



• • • Seminar

Claude Code × GAS × clasp 超入門！

非エンジニアでもかんたん「備品レンタルアプリ」を作ってみよう

もうコピペしない。AI が GAS を直接書く「clasp」入門

2026 年 9 月 11 日（金）21:00 | オンライン（Zoom）

講師: 久野 晃平（くのーる） | 株式会社フロント・ワークス

チャット・リアクション、大歓迎です

セミナー中のチャットとリアクションは、いつでも歓迎です！

- 質問は、ぜひ気軽に書いてください
- 「わかった」「早い」「もう一度」といったコメントも OK
- あとで再現できるように、アーカイブ動画付きのチケットをご用意しています

わかった

早い

もう一度





PROFILE

久野 晃平

くの こうへい

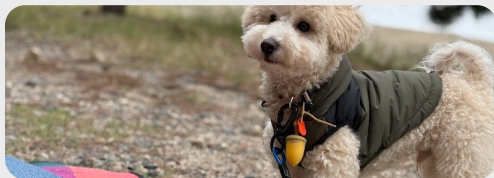
株式会社フロント・ワークス
代表取締役

front-works.co.jp

X @Knorr_CS

HOBBY

愛犬と遊ぶ・キャンプ・車



事業内容

AI 導入支援

Claude Code 専門

業務用システム受託開発

Google Cloud、Google Apps Script (GAS)、Data Studio (旧: Looker Studio)、AppSheet

情報セキュリティ対策支援

BPO サービス

コールセンター、バックオフィス業務全般、映像検証 / QC

私について

● コールセンターのスタッフからキャリアをスタート

Windows 用アプリケーションのテクニカル サポートとして、お客さまの「わからない」を 15 万件以上対応。1 万時間以上「人に教える仕事」に向きあってきました

● 前職: 取締役 CIO (情報システム担当)

ISMS (JIS Q 27001) 取得を主導 / 内部監査責任者。社内エンジニアとして GWS 導入・社内システム内製開発など

● 現在: 経営をはじめ、PM・開発・セールスを担当

多数の PM・システム納品実績あり

じつは・・・ 5 年ほど前までは**非エンジニア**でした

だからこそ「非エンジニアのつまずき」がわかります。今日はそこに寄り添って進めます。

主な開発実績

農家向け 注文受注システム

美容業界向け タレントマネジメント

建設業向け 案件管理・測量記録管理

エンタメ業界向け 配信映像管理システム

インサイドセールス向け 営業管理システム



自社でも、Claude Code で開発しています

100+

自社で開発してきた
GAS プロジェクト

今日お見せする方法は、私たちが毎日の仕事で使っている方法そのものです。

- 自社で百を超える GAS プロジェクトを開発してきました
- Claude Code をつかって、社内内製のシステムも開発をおこなっています
- 今日の「くのー式 AI GAS コーディングキット」は、その開発現場で生まれたノウハウを Claude Code 用に凝縮したものです



有料チケット特典

有料チケットには豪華 3 大特典

01

動画アーカイブ

あとからゆっくり、お手元で復習できます

02

くのーる式 AI GAS コーディングキット（Claude Code 用）

フォルダに置くだけで、Claude Code が「GAS を知り尽くした、安全第一の開発パートナー」に変わります

03

30 分 1on1 相談

1on1 付きチケット限定

Claude Code ・ GAS ・ 業務課題 ・ 生成 AI 活用まで、個別にご相談いただけます



最初に、3つの質問

チャットで教えてください！

1

Claude Code を使ったことがある方

2

GAS (Google Apps Script) を使ったことがある方

3

clasp を使ったことがある方



本日の流れ

90 分で、clasp を軸に「作れる」から「安心して使える」までお届けします。

章	内容	時間
0	オープニング	5 分
1	目的とゴール。完成アプリを、先にお見せします	5 分
2	Claude Code × GAS の基礎知識	12 分
3	今日の主役「clasp」。AI の新しい「手」	15 分
4	くの一の式 AI GAS コーディングキット	5 分
5	【ライブ】 備品レンタルアプリを作る	30 分
6	まとめ・特典のご案内・サービス紹介	11 分
—	Q&A	7 分



