

OpenCode

OpenCode CLI で Z.AI の GLM モデルを統合する完全ガイド

OpenCode は強力な AI コーディングエージェントであり、Z.AI の GLM モデルを使用するように設定できます。

ステップ 1: OpenCode のインストール

OpenCode をインストールする最も簡単な方法は、インストールスクリプトを使用することです。

```
curl -fsSL https://opencode.ai/install | bash
```

npm を使ってインストールすることもできます：

```
npm install -g opencode-ai
```

ステップ 2: はじめに

1. Z.AI API コンソールにアクセスし、API キーを取得します。
2. opencode auth login を実行し、Z.AI を選択します。

```
$ opencode auth login
  ┌ Add credential
  └◆ Select provider
    • Z.AI
    ...
  ...
```

GLM コーディングプランをご利用の場合は、「Z.AI Coding Plan」を選択してください。

```
$ opencode auth login
  ┌ Add credential
  └◆ Select provider
    • Z.AI Coding Plan
    ...
  ...
```

3. Z.AI の API キーを入力します。

```
$ opencode auth login
```

```
└─ Add credential
    └─ ◇ Select provider
        └─ Z.AI Coding Plan
    └─ ◇ Enter your API key
        └─ -
```

4. opencode コマンドを実行して、OpenCode を起動します。

```
$ opencode
```

/models コマンドで、GLM-4.6 などのモデルを選択できます。

```
/models
```

共有機能 (Share)

OpenCode の共有機能を使えば、会話内容をパブリックなリンクとして作成し、チームメンバーと協力したり、他者に助けを求めたりできます。

動作の仕組み

共有時に OpenCode は以下の処理を行います：

1. セッション用の一意な公開 URL を生成
2. 会話履歴をサーバーに同期
3. 共有リンク（例: opencode.ai/s/）経由で誰でもアクセス可能に

共有方法

デフォルトでは会話は自動で共有されません。手動で /share コマンドを使用してください：

```
/share
```

共有を停止するには

公開アクセスを停止し、共有リンクを無効にするには：

```
/unshare
```

この操作により、共有リンクが削除され、関連データもサーバーから削除されます。

詳細は会話の共有についての[ドキュメント](#)をご覧ください。

IDE 拡張機能

OpenCode は VS Code・Cursor など、ターミナルをサポートするあらゆる IDE と連携します。

インストール方法

VS Code や Cursor、Windsurf、VSCodium などの派生版にインストールするには：

1. VS Code を開く
2. 統合ターミナルを開く
3. `opencode` を実行 → 拡張機能が自動でインストールされます

使い方

- **クリック起動:** `Cmd + Esc` (Mac) または `Ctrl + Esc` (Windows/Linux) で分割ターミナルに OpenCode を起動。すでにターミナルが開いている場合はフォーカスを移動します。
- **新規セッション:** `Cmd + Shift + Esc` (Mac) または `Ctrl + Shift + Esc` (Windows/Linux) で新しいセッションを開始（常時）。UI の OpenCode ボタンからも起動可能です。
- **コンテキスト認識:** 現在選択中のコードや開いているタブを自動で OpenCode に送信します。
- **ファイル参照ショートカット:** `Cmd + Option + K` (Mac) または `Alt + Ctrl + K` (Windows/Linux) でファイル参照を挿入できます。例：`@File#L37-42`.

詳細は IDE 連携の[ドキュメント](#)をご確認ください。

GitHub ワークフロー

OpenCode は GitHub ワークフローと連携します。Issue や PR のコメントで `/opencode` または `/oc` と記述すると、OpenCode が GitHub Actions ランナー内でタスクを実行します。

主な機能

- **Issue のトリアージ:** OpenCode に Issue を調査させ、内容をわかりやすく説明してもらいます。
- **バグ修正・機能実装:** OpenCode が新しいブランチで修正・実装を行い、すべての変更を含む PR を自動で送信します。
- **セキュリティ対応:** OpenCode はあなたの GitHub ランナー内で実行されるため、コードやシークレットが外部に漏れることはありません。

インストール方法

GitHub リポジトリ内のプロジェクトで、以下のコマンドを実行します：

```
opencode github install
```

このコマンドは、GitHub アプリのインストール、ワークフローの作成、Secrets の設定をガイド付きで行います。

使用例

以下は、GitHub で OpenCode を使用する方法の例です。

- **Issue の内容を説明してもらう**

GitHub Issue に以下のコメントを投稿します：

```
/opencode explain this issue
```

OpenCode がスレッド全体（すべてのコメントを含む）を読み取り、わかりやすい要約を返信します。

- **Issue を修正してもらう**

Issue に次のように記述します：

```
/opencode fix this
```

新しいブランチで修正を実装し、変更内容を含む PR を自動作成します。

- **PR のレビューと変更依頼**

GitHub PR のコメントに次のように記述します：

```
Delete the attachment from S3 when the note is removed /oc
```

指定された変更を実装し、同じ PR にコミットします。

詳細は [GitHub ワークフローのドキュメント](#)をご覧ください。

リソース

- **ドキュメント:** [opencode.ai/docs](#)
- **GitHub Issues:** [github.com/sst/opencode/issues](#)
- **Discord:** [opencode.ai/discord](#)