

lzwjava.github.io

This is the README.md from github project <https://github.com/lzwjava/lzwjava.github.io>.

lzwjava.github.io

Welcome to my personal blog! Here, I share my thoughts, experiences, and knowledge on various topics.

Key Improvements

This blog incorporates several enhancements compared to a standard Jekyll blog using the jekyll-theme-cayman:

- **AI-Powered Translation:** Leverages advanced language models for accurate and contextually relevant translations, expanding content accessibility to a global audience.
- **XeLaTeX PDF Generation:** Integrates XeLaTeX to produce high-quality, print-ready PDFs for offline reading and sharing.
- **Google Cloud Text-to-Speech:** Utilizes Google Cloud's Text-to-Speech service to generate audio versions of posts, improving accessibility for visually impaired users and those who prefer audio content.
- **Enhanced CSS Styling:** Features a refined and custom CSS design for a visually appealing and user-friendly experience.
- **MathJax Support:** Implements MathJax for rendering complex mathematical expressions and equations, making technical content more accessible.
- **Night Mode:** Includes a night mode option to reduce eye strain and improve readability in low-light conditions.
- **Flexible Post Selection:** Offers various post selection options, such as filtering by category or tag, to enhance navigation.
- **Regular Updates:** Ensures the blog's library and dependencies are up-to-date for optimal performance and security.
- **awesome-cv Integration:** Uses `awesome-cv` to generate professional CVs directly from the blog.
- **RSS Feed Support:** Provides RSS feeds via `feed.xml`, allowing users to subscribe to the blog.
- **Bilingual Content:** Supports both Chinese and English content to cater to a diverse audience.
- **GitHub Workflow Automation:** Implements GitHub Actions for automated building, testing, and deployment, ensuring a streamlined development process.
- **Automatic Translation Workflow:** Automatically translates new or updated posts into multiple languages using GitHub Actions.
- **EPUB Support:** Converts Markdown to EPUB for ebook readers.

Getting Started

To set up a local Jekyll environment, follow these steps:

```
gem install jekyll bundler
```

```
jekyll new myblog
```

```
cd myblog
```

```
bundle install
```

```
bundle exec jekyll serve
```

```
bundle exec jekyll serve --draft
```

```
bundle add webrick
```

```
jekyll serve
```

Markdown Styling

To customize syntax highlighting, use `rougify`:

```
rougify help style
```

```
rougify style github > _sass/syntax.css
```

```
rougify style gruvbox.dark > _sass/syntax.css
```

```
rougify style base16.monokai.dark > _sass/syntax.css
```

Audio and PDF Pipelines

To generate audio and PDF files, use the following commands:

```
conda activate google-cloud-env
```

```
python scripts/audio_pipeline.py --task posts --n 10
```

```
python scripts/audio_pipeline.py --task pages
```

```
python scripts/audio_pipeline.py --task notes
```

```
python scripts/audio_pipeline.py --task notes --n 100
```

```
python scripts/pdf_pipeline.py --task pages
```

```
python scripts/pdf_pipeline.py --task posts --n 10
```

```
python scripts/pdf_pipeline.py --task notes
```

```
python scripts/generate_notes_link.py
```

```
python scripts/auto-ss-config/upload_configs.py
```

```
python scripts/epub_book.py notes
```

```
python scripts/epub_book.py _posts/en
```

```
python scripts/epub_book.py _posts/ja
```

Scripts

To create a new draft post, use:

```
python scripts/draft.py <name>
```

Content Guide

- `-en`: Denotes English posts.
- `-zh`: Denotes Chinese posts.

The `_posts` and `pages` directories support both `-en` and `-zh` suffixes.

The `notes` directory primarily contains English content for personal learning purposes and does not require translations.

Reference

Syntax Highlighting with Jekyll