```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
"""
Analyze the structure and broad properties of MMWR online archive

@author: chadheilig

Sections of this script, based on levels of MMWR archive:
0. MMWR home https://www.cdc.gov/mmwr/about.html
1. List and contents of series
2. List and contents of volumes (issues are integrated into volumes)
3. List of articles
4. Complete list of MMWR files

Main product: mmwr_dframe
"""

#%% Import modules and set up environment
# import from 0_cdc-corpora-header.py

os.chdir('/Users/cmheilig/cdc-corpora/_test')

#%% 0. Start with MMWR home https://www.cdc.gov/mmwr/about.html
base_url = 'https://www.cdc.gov/mmwr/about.html'
home_a = BeautifulSoup(get_html_from_url(base_url), 'lxml').\
   find('a', href=re.compile('about.html'))
# process_aTag(home_a, base_url)
# {'base': 'https://www.cdc.gov/mmwr/about.html',
#  'href': '/mmwr/about.html',
#  'url': 'https://www.cdc.gov/mmwr/about.html',
#  'path': '/mmwr/about.html',
#  'filename': 'about.html',
#  'mirror_path': '/mmwr/about.html',
#  'string': 'About|MMWR'}

home_dframe = pd.DataFrame(process_aTag(home_a, base_url), index = [0])
home_html = get_html_from_url_(home_dframe.url[0]) # len(home_html) # 194413
home_soup = BeautifulSoup(home_html, 'lxml')

# review all anchor-hrefs from home URL
# len(home_soup.find_all('a', href=True)) # 130
pd.DataFrame([process_aTag(aTag, home_dframe.url[0])
    for aTag in home_soup.find_all('a', href=True)]).\
    to_excel('mmwr-home-anchors.xlsx', engine='openpyxl')
# [130 rows x 7 columns]

#%% 1. List and contents of series (and some volumes)
#      Series: WR, RR, SS, SU, ND, NNC (top)

# Review of anchor elements in home page, mmwr-home-anchors.xlsx
# limit to regexes for series-specific volume lists; omit ND, NNC
series_a = home_soup.find_all('a', string=re.compile('Past Volumes'))
# [<a href="/mmwr/mmwr_wk/wk_pvol.html">Past Volumes (1982-2021)</a>,
#  <a href="/mmwr/mmwr_rr/rr_pvol.html">Past Volumes (1990-2020)</a>,
#  <a href="/mmwr/mmwr_ss/ss_pvol.html">Past Volumes (1983-2021)</a>,
#  <a href="/mmwr/mmwr_su/index.html">Past Volumes (1985-2020)</a>]
```

```
 57
 58 series_dframe = pd.DataFrame(
 59    [process_aTag(aTag, home_dframe.url[0]) for aTag in series_a])
 60 # series_dframe.loc[:, ['path', 'string']]
 61 #                          path                    string
 62 # 0  /mmwr/mmwr_wk/wk_pvol.html  Past Volumes (1982-2021)
 63 # 1  /mmwr/mmwr_rr/rr_pvol.html  Past Volumes (1990-2020)
 64 # 2  /mmwr/mmwr_ss/ss_pvol.html  Past Volumes (1983-2021)
 65 # 3    /mmwr/mmwr_su/index.html  Past Volumes (1985-2020)
 66
 67 # pool = multiprocessing.Pool(processes=multiprocessing.cpu_count() * 1) # * 3
 68 # series_html = list(pool.imap(get_html_from_url_, series_dframe.url)) # list of 4
 69 series_html = [get_html_from_url_(url) for url in series_dframe.url] # list of 4
 70 # [len(x) for x in series_html]
 71 # [192509, 191602, 192541, 190416]
 72 series_soup = [BeautifulSoup(html, 'lxml') for html in tqdm(series_html)]
 73
 74 # review all anchor-hrefs from series URLs
 75 # [len(soup.find_all('a', href=True)) for soup in series_soup]
 76 # [165, 156, 159, 142] # sum(_) # 622
 77 pd.DataFrame([process_aTag(aTag, url)
 78     for soup, url in zip(series_soup, series_dframe.url)
 79     for aTag in soup.find_all('a', href=True)]).\
 80     to_excel('mmwr-series-anchors.xlsx', engine='openpyxl')
 81 # [426 rows x 7 columns]
 82
 83 #%% 2. List and contents of volumes
 84
 85 # Review of anchor elements in volumes page, mmwr-series-anchors.xlsx
 86 # regexes for index files, i.e., volume-specific issue lists
 87 mmwr_ind_re0 = re.compile(r'/(ind\w*\d{2,4}\w*\.html?)')
