起動。 Claude を右クリックで「タスクの終了」を選択)

タスク マネージャー

検索する名前、発行元、PID...

プロセス

新しいタスクを実行する

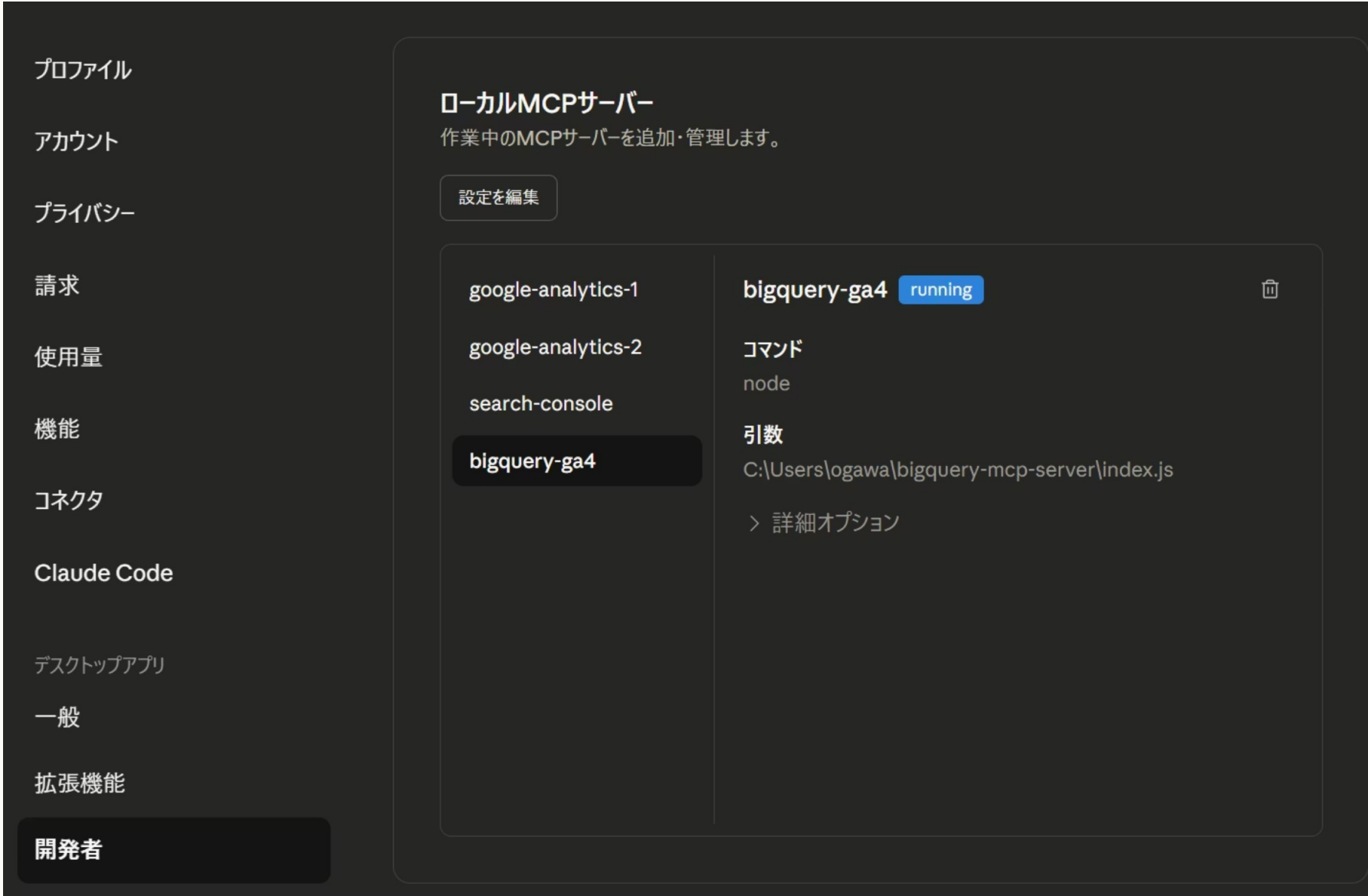
タスクを終了する

効率モード

名前	状態	6% CPU	56% メモリ	1% ディスク	0% ネットワ...
> Claude (14)		1.3%	447.8 MB	0 MB/秒	0 Mbps
> Google Chrome (26)		1.7%	2,494.6 ...	0.1 MB/秒	0 Mbps

