

---

# **Yandex Market Language (YML) for Python Documentation**

*Release \_\_version\_\_ = '0.6.2'*

**Alexandr Stefanitsky-Mozdor**

Sep 10, 2021



---

## Contents:

---

<b>1</b>	<b>Project is deprecated, instead of this you can simply use Pydantic</b>	<b>1</b>
1.1	Yandex Market Language (YML) for Python . . . . .	1
<b>2</b>	<b>Installation</b>	<b>3</b>
2.1	Stable release . . . . .	3
2.2	From sources . . . . .	3
<b>3</b>	<b>Usage</b>	<b>5</b>
3.1	Models . . . . .	5
3.2	Parser / Converter . . . . .	5
<b>4</b>	<b>yandex_market_language</b>	<b>7</b>
4.1	yandex_market_language package . . . . .	7
<b>5</b>	<b>Contributing</b>	<b>37</b>
5.1	Types of Contributions . . . . .	37
5.2	Get Started! . . . . .	38
5.3	Pull Request Guidelines . . . . .	39
5.4	Tips . . . . .	39
5.5	Deploying . . . . .	39
<b>6</b>	<b>Credits</b>	<b>41</b>
6.1	Development Lead . . . . .	41
6.2	Contributors . . . . .	41
<b>7</b>	<b>History</b>	<b>43</b>
7.1	0.6.2 (2021-30-03) . . . . .	43
7.2	0.6.1 (2020-07-28) . . . . .	43
7.3	0.6.0 (2020-04-06) . . . . .	43
7.4	0.5.0 (2020-04-02) . . . . .	43
7.5	0.4.0 (2020-04-01) . . . . .	43
7.6	0.3.0 (2020-03-30) . . . . .	44
7.7	0.2.0 (2020-03-29) . . . . .	44
7.8	0.1.0 (2020-03-28) . . . . .	44
<b>8</b>	<b>Indices and tables</b>	<b>45</b>
	<b>Python Module Index</b>	<b>47</b>



---

Project is deprecated, instead of this you can simply use Pydantic

---

## 1.1 Yandex Market Language (YML) for Python

Yandex Market Language for Python provides user-friendly interface for parsing or creating XML files.

- Free software: MIT
- Documentation: <https://yandex-market-language.readthedocs.io>.
- Usage: <https://yandex-market-language.readthedocs.io/en/latest/usage.html>

### 1.1.1 Features

- The ability to convert from XML to dictionary and vice versa using models.
- Parse XML file into the ready-to-use Feed model.
- Convert dict-like objects into the ready-to-use XML file.
- Validation for in-out data.

### 1.1.2 TODO

- [ ] Full models validation support as in the Yandex.Market.
- [ ] Simplify models (code refactoring + DRY).
- [ ] Add custom exceptions on parse / convert to get more information.

- [\[ \] Documentation in Russian.](#)

### 1.1.3 Credits

This package was created with [Cookiecutter](#) and the [audreyr/cookiecutter-pypackage](#) project template.

### 2.1 Stable release

To install Yandex Market Language (YML) for Python, run this command in your terminal:

```
$ pip install yandex_market_language
```

This is the preferred method to install Yandex Market Language (YML) for Python, as it will always install the most recent stable release.

If you don't have `pip` installed, this [Python installation guide](#) can guide you through the process.

### 2.2 From sources

The sources for Yandex Market Language (YML) for Python can be downloaded from the [Github repo](#).

You can either clone the public repository:

```
$ git clone git://github.com/stefanitsky/yandex_market_language
```

Or download the [tarball](#):

```
$ curl -OJL https://github.com/stefanitsky/yandex_market_language/tarball/master
```

Once you have a copy of the source, you can install it with:

```
$ python setup.py install
```



To use Yandex Market Language (YML) for Python in a project:

```
from yandex_market_language import parse, convert
```

## 3.1 Models

You can create models manually:

```
>>> from yandex_market_language import models
>>> category = models.Category(category_id="1", name="Shoes")
```

And then transform them into a dictionary:

```
>>> d = category.to_dict()
>>> d
{'id': '1', 'name': 'Shoes', 'parent_id': None}
```

Or XML element:

```
>>> el = category.to_xml()
>>> el
<Element 'category' at 0x10ecda900>
>>> from xml.etree import ElementTree as ET
>>> ET.tostring(el)
b'<category id="1">Shoes</category>'
```

