

# parseStringToDataclass Function

## Opis

Ta funkcija sprejme niz kot vhod in ga razčleni, da ustvari objekt podatkovnega razreda ShopItem. Niz mora imeti specifično oblikovanje, kjer vsaka vrstica predstavlja drugačen atribut predmeta ShopItem. Funkcija razdeli niz v ločene vrstice, izvleče ustrezne informacije in jih dodeli ustreznim atributom predmeta ShopItem.

## Parametri

niz (str): vhodni niz, ki ga je treba razčleniti. Upoštevati mora določeno obliko.

## Povratna vrednost

Funkcija vrne predmet ShopItem, napolnjen s podatki, ekstrahiranimi iz vhodnega niza.

## Primer uporabe

```
string = "Apple\niPhone,5\n€1020.99\n"
parsed_item = parseStringToDataclass(string)
print(parsed_item)
```

Output:

```
ShopItem(znamka='Apple', vrsta='iPhone', kolicina='5', cena='€1020.99')
```

# getItemText Function

## Opis

Ta funkcija sprejme seznam nizov (predmetov) kot vhod in odstrani oznake HTML iz vsakega niza z uporabo regularnih izrazov. Vrne nov seznam (vsebina), ki vsebuje spremenjene nize z odstranjenimi oznakami HTML.

## Parametri

predmeti (List[str]): seznam nizov, iz katerih je treba odstraniti oznake HTML.

## Povratna vrednost

Funkcija vrne nov seznam nizov (vsebine), kjer so bile oznake HTML odstranjene.

## Primer uporabe

```
items = ["<p>Example text 1</p>", "<h1>Example text 2</h1>", "<div>Example text 3</div>"]
modified_items = getItemText(items)
print(modified_items)
```

### Output:

```
['Example text 1', 'Example text 2', 'Example text 3']
```

# get\_address Function

## Opis

Ta funkcija ima tri parametre: naslov ulice, poštno številko in kraj. Na podlagi navedenih parametrov sestavi formatiran naslovni niz in ga vrne. Funkcija obravnava različne scenarije, kjer so lahko nekateri ali vsi parametri Nič ali prazni nizi.

## Parametri

street\_address (str): naslov ulice ali informacije o lokaciji.

postal\_code (str): poštna številka ali poštna številka.

kraj (str): Ime kraja ali mesta.

## Povratna vrednost

Funkcija vrne oblikovan naslovni niz, ki vključuje navedene parametre. Format je ulica\_naslov, poštna\_številka, kraj, Slovenija.

## Primer uporabe

```
address_1 = get_address("Main Street 123", "1000", "Ljubljana")
print(address_1)
```

### Output:

```
string = "Ljubljanska ulica 13, 1000, Ljubljana, Slovenia"

address_2 = get_address("", "1000", "Ljubljana")
print(address_2)
```

Ali

```
1000, Ljubljana, Slovenia  
  
address_3 = get_address("Ljubljanska ulica 13", "", "")  
print(address_3)
```

## Dodatne opombe

Ta funkcija predpostavlja, da je naslov za Slovenijo, saj naslovnemu nizu doda "Slovenija". Če morate uporabiti drugo državo, lahko kodo ustrezno spremenite.

Prepričajte se, da ste posredovali ustrezne tipe podatkov (nize), ko kličete funkcijo `get_address`.

# generate\_unique\_id Function

## Opis

Ta funkcija ustvari edinstven ID na podlagi podanega vhodnega niza. Uporablja algoritem zgoščevanja SHA-256 za ustvarjanje predmeta zgoščevanja iz vhodnega niza, pretvori izvleček zgoščene vrednosti v šestnajstiško predstavitev in nato pretvori šestnajstiško zgoščeno vrednost v decimalno število. Funkcija vrne prvih 16 števk decimalnega ID-ja kot niz.

## Parametri

`input_string (str)`: Vhodni niz, iz katerega je ustvarjen enolični ID.

## Povratna vrednost

Funkcija vrne 16-mestni decimalni ID kot niz.

## Primer uporabe

```
id_1 = generate_unique_id("example_string")  
print(id_1)
```

## Output:

```
1168886167985459
```

## Dodatne opombe

Ta funkcija zahteva modul `hashlib`, zato ga uvozite pred uporabo funkcije `generate_unique_id`.

Upoštevajte, da čeprav ta funkcija ustvari edinstvene ID-je na podlagi vhodnega niza, so kolizije (dva različna vhodna niza, ki ustvarita isti ID) teoretično možne, vendar zelo malo verjetne zaradi kriptografskih lastnosti algoritma SHA-256.

# get\_stores\_on\_page Function

## Opis

Ta funkcija pridobi informacije o trgovini s spletnega mesta "odpiralnicasi.com" na podlagi navedene številke strani in imena trgovine. Spletnemu mestu pošlje zahtevo HTTP, razčleni odgovor s pomočjo BeautifulSoup in izvleče ustrezne podatke, kot so naslov, poštna številka, kraj, zemljepisna širina in dolžina. Nato ustvari edinstven ID za vsako trgovino in sestavi seznam slovarjev, ki vsebujejo informacije o trgovini. Funkcija vrne seznam shranjenih podatkov.

## Parametri

stran (int): številka strani, s katere želite pridobiti informacije o trgovini.

store\_name (str): ime trgovine, za katero želite pridobiti informacije.

## Povratna vrednost

Funkcija vrne seznam slovarjev, kjer vsak slovar predstavlja shrambo in vsebuje naslednje ključne:

"id" (str): Enolični ID trgovine.

"storeType" (str): Vrsta ali ime trgovine.

"address" (str): Oblikovan naslov trgovine.

"\_geo" (dict): Slovar, ki vsebuje koordinate zemljepisne širine in dolžine trgovine. Ima naslednje ključne:

"lat" (float): koordinata zemljepisne širine.

"lng" (float): koordinata dolžine.

## Primer uporabe

```
stores = get_stores_on_page(1, "grocery")
for store in stores:
    print(store)
```

## Output:

## JSON

```
{  
  "id": "6540962400889265",  
  "storeType": "grocery",  
  "address": "Main Street 123, 1000, Ljubljana, Slovenia",  
  "_geo": {  
    "lat": 46.056947,  
    "lng": 14.505751  
  }  
}  
  
{  
  "id": "1168886167985459",  
  "storeType": "grocery",  
  "address": "Second Avenue 456, 2000, Maribor, Slovenia",  
  "_geo": {  
    "lat": 46.5547,  
    "lng": 15.6456  
  }  
}
```

# remove\_duplicates Function

## Opis

Ta funkcija odstrani dvojnike z danega seznama podatkov na podlagi polja 'naslov' vsakega elementa. Ponavlja se po seznamu podatkov, preverja, ali je naslov trenutnega elementa edinstven v primerjavi z naslovi elementov, ki so se prej srečali, in shrani samo edinstvene elemente na nov seznam. Funkcija vrne seznam edinstvenih elementov.

## Parametri

podatki (List[Dict[str, Union[str, Dict[str, Union[str, float]]]]]): seznam podatkov, ki vsebuje slovarje, ki predstavljajo elemente s poljem 'naslov'.

## Povratna vrednost

Funkcija vrne nov seznam (`unique_data`), ki vsebuje slovarje, ki predstavljajo edinstvene elemente na podlagi polja 'address'.

## Primer uporabe

```
unique_data = remove_duplicates(data)
for element in unique_data:
    print(element)
```