

# Python Index Generator Documentation

**Document Title:** Python Index Generator Documentation

**Author:** Joseph Malone

**Program:** `indexer.py`

**Date:** 08/18/2025

**Classification:** Technical Reference

**Note:** Applies to versions **beta-3.0, 3.1, 3.2**

## Troubleshooting Guide

### 1. Verify File Placement

- Open **File Explorer**
- Navigate to: `C:\mgu\`
- Confirm your script is present → e.g., `beta_3.py`
- If the file appears as `beta_3.py.txt` →
  - Enable *View* → *File name extensions* in Explorer
  - Rename file to `beta_3.py`

### 2. Run from PowerShell

If located in `C:\mgu\`:

```
cd C:\mgu
python beta_3.py
```

### 3. If Script is in Another Folder

Example: saved in **Downloads**

```
cd C:\Users\YourName\Downloads
python beta_3.py
```

Or bypass `cd` with a full path:

```
python C:\Users\YourName\Downloads\beta_3.py
```

### 4. Verify Python Availability

```
python --version
```

If version displays (e.g., *Python 3.13.0*), environment is correctly configured.

## 5. Expected Script Output

```
Created departments\political-science\index.html
Created departments\law\index.html
...
```

Preview generated site locally:

```
python -m http.server 8000
```

Access at → <http://localhost:8000>

## Metadata Approaches

### README (Markdown/Text) Approach

#### Advantages

- Human-friendly, faculty-readable
- Inline documentation doubles as narrative
- Lightweight & forgiving of errors
- Immediate value on GitHub/File Explorer

#### Limitations

- Parsing is heuristic (script must “guess”)
- Limited structure → poor for complex data
- Weak machine-readability

### JSON Metadata File

#### Advantages

- Structured, precise, no ambiguity
- Scalable (add new fields easily)
- Machine-reliable → no guessing
- Integrates with APIs/dashboards

#### Difficulties

- Less friendly for non-technical staff
- Syntax-sensitive (errors break parsing)
- No narrative/documentation value

### Strategic Recommendation: Hybrid Model

- **README.md** → narrative, human-facing

- **meta.json** → structured, machine-facing

### Script Behavior:

1. Check for `meta.json` first
2. Fallback to `README.md`
3. If absent, fallback to `README.txt`

## Beta Version Updates

### Beta 3.1

- Introduced priority parsing (`meta.json` → `README.md` → `README.txt`)
- Supported thumbnail in hero
- Added author + updated date in footer

### Beta 3.2

- Read tags (from JSON or README)
- Render as clickable tag badges (styled in `pyc.css`)
- Maintains 3.1 functionality (JSON-first logic, thumbnail, metadata injection, timestamping)

## CSS Snippet for Tag Badges (`/components/css/pyc.css`)

```
.tags {  
  margin-top: 1rem;  
}  
.tag {  
  display: inline-block;  
  background: var(--accent);  
  color: #fff;  
  padding: 0.2rem 0.6rem;  
  margin: 0 0.3rem 0.3rem 0;  
  border-radius: 999px;  
  font-size: 0.85rem;  
}
```

## Example Implementations

### **README.md (human-friendly, fallback)**

# Political Science

Exploring governance, power, and global relations.

Author: Faculty of Social Sciences

Updated: 2025-08-18

Tags: politics, governance, law, research

Thumbnail: /images/departments/political-science.png

### **meta.json (structured, primary source)**

```
{
  "title": "Political Science",
  "description": "Exploring governance, power, and global relations.",
  "author": "Faculty of Social Sciences",
  "updated": "2025-08-18",
  "tags": ["politics", "governance", "law", "research"],
  "thumbnail": "/images/departments/political-science.png"
}
```

### **Summary:**

- Use meta.json for **structured reliability**.
- Use README.md for **human readability**.
- The **hybrid strategy** future-proofs your index generator for both **faculty adoption** and **machine scalability**.