6.Claude Desktop の設定

再度 Claude を起動し、設定＞開発者と確認し、 bigquery-ga4 が追加されていれば完了です。



6.Claude Desktop の設定

プロンプトを投げて確認してみましょう

直近1週間の日別セッション数を取得しました！

日別セッション数（2025年10月9日～14日）

日付	セッション数
10月9日（木）	1,035
10月10日（金）	1,038
10月11日（土）	1,204
10月12日（日）	757
10月13日（月）	751
10月14日（火）	1,025

合計: 5,810セッション

10月15日（本日）のデータはまだBigQueryに反映されていないようです。GA4のデータは通常、24～48時間の遅延があります。

週末の10月11日が最も多く、その後の日曜日と月曜日が少なくなっている傾向が見られますね。



再試行

Claudeは間違えることがあります。回答内容を必ずご確認ください。

設定に関する FAQ

よくある質問

エラー 1-1: サービスアカウントが作成できない

✖ エラー症状

「+ サービスアカウントを作成」ボタンがない
ボタンを押しても反応しない
「権限がありません」エラー

🔍 原因

IAM でサービスアカウント作成権限がない
プロジェクトの権限が不足
組織ポリシーで制限されている

✔ 対処方法

【権限確認】
GCP → IAM と管理 → IAM

自分のアカウントに以下のいずれかがあるか確認:

- ✔ 「オーナー」
- ✔ 「編集者」
- ✔ 「Service Account Admin」

【管理者に依頼する場合】

必要なロール:

- Service Account Admin
- Service Account Key Admin

【確認コマンド (Cloud Shell) 】

```
gcloud projects get-iam-policy [PROJECT_ID] \  
--flatten="bindings[].members" \  
--filter="bindings.members:[YOUR_EMAIL]"
```

よくある質問

エラー 1-2: 「BigQuery データ閲覧者」ロールが付与できない

✖ エラー症状

サービスアカウントにロールを追加しようとしてもエラー
「ロールを付与」ボタンが押せない
保存時にエラー

🔍 原因

自分に BigQuery 管理権限がない
サービスアカウントの選択ミス
画面の操作手順ミス

✔ 対処方法

【正しい手順】

1. GCP → IAM と管理 → IAM （※サービスアカウントではない）
2. 「アクセス権を付与」をクリック
3. 「新しいプリンシパル」に以下を入力：
[サービスアカウント名]@[プロジェクト ID].iam.gserviceaccount.com
4. 「ロールを選択」で以下の2つを追加：
 - ✔ BigQuery データ閲覧者
 - ✔ BigQuery ジョブユーザー
5. 「保存」をクリック

【必須ロール2つ】

- BigQuery データ閲覧者：データを読むため
- BigQuery ジョブユーザー：クエリを実行するため

※ 両方ないとデータ取得時にエラーになります

よくある質問

エラー 1-3: ロールを追加しても権限が反映されない

✖ エラー症状

ロール追加は成功したがデータにアクセスできない
「Permission denied」エラーが出続ける

🔍 原因

権限の反映に時間がかかっている（最大 5 分）
データセット単位の権限が不足
プロジェクトが間違っている

✔ 対処方法

【待機時間】

ロール付与後、5 分程度待ってから再試行

【データセットレベルの権限確認】

1. BigQuery → 該当データセットを選択
2. 「共有」→「権限」をクリック
3. サービスアカウントが表示されているか確認

なければ追加：

- プリンシパル：サービスアカウントのメール
- ロール：BigQuery データ閲覧者

【プロジェクト ID 確認】

左上のプロジェクト名が正しいか確認

GA4 データがエクスポートされているプロジェクトか？

よくある質問

エラー 2-1: JSON キーがダウンロードできない

✖ エラー症状

「キーを追加」→「新しい鍵を作成」をクリックしてもファイルが保存されない
ブラウザがブロックしている

🔍 原因

ブラウザのダウンロード設定
ポップアップブロッカー
セキュリティソフトの制限

✔ 対処方法

【Chrome の場合】

1. 設定 → プライバシーとセキュリティ → サイトの設定
2. 「追加のコンテンツ設定」→「自動ダウンロード」
3. console.cloud.google.com を許可

