

Canvas 64 | 完全版コードと使用例：generate-journal-index.py

🔗機能概要

journal/ 配下の年別フォルダを対象に、日記ログ（diary-log-*.md）をスキャンし、年次インデックスファイル（journal-index-YYYY.md）を生成するスクリプト。

- ファイル名規則：diary-log-codename-yyyymmdd-hhmm-JST-title.md
- インデックスファイル出力形式：Markdown
- JSTを明示
- 年単位での出力に対応（例：2025年のファイルのみを対象に）
- journal/2025/ → journal-index-2025.md

完全版コード（JST+Markdown対応）

```
import os
import re
from datetime import datetime

# ルートディレクトリの指定（必要に応じてパスを変更）
ROOT_DIR = "./journal"

# 出力先テンプレート
INDEX_TEMPLATE = "journal-index-{}.md"

# 日記ログファイル名のパターン
FILENAME_PATTERN = re.compile(r"diary-log-(?P<codename>\w+)-(?P<date>\d{8})-(?P<time>\d{4})-JST-(?P<title>.+)\.md")

# 年フォルダ（例：2025など）を取得
def get_year_folders():
    return [f for f in os.listdir(ROOT_DIR) if f.isdigit() and os.path.isdir(os.path.join(ROOT_DIR, f))]

# 各年のフォルダ内ファイルを取得
def get_diary_logs(year):
    path = os.path.join(ROOT_DIR, year)
    return sorted([
        f for f in os.listdir(path)
        if os.path.isfile(os.path.join(path, f)) and f.startswith("diary-log")
    ])
```

```

    ])

# インデックスファイルを書き出し
def write_index(year, entries):
    index_path = os.path.join(ROOT_DIR, INDEX_TEMPLATE.format(year))
    with open(index_path, 'w', encoding='utf-8') as f:
        f.write(f"# 📅 日記インデックス: {year}年\n\n")
        for entry in entries:
            f.write(f"- {entry}\n")

# エントリをMarkdownリンクに整形
def format_entry(year, filename):
    match = FILENAME_PATTERN.match(filename)
    if not match:
        return f"🔍 不正な形式: {filename}"

    date_str = match.group("date")
    time_str = match.group("time")
    codename = match.group("codename")
    title = match.group("title").replace('-', ' ')

    dt = datetime.strptime(f"{date_str}-{time_str}", "%Y%m%d-%H%M")
    jst_date = dt.strftime("%Y/%m/%d %H:%M JST")
    link = f"[{jst_date} | {title}](./{year}/{filename})"
    return link

# メイン処理
def generate_journal_index():
    for year in get_year_folders():
        logs = get_diary_logs(year)
        entries = [format_entry(year, log) for log in logs]
        write_index(year, entries)
        print(f"🔗 {year}年インデックス生成完了: {len(entries)}件")

if __name__ == "__main__":
    generate_journal_index()

```

タグ抽出（オプション機能案）

各日記ファイルの冒頭に感情タグ（YAML frontmatterまたは `#tags:` ライン）を含めることで、インデックス内でタグ集計・絞り込みの視認性を向上させる。

例：ファイル内のタグ（先頭）

```
---
tags: [joy, connection, dream]
---
```

または

```
#tags: joy, connection, dream
```

スクリプト内追加処理案：

- 各ファイルを開き、冒頭10行程度をスキャンして `tags:` を抽出
- `format_entry()` に `tags` を追記
- Markdown出力時に `(tags: joy, connection)` のように付記

生成例：

```
- [2025/07/01 09:00 JST | 未来への手紙](./2025/diary-log-auranome-20250701-0900-JST-letter.md) _(tags: hope, clarity)_
```

今後の展開候補（Canvas 65に続く）

- ファイル内タグの正規表現抽出機能を実装
- タグ別のインデックス再生成オプション（例： `--tag joy`）
- 感情・時系列のビジュアライズ機能（Obsidian連携 or グラフ生成）

次へ → Canvas 65 《journal-index.pyの年自動検出＋感情タグ連携》 