## 3.2 Parser / Converter

You can parse XML files into ready-to-use Feed model instance with parser:

```
>>> from yandex_market_language import parse, convert
>>> feed = parse("tests/fixtures/valid_feed.xml")
>>> feed
<yandex_market_language.models.feed.Feed object at 0x10d99fdf0>
>>> feed.to_dict()
{
  'shop': {
    'name': 'ZS',
    'company': 'ZoneSmart',
    'url': 'https://zonesmart.ru',
    ...
    'offers': [
      'type': None,
      'vendor': 'Brother',
      'vendor_code': 'ABC1234'
      ...
    ]
    ...
  }
}
```

And convert Feed model instances into XML files:

```
>>> convert("converted_from_feed_model.xml", feed)
>>> feed = parse("converted_from_feed_model.xml")
>>> feed
<yandex_market_language.models.feed.Feed object at 0x10d8bdee0>
>>> feed.to_xml()
<Element 'yml_catalog' at 0x000002121B634E00>
>>> from xml.etree import ElementTree as ET
>>> ET.tostring(feed.to_xml())
b'<yml_catalog date="2019-11-01 17:22">
  <shop>
    <name>ZS</name>
    <company>ZoneSmart</company>
    <url>https://zonesmart.ru</url>
    ...
    <offers>
      <offer>
        <name>...</name>
        <vendor>...</name>
        <vendorCode>...</vendorCode>
      </offer>
      ...
    </offers>
    ...
  </shop>
</yml_catalog>
```

---

## yandex\_market\_language

---

### 4.1 yandex\_market\_language package

#### 4.1.1 Subpackages

yandex\_market\_language.models package

##### Subpackages

yandex\_market\_language.models.fields package

##### Submodules

yandex\_market\_language.models.fields.enable\_auto\_discounts module

```
class yandex_market_language.models.fields.enable_auto_discounts.EnableAutoDiscountField
    Bases: object
    enable_auto_discounts
```

##### Module contents

```
class yandex_market_language.models.fields.EnableAutoDiscountField
    Bases: object
    enable_auto_discounts
```

```
class yandex_market_language.models.fields.DeliveryOptionsField
    Bases: object
    delivery_options
```

```
class yandex_market_language.models.fields.PickupOptionsField
```

```
    Bases: object
```

```
    pickup_options
```

```
class yandex_market_language.models.fields.YearField
```

```
    Bases: object
```

```
    year
```

## Submodules

### yandex\_market\_language.models.base module

### yandex\_market\_language.models.category module

```
class yandex_market_language.models.category.Category(category_id, name, parent_id=None)
```

```
    Bases: yandex_market_language.models.abstract.AbstractModel
```

```
    Category model for shop.
```

```
    Docs: https://yandex.ru/support/partnermarket/elements/categories.html
```

```
    category_id
```

```
    create_dict(**kwargs) → dict
```

```
        Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.
```

```
    create_xml(**kwargs) → xml.etree.ElementTree.Element
```

```
        Must be inherited by each child class. Describes the logic for creating a xml with data from a model.
```

```
    static from_xml(el: xml.etree.ElementTree.Element) → yandex_market_language.models.category.Category
```

```
        Must be inherited by each child class. Describes the logic for creating a model from an xml element.
```

```
    name
```

```
    parent_id
```

### yandex\_market\_language.models.currency module

```
class yandex_market_language.models.currency.Currency(currency, rate, plus=None)
```

```
    Bases: yandex_market_language.models.abstract.AbstractModel
```

```
    Currency model. Used to create a list of shop currency rates.
```

```
    Docs: https://yandex.ru/support/partnermarket/elements/currencies.html
```

```
    create_dict(**kwargs) → dict
```

```
        Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.
```

```
    create_xml(**kwargs) → xml.etree.ElementTree.Element
```

```
        Must be inherited by each child class. Describes the logic for creating a xml with data from a model.
```

```
    currency
```

```
    static from_xml(el: xml.etree.ElementTree.Element) → yandex_market_language.models.currency.Currency
```

```
        Must be inherited by each child class. Describes the logic for creating a model from an xml element.
```

**plus**

**rate**

### yandex\_market\_language.models.feed module

**class** `yandex_market_language.models.feed.Feed` (*shop: yandex\_market\_language.models.shop.Shop, date: None.datetime.date = None*)

Bases: `yandex_market_language.models.abstract.AbstractModel`

YML Feed model.