【ダウンロード確認】

Ctrl + J でダウンロード履歴を開く
→ 最新のJSON ファイルを確認

【再ダウンロード】

1. サービスアカウントの詳細画面を開く
2. 「キー」タブ
3. 既存のキーの右側「⋮」→「キーをダウンロード」

【ファイル名】

ダウンロードされるファイル名：
[プロジェクト名]-[ランダム文字列].json

これを「bigquery-key.json」にリネーム

よくある質問

エラー 2-2: JSON ファイルの保存場所が分からない

✖ エラー症状

ダウンロードしたはずだが見つからない
どこに保存したか不明

🔍 原因

デフォルトのダウンロードフォルダに保存されている
ファイル名を忘れた

✔ 対処方法

【Windows 検索】

1. エクスプローラーを開く
2. ダウンロードフォルダで検索 :
*.json
3. 更新日時で並び替え → 最新のものを確認

【推奨保存場所】

C:\Users\[ユーザー名]\bigquery-mcp-server\
フォルダを作成して、そこに移動

ファイル名 : bigquery-key.json

よくある質問

エラー 3-1: データセット「analytics_XXXXXXXX」が見つからない

✖ エラー症状

BigQuery を開いても GA4 のデータセットが表示されない。プロジェクトにデータがない

🔍 原因

最頻出: GA4 から BigQuery へのエクスポート設定がされていない

別のプロジェクトにエクスポートされている

エクスポート設定は完了しているがデータがまだない（初日は遅延あり）

✅ 対処方法

【GA4 でエクスポート設定確認】

1. GA4 の「管理」を開く
2. プロパティ列の「BigQuery のリンク」をクリック
3. リンクがあるか確認

【エクスポート設定がない場合】

1. 「リンクを作成」をクリック
2. GCP プロジェクトを選択
3. エクスポート設定:
 - ✅ 毎日のエクスポート（推奨） ✅ ストリーミング エクスポート（リアルタイム・有料）

4. 保存後、翌日までデータ反映待ち

【プロジェクト確認】

BigQuery の左上でプロジェクトを切り替えて確認 GA4 でリンクしたプロジェクトと一致しているか？

【データセット名の形式】

analytics_[プロパティ ID]

例: analytics_123456789

※ プロパティ ID は GA4 の管理→プロパティの詳細で確認

よくある質問

エラー 3-2: テーブル名が分からない

✖ エラー症状

データセットは見つかったがテーブルがたくさんある
どのテーブルを使えばいいか不明

🔍 原因

GA4 の BigQuery エクスポートは日別にテーブルを作成
テーブル命名規則を理解していない

✔ 対処方法

【テーブル命名規則】
events_YYYYMMDD ← 確定データ（翌日作成）
events_intraday_YYYYMMDD ← 当日データ（リアルタイム）

【例】
2025 年 10 月 17 日のデータ：
- events_20251017 ← これを使う（推奨）
- events_intraday_20251017 ← 当日のみ

【推奨の使い方】
スクリプト内でワイルドカードを使用：
`analytics_123456789.events_*

これで全期間のデータにアクセス可能

よくある質問

エラー 4-1: npm install でエラー「EACCES: permission denied」

✖ エラー症状

```
npm ERR! code EACCES
npm ERR! syscall mkdir
npm ERR! path C:\...\node_modules
npm ERR! errno -4048
npm ERR! Error: EACCES: permission denied
```

🔍 原因

管理者権限が不足
フォルダへの書き込み権限がない
ウイルス対策ソフトがブロック

✔ 対処方法

【管理者権限で実行】

1. スタートメニューで「cmd」と入力
2. 「コマンドプロンプト」を右クリック
3. 「管理者として実行」

4. フォルダに移動：

```
> cd C:\Users\[ユーザー名]\bigquery-mcp-server
```

5. 再度インストール：

```
> npm install @google-cloud/bigquery @modelcontextprotocol/sdk
```

【フォルダ権限確認】

1. フォルダを右クリック → プロパティ
2. セキュリティタブ
3. 自分のユーザーに「フルコントロール」があるか確認

【ウイルス対策ソフト】

一時的に除外リストに追加：

```
C:\Users\[ユーザー名]\bigquery-mcp-server
```

よくある質問

エラー 4-2: 「Cannot find module」エラー

✖ エラー症状

Error: Cannot find module '@google-cloud/bigquery'

Error: Cannot find module '@modelcontextprotocol/sdk'

🔍 原因

npm install が失敗している

node_modules フォルダがない

実行場所が間違っている

✅ 対処方法

【node_modules 確認】

> dir

以下があるか確認：

- node_modules フォルダ

- package.json

- package-lock.json

【再インストール】

> npm install @google-cloud/bigquery @modelcontextprotocol/sdk

【実行場所確認】

> cd

現在のパスを表示

C:\Users\[ユーザー名]\bigquery-mcp-server

になっているか確認

よくある質問

エラー 5-1: 「SyntaxError: Cannot use import statement」

✖ エラー症状

SyntaxError: Cannot use import statement outside a module
at Object.compileFunction (node:vm:360:18)

