

Language Support: Fonts and Text-to-Speech

My blog now supports nine languages: Japanese (ja), Spanish (es), Hindi (hi), Chinese (zh), English (en), French (fr), German (de), Arabic (ar), and Traditional Chinese (hant). You can find the site at <https://lzwjava.github.io>

When dealing with multiple languages in a computer environment, several aspects need consideration.

Font Handling

Different languages require specific fonts for correct rendering, especially when generating PDFs using LaTeX. The following Python code demonstrates how to select appropriate fonts based on the operating system and language:

```
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', 'CJKnptions=Scale=1.1',
```

```
    '-V', 'linestretch=1.5'
```

```
]
```

It's important to note that this solution isn't perfect. For instance, Hindi text within code block comments might not render as expected.

Text-to-Speech

I utilize Google Text-to-Speech to generate audio versions of my blog posts. The following code snippet illustrates how I select the appropriate language code for the text-to-speech engine:

```
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  
)
```

Currently, audio is generated for Chinese and English content. To extend support to other languages, the corresponding language codes must be configured.