Docs: <https://yandex.ru/support/partnermarket/export/yml.html>

**create\_dict** (*\*\*kwargs*) → dict

Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.

**create\_xml** (*\*\*kwargs*) → `xml.etree.ElementTree.Element`

Must be inherited by each child class. Describes the logic for creating a xml with data from a model.

**date**

**static from\_xml** (*el: xml.etree.ElementTree.Element*) → `yandex_market_language.models.feed.Feed`

Must be inherited by each child class. Describes the logic for creating a model from an xml element.

**shop**

### yandex\_market\_language.models.offers module

**class** `yandex_market_language.models.offers.AbstractBookOffer` (*name: str, publisher: str, age: yandex\_market\_language.models.age.Age, isbn: str = None, author: str = None, series: str = None, year=None, volume=None, part=None, language: str = None, table\_of\_contents=None, \*\*kwargs*)

Bases: `yandex_market_language.models.fields.year.YearField, yandex_market_language.models.offers.AbstractOffer, abc.ABC`

Abstract book offer for book & audio book offer types.

**age**

**author**

**barcodes**

**bid**

**category\_id**

**condition**

**create\_dict** (*\*\*kwargs*) → dict

Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.

**create\_xml** (*\*\*kwargs*) → xml.etree.ElementTree.Element

Must be inherited by each child class. Describes the logic for creating a xml with data from a model.

**credit\_template\_id**

**currency**

**description**

**dimensions**

**static from\_xml** (*offer\_el: xml.etree.ElementTree.Element, \*\*mapping*) → dict

Abstract method for parsing the xml element into a dictionary.

**isbn**

**language**

**name**

**offer\_id**

**old\_price**

**part**

**pictures**

**price**

**publisher**

**sales\_notes**

**series**

**supplier**

**table\_of\_contents**

**url**

**vendor**

**vendor\_code**

**volume**

```

class yandex_market_language.models.offers.AbstractOffer(offer_id, url, price: yan-
dex_market_language.models.price.Price,
currency: str, cate-
gory_id, vendor=None,
vendor_code=None,
bid=None, cbid=None,
old_price=None, en-
able_auto_discounts=None,
pictures: List[str]
= None, sup-
plier=None, deliv-
ery=True, pickup=True,
delivery_options:
List[yandex_market_language.models.option.Opti
= None, pickup_options:
List[yandex_market_language.models.option.Opti
= None, store=None, de-
scription: str = None,
sales_notes: str =
None, min_quantity=1,
manufac-
turer_warranty=None,
coun-
try_of_origin=None,
adult=None, bar-
codes: List[str] =
None, parameters:
List[yandex_market_language.models.parameter.I
= None, condition: yan-
dex_market_language.models.condition.Condition
=
None,
credit_template_id: str =
None, expiry=None,
weight=None, di-
mensions: yan-
dex_market_language.models.dimensions.Dimens
= None, download-
able=None, avail-
able=None, age: yan-
dex_market_language.models.age.Age
=
None,
group_id=None)

```

Bases: `yandex_market_language.models.fields.enable_auto_discounts.EnableAutoDiscountField`, `yandex_market_language.models.fields.options.DeliveryOptionsField`, `yandex_market_language.models.fields.options.PickupOptionsField`, `yandex_market_language.models.abstract.AbstractModel`, `abc.ABC`

Abstract offer model for all other offer models.

**adult**

**age**

**available**

**barcodes**

`bid`

`category_id`

`cbid`

`condition`

`create_dict` (*\*\*kwargs*) → dict

Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.

`create_xml` (*\*\*kwargs*) → xml.etree.ElementTree.Element

Must be inherited by each child class. Describes the logic for creating a xml with data from a model.

`credit_template_id`

`currency`

`delivery`

`description`

`dimensions`

`downloadable`

`expiry`

`static from_xml` (*offer\_el: xml.etree.ElementTree.Element, \*\*mapping*) → dict

Abstract method for parsing the xml element into a dictionary.

