oerforge.convert

Content Conversion Utilities for OERForge

Overview

`oerforge.convert` provides functions for converting Jupyter notebooks (`.ipynb`) and Markdown files to various formats, managing associated images, and updating a SQLite database with conversion status. It supports batch and single-file conversion, image extraction and copying, and database logging.

Functions

setup_logging

```python
def setup_logging()
```

Configure logging for conversion actions. Logs to `log/export.log` and the console.

---

query_images_for_content

```python
def query_images_for_content(content_record, conn)
```

Query the database for all images associated with a content file.

Parameters - `content_record` (dict): Content record dictionary. - `conn`: SQLite connection object.

Returns - `list[dict]`: List of image records.

---

copy_images_to_build

```python
def copy_images_to_build(images, images_root=IMAGES_ROOT, conn=None)
```

Copy images to the build images directory. Returns a list of new build paths.

Parameters - `images` (list[dict]): List of image records. - `images_root` (str): Destination directory for images. - `conn`: SQLite connection object (optional).

Returns - `list[str]`: List of copied image paths.

---

update_markdown_image_links

```
def update_markdown_image_links(md_path, images, images_root=IMAGES_ROOT)
```

Update image links in a Markdown file to point to copied images in the build directory.

Parameters - `md_path` (str): Path to the Markdown file. - `images` (list[dict]): List of image records. - `images_root` (str): Images directory.

---

handle_images_for_markdown

```
def handle_images_for_markdown(content_record, conn)
```

Orchestrate image handling for a Markdown file: query, copy, and update links.

Parameters - `content_record` (dict): Content record dictionary. - `conn`: SQLite connection object.

---

convert_md_to_docx

```
def convert_md_to_docx(src_path, out_path, record_id=None, conn=None)
```

Convert a Markdown file to DOCX using Pandoc. Updates DB conversion status if `record_id` and `conn` are provided.

Parameters - `src_path` (str): Source Markdown file path. - `out_path` (str): Output DOCX file path. - `record_id` (int, optional): Content record ID. - `conn`: SQLite connection object (optional).

---

convert_md_to_pdf

```
def convert_md_to_pdf(src_path, out_path, record_id=None, conn=None)
```

Convert a Markdown file to PDF using Pandoc. Updates DB conversion status if `record_id` and `conn` are provided.

Parameters - `src_path` (str): Source Markdown file path. - `out_path` (str): Output PDF file path. - `record_id` (int, optional): Content record ID. - `conn`: SQLite connection object (optional).

---

convert_md_to_tex

```
def convert_md_to_tex(src_path, out_path, record_id=None, conn=None)
```

Convert a Markdown file to LaTeX using Pandoc. Updates DB conversion status if `record_id` and `conn` are provided.

Parameters - `src_path` (str): Source Markdown file path. - `out_path` (str): Output LaTeX file path. - `record_id` (int, optional): Content record ID. - `conn`: SQLite connection object (optional).

---

convert_md_to_txt

```python
def convert_md_to_txt(src_path, out_path, record_id=None, conn=None)
```

Convert a Markdown file to plain TXT (extracts readable text). Updates DB conversion status if `record_id` and `conn` are provided.

Parameters - `src_path` (str): Source Markdown file path. - `out_path` (str): Output TXT file path. - `record_id` (int, optional): Content record ID. - `conn`: SQLite connection object (optional).

---

batch_convert_all_content

```python
def batch_convert_all_content(config_path=None)
```

Batch process all files in the content table. For each file, checks conversion flags and calls appropriate conversion functions. Organizes output to mirror TOC hierarchy.

Parameters - `config_path` (str, optional): Path to `_content.yml` config file.

CLI Usage

```
python convert.py batch
python convert.py single --src <source> --out <output> --fmt <format> [--record_id <id>]
```

Requirements

- Python 3.7+
- Pandoc (for docx, pdf, tex conversions)
- nbconvert
- markdown-it-py
- SQLite3

3

See Also

- Pandoc Documentation
- nbconvert Documentation

License

See `LICENSE` in the project root.