01

目的とゴール



今日の目的とゴール

非エンジニアでも、AI に GAS を「直接」書かせて、業務に便利なアプリを安心して作れるようになること。

- 「AI がコードを書く」はもう当たり前。今日の焦点は **clasp**。GAS のコピー作業から解放されます
- スプレッドシートを台帳に、GAS で Web アプリを公開するまでの全体像が分かる
- Claude Code への実際の指示文を、そのまま見られる
- 「作れる」の一步先、「安心して使える」ための作法とセキュリティが分かる

「あ、これ自分の仕事にも使えるかも」というヒントを、ひとつ持ち帰ってください。



完成イメージ「備品レンタルアプリ」



今日作るのは、社内の備品の「借りる・返す」をひと押しで記録できるアプリです。

- スマホでもパソコンでも、ブラウザで開けます
- 管理台帳は、おなじみの Google スプレッドシート。いつ・誰が・何を借りて、いつ返したかが自動で残ります
- 画面は GAS で作るので、サーバーの契約は不要。ランニング費用はゼロ円

使い方は、たった 3 ステップ

1 アプリを開く

2 借りたい備品を選び、名前と返却予定日を入れて「借りる」

3 返すときは「返す」を押す



02

Claude Code × GAS の基礎知識



Google Apps Script (GAS) とは

GAS は、Google が提供する「Google のサービスを自動で動かすためのプログラム環境」です。

- スプレッドシート・ドライブ・Gmail・カレンダーなどを、プログラムから操作できます
- Google アカウントがあれば、追加の契約なしで使い始められます
- 自動化だけでなく、ブラウザで開ける Web アプリの画面も作れます。今日使うのはこの機能です

スプレッドシート

ドライブ

Gmail

カレンダー

Web アプリ



Claude Code とは

Claude Code は、Anthropic 社が提供する、あなたのパソコンの中で「働く」AI エージェントです。

- 日本語で「〇〇を作りたい」と話しかけると、ファイルを読み、コードを書き、コマンドを実行します
- 動かし方は 4 つ (Web / デスクトップ アプリ / VS Code 拡張 / ターミナル)。今日はもっともパワフルなターミナル版を、VS Code と組み合わせて使います
- 見た目は難しそうですが、ふだんお使いの AI と同じく、日本語の対話で動きます

Web

デスクトップ アプリ

VS Code 拡張

ターミナル (今日はこれ)



「Claude」と「Claude Code」の違い

同じ AI モデルでも、製品が違えばできることが違います。

	Claude（チャット）	Claude Code
主な使い方	質問に答える・文章を書く	あなたの代わりに作業する
動く場所	ブラウザの中	あなたのパソコンの中
コードの扱い	書いて見せる。貼るのは人	書いて、保存して、実行する

- ブラウザのチャット型 AI（Claude・ChatGPT・Gemini）でも GAS のコードは書けますが、人間によるコピペが必要です
- 今日の話は右側。「AIの手」がどこまで届くのかを見ていただきます



Web アプリは「3 つの層」でできている

どんな Web アプリも 3 つの層の組み合わせです。これを自分で用意すると、考えることが山ほど出てきます。

1

ユーザーが操作する画面

プレゼンテーション層

2

業務の処理をおこなう部分

アプリケーション層

3

データを保管する部分

データ層。データベース

- 層ごとに設計・開発・運用が必要で、非エンジニアが取り組むには現実的ではありません
- とくに怖いのがセキュリティ対策。専門家が作ったシステムでも事故は起き、責任は作った人に及びます



GAS なら、3 つの層を Google が肩代わりする

GAS の Web アプリ作成機能を使うと、3 層をぜんぶ Google が持ってくれます。これが「サーバーレス」です。

1 画面

GAS が配信する

2 処理

GAS が実行する

3 データ

いつものスプレッドシートが、
そのままデータベースになる

- 「サーバーレス」は、サーバーが「無い」のではなく「自分で用意・管理しなくてよい」という意味。サーバーは Google が動かし、守っています
- だから契約も、月額費用も、保守も要りません。「ゼロ円」の理由はここにあり、非エンジニアにピッタリです



