zyte-parsers

Release 0.1.0

Zyte Group Ltd

Apr 16, 2024

CONTENTS

`zyte-parsers` is a Python 3.7+ library that contains functions to extract data from webpage parts.

CHAPTER

ONE

INTRO

zyte-parsers provides functions that extract specific data from HTML elements. The input element can be an instance of either parsel.selector.Selector or lxml.html.HtmlElement. Some functions can also take a string with text (e.g. extracted from HTML or JSON) as input.

zyte_parsers.**SelectorOrElement**

alias of Union[Selector, HtmlElement, HtmlComment]

1.1 Parsers

1.1.1 Brand

zyte_parsers.**extract_brand_name**(*node: Selector | HtmlElement | HtmlComment*, *search_depth: int = 0*) → str | None

Extract a brand name from a node that contains it.

It tries element text and image alt and title attributes.

Parameters

- **node** – Node including the brand name.
- **search_depth** – Max depth for searching images.

Returns

The brand name or None.

1.1.2 Breadcrumbs

class zyte_parsers.**Breadcrumb**(*name: str | None = None*, *url: str | None = None*)

name: str | None

url: str | None

zyte_parsers.**extract_breadcrumbs**(*node: Selector | HtmlElement | HtmlComment*, ***, *base_url: str | None*, *max_search_depth: int = 10*) → Tuple[*Breadcrumb*, ...] | None

Extract breadcrumb items from node that represents breadcrumb component.

It finds all anchor elements to specified maximal depth. Anchors are collected in pre-order traversal. Such strategy of traversing supports cases where structure of nodes representing breadcrumbs is flat, which means that breadcrumb's anchors are on the same depth of HTML structure and where breadcrumb items are nested,

which means that element with next item can be a child of element with previous breadcrumb item. It also post-processes extracted breadcrumbs by using semantic markup or the location of breadcrumb separators.

Parameters

- **node** – Node representing and including breadcrumb component.
- **base_url** – Base URL of site.
- **max_search_depth** – Max depth for searching anchors.

Returns

Tuple with breadcrumb items.

1.1.3 GTIN

class zyte_parsers.**Gtin**(*type: str*, *value: str*)

type: str

value: str

zyte_parsers.**extract_gtin**(*node: Selector | HtmlElement | HtmlComment | str*) → *Gtin* | None

Extract a GTIN (Global Trade Item Number) from a node or a string that contains its text.

It detects the GTIN type and returns it together with the cleaned GTIN value. The following types are supported: *isbn10*, *isbn13*, *issn*, *ismn*, *upc*, *gtin8*, *gtin13*, *gtin14*.

Parameters

node – A node or a string that includes the GTIN text.

Returns

A GTIN item.

1.1.4 Price

zyte_parsers.**extract_price**(*node: Selector | HtmlElement | HtmlComment | str*, ***, *currency_hint: Selector | HtmlElement | HtmlComment | str | None = None*) → Price

Extract a price value from a node or a string that contains it.

Parameters

- **node** – A node or a string that includes the price text.
- **currency_hint** – A string or a node that can contain currency. It will be passed as a hint to `price-parser`. If currency is present in the price string, it could be preferred over the value extracted from `currency_hint`.

Returns

The price value as a `price_parser.Price` object.

1.1.5 Ratings and review count

class zyte_parsers.**AggregateRating**(*bestRating: float | None = None*, *ratingValue: float | None = None*)

bestRating: float | None

ratingValue: float | None

zyte_parsers.**extract_rating**(*node: Selector | HtmlElement | HtmlComment*) → *AggregateRating*

Extract rating data from a node.

Parameters
node – Node that includes the rating data.

Returns
AggregateRating item.

zyte_parsers.**extract_rating_stars**(*node: Selector | HtmlElement | HtmlComment*) → float | None

Extract a rating value from a node containing rating stars.

Parameters
node – Node that includes the rating stars.

Returns
Rating value as a float or None.

zyte_parsers.**extract_review_count**(*node: Selector | HtmlElement | HtmlComment*) → int | None

Extract review count from a node containing it.

Parameters
node – Node that includes the review count.

Returns
Review count as an int or None.

CHAPTER

TWO

CHANGES

2.1 0.5.0 (2024-01-24)

- Add the `extract_rating` and `extract_rating_stars` functions for extracting values.
- Add the `extract_review_count` function for extracting review counts.

2.2 0.4.0 (2023-12-26)

- New dependencies:
  - `gtin-validator >= 1.0.3`
  - `python-stdnum >= 1.19`
  - `six`
- Add the `extract_gtin` function for extracting GTIN values of various types.
- Add support for text input to `extract_price`.
- Add support for Python 3.12.
- CI improvements.

2.3 0.3.0 (2023-07-28)

- Now requires `price-parser >= 0.3.4`.
- Add the `extract_price` function for extracting prices and currencies.

2.4 0.2.0 (2023-07-07)

- Add the `extract_brand_name` function for extracting brands.
- Drop Python 3.7 support.

2.5 0.1.1 (2023-05-24)

- Fix building documentation.

2.6 0.1.0 (2023-05-24)

- Initial version.
- Includes extraction of `Breadcrumb` objects.

INDEX