

言語サポート：フォントとテキスト読み上げ

私のブログは現在、9つの言語をサポートしています：日本語 (ja)、スペイン語 (es)、ヒンディー語 (hi)、中国語 (zh)、英語 (en)、フランス語 (fr)、ドイツ語 (de)、アラビア語 (ar)、そして繁体字中国語 (hant)。サイトは<https://lzwjava.github.io>で見つけることができます。

コンピュータ環境で複数の言語を扱う際には、いくつかの側面を考慮する必要があります。

フォントの扱い

異なる言語は、特に LaTeX を使用して PDF を生成する際に、正しいレンダリングのために特定のフォントを必要とします。以下の Python コードは、オペレーティングシステムと言語に基づいて適切なフォントを選択する方法を示しています：

```
if platform.system() == "Darwin":
    if lang == "hi":
        CJK_FONT = "Kohinoor Devanagari"
    elif lang == "ar":
        CJK_FONT = "Geeza Pro"
    elif lang in ["en", "fr", "de", "es"]:
        CJK_FONT = "Helvetica"
    elif lang == "zh":
        CJK_FONT = "PingFang SC"
    elif lang == "hant":
        CJK_FONT = "PingFang TC"
    elif lang == "ja":
        CJK_FONT = "Hiragino Sans"
    else:
        CJK_FONT = "Arial Unicode MS"
else:
    if lang == "hi":
        CJK_FONT = "Noto Sans Devanagari"
    elif lang == "ar":
        CJK_FONT = "Noto Naskh Arabic"
    elif lang in ["en", "fr", "de", "es"]:
        CJK_FONT = "DejaVu Sans"
    elif lang == "zh":
        CJK_FONT = "Noto Sans CJK SC"
    elif lang == "hant":
        CJK_FONT = "Noto Sans CJK TC"
    elif lang == "ja":
```

```

CJK_FONT = "Noto Sans CJK JP"
else:
    CJK_FONT = "Noto Sans"
command = [
    'pandoc',
    input_markdown_path,
    '-o', output_pdf_path,
    '-f', 'markdown',
    '--pdf-engine', 'xelatex',
    '-V', f'romanfont={CJK_FONT}',
    '-V', f'mainfont={CJK_FONT}',
    '-V', f'CJKmainfont={CJK_FONT}',
    '-V', f'CJKsansfont={CJK_FONT}',
    '-V', f'CJKmonofont={CJK_FONT}',
    '-V', f'geometry:{GEOMETRY}',
    '-V', 'classoption=16pt',
    '-V', 'CJKoptions=Scale=1.1',
    '-V', 'linestretch=1.5'
]

```

この解決策は完璧ではないことに注意してください。例えば、コードブロックのコメント内のヒンディー語テキストは期待通りにレンダリングされないかもしれません。

テキスト読み上げ

私は Google Text-to-Speech を使用してブログ記事の音声版を生成しています。以下のコードスニペットは、テキスト読み上げエンジンに適切な言語コードを選択する方法を示しています：

```

if filename.endswith('-zh.md'):
    language_code = "cmn-CN"
    voice_language_code = "cmn-CN"
else:
    language_code = "en-US"
    voice_language_code = "en-US"

text_to_speech(
    text=article_text,
    output_filename=output_filename,
    task=task,
    language_code=language_code,

```

```
    dry_run=dry_run,  
    progress=progress  
)
```

現在、中国語と英語のコンテンツに対して音声が生成されています。他の言語のサポートを拡張するためには、対応する言語コードを設定する必要があります。