Google が守ってくれるのは「建物」まで

サーバーの安全は Google に任せられます。ただし「部屋の鍵のかけ方」は作る人の責任です。

Google が守る

建物

サーバー本体の安全。世界最高水準の情報セキュリティを持つ Google が守ります

作る人が決める

部屋の鍵

公開範囲の設定と、コードの中身。ここは作った人が決めます

4 章で、その鍵のかけ方を最初から AI に守らせる道具をご紹介します



まとめ

GAS が「サーバー」の壁を消し、残った「コーディング」の壁は Claude Code が引き受けます。

- **GAS:** Google のサービスを動かすプログラム環境。Web アプリも作れて、サーバー不要・ゼロ円
- **Claude Code:** パソコンの中で働く AI エージェント。日本語で話しかけると、書いて・保存して・実行する
- 鍵のかけ方（公開範囲・コードの中身）だけは、作る人の責任として残る

ただし、じつはこれだけでは足りません。3 章に続きます。



03

今日の主役「clasp」



Claude Code だけでは、まだ足りません

Claude Code の手が届くのは、パソコンの中だけです。



- Claude Code はパソコンの中のファイルを読み書きし、コマンドを実行できます
- いっぽう GAS のコードが動く場所は、Google のクラウドの上。まったくの別世界です
- だから AI が書いたコードを、結局は人がコピーして貼り付けてきました。直すたびに、もう一度



だから、これまでは「コピペ」だった

人間が「AIの手」の代わりにしてきました。

1

Claude Code がパソコン
の中にコードを書く

AI

2

人がそれをコピーする

人

3

ブラウザで GAS のエディ
タを開いて、貼り付けて、
保存する

人

4

直したいところがあれば、1に戻る

もう一度

直すたびに、コピーして・開いて・貼って・保存する



コピペ方式の限界

これは AI が賢くなれば解決する種類の問題ではありません。

手間

直すたびに、コピーして・開いて・貼って・保存する

ずれ

人が GAS 側で手直しすると、AI の持っているコードと食い違う。
AI は最新を知らずに次を書く

事故

貼り間違い・貼り忘れ・古いコードの上書き

届かない手をどれだけ賢くしても、効果はありません



clasp（クラスプ）とは

clasp は、Google が公開している、GAS をパソコンのコマンドから操作するための公式ツールです。

- 正式名は `@google/clasp` 。Node.js 上で動くコマンドライン ツール
- 2026 年 9 月時点の最新は v3 系
- 入れると、パソコンの中のファイルと Google 上の GAS プロジェクトを「同期」できるようになります

clasp

Google 公式

Node.js

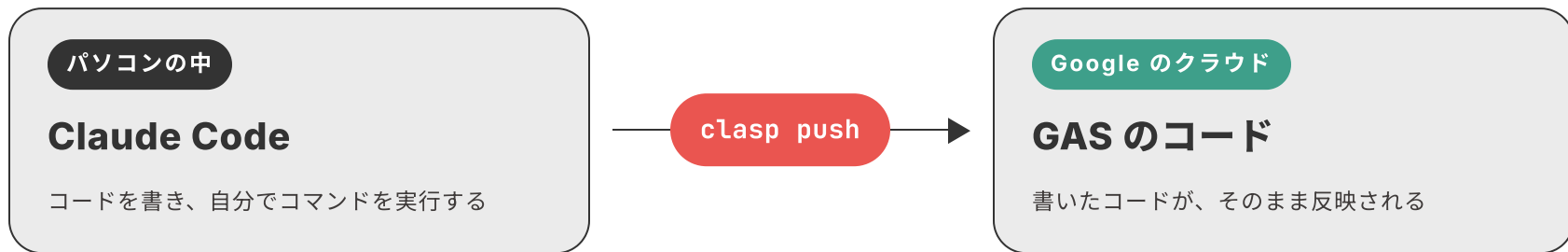
v3 系

パソコン ⇄ Google 上の GAS
ファイルを「同期」する



clasp で、AI の手が Google まで届く

Claude Code はコマンドを実行できる。clasp はコマンドで GAS を動かせる。つなぐと、手が届きます。



- 1 Claude Code がパソコンの中にコードを書く
- 2 Claude Code が自分で `clasp push` を実行する
- 3 コードが Google 上の GAS に反映される