`group_id`

`manufacturer_warranty`

`min_quantity`

`offer_id`

`old_price`

`parameters`

`pickup`

`pictures`

`price`

`sales_notes`

`store`

`supplier`

`url`

`vendor`

`vendor_code`

`weight`

```
class yandex_market_language.models.offers.AlcoholOffer (name: str, vendor: str, barcodes: List[str], parameters: List[yandex_market_language.models.parameter.Parameter], **kwargs)
```

Bases: `yandex_market_language.models.offers.AbstractOffer`

Alcohol offer.

Docs: <https://yandex.ru/support/partnermarket/export/alcohol.html>

**age**

**barcodes**

**bid**

**category\_id**

**condition**

**create\_dict** (*\*\*kwargs*) → dict

Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.

**create\_xml** (*\*\*kwargs*) → xml.etree.ElementTree.Element

Must be inherited by each child class. Describes the logic for creating a xml with data from a model.

**credit\_template\_id**

**currency**

**description**

**dimensions**

**static from\_xml** (*offer\_el: xml.etree.ElementTree.Element, \*\*mapping*) → yandex\_market\_language.models.offers.AlcoholOffer

Abstract method for parsing the xml element into a dictionary.

**name**

**offer\_id**

**old\_price**

**parameters**

**pictures**

**price**

**sales\_notes**

**supplier**

**url**

**vendor**

**vendor\_code**

```
class yandex_market_language.models.offers.ArbitraryOffer (model: str, vendor: str,
                                                         type_prefix: str = None,
                                                         **kwargs)
```

Bases: *yandex\_market\_language.models.offers.AbstractOffer*

Arbitrary offer. In an arbitrary type, the manufacturer, type and name of the product must be indicated in separate elements - model, vendor & typePrefix.

Yandex.Market docs: <https://yandex.ru/support/partnermarket/export/vendor-model.html>

**age**

**barcodes**

**bid**

`category_id`

`condition`

`create_dict` (*\*\*kwargs*) → dict

Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.

`create_xml` (*\*\*kwargs*) → xml.etree.ElementTree.Element

Must be inherited by each child class. Describes the logic for creating a xml with data from a model.

`credit_template_id`

`currency`

`description`

`dimensions`

`static from_xml` (*offer\_el*: `xml.etree.ElementTree.Element`, *\*\*mapping*) → `yandex_market_language.models.offers.ArbitraryOffer`

Abstract method for parsing the xml element into a dictionary.

`model`

`offer_id`

`old_price`

`pictures`

`price`

`sales_notes`

`supplier`

`type_prefix`

`url`

`vendor`

`vendor_code`

```
class yandex_market_language.models.offers.AudioBookOffer (performed_by: str  
= None, performance_type: str =  
None, storage: str =  
None, audio_format:  
str = None, recording_length: str = None,  
**kwargs)
```

Bases: `yandex_market_language.models.offers.AbstractBookOffer`

Audio book offer.

Docs: <https://yandex.ru/support/partnermarket/export/audiobooks.html>

`age`

`audio_format`

`author`

`barcodes`

`bid`

**category\_id**

**condition**

**create\_dict** (*\*\*kwargs*) → dict

Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.

**create\_xml** (*\*\*kwargs*) → xml.etree.ElementTree.Element

Must be inherited by each child class. Describes the logic for creating a xml with data from a model.

**credit\_template\_id**

**currency**

**description**

**dimensions**

**static from\_xml** (*offer\_el: xml.etree.ElementTree.Element, \*\*mapping*) → yandex\_market\_language.models.offers.AudioBookOffer

Abstract method for parsing the xml element into a dictionary.

**isbn**

**language**

**name**

**offer\_id**

**old\_price**

**performance\_type**

**performed\_by**

**pictures**

**price**

**publisher**

**recording\_length**

**sales\_notes**

**series**

**storage**

**supplier**

**table\_of\_contents**

**url**

**vendor**

**vendor\_code**

**class** yandex\_market\_language.models.offers.**BookOffer** (*binding=None, page\_extent=None, \*\*kwargs*)

Bases: *yandex\_market\_language.models.offers.AbstractBookOffer*

Special offer type for books.

Yandex.Market docs: <https://yandex.ru/support/partnermarket/export/books.html>

**age**

**author**

**barcodes**

**bid**

**binding**

**category\_id**

**condition**

**create\_dict** (*\*\*kwargs*) → dict

Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.