🔍 原因

最頻出: package.json に "type": "module" がない
Node.js が CommonJS 形式で実行している
Node.js のバージョンが古い (v14 以下)

✔ 対処方法

【package.json の修正】

ファイルをメモ帳で開いて以下を追加:

```
{  
  "type": "module", ← この行を追加  
  "dependencies": {  
    "@google-cloud/bigquery": "^7.0.0",  
    "@modelcontextprotocol/sdk": "^1.0.0"  
  }  
}
```

保存後、再実行:

> node index.js

【Node.js バージョン確認】

> node --version

v18.0.0 以上が必要

古い場合: <https://nodejs.org/> から最新 LTS 版をインストール

【代替方法 (非推奨)】

ファイル名を index.mjs に変更

> node index.mjs

よくある質問

エラー 5-2: 認証エラー「 **Could not load the default credentials** 」

✖ エラー症状

Error: Could not load the default credentials

Unable to detect a Project Id in the current environment

🔍 原因

超重要: 環境変数 `GOOGLE_APPLICATION_CREDENTIALS` が設定されていない

JSON キーファイルのパスが間違っている

ファイル名が違う

よくある質問

エラー 5-2: 認証エラー「 Could not load the default credentials 」

✓ 対処方法

【環境変数設定（Windows）】

1. システムのプロパティ → 環境変数
2. 「ユーザー環境変数」で「新規」
3. 変数名: GOOGLE_APPLICATION_CREDENTIALS
変数値: C:\Users\[ユーザー名]\bigquery-mcp-server\bigquery-key.json
4. OK → OK
5. コマンドプロンプトを再起動

【確認】

```
> echo %GOOGLE_APPLICATION_CREDENTIALS%  
C:\Users\...\bigquery-key.json と表示されれば OK
```

【パス確認】

エクスプローラーでファイルを開く
→ アドレスバーをコピー
→ ファイル名を追加

【 Claude Desktop 設定の場合 】

claude_desktop_config.json に直接記載:

```
"env": {  
  "GOOGLE_APPLICATION_CREDENTIALS": "C:\\Users\\...\\bigquery-key.json",  
  "GCP_PROJECT_ID": "your-project-id",  
  "GA4_DATASET_ID": "analytics_123456789"  
}
```

※ バックスラッシュ 2 つ (\\) でエスケープ

よくある質問

エラー 5-3: 「BigQuery: Permission denied」

✖ エラー症状

Error: Permission denied on dataset/table
Caller does not have permission

🔍 原因

サービスアカウントに必要な権限がない
データセット / テーブルへのアクセス権限不足
プロジェクト ID が間違っている

✔ 対処方法

【権限再確認】

GCP → IAM と管理 → IAM

サービスアカウントに以下の両方があるか確認:

✔ BigQuery データ閲覧者

✔ BigQuery ジョブユーザー

【データセット権限確認】

BigQuery → データセットを選択

→ 「共有」 → 「権限」

サービスアカウントが表示されているか?

【プロジェクト ID 確認】

環境変数 GCP_PROJECT_ID が正しいか確認

```
> echo %GCP_PROJECT_ID%
```

GCP の左上に表示されているプロジェクト ID と一致しているか?

【API 有効化確認】

GCP → API とサービス → 有効な API

→ BigQuery API が有効になっているか確認

よくある質問

エラー 5-4: 「Table not found」エラー

✖ エラー症状

Error: Not found: Table project:dataset.table

🔍 原因

データセット名やテーブル名が間違っている
環境変数の設定ミス
データがまだ存在しない

✅ 対処方法

【環境変数確認】

```
> echo %GA4_DATASET_ID%
```

analytics_123456789 のように表示されるか確認

【BigQuery で直接確認】

BigQuery コンソールで以下のクエリを実行：

```
SELECT table_name
FROM `project-id.analytics_123456789.INFORMATION_SCHEMA.TABLES`
WHERE table_name LIKE 'events_%'
LIMIT 10;
```