コピーする人が、この流れから消えます。



何が変わるか

「AI が書く → 人が運ぶ」から、「AI が書いて、AI が運ぶ」へ。

	コピペ方式	clasp 方式
反映する人	人間	Claude Code
直すたびの手間	毎回コピペ	「直して」と言うだけ
AI が知っている状態	貼った後はずれていく	パソコンの中が常に正



clasp v3 のコマンド一覧

全部で 28 個。役割ごとに並べると、こうなります。

役割	コマンド	何をするか
アカウント	login / logout / show-authorized-user	Google アカウントでログイン・ログアウト・確認
プロジェクト	create-script / clone-script / delete-script / list-scripts	GAS プロジェクトを新規作成・既存に接続・削除・一覧
同期	push / pull / show-file-status	パソコン→Google へ反映・Google→パソコンへ取得・反映対象の確認
開く	open-script / open-web-app / open-container / open-credentials-setup	エディタ・Web アプリ・親ファイル・認証設定画面をブラウザで開く
デプロイ	create-deployment / update-deployment / delete-deployment / list-deployments	公開の作成・更新・削除・一覧
バージョン	create-version / list-versions	変更履歴の固定と一覧
ログ・実行	setup-logs / tail-logs / run-function	ログの設定・表示、関数のリモート実行
API	list-apis / enable-api / disable-api / open-api-console	Google API の一覧・有効化・無効化・コンソールを開く
実験的	mcp	AI エージェント接続用モード



でも、ふだん使うのは 2 つだけ

26 個を覚える必要はありません。開発で使うのはこの 2 つ。しかも打つのは Claude Code です。

接続する 最初に 1 回

```
clasp clone-script <URL>
```

Google 上の GAS プロジェクトとパソコンのフォルダをつなぐ

置く 直すたびに

```
clasp push
```

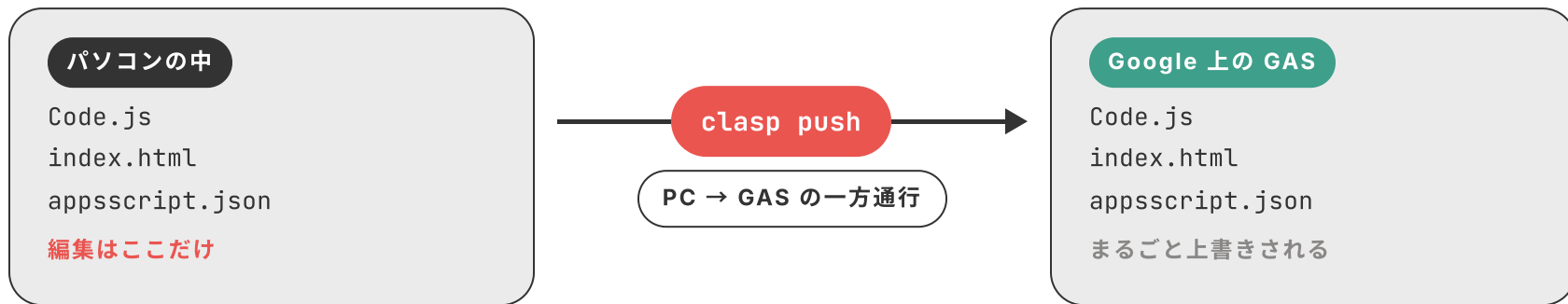
パソコンで書いたコードを Google へ反映する

- 「反映して」と言えば、Claude Code が `clasp push` を実行します。人がやるのは、最初に接続先（GAS プロジェクトの URL）を渡すことだけ
- パソコンの中がツネに正なので、`pull`（Google 側から取ってくる）は基本的に使いません
- コマンド一覧は「AI が何をしているか」を眺めて分かるための地図です



push は「まるごと入れ替え」

`clasp push` は、パソコンの中のファイルで Google 側を上書きします。



- PC → GAS への一方通行と考えてください
- 編集はパソコンの中だけでおこない、反映は push に統一します
- push のときにサーバー側で文法チェックが走るので、成功すれば最低限の検証になります



