

# Project Sol: コマンドライン 虎の巻 (Ver 10.2)

このファイルは、開発・運用中に使用する全ての自動化コマンドを一覧にしたチートシートです。

## ⚠ 実行前の重要事項 (AWSサーバー運用)

- 1. 仮想環境に入る (必須) AWSサーバー上で cli.py を実行する際は、必ず仮想環境を有効化してください。

```
# ログイン後、まずこれを実行
source ~/docker-first-ctf/venv/bin/activate
```

- 2. Pythonのパス 仮想環境が有効化されている限り、コマンドは python3 tools/cli.py <command> ではなく、python tools/cli.py <command> で実行可能です。(python3 の代わりに python を使います)

## 1. 🚀 完全自動化コマンド

コマンド	説明	実行タイミング
python tools/cli.py auto-add	<b>【最強】</b>  AIによる問題作成 → 解説生成 → DB登録 → ビルド → SNS投稿生成まで、全てを一括で実行します。	新しい問題を追加したい時。  (毎週金曜にCronが実行しているのもこれです)

## 2. 🧹 環境リセット・クリーンアップ

開発中のトラブル解消や、本番前のデータ一掃に使います。

コマンド	引数	説明	使う場面
python tools/cli.py reset	なし	<b>【超重要】</b>  DBの全データ、関連するDockerコンテナ、Dockerイメージを <b>全て削除</b> します。  ※実行前に確認プロンプトが出ます。	開発環境をきれいにしたい時。  データベースの型を変更した後のデータ初期化。
docker rm -f \$(docker ps -aq)	なし	実行中のDockerコンテナを全て停止・削除します。システムが不安定な時に実行してください。	サーバーが重くなった時。  ポートが解放されない時。

`docker compose restart  
api frontend`

なし

APIとフロントエンドを再起動し、最新のコードと環境変数を読み込ませます。

`.env` ファイルを変更した後。

3. 🛠️ 開発・デバッグ用コマンド（個別実行）

auto-add の中身を個別に実行したり、生成内容を確認したりするためのコマンドです。

コマンド	引数	説明
<code>python tools/cli.py draft</code>	なし	AIにシナリオ・コード・解説を考えさせ、JSONファイルだけを生成します。（DB登録はしません）
<code>python tools/cli.py deploy &lt;JSONパス&gt;</code>	JSONファイルへのパス	指定したJSONファイルの内容をDB（Supabase）に登録します。
<code>python tools/cli.py build &lt;JSONパス&gt;</code>	JSONファイルへのパス	指定したJSONに基づいて、Dockerイメージをビルドします。
<code>python tools/cli.py generate &lt;JSONパス&gt; sns</code>	JSONファイルへのパス	指定したJSONの内容から、SNS投稿文（宣伝）を生成します。