**create\_xml** (*\*\*kwargs*) → xml.etree.ElementTree.Element

Must be inherited by each child class. Describes the logic for creating a xml with data from a model.

**credit\_template\_id**

**currency**

**description**

**dimensions**

**static from\_xml** (*offer\_el: xml.etree.ElementTree.Element, \*\*mapping*) → yandex\_market\_language.models.offers.BookOffer

Abstract method for parsing the xml element into a dictionary.

**isbn**

**language**

**name**

**offer\_id**

**old\_price**

**page\_extent**

**pictures**

**price**

**publisher**

**sales\_notes**

**series**

**supplier**

**table\_of\_contents**

**url**

**vendor**

**vendor\_code**

```
class yandex_market_language.models.offers.EventTicketOffer (name: str, place: str, date, hall: str = None, hall_part: str = None, is_premiere=None, is_kids=None, **kwargs)
```

Bases: `yandex_market_language.models.offers.AbstractOffer`

EventTicket offer.

Docs: <https://yandex.ru/support/partnermarket/export/event-tickets.html>

**age**

**barcodes**

**bid**

**category\_id**

**condition**

**create\_dict** (*\*\*kwargs*) → dict

Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.

**create\_xml** (*\*\*kwargs*) → xml.etree.ElementTree.Element

Must be inherited by each child class. Describes the logic for creating a xml with data from a model.

**credit\_template\_id**

**currency**

**date**

**description**

**dimensions**

**static from\_xml** (*offer\_el: xml.etree.ElementTree.Element, \*\*mapping*) → `yandex_market_language.models.offers.EventTicketOffer`

Abstract method for parsing the xml element into a dictionary.

**hall**

**hall\_part**

**is\_kids**

**is\_premiere**

**name**

**offer\_id**

**old\_price**

**pictures**

**place**

**price**

**sales\_notes**

**supplier**

**url**

**vendor**

**vendor\_code**

**class** `yandex_market_language.models.offers.MedicineOffer` (*name, delivery, pickup, \*\*kwargs*)

Bases: `yandex_market_language.models.offers.AbstractOffer`

Medicine offer.

Docs: <https://yandex.ru/support/partnermarket/export/medicine.html>

**age**

**barcodes**

**bid**

**category\_id**

**condition**

**create\_dict** (*\*\*kwargs*) → dict

Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.

**create\_xml** (*\*\*kwargs*) → xml.etree.ElementTree.Element

Must be inherited by each child class. Describes the logic for creating a xml with data from a model.

**credit\_template\_id**

**currency**

**description**

**dimensions**

**static from\_xml** (*offer\_el: xml.etree.ElementTree.Element, \*\*mapping*) → yandex\_market\_language.models.offers.MedicineOffer

Abstract method for parsing the xml element into a dictionary.

**name**

**offer\_id**

**old\_price**

**pictures**

**price**

**sales\_notes**

**supplier**

**url**

**vendor**

**vendor\_code**

```
class yandex_market_language.models.offers.MusicVideoOffer (title: str, artist: str
                                                    = None, year=None,
                                                    media: str = None,
                                                    starring: str = None,
                                                    director: str = None,
                                                    original_name: str =
                                                    None, country: str =
                                                    None, **kwargs)
```

Bases: `yandex_market_language.models.fields.year.YearField`,  
`yandex_market_language.models.offers.AbstractOffer`

Music or video offer.

Docs: <https://yandex.ru/support/partnermarket/export/music-video.html>

**age**

**artist**

**barcodes**

**bid**

**category\_id**

**condition**

**country**

**create\_dict** (*\*\*kwargs*) → dict

Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.

**create\_xml** (*\*\*kwargs*) → xml.etree.ElementTree.Element

Must be inherited by each child class. Describes the logic for creating a xml with data from a model.

**credit\_template\_id**

**currency**

**description**

**dimensions**

**director**

**static from\_xml** (*offer\_el: xml.etree.ElementTree.Element, \*\*mapping*) → yandex\_market\_language.models.offers.MusicVideoOffer

Abstract method for parsing the xml element into a dictionary.

