save_to_file

```
.save_to_file(
   data, file_name
)
```

Funkcija koju pozivamo kada podatke zelimo trajno pohraniti u datoteku

Args

- **data** (str) : podaci za zapisati u datoteku
- **file_name** (str) : ime datooteke u koju ce se zapisati podaci

Returns

None

---

extract_news_from_last_year

```
.extract_news_from_last_year(
   category
)
```

U ovoj funkciji imami logiku koja pronalazi sve clanke na index.hr stranice u proteklih dana. Funkcija radi na nacin da koristi trazilizu index portala, trazilicu ispuni sa imenom kategorije kao npr. "vijesti" te se koristi order_by opcija za sortiranje clanaka od najnovijeg prema najstarijem. Iteriramo kroz rezultate i izvlacimo rezultate do trenutka kada naidemo na clanak koji je stariji od godine dana. Za parsiranje stranice koristi se BeautifulSoup. Nakon sto smo iterirali kroz sve clanke rezultate spremamo u novu datoteku kako bi bili dostupni za kasniju analizu Notes: Ova funkcija se moze izvrsavati jako dugo vremena zbog mnogobrojnih zahtjeva prema portalu, nase zabiljeno vrijeme je cca 5 sati!

Args

- **category** (str) : kategorija koja se pretrazuje i za koju extractamo clanke sa portala

Returns

- **titles** (list) : lista svih pronadenih naslova za danu kategoriju

---

remove_last_h3_item

```
.remove_last_h3_item(
   html: str
)
```

Funkcija sa kojim prociscavamo preuzete clanke. Konkretno uklanjamo h3 html oznake.

Args

- **html** (str) : html string koji prociscavamo od h3 oznaka

Returns

- **clean_html** (str) : procisceni html string

---

find_category

```
.find_category(
   url: str
)
```

Funkcija pomocu koj iz URL-a clanka izdvajamo kategoriju u koju clanak pripada (kategorija clanka je navedena u URL-u)

Args

- **url** (str) : URL clanka iz kojeg citamo kategoriju u koju clanak pripada

Returns

- **category** (str) : detektirana kategorija u koju pripada clanak uz pripadajuci URL

---

find_title

```
.find_title(
   soup
)
```

Funkcija pomocu koje pronalazimo naslov clanka sluzeci se title html oznakom u html-u clanka.

Args

- **soup** : bs objekt koji sadrzava cijeloviti html clanka

Returns

- **title** (str) : detektirani naslov clanka

---

find_date_and_author

```
.find_date_and_author(
   info
)
```

Funkcija pomocu koje pronalazimo autora i datum clanka. Ova funkcija ucitava html clanka te u javascript skriptama dostupnima u html-u pronalazi podatke o autoru i datumu objavljivanja clanka. Takoder prilagodava datum u zeljeni format

Args

- **info** : bs objekt koji sadrzava potrebne html podatke

Returns

- **author_full_name** (str, str) : datum objave clanka te autor clanka

---

find_keywords

```
.find_keywords(
   info
)
```

Funkcija pomocu koje iz html-a izvlacimo kljucne rijeci. Slicno find_date_and_author funkciji ova funkcija trazi javascript kod unutar htmla koji sadrzi kljucne rijeci koje je naveo autor clanka

Args

- **info** : bs objekt koji sadrzava potrebne html podatke

Returns

- **found_keywords** (list) : pronadene kljucne rijeci clanka

---

find_text_in_html_format

```
.find_text_in_html_format(
   soup
)
```

Funkcija koja pretvara bs objekt u string koji spremamo kao tekst clanka

Args

- **soup** : bs objekt sa html-om clanka

Returns

- **full_text_in_html** (str) : string generiran iz bs objekta

---

remove_html_tags

```
.remove_html_tags(
   html: str, tags: List[str]
)
```

Ova funkcija ce pretvarati clanke iz html formata u obican string. (HTML oznake biti ce izbrisane).
Notes: Koristenjem ove funkcije biti ce teze prepoznati podnaslove unutar clanka

Args

- **html** (str) : html string clanka
- **tags** (list) : html oznake koje zelimo ukloniti iz html-a

Returns

- **html** (str) : string sa uklojenim html tagovima

---

load_all_links

```
.load_all_links(
   categories_list, links_suffix = '_links'
)
```

Funkcija koja iz vec spremljene datoteke ucitava sve spremljene linkove clanaka. Pretpostavka je da za svaku kategoriju postoji zasebna datoteka te se ucitava kategorija po kategorija.

Args

- **categories_list** (str) : lista kategorija za koju zelimo ucitati linkove iz datoteke
- **links_suffix** (str) : sufiks koji smo zadali za datoteku koja sadrzi iskljucivo linkove na clanke (ne i sam sadrzak clanaka)

Returns

- **links** (list) : lista koja sazdrava sve pohranjene linkove u datotekama za dane kategorije

---

load_all_articles

```
.load_all_articles(
   categories_list, links_suffix = '_extracted_data'
)
```

Funkcija koja iz vec spremljenih datoteka ucitava sve clanke tj sadrzaj clanaka. Pretpostavka je da za svaku kategoriju postoji zasebna datoteka te se ucitava kategorija po kategorija.