 88 volumes_a = [soup.find_all('a', href=mmwr_ind_re0) for soup in series_soup]
 89 # a list of 4 lists; make a single, concatenated list
 90 volumes_a_n = [len(x) for x in volumes_a]
 91 # [44, 35, 38, 21] # 138
 92 # reorganize 4 nested lists as a sinlge list of 134
 93
 94 volumes_dframe = pd.DataFrame([process_aTag(aTag, url)
 95    for a_list, url in zip(volumes_a, series_dframe.url)
 96    for aTag in a_list])
 97 # [138 rows x 7 columns]
 98 # volumes_dframe.loc[:, ['path', 'string']]
 99 #                               path                        string
100 # 0             /mmwr/index2021.html             Volume 70 (2021)
101 # 1             /mmwr/index2020.html             Volume 69 (2020)
102 # 2             /mmwr/index2019.html             Volume 68 (2019)
103 # 3             /mmwr/index2018.html             Volume 67 (2018)
104 # 4             /mmwr/index2017.html             Volume 66 (2017)
105 # ..                             ...                         ...
106 # 133  /mmwr/preview/ind1985_su.html             Volume 34 (1985)
107 # 134           /mmwr/index2022.html                Weekly Report
108 # 135          /mmwr/indrr_2021.html  Recommendations and Reports
109 # 136          /mmwr/indss_2022.html       Surveillance Summaries
110 # 137          /mmwr/ind2022_su.html                  Supplements
111
112 # Check for duplicate values
```

```
# volumes_dframe.path.drop_duplicates() # drops from 138 to 126
volumes_repeated = volumes_dframe.loc[volumes_dframe.path.duplicated(keep = False)].index
# (16,)
# 16 rows containing duplicate path values, with indices:
# [ 40,  41,  42,  43,  75,  76,  77,  78,
#  113, 114, 115, 116, 134, 135, 136, 137]
volumes_dframe.loc[volumes_repeated, ['path', 'string']]
# on inspection, keep indices 40, 41, 42, 43 - vol type corresponds to ser type
# delete 12 indices: 75,  76,  77,  78, 113, 114, 115, 116, 134, 135, 136, 137
volumes_dframe = volumes_dframe.drop(\
   [75,  76,  77,  78, 113, 114, 115, 116, 134, 135, 136, 137])
# Check again for duplicate values
# volumes_dframe.loc[volumes_dframe.path.duplicated(keep = False)].index # []
volumes_dframe.index = list(range(126))

volumes_html = [get_html_from_url(url) for url in tqdm(volumes_dframe.url)] # list of 126
# 126/126 [00:28<00:00,  4.46it/s]
# [len(x) for x in volumes_html]
# [294414, 300349, 273408, 283545, 303795, 389800, 385758, 134350, 117893, ...]
volumes_soup = [BeautifulSoup(html, 'lxml') for html in tqdm(volumes_html)]
# 126/126 [00:03<00:00, 36.31it/s]

# review all anchor-hrefs from volumes URLs
# [len(soup.find_all('a', href=True)) for soup in volumes_soup]
# [598, 645, 517, 581, 728, 743, 688, 722, 627, 592, ...] # len 126, sum 31141
# pd.DataFrame([process_aTag(aTag, url)
#     for soup, url in zip(volumes_soup, volumes_dframe.url)
#     for aTag in soup.find_all('a', href=True)]).\
#     to_excel('mmwr-volumes-anchors.xlsx', engine='openpyxl')
# [29010 rows x 7 columns]

#%% 3. List of articles

# Review of anchor elements in volumes page, mmwr-volumes-anchors.xlsx
# all article URLs contain /preview/mmwrhtml/ or /volumes/ and end with .htm
mmwr_art_re0 = re.compile(r'(mmwrhtml|volumes)/(\w|-|/)+.html?')
