

Directory Index Building - 3

Never finished

engelbert gruber

08.04.2020

3RD GRINDED

ToDo / Next

- system-Verzeichnis und info Dateien ignorieren
- Umgekehrte Sortierung.
- Mehr als eine Sprache bei einem Dokument, das bedeutet es muss mehrfach gelistet werden.
- .info-Dateien für remote Dateien via URL.

INFO Dateien anlegen

Und crash:

```
Traceback (most recent call last):
  File "mkindexes.py", line 52, in <module>
    liste.append(Document(e1))
  File "mkindexes.py", line 25, in __init__
    self.read_attributes()
  File "mkindexes.py", line 36, in read_attributes
    a = json.load(iPath.open())
  File "/usr/lib/python3.6/json/__init__.py", line 299, in load
    parse_constant=parse_constant, object_pairs_hook=object_pairs_hook, **kw)
  File "/usr/lib/python3.6/json/__init__.py", line 354, in loads
```

```

        return _default_decoder.decode(s)
File "/usr/lib/python3.6/json/decoder.py", line 339, in decode
    obj, end = self.raw_decode(s, idx=_w(s, 0).end())
File "/usr/lib/python3.6/json/decoder.py", line 355, in raw_decode
    obj, end = self.scan_once(s, idx)
json.decoder.JSONDecodeError: Expecting ':' delimiter: line 3 column 12 (char
33)

```

Ok ... irgendwo fehlt ein ":" in einer info/json-Datei, aber in welcher ?

```

def read_attributes(self):
    """
    read attribute file if present.
    """
    iPath = self.path.with_suffix(".info")
    if iPath.exists():
        # TODO with context ?
        try:
            a = json.load(iPath.open())
        except:
            print("Error: json.load(%s)" % iPath)
            raise
        # TODO how to move all into attributes ?
        if "author" in a:
            self.author = a["author"]
        if "complang" in a:
            self.complang = a["complang"]
    # TODO ignore info files in liste

```

“Json.load” löst eine Exception aus, weil etwas nicht passt. Wir packen den json.load-Aufruf in ein “try”-Statement. Wenn etwas außergewöhnliches passiert kommt der Programmteil in “except:” zur Ausführung und gibt den Dateinamen aus. Das nachfolgende “raise” bringt dann die selbe Fehlermeldung wie vorher.

INFO Dateien nicht anzeigen

```

liste = []
for e in p.iterdir():
    # nicht anzeigen wenn etwas mit "_" beginnt
    if str(e).startswith("_"):

```

```

        continue
    if e.suffix == ".info":    # nicht anzeigen
        continue
    if e.is_dir(): # Unterverzeichnisinhalt holen
        for e1 in e.iterdir():
            if e1.suffix != ".info": # nicht anzeigen
                liste.append(Document(e1))
    else:
        liste.append(Document(e))

```

Systemverzeichnis nicht einlesen

Wir machen aus dem Path “e” einen String mittels “str(e)” und ein String hat einen Methode “startswith” die “true” zurückgibt wenn der String mit dem angegeben String beginnt.

info-Dateien nicht auflisten

Wenn das suffix/die Dateierweiterung “.info” ist “continue”, mit continue wird zum nächsten Element einer for-Schleife gesprungen.

Regression test

Ein regression test ist (meiner Interpretation) ein Test der sicher stellt, dass die Ausgabe des Programms so ist wie vorher.

.Dazu legen wir im system-Verzeichnis “_” (weil dort wird es nicht vom Programm gelesen) ein Verzeichnis “expect” and und dorthin kopieren wir die “index-*.html” Dateien. (mkdir _/expect ; copy index-*.html _/expect/).

Dann vergleichen wir das was dort im “_/_expect” ist mit dem im “.” (wir befinden uns in “blaetter”).

```
diff -u -s _/expect/ .
```

Es wird etwas viel ausgegeben, wichtig ist der Teil:

```
Files _/expect/index-author.html and ./index-author.html are identical
Files _/expect/index-complang.html and ./index-complang.html are identical
Files _/expect/index-date.html and ./index-date.html are identical
Files _/expect/index-name.html and ./index-name.html are identical
Files _/expect/index.html and ./index.html are identical

```

Das “diff”-Programm zeigt Unterschiede an (differences).

Die Option “-s” führt dazu, dass unveränderte Dateien auch angezeigt werden, mit der Meldung “are identical”.

“Die Option “-u” zeigt geänderte Zeilen (meiner Meinung nach) übersichtlicher an.

Umgekehrt Sortieren

Das muss irgendwo bei “liste.sort” passieren.

Hier <https://docs.python.org/3/library/stdtypes.html#list.sort> steht

```
sort(*, key=None, reverse=False)
```

Also machen wir:

```
# Ausgabe html: sortiert nach Datum (das Verzeichnis)
liste.sort(key=lambda p: str(p.parent).lower())
wr_html_file("index-date.html", liste, lambda p: p.parent)
liste.sort(key=lambda p: str(p.parent).lower(), reverse=True)
wr_html_file("index-date-r.html", liste, lambda p: p.parent)
```

Und bauen das zum menu dazu:

```
<a href="index-date.html">byDate </a>
<a href="index-date-r.html"></a>
```

Die Bilddateien kopieren wir von der apache2-Installation “/usr/share/apache2/icons”.

Im stylesheet verschieben wir die Bilder etwas:

```
#mainmenu a img {
  transform: translate(10%, 25%);
}
```

Der Test: runtest.sh meldet dann (für alle index-Dateien) eine Änderung des Menüs.

```
--- _/expect/index.html 2020-04-08 19:55:12.430478385 +0200
+++ ./index.html        2020-04-08 20:42:53.950972009 +0200
@@ -7,7 +7,8 @@
  <body>
```

```

    <div id="mainmenu">
        <a href="index-name.html">byName</a>
-       <a href="index-date.html">byDate</a>
+       <a href="index-date.html">byDate </a>
+       <a href="index-date-r.html"></a>
        <a href="index-complang.html">byLanguage</a>
        <a href="index-author.html">byAuthor</a>
    </div>
@@ -15,6 +16,7 @@
<div class="sep">.</div><ul>
<li><a href="index-author.html">index-author.html</a></li>
<li><a href="index-complang.html">index-complang.html</a></li>
+<li><a href="index-date-r.html">index-date-r.html</a></li>
<li><a href="index-date.html">index-date.html</a></li>
<li><a href="index-name.html">index-name.html</a></li>
<li><a href="index.html">index.html</a></li>

```

Die Zeilen mit vorangestelltem “-” sind in der alten Datei , die die am Anfang der Ausgabe steht “--- _/expect/index.html ...” und in der neuen nicht mehr , die mit “+” am Zeilenanfang sind in der neuen.

Das passt so, wir kopieren die neuen index-html ins expect-Verzeichnis.

REFERENCES

1. <https://docs.python.org/3/library/stdtypes.html#list.sort>