デプロイ（アプリの公開）だけは、あえて人がやります

「誰に公開するか」の判断は、AI に渡さない

- clasp にもデプロイのコマンドはありますが、公開範囲（自分のみ／組織内／全員）はコマンドの引数では決められません
- そこでデプロイ操作自体は GAS の画面でおこないます
- 初回は必ず「自分のみ」で公開し、自分で動作を確かめてから共有へ進みます

アクセスできるユーザー

自分のみ

初回は必ずここ

組織内

全員

GAS の画面で、人が選ぶ



まとめ

clasp は、AI の手を Google まで届けさせます

- PC から Google に手を届かせる道具が clasp
- 覚えるのは「接続する」「置く」の 2 つ。しかも打つのは Claude Code
- push はまるごと入れ替え。編集はパソコンの中だけに統一する
- 公開範囲だけは人が決める。初回は「自分のみ」

```
clasp clone-script <URL>
```

```
clasp push
```



04

くの一 式 AI GAS コーディングキット



GAS には、AI が知らない「クセ」がある

AI は世の中の一般的なプログラムを学んでいて、GAS とスプレッドシート特有の「クセ」を知りません。知らずに書くと「動かない」「想定の問題」が起きるリスクがあります。

「クセ」を知らずに書くこと

キットの決めごと

それらしいコードなのに動かない。部品の読み込みや外部との通信など、ふつうの JavaScript の書き方が GAS には無い

GAS 流の正解を一覧にして、最初から教えておく

電話番号や社員番号の先頭の「0」が消える。シートは数字に見える文字を勝手に数値にする

計算しない数字は「文字」として扱い、列を「書式なしテキスト」にする

記録した日時がずれる。シートとスクリプトは別々にタイムゾーンの設定を持ち、食い違ったまま書いた日時はあとで直しても戻らない

最初に設定が一致しているか確かめ、違えば止まる

「=」で始まる入力計算式として動いてしまう。シートは「=」始まりを数式と読む

書き込む直前に先頭へ「'」を付けて、ただの文字にする

件数が増えると途中で止まる。1 回の実行は 6 分まで

セルを 1 つずつ読み書きせず、まとめて処理する

くの一 式 AI GAS コーディングキット

その落とし穴をまとめて防ぐために作ったのが「くの一 式 AI GAS コーディングキット」です。



- テキストファイルのセットをフォルダに置くだけ
- 今日のライブコーディングは、このキットを使って進めます
- キットの中身は、最後に改めてご紹介します



「○○を作りたい」から始まる 6 フェーズ

Claude Code が、先走らずに一步步進めます。各フェーズの終わりに、あなたの合意を取ります。

1 ヒアリング

コーチングの姿勢で、作りたいものと困りごとを引き出す

2 要件定義シート

非エンジニアが読める言葉だけで合意文書を作る

3 テーブル設計

もっとも時間をかけてよい、成否を決めるフェーズ

4 接続確認

clasp

clasp で Google とつなぎ、シートが読めることを確かめる

5 実装とテスト

小さく push し、「できたら完成」の項目を一つずつ確認する

6 リリース

デプロイ手順を案内し、使い方・「困ったときは」・担当が代わるときの手順を残す



05

【ライブ】 備品レンタルアプリを作る



「会議室の鍵、いま誰が持ってるんだっけ？」

備品の貸し借りが「みんなの記憶」だけで回っている職場は、少なくありません。

会議室の鍵を借りたいのに、いま誰が持っているのか分からない

あのノート PC、そういえば返ってきていないような.....