**media**

**offer\_id**

**old\_price**

**original\_name**

**pictures**

**price**

**sales\_notes**

**starring**

**supplier**

**title**

**url**

**vendor**

**vendor\_code**

**class** yandex\_market\_language.models.offers.**SimplifiedOffer** (*name, \*\*kwargs*)

Bases: *yandex\_market\_language.models.offers.AbstractOffer*

Simplified offer. In a simplified type, the manufacturer, type and name of the goods are indicated in one element - name.

Yandex.Market docs: <https://yandex.ru/support/partnermarket/offers.html>

**age**

**barcodes**

**bid**

**category\_id**

**condition**

**create\_dict** (*\*\*kwargs*) → dict

Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.

**create\_xml** (*\*\*kwargs*) → xml.etree.ElementTree.Element

Must be inherited by each child class. Describes the logic for creating a xml with data from a model.

**credit\_template\_id**

**currency**

**description**

**dimensions**

**static from\_xml** (*offer\_el: xml.etree.ElementTree.Element, \*\*mapping*) → yandex\_market\_language.models.offers.SimplifiedOffer

Abstract method for parsing the xml element into a dictionary.

**name**

**offer\_id**

**old\_price**

**pictures**

**price**

**sales\_notes**

**supplier**

**url**

**vendor**

**vendor\_code**

## yandex\_market\_language.models.option module

**class** yandex\_market\_language.models.option.**Option** (*cost, days, order\_before=None*)

Bases: yandex\_market\_language.models.abstract.AbstractModel

Option model for both the delivery option and the pickup option.

Docs: <https://yandex.ru/support/partnermarket/elements/delivery-options.html> <https://yandex.ru/support/partnermarket/elements/pickup-options.html>

**cost**

**create\_dict** (*\*\*kwargs*) → dict

Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.

**create\_xml** (*\*\*kwargs*) → xml.etree.ElementTree.Element

Must be inherited by each child class. Describes the logic for creating a xml with data from a model.

**days**

**static from\_xml** (*el*: `xml.etree.ElementTree.Element`) → `yandex_market_language.models.option.Option`  
 Must be inherited by each child class. Describes the logic for creating a model from an xml element.

**order\_before**

### yandex\_market\_language.models.price module

**class** `yandex_market_language.models.price.Price` (*value*, *is\_starting=False*)  
 Bases: `yandex_market_language.models.abstract.AbstractModel`

Actual offer price model.

**create\_dict** (*\*\*kwargs*) → dict  
 Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.

**create\_xml** (*\*\*kwargs*) → `xml.etree.ElementTree.Element`  
 Must be inherited by each child class. Describes the logic for creating a xml with data from a model.

**static from\_xml** (*el*: `xml.etree.ElementTree.Element`) → `yandex_market_language.models.abstract.AbstractModel`  
 Must be inherited by each child class. Describes the logic for creating a model from an xml element.

**is\_starting**

**value**

### yandex\_market\_language.models.shop module

**class** `yandex_market_language.models.shop.Shop` (*name*: *str*, *company*: *str*, *url*: *str*, *currencies*: *List[models.Currency]*, *categories*: *List[models.Category]*, *offers*: *List[models.offers.AbstractOffer]*, *platform*: *str = None*, *version*: *str = None*, *agency*: *str = None*, *email*: *str = None*, *delivery\_options*: *List[models.Option] = None*, *pickup\_options*: *List[models.Option] = None*, *enable\_auto\_discounts*: *None*, *gifts*: *List[models.Gift] = None*, *promos*: *List[models.Promo] = None*)

Bases: `yandex_market_language.models.fields.enable_auto_discounts.EnableAutoDiscountField`, `yandex_market_language.models.fields.options.DeliveryOptionsField`, `yandex_market_language.models.fields.options.PickupOptionsField`, `yandex_market_language.models.abstract.AbstractModel`

Shop model.

Docs: <https://yandex.ru/support/partnermarket/elements/shop.html>

**agency**

**categories**

**company**

**create\_dict** (*\*\*kwargs*) → dict  
 Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.

**create\_xml** (*\*\*kwargs*) → `xml.etree.ElementTree.Element`  
Must be inherited by each child class. Describes the logic for creating a xml with data from a model.

**currencies**

**email**

**static from\_xml** (*shop\_el:* `xml.etree.ElementTree.Element`) → `yandex_market_language.models.shop.Shop`  
Must be inherited by each child class. Describes the logic for creating a model from an xml element.