Zbog potencialnog velikog broja clanaka datoteke mogu biti relativno velike sto ce utjecati na performanse tj. brzinu rada ovog koda.

Args

- **categories_list** (str) : lista kategorija za koju zelimo ucitati sadrzaj clanaka iz datoteke
- **links_suffix** : sufiks koji smo zadali za datoteku koja sadrzi sadrzaje clanaka

Returns

- **articles** (list) : lista sa svim pohranjemim clancima i njihovim sadrzajem

---

extract_corona_articles

```
.extract_corona_articles(
   articles
)
```

Funkcija pomocu koje detektiramo sve clanke vezane uz korona virus tematiku te takve clanke izvlacimo za daljnu obradu. za kasniju obradu. Korona clanci prepoznati su na nacin da pretrazujemo tekst naslova clanka, tekst samog sadrzaja clanka i spomenute kljucne rijeci u clanku te s obzirom na predefiniranu listu rijeci koje se asociraju sa korona virusom odlucujemo radi li se o clanku koji je vezan za korona virus ili ne

Args

- **articles** (list) : lista svih clanaka koji su pronadeni za period od zadnjih godinu dana

Returns

- **corona_articles** (list) : lista svih clanaka koji su detekirani da sadrze tekst koji se dotice teme korona virusa

---

extract_vaccination_articles

```
.extract_vaccination_articles(
   corona_articles
)
```

Funkcija pomocu koje detektiramo sve clanke vezane uz podtemu cijepljenja protiv korona virusa te takve clanke izvlacimo za daljnu obradu. Clanci o cijepljenju prepoznati su na nacin da pretrazujemo tekst naslova clanka, tekst samog sadrzaja clanka i spomenute kljucne rijeci u clanku te s obzirom na predefiniranu listu rijeci koje asociraju na cijepljenjem protiv korona virusa odlucujemo radi li se o clanku koji je vezan za cijepljenje protiv virusa ili ne

Args

- **corona_articles** (list) : lista svih clanaka koji su pronadeni na temu korona virusa za period od zadnjih godinu dana

Returns

- **vaccination_keywords** (list) : lista svih clanaka koji su detekirani da sadrze tekst koji se dotice teme cijepljenja protiv korona virusa

---

extract_isolation_articles

```
.extract_isolation_articles(
   corona_articles
)
```

Funkcija pomocu koje detektiramo sve clanke vezane uz podtemu samoizolacije te takve clanke izvlacimo za daljnu obradu. Clanci o samoizolaciji prepoznati su na nacin da pretrazujemo tekst naslova clanka, tekst samog sadrzaja clanka i spomenute kljucne rijeci u clanku te s obzirom na predefiniranu listu rijeci koje asociraju na samoizolaciju te odlucujemo radi li se o clanku koji je vezan za samoizolaciju ili ne

Args

- **corona_articles** (list) : lista svih clanaka koji su pronadeni na temu korona virusa za period od zadnjih godinu dana

Returns

- **isolation_articles** (list) : lista svih clanaka koji su detekirani da sadrze tekst koji se dotice teme samoizolacije

---

extract_symptoms_articles

```
.extract_symptoms_articles(
   corona_articles
)
```

Funkcija pomocu koje detektiramo sve clanke vezane uz podtemu simptoma korona virusa te takve clanke izvlacimo za daljnu obradu. Clanci o simptomima korone prepoznati su na nacin da pretrazujemo tekst naslova clanka, tekst samog sadrzaja clanka i spomenute kljucne rijeci u clanku te s obzirom na predefiniranu listu rijeci koje asociraju na simptome korone te odlucujemo radi li se o clanku koji je vezan za simptome korone ili ne

Args

- **corona_articles** (list) : lista svih clanaka koji su pronadeni na temu korona virusa za period od zadnjih godinu dana

Returns

- **symptoms_articles** (list) : lista svih clanaka koji su detekirani da sadrze tekst koji se dotice teme simptoma korone

---

remove_duplicated_dicts_from_list

```
.remove_duplicated_dicts_from_list(
   items
)
```

Pomocna funkcija kojom micemo duplikate iz dane liste

Args

- **items** (list) : lista koju zelimo procistiti od duplikata

Returns

- **items_without_duplicates** (list) : lista bez duplikata

---

generate_monthly_statistic

```
.generate_monthly_statistic(
   articles
)
```

Funkcija koju koristimo za generiranje mjesecne statistike odnosno racunanje broja clanaka po mjesecima

Args

- **articles** (list) : lista clanaka za koju zelimo provesti generiranje statistike po mjesecima

Returns

- **month_statistic** (dict) : dict objekt koji sadrze sve mjesece (kao kljucevi dict objekta) te broj objavljenih clanaka za taj mjesec

---

replace_croatian_chars

```
.replace_croatian_chars(
   string
)
```

Prilikom spremanja clanaka znakovi karakteristicni za hrvatsku abecedu su spremljeni kao unicode blok. Ova funkcija pretvara unicode blokove za hrvatska slova u najslicnija internacionalna slova (npr. Š->S...)

Args

- **string** (str) : string u kojem zelimo pretvoriti unicode blokove u slova

Returns

- **string** (str) : string bez unicode blokova