貸出簿はあるけれど、書く人と書かない人がいる

今日はこの困りごとをなくすアプリを、その場でゼロから作ります。



このパートで、やること・やらないこと

30 分で「最初の一言」から「自分だけに公開」まで通します。

やること

- Claude Code に日本語で話しかけ、キットの 6 フェーズを進める
- コピペを一度もしないまま、clasp で Google に反映する
- 完成した Web アプリを開いて、借りる・返すを実際に動かす
- 動かない場面が出たら、事前に動作確認した完成品に切り替えて先へ進む

やらないこと

- Git・GitHub などのバージョン管理、コードエディタの使い方、コマンドの細かい解説
- 組織外への公開



事前準備

1 Node.js の最新 LTS (v24 以上) を入れる

2 clasp (v3 系) を入れる

```
npm install -g @google/clasp
```

3 Apps Script API を有効にする

```
script.google.com/home/usersettings
```

4 Google アカウントでログインする

```
clasp login
```

5 Claude Code を入れて、初回ログインする

- Windows の場合は Git for Windows も入れます。Mac は不要です
- 今日はここまで済ませた状態から始めます



最初の一言

キットのフォルダで Claude Code を起動し、こう話しかけます。

> 社内の備品の貸し出しを管理したいです

- プロンプトの工夫はしません。この一言から、Claude Code がヒアリングを始めます
- ここからの指示文は、すべて画面のまま見せます



フェーズ 1~2 ヒアリングと要件定義シート

Claude Code が、まず 4 つの「決めごと」を聞いてきます。今日はこう答えます。

決めごと	今日の答え
画面が必要か	必要。スマホでも使う Web アプリ
誰が使うか	まず自分だけ。動いたら同じ組織内へ
定期実行が必要か	不要
シートは既存か新規か	新規に作る

- 合意すると要件定義シート（`REQUIREMENTS.md`）が作られ、進行状況のチェックリストが残ります



フェーズ 3 テーブル設計

もっとも時間をかけてよいフェーズです。台帳の形がアプリの出来を決めます。

- 備品の一覧（マスタ）と、貸し借りの記録（トランザクション）を分けます
- シート名の頭に種別を付けます。M_ = マスタ、T_ = トランザクション
- 見出し 1 行・A1 起点・結合セルなし。スクリプトが読めるテーブルの形にします
- 列ごとに「文字・数値・日付」の型も決めます。備品番号のような計算しない数字は「文字」にします（先頭の「0」が消える事故を防ぐ）

実際のシート名・列構成 **要確認**

■ 備品レンタル台帳					
	A	B	C	D	E
1	備品番号	備品名	分類	保管場所	状態
2	PC-001	ノート PC A	PC	2F 倉庫	貸出可
3	PC-002	ノート PC B	PC	2F 倉庫	貸出中
4	PJ-001	プロジェクター	AV 機器	会議室 A	貸出可
5	CM-001	ビデオカメラ	AV 機器	2F 倉庫	貸出可
6	MB-001	モバイルルーター	通信	総務デスク	貸出可
M_備品 T_貸出					

事前に作った完成品で確定後に記載



フェーズ 4 接続確認。ここで clasp が登場します

Google ドライブにスプレッドシートと Apps Script を作り、URL を Claude Code に渡します。

1

ドライブで、台帳のスプレッドシートと、スタンドアロンの Apps Script を作る（共有設定は「制限付き」のまま）

人

2

両方の URL を Claude Code に渡す

人

3

Claude Code が `clasp clone-script <URL>` を実行し、パソコンのフォルダと Google をつなぐ

Claude Code

4

疎通確認用のコードを push し、GAS エディタで 1 回実行して権限を承認する

Claude Code

人

- 人がやるのは 1 と 2 と 4 の実行ボタンだけです



フェーズ 5 実装、そして push

ここが今日いちばん見ていただきたい瞬間です。

- Claude Code が画面・処理・テストのコードを書き、自分で `clasp push` を実行します
- ブラウザで GAS エディタを開くと、コードがもう入っています。
誰もコピペしていません
- 「〇〇を直して」と言えば、また書いて、また push します