articles_a = [soup.find_all('a', href=mmwr_art_re0) for soup in tqdm(volumes_soup)]
articles_a_n = [len(x) for x in articles_a]
# sum(articles_a_n) # 14630
# reorganize 126 nested lists as a single list of 14630

articles_dframe = pd.DataFrame([process_aTag(aTag, url)
   for a_list, url in zip(articles_a, volumes_dframe.url)
   for aTag in a_list])
# articles_dframe.shape # (14630, 7)
# with pd.option_context("display.max_colwidth", 36):
#    display(articles_dframe.loc[:, ['path', 'string']])
#                                        path                                 string
# 0       /mmwr/volumes/70/wr/mm705152a1.htm  COVID-19 Vaccine Safety in Child...
# 1       /mmwr/volumes/70/wr/mm705152a2.htm  Interim Estimate of Vaccine Effe...
# 2       /mmwr/volumes/70/wr/mm705152a3.htm  Characteristics and Clinical Out...
# 3       /mmwr/volumes/70/wr/mm705152e1.htm  Evaluation of a Test to Stay Str...
# 4       /mmwr/volumes/70/wr/mm705152e2.htm  Evaluation of Test to Stay Strat...
#                                       ...                                  ...
# 14625  /mmwr/preview/mmwrhtml/00026330.htm  Guidelines for the Prevention an...
# 14626  /mmwr/preview/mmwrhtml/00014715.htm  Human Immunodeficiency Virus Inf...
```

```
168 # 14627  /mmwr/preview/mmwrhtml/00023587.htm  Recommendations for Prevention o...
169 # 14628  /mmwr/preview/mmwrhtml/00001773.htm  Premature Mortality in the Unite...
170 # 14629  /mmwr/preview/mmwrhtml/00001712.htm  Summaries of Current Intelligenc...
171
172 # articles_dframe.to_excel("articles_dframe.xlsx", engine='openpyxl')
173
174 articles_repeated = {
175    label: content.loc[content.duplicated(keep = False)].index.to_list()
176       for label, content
177       in articles_dframe.loc[:, ['href', 'url', 'path', 'filename', 'string']].items() }
178 # { k: len(v) for k, v in articles_repeated.items() }
179 # {'href': 16, 'url': 18, 'path': 18, 'filename': 18, 'string': 1884}
180 articles_dframe.iloc[articles_repeated['path'], 3] # 18 rows containing duplicate path
180 values
181 articles_dframe.iloc[articles_repeated['path']].index
182 # [ 4470,  4513,  7235,  7236,  8013,  8020,  8029,  8036,  8387,
183 #   8413,  8743,  8744, 13575, 13577, 14290, 14293, 14508, 14509]
184 # on inspection, keep rows 4470, 7235, 8013, 8387, 8743, 13575, 14290, 14508
