

# build.py — OERForge Build Orchestrator

## Overview

build.py is the main workflow orchestrator for the OERForge static site generator. It coordinates the entire build process, including database initialization, content scanning, conversion, HTML generation, exporting, and deployment preparation. This script is designed for new users and programmers to provide a clear, step-by-step build pipeline for open educational resources.

- **Initializes the content database**
  - **Scans the table of contents and populates the database**
  - **Converts all source content to output formats**
  - **Exports all content and assets to the build directory**
  - **Builds HTML pages and section indexes**
  - **Copies build outputs to docs/ for publishing**
  - **Logs all major steps and errors**
- 

## Functions

### run()

Runs the complete OERForge build workflow.

**Parameters** - None

**Returns** - None

**Workflow Steps** 1. Initializes the SQLite database for content and assets. 2. Scans the table of contents (`_content.yml`) and populates the database. 3. Converts all source content (Markdown, Jupyter, DOCX) to output formats. 4. Exports all content and assets to the build directory. 5. Builds HTML pages and section indexes using templates. 6. Copies the build output to `docs/` for publishing (e.g., GitHub Pages). 7. Logs all steps and errors to the build log.

---

## Constants

- `BUILD_DIR`: Path to the build output directory.
  - `FILES_DIR`: Subdirectory for files in build.
  - `PROJECT_ROOT`: Absolute path to the project root.
  - `BUILD_FILES_DIR`: Path to files in build.
  - `BUILD_HTML_DIR`: Path to HTML output in build.
-

## Logging

All major operations and errors are logged for debugging and auditing. The build log can be found in the `log/` directory.

## Error Handling

Robust error handling is implemented in each step of the workflow modules. All failures are logged with context.

## Example Usage

```
# Run the build workflow from the command line  
python build.py
```

```
# Or import and run from another script  
from build import run  
run()
```

---

## License

See LICENSE in the project root.