実装する機能の一覧 **要確認**

事前に作った完成品で確定後に記載



フェーズ 6 デプロイ。ここは「人」

公開範囲を決めるのは人の仕事です。

1 GAS エディタで「デプロイ」→「新しいデプロイ」

2 種類は「ウェブアプリ」。実行ユーザーは「自分」

3 アクセスできるユーザーは、必ず「**自分のみ**」

4 発行された URL をスマホで開く

アクセスできるユーザー

自分のみ

組織内の全員

動作を確かめたあとで広げる

- 動作を自分で確かめたあとで、「組織内の全員」へ広げます。この選択肢があるのは Google Workspace のアカウントだけです



完成。備品レンタルアプリ

スマホで開いて、借りて、返します。台帳のスプレッドシートに記録が残ります。



■ 備品レンタル台帳							
	A	B	C	D	E	F	G
1	貸出 ID	備品番号	備品名	借りた人	貸出日時	返却予定日	返却日時
2	R-0004	PJ-001	プロジェクター	佐藤 美和	09/09 13:05	09/09	09/09 17:22
3	R-0005	MB-001	モバイルルーター	田中 亮	09/10 08:44	09/11	09/10 19:03
4	R-0006	CM-001	ビデオカメラ	鈴木 健	09/10 15:30	09/12	09/11 09:58
5	R-0007	PC-002	ノート PC B	久野 晃平	09/11 10:24	09/18	貸出中
M_備品 T_貸出							

借りる

名前と返却予定日 → 台帳に 1 行増える

返す

返却日が入る



今日、目の前で起きたこと

日本語の一言から、公開まで。コピペは一度もしませんでした。

- 1 「社内の備品の貸し出しを管理したいです」と話しかけた
- 2 Claude Code が決めごとを聞き、要件定義シートと台帳の設計を作った
- 3 clasp で Google とつなぎ、Claude Code が自分でコードを反映した
- 4 直しも push も Claude Code。人は確認と合意だけ
- 5 公開範囲だけは人が決めて、「自分のみ」で公開した



06

まとめ・特典のご案内・サービス紹介



今日持ち帰ってほしい 3 つのこと

1 GAS の Web アプリなら、サーバーも費用も要らない。残る壁はコーディングだけで、それは Claude Code が引き受ける

2 clasp を入れると、AI の手が Google まで届く。コピペから卒業できる

3 AI に GAS を書かせるなら、不具合やセキュリティの問題がおきない「勘所」を最初から教えておく。それが「くの一の式キット」



くのー式 AI GAS コーディングキット、その中身

中身はテキストファイル 4 つ。フォルダに置くだけで、Claude Code が「GAS を知り尽くした、安全第一の開発パートナー」に変わります。

MANUAL.md

人間が読む

初回セットアップと導入の手順書。人が読むのはこれだけ

CLAUDE.md

Claude が読む

毎回自動で読み込まれる指示書

GAS_GUIDELINES.md

Claude・人間

GAS のコーディング規約。「クセ」の正解とセキュリティの決めごと

PROCESS.md

Claude が読む

開発を進める 6 フェーズの定義

- 自社の開発現場で「ハマったこと」「GAS 特有のクセ」にぶつかるたびに書き足してきたノウハウを、Claude Code 用に凝縮しました
- 使い方は「〇〇を作りたい」と話しかけるだけ。プロンプトの工夫も、専門用語も要りません
- 有料チケットの特典としてお贈りします



入れたのは「AI が自力では気づけないこと」だけ

AI への指示は、増やすほど一つ一つが守られなくなります。だから規則は「GAS の落とし穴」と「セキュリティの決めごと」に絞っています。

決めごと	中身
GAS のクセ	Claude Code が最初から知っている状態で始める
秘密情報	API キーやパスワードはコードに書かない。GAS の「スクリプト プロパティ」に入れる
権限と公開範囲	使う権限だけを求める。公開は「自分のみ」が基本で、広げても同じ組織内まで
作り方の 3 原則	同じことを 2 か所には書かない。頼まれていない機能を作らない。いちばん単純な構造を選ぶ