185 # drop 10 rows: 4513, 7236, 8020, 8029, 8036, 8413, 8744, 13577, 14293, 14509
186 articles_dframe = articles_dframe.drop(\
187    [4513, 7236, 8020, 8029, 8036, 8413, 8744, 13577, 14293, 14509])
188 articles_dframe.index = list(range(14620))
189
190
191 #%% 4. Complete list of MMWR HTML files
192 mmwr_dframe = pd.concat([
193    home_dframe.assign(level='home'),
194    series_dframe.assign(level='series'),
195    volumes_dframe.assign(level='volume'),
196    articles_dframe.assign(level='article')],
197    axis = 0, ignore_index = True) # mmwr_dframe.index = list(range(13800))
198 # (14226, 7)
199
200 # pickle
201 pickle.dump(mmwr_dframe, open("mmwr_dframe.pkl", "wb"))
202 # mmwr_dframe_ = pickle.load(open("mmwr_dframe.pkl", "rb"))
203 # mmwr_dframe.equals(mmwr_dframe_)
204
205 # Excel; coulad also use engine=
206 mmwr_dframe.to_excel('mmwr_dframe.xlsx', engine='openpyxl')
207 # Excelternatives
208 # mmwr_dframe.to_excel('mmwr_dframe.xlsx', engine='xlsxwriter') # pd default
209 # mmwr_dframe.to_excel('mmwr_dframe.xls', engine='xlwt')
210
211 #%% 5. Complete list of MMWR PDF files
212 mmwr_ind_pdf_re1 = re.compile(r'\w*\.pdf')
213 volumes_p_a = [soup.find_all('a', href=mmwr_ind_pdf_re1) for soup in series_soup]
214 # a list of 4 lists, all empty; [[], [], [], []]
215
216 # all occurrences of *.pdf in a subdirectory
217 mmwr_art_pdf_re1 = re.compile(r'.+?\.pdf')
218 articles_p_a = [soup.find_all('a', href=mmwr_art_pdf_re1) for soup in tqdm(volumes_soup)]
219 articles_p_a_n = [len(x) for x in articles_p_a]
220 # [51, 51, 52, 51, 51, 51, 0, 105, 103, 91, 51, 51, 51, 53, 51, 53, 52, 51, 52, 52, 51,
220 54, 51, 52, 53, 52, 52, 52, 52, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 21, 0, 0, 3, 0, 0, 0, 0,
220 0, 0, 6, 10, 5, 7, 12, 12, 10, 9, 17, 17, 15, 17, 19, 22, 16, 14, 20, 18, 15, 14, 15, 16,
```

```
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 14, 9, 10, 16, 10, 10, 13, 10, 12, 8, 9, 12, 11, 5, 10, 8,
    5, 6, 6, 6, 3, 6, 0, 0, 0, 0, 0, 1, 3, 1, 4, 3, 4, 4, 2, 1, 1, 1, 1, 0, 0, 1, 0, 0]
# len(articles_p_a_n) # 126 # sum(articles_p_a_n) # 2148

articles_pdf_dframe = pd.DataFrame([process_aTag(aTag, url)
   for a_list, url in zip(articles_p_a, volumes_dframe.url)
   for aTag in a_list])
# articles_pdf_dframe.shape # (2148, 7)
with pd.option_context("display.max_colwidth", 36):
    display(articles_pdf_dframe.loc[:, ['path', 'string']])