テーブルが表示されるか？

【データセット名の確認】

正しい形式：

✅ analytics_123456789 ← プロパティ ID の数字

✖ ga4_data

✖ google_analytics_4

【データ存在確認】

GA4 の BigQuery エクスポートは：

- 毎日のエクスポート：翌日午前中に反映
- ストリーミング：リアルタイム（有料）

新規設定の場合は翌日まで待つ必要あり

よくある質問

エラー 6-1: `claude_desktop_config.json` の編集エラー

✖ エラー症状

保存後、Claude Desktop が起動しない
「設定ファイルにエラーがあります」と表示

🔍 原因

JSON 構文エラー（GA4 ・サーチコンソールと同じ）
パスのエスケープミス
環境変数の記載ミス

よくある質問

エラー 6-1: `claude_desktop_config.json` の編集エラー

✔ 対処方法

【正しい設定例】

```
{
  "mcpServers": {
    "google-analytics": {
      "command": "npx",
      "args": ["-y", "mcp-server-google-analytics"],
      "env": { ... }
    },
    "search-console": {
      "command": "node",
      "args": ["C:\\search-console-mcp\\server.js"],
      "env": {}
    },
    "bigquery-ga4": {
      "command": "node",
      "args": ["C:\\Users\\YourName\\bigquery-mcp-server\\index.js"],
      "env": {
        "GOOGLE_APPLICATION_CREDENTIALS": "C:\\Users\\YourName\\bigquery-mcp-server\\bigquery-key.json",
        "GCP_PROJECT_ID": "your-project-id",
        "GA4_DATASET_ID": "analytics_123456789"
      }
    }
  }
}
```

【重要ポイント】

1. Windows パスは \ (バックスラッシュ 2 つ)

× C:\Users\...

✔ C:\\Users\\...

2. カンマの位置

各サーバー定義の間にカンマ必要

最後のサーバーの後はカンマ不要

3. env の値はすべてダブルクォートで囲む

【検証】

<https://jsonlint.com/> に貼り付けて構文チェック

よくある質問

エラー 6-2: 「bigquery-ga4」がコネクタに表示されない

✖ エラー症状

Claude Desktop を再起動しても
「コネクタ」に bigquery-ga4 が表示されない

🔍 原因

index.js のパスが間違っている
index.js が正常に実行できない
環境変数が読み込めていない
プロセスが残っている

よくある質問

エラー 6-2: 「bigquery-ga4」がコネクタに表示されない

✓ 対処方法

【パス確認】

1. エクスプローラーで index.js を開く
2. アドレスバーをクリックしてパスをコピー
3. index.js を追加

例 : C:\Users\ogawa\bigquery-mcp-server\index.js

4. claude_desktop_config.json に記載する際は :
C:\\Users\\ogawa\\bigquery-mcp-server\\index.js
(バックスラッシュ 2 つ)

【index.js の動作確認】

コマンドプロンプトで単独実行 :

```
> cd C:\Users\ogawa\bigquery-mcp-server
> set GOOGLE_APPLICATION_CREDENTIALS=C:\Users\ogawa\bigquery-mcp-server\bigquery-key.json
> set GCP_PROJECT_ID=your-project-id
> set GA4_DATASET_ID=analytics_123456789
> node index.js
```

エラーが出ないか確認

【プロセス完全終了】

タスクマネージャー (Ctrl+Shift+Esc) で :

- "Claude" プロセスを全て終了
- "node" プロセスも全て終了

再起動

【環境変数の確認】

index.js に以下を追加してデバッグ :

```
console.log('GOOGLE_APPLICATION_CREDENTIALS:', process.env.GOOGLE_APPLICATION_CREDENTIALS);
console.log('GCP_PROJECT_ID:', process.env.GCP_PROJECT_ID);
console.log('GA4_DATASET_ID:', process.env.GA4_DATASET_ID);
```

よくある質問

エラー 7-1: プロンプトに反応しない・データが取得できない

✖ エラー症状

プロンプトを入力しても無反応

「データが見つかりません」と返る

「エラーが発生しました」と表示

🔍 原因

環境変数の設定ミス

BigQuery への接続失敗

データセット / テーブルの指定ミス

期間指定のミス

よくある質問

エラー 7-1: プロンプトに反応しない・データが取得できない

✓ 対処方法

【簡単なプロンプトでテスト】

「BigQuery に接続できているか確認して」

「GA4_DATASET_ID の値を教えて」

これで環境変数が読めているか確認

【データ存在確認】

BigQuery コンソールで直接クエリ：

```
SELECT COUNT(*) as event_count
FROM `your-project.analytics_123456789.events_*`
WHERE _TABLE_SUFFIX = FORMAT_DATE('%Y%m%d', DATE_SUB(CURRENT_DATE(), INTERVAL 1 DAY));
```

データが返ってくるか確認

【プロンプト調整】

× 「今日のデータ」（まだない可能性）

× 「リアルタイムデータ」（ストリーミング有効化必要）

✓ 「昨日のデータ」

✓ 「過去 7 日間のデータ」

✓ 「先月のデータ」

【環境変数の再設定】

1. claude_desktop_config.json を再確認

2. 全ての env 値が正しいか確認

3. プロジェクト ID は数字ではなくプロジェクト名

例: "my-ga4-project-123456"