- 全規則は、実際に問題を確認できたことだけを採用しています
- 対応範囲は、自分ひとり、または同じ組織（Google Workspace）内で使うアプリです



キットなし／キットあり

同じ「社内の備品の貸し出しを管理したいです」でも、始まり方が変わります。

	キットなし	キットあり
最初の反応	いきなりコードを書き始める	「誰が使いますか」と聞いてくる
GAS のクセ	一般的な JavaScript で書き、動かないことがある	規約に沿って GAS の正解で書く
秘密情報	コードに書いてしまうことがある	スクリプト プロパティに入れる
公開範囲	指示しなければ決めない	初回は「自分のみ」を案内する
残るもの	コードだけ	要件定義シート・使い方・困ったときは



次にできること

アーカイブとキットを受け取ったら、この順で試してください。

1 キットの `MANUAL.md` を開き、PC ごとに 1 回のセットアップを済ませる

2 キットのフォルダで `claude` を起動し、「○○を作りたい」と話しかける

3 最初のアプリは「自分だけが使うもの」から。公開は「自分のみ」で



サービス紹介 DENJU（伝授）

Claude Code を会社を導入し、定着まで伴走する、法人向けのオンライン導入支援・コンサルティングです。

01

導入・構築・マンツーマン指導

環境構築から、貴社の実務を題材にした実践指導まで

02

運用・伴走コンサルティング

使い続けられる習慣を組織に定着させます

03

業務システムの受託開発・自動化

今日のような GAS アプリを、貴社専用開発します

道具を配って終わりにはしません。頭の中にしかない知を言葉にし、出所つきの台帳に納め、次の人へ渡せるかたちにするとところまでを引き受けます。



安心してお任せいただける理由

「作れる」の先にある「安心して使える」を、会社として担保します。

- ISMS（JIS Q 27001）認証取得事業者。機密データを扱う前に NDA を締結します
- 自社で百を超える GAS プロジェクトを開発し、社内プロダクトも Claude Code で開発・運用しています
- 5 年ほど前まで非エンジニアだった講師が、非エンジニアのつまずきに寄り添います



ISMS（JIS Q 27001）認証取得

100+ GAS プロジェクト

まずは 60 分の無料相談から

60分

オンライン・無料
売り込みはしません

オンライン 60 分・無料。売り込みはしません。

- いまの業務と記録の状態を見立て、どこから始めるかを一緒に決めます
- Claude Code を法人で導入するときの環境・権限・セキュリティの考え方
- 個人の方は、1on1 チケットまたは X の DM (@Knorr_CS) からどうぞ

<https://front-works.co.jp>

X @Knorr_CS



Q & A

チャットに書いていただいた質問から、順にお答えします。

- 時間内にお答えできなかった質問は、アーカイブのご案内とあわせて回答します

Q&A



参考資料・リンク集

今日の内容の一次情報です。

clasp (Google 公式リポジトリ)	https://github.com/google/clasp
------------------------	---

clasp の使い方 (Google 公式ガイド)	https://developers.google.com/apps-script/guides/clasp
---------------------------	---

Apps Script の割り当てと上限	https://developers.google.com/apps-script/guides/services/quotas
----------------------	---

Apps Script の Web アプリ	https://developers.google.com/apps-script/guides/web
-----------------------	---

Claude Code クイックスタート	https://code.claude.com/docs/ja/quickstart
----------------------	---

Fuwapp X	https://fuwapp.com @Knorr_CS
------------	---





... Thank you

ご参加、ありがとうございました。

Claude Code × GAS × clasp 超入門！

久野 晃平（くのーる） | 株式会社フロント・ワークス

X: @Knorr_CS | <https://front-works.co.jp> | <https://fuwapp.com>

動画アーカイブ

くのーる式 AI GAS コーディングキット

30 分 1on1 相談