#                                     path                                  string
# 0     /mmwr/volumes/70/wr/pdfs/mm70515...           PDF of this issue|pdf icon
# 1     /mmwr/volumes/70/wr/pdfs/mm7050-...           PDF of this issue|pdf icon
# 2     /mmwr/volumes/70/wr/pdfs/mm7049-...           PDF of this issue|pdf icon
# 3     /mmwr/volumes/70/wr/pdfs/mm7048-...           PDF of this issue|pdf icon
# 4     /mmwr/volumes/70/wr/pdfs/mm7047-...           PDF of this issue|pdf icon
#                                       ...                                   ...
# 2143            /mmwr/pdf/wk/mm54su01.pdf  Download.pdf document of this is...
# 2144            /mmwr/pdf/wk/mm53su01.pdf  Download .pdf document of this i...
# 2145            /mmwr/pdf/wk/mmSU5201.pdf  Download .pdf document of this i...
# 2146         /mmwr/pdf/other/highlite.pdf  Highlights in Public Health -|MM...
# 2147         /mmwr/pdf/other/mmsu3601.pdf  Revision of the CDC Surveillance...

mmwr_art_pdf_re2 = re.compile(r'(mm\d{4}md|highlite)\.pdf')
articles_pdf_dframe.loc[articles_pdf_dframe.filename.str.match(mmwr_art_pdf_re2),
'filename']

# articles_pdf_dframe.loc[articles_pdf_dframe.filename.str.match(mmwr_art_pdf_re2)]
# 146 x 7

articles_pdf_dframe = articles_pdf_dframe.loc[\
    ~(articles_pdf_dframe.filename.str.match(mmwr_art_pdf_re2) | \
      (articles_pdf_dframe.string == ''))]
# 1999 x 7

# articles_pdf_dframe.to_excel("articles_pdf_dframe.xlsx", engine='openpyxl')

articles_pdf_dframe['series'] = articles_pdf_dframe.filename.str[:2]
articles_pdf_dframe['volume'] = articles_pdf_dframe.filename.str[2:4]

# articles_pdf_dframe.loc[\
#     articles_pdf_dframe.filename.str.fullmatch('mm(501|su3601|SU5201).pdf'),
#        ['url', 'series', 'volume']]

# ad hoc adjustments to volume number
# mm501 -> 54; mmsu3601 -> 36; mmSU5201 -> 52
articles_pdf_dframe.loc[\
    articles_pdf_dframe.filename.str.fullmatch('mm(501|su3601|SU5201).pdf'), 'volume'] = \
    ['54', '52', '36']
# 1999 x 9

# ad hoc inclusion of volume 64, as base files don't contain PDF hrefs
# base wk https://www.cdc.gov/mmwr/index2015.html
#      rr https://www.cdc.gov/mmwr/indrr_2015.html
#      ss https://www.cdc.gov/mmwr/indss_2015.html
# href wk /mmwr/pdf/wk/mm6401.pdf ... /mmwr/pdf/wk/mm6450.pdf, mm6452
```

```
274 #      rr /mmwr/pdf/rr/rr6401.pdf ... /mmwr/pdf/rr/rr6404.pdf
275 #      ss /mmwr/pdf/ss/ss6401.pdf ... /mmwr/pdf/ss/ss6412.pdf
276 # url  'https://www.cdc.gov' + href
277 # path href
278 # filename  wk mm6401.pdf ... mm6450.pdf, mm6452.pdf
279 #      rr rr6401.pdf ... rr6404.pdf
280 #      ss ss6401.pdf ... ss6412.pdf
281 # mirror_path href # ignore in favor of /mmwr/pdfs/<vol>/<filename>
282 # string ''
283 # series mm, rr, or ss
284 # volume 64
285
286 _mm_list = [f'{x:02d}' for x in list(range(1, 51)) + [52] ] # 51
287 _rr_list = [f'{x:02d}' for x in range(1, 5)]                #  4
288 _ss_list = [f'{x:02d}' for x in range(1, 13)]               # 12
289 _href = ['/mmwr/pdf/wk/mm64' + iss + '.pdf' for iss in _mm_list] + \
290         ['/mmwr/pdf/rr/rr64' + iss + '.pdf' for iss in _rr_list] + \
291         ['/mmwr/pdf/ss/ss64' + iss + '.pdf' for iss in _ss_list]
292 _flnm = ['mm64' + iss + '.pdf' for iss in _mm_list] + \
293         ['rr64' + iss + '.pdf' for iss in _rr_list] + \
294         ['ss64' + iss + '.pdf' for iss in _ss_list]
295
296 articles_vol64_pdf_dframe = pd.DataFrame(dict(
297     base = ['https://www.cdc.gov/mmwr/' + iss for iss in
298                 ['index2015.html' for iss in _mm_list] +
299                 ['indrr_2015.html' for iss in _rr_list] +
300                 ['indss_2015.html' for iss in _ss_list]],
301     href = _href,
302     url = ['https://www.cdc.gov' + href for href in _href],
303     path = _href,
304     filename = _flnm,
305     mirror_path = ['/mmwr/pdfs/64/' + flnm for flnm in _flnm],
306     string = ['' for flnm in _flnm],
307     series = ['mm' for iss in _mm_list] + \
308                 ['rr' for iss in _rr_list] + \
309                 ['ss' for iss in _ss_list],
310     volume = ['64' for flnm in _flnm]
311    ))
312
313 mmwr_pdf_dframe = pd.concat([articles_pdf_dframe, articles_vol64_pdf_dframe],
314                                 axis=0) # 2066 x 9
315
316 # pickle
317 pickle.dump(mmwr_pdf_dframe, open("mmwr_pdf_dframe.pkl", "wb"))
318 # mmwr_pdf_dframe_ = pickle.load(open("mmwr_pdf_dframe.pkl", "rb"))
319 # mmwr_pdf_dframe.equals(mmwr_pdf_dframe_)
320
321 mmwr_pdf_dframe.to_excel("mmwr_pdf_dframe.xlsx", engine='openpyxl')
```