**gifts**

**name**

**offers**

**platform**

**promos**

**url**

**version**

## Module contents

**class** `yandex_market_language.models.AbstractModel`

Bases: `abc.ABC`

Abstract model for creating child models.

**clean\_dict**

A helper property to get clean dictionary with data.

**create\_dict** (*\*\*kwargs*) → `dict`

Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.

**create\_xml** (*\*\*kwargs*) → `xml.etree.ElementTree.Element`

Must be inherited by each child class. Describes the logic for creating a xml with data from a model.

**static from\_xml** (*el:* `xml.etree.ElementTree.Element`) → `yandex_market_language.models.abstract.AbstractModel`

Must be inherited by each child class. Describes the logic for creating a model from an xml element.

**to\_dict** (*clean: bool = False*) → `dict`

Calls the inherited method to create the dictionary and returns a clean dictionary if the parameter was set.

**to\_xml** (*root\_el: xml.etree.ElementTree.Element = None*) → `xml.etree.ElementTree.Element`

Calls the inherited method to create the element and appends it to the parent, if it was set.

**class** `yandex_market_language.models.Feed` (*shop: yandex\_market\_language.models.shop.Shop,*  
*date: None.datetime.date = None*)

Bases: `yandex_market_language.models.abstract.AbstractModel`

YML Feed model.

Docs: <https://yandex.ru/support/partnermarket/export/yml.html>

**create\_dict** (*\*\*kwargs*) → `dict`

Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.

**create\_xml** (*\*\*kwargs*) → `xml.etree.ElementTree.Element`

Must be inherited by each child class. Describes the logic for creating a xml with data from a model.

**date**

**static from\_xml** (*el*: `xml.etree.ElementTree.Element`) → `yandex_market_language.models.feed.Feed`

Must be inherited by each child class. Describes the logic for creating a model from an xml element.

**shop**

**class** `yandex_market_language.models.Shop` (*name*: `str`, *company*: `str`, *url*: `str`, *currencies*: `List[models.Currency]`, *categories*: `List[models.Category]`, *offers*: `List[models.offers.AbstractOffer]`, *platform*: `str = None`, *version*: `str = None`, *agency*: `str = None`, *email*: `str = None`, *delivery\_options*: `List[models.Option] = None`, *pickup\_options*: `List[models.Option] = None`, *enable\_auto\_discounts*: `bool = None`, *gifts*: `List[models.Gift] = None`, *promos*: `List[models.Promo] = None`)

Bases: `yandex_market_language.models.fields.enable_auto_discounts.EnableAutoDiscountField`, `yandex_market_language.models.fields.options.DeliveryOptionsField`, `yandex_market_language.models.fields.options.PickupOptionsField`, `yandex_market_language.models.abstract.AbstractModel`

Shop model.

Docs: <https://yandex.ru/support/partnermarket/elements/shop.html>

**agency**

**categories**

**company**

**create\_dict** (*\*\*kwargs*) → `dict`

Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.

**create\_xml** (*\*\*kwargs*) → `xml.etree.ElementTree.Element`

Must be inherited by each child class. Describes the logic for creating a xml with data from a model.

**currencies**

**email**

**static from\_xml** (*shop\_el*: `xml.etree.ElementTree.Element`) → `yandex_market_language.models.shop.Shop`

Must be inherited by each child class. Describes the logic for creating a model from an xml element.

**gifts**

**name**

**offers**

**platform**

**promos**

**url**

**version**

**class** `yandex_market_language.models.Currency` (*currency*, *rate*, *plus=None*)

Bases: `yandex_market_language.models.abstract.AbstractModel`

Currency model. Used to create a list of shop currency rates.

Docs: <https://yandex.ru/support/partnermarket/elements/currencies.html>

**create\_dict** (\*\*kwargs) → dict

Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.

**create\_xml** (\*\*kwargs) → xml.etree.ElementTree.Element

Must be inherited by each child class. Describes the logic for creating a xml with data from a model.

**currency**

**static from\_xml** (el: *xml.etree.ElementTree.Element*) → *yandex\_market\_language.models.currency.Currency*

Must be inherited by each child class. Describes the logic for creating a model from an xml element.

**plus**

**rate**

**class** `yandex_market_language.models.Category` (*category\_id, name, parent\_id=None*)

Bases: `yandex_market_language.models.abstract.AbstractModel`

Category model for shop.

Docs: <https://yandex.ru/support/partnermarket/elements/categories.html>

**category\_id**

**create\_dict** (\*\*kwargs) → dict

Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.

**create\_xml** (\*\*kwargs) → xml.etree.ElementTree.Element

Must be inherited by each child class. Describes the logic for creating a xml with data from a model.

**static from\_xml** (el: *xml.etree.ElementTree.Element*) → *yandex\_market\_language.models.category.Category*

Must be inherited by each child class. Describes the logic for creating a model from an xml element.

**name**

**parent\_id**

**class** `yandex_market_language.models.Option` (*cost, days, order\_before=None*)

Bases: `yandex_market_language.models.abstract.AbstractModel`

Option model for both the delivery option and the pickup option.

Docs: <https://yandex.ru/support/partnermarket/elements/delivery-options.html> <https://yandex.ru/support/partnermarket/elements/pickup-options.html>

**cost**

**create\_dict** (\*\*kwargs) → dict

Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.

**create\_xml** (\*\*kwargs) → xml.etree.ElementTree.Element

Must be inherited by each child class. Describes the logic for creating a xml with data from a model.

**days**

**static from\_xml** (el: *xml.etree.ElementTree.Element*) → *yandex\_market\_language.models.option.Option*

Must be inherited by each child class. Describes the logic for creating a model from an xml element.

**order\_before**

```
class yandex_market_language.models.Price (value, is_starting=False)
    Bases: yandex_market_language.models.abstract.AbstractModel

    Actual offer price model.

    create_dict (**kwargs) → dict
        Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.

    create_xml (**kwargs) → xml.etree.ElementTree.Element
        Must be inherited by each child class. Describes the logic for creating a xml with data from a model.

    static from_xml (el: xml.etree.ElementTree.Element) → yandex_market_language.models.abstract.AbstractModel
        Must be inherited by each child class. Describes the logic for creating a model from an xml element.

is_starting
value

class yandex_market_language.models.SimplifiedOffer (name, **kwargs)
    Bases: yandex_market_language.models.offers.AbstractOffer

    Simplified offer. In a simplified type, the manufacturer, type and name of the goods are indicated in one element
    - name.

    Yandex.Market docs: https://yandex.ru/support/partnermarket/offers.html

age
barcodes
bid
category_id
condition
create_dict (**kwargs) → dict
    Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.

create_xml (**kwargs) → xml.etree.ElementTree.Element
    Must be inherited by each child class. Describes the logic for creating a xml with data from a model.

credit_template_id
currency
description
dimensions
static from_xml (offer_el: xml.etree.ElementTree.Element, **mapping) → yandex_market_language.models.offers.SimplifiedOffer
    Abstract method for parsing the xml element into a dictionary.

name
offer_id
old_price
pictures
price
sales_notes
supplier
```

`url`

`vendor`

`vendor_code`

```
class yandex_market_language.models.ArbitraryOffer (model: str, vendor: str,
                                                    type_prefix: str = None,
                                                    **kwargs)
```

Bases: `yandex_market_language.models.offers.AbstractOffer`

Arbitrary offer. In an arbitrary type, the manufacturer, type and name of the product must be indicated in separate elements - model, vendor & typePrefix.

Yandex.Market docs: <https://yandex.ru/support/partnermarket/export/vendor-model.html>

`age`

`barcodes`

`bid`

`category_id`

`condition`

`create_dict` (*\*\*kwargs*) → dict

Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.

`create_xml` (*\*\*kwargs*) → xml.etree.ElementTree.Element

Must be inherited by each child class. Describes the logic for creating a xml with data from a model.

`credit_template_id`

`currency`

`description`

`dimensions`

`static from_xml` (*offer\_el*: xml.etree.ElementTree.Element, *\*\*mapping*) → yandex\_market\_language.models.offers.ArbitraryOffer

Abstract method for parsing the xml element into a dictionary.

`model`

`offer_id`

`old_price`

`pictures`

`price`

`sales_notes`

`supplier`

`type_prefix`

`url`

`vendor`

`vendor_code`

```
class yandex_market_language.models.BookOffer (binding=None, page_extent=None,  
                                               **kwargs)  
    Bases: yandex_market_language.models.offers.AbstractBookOffer  
    Special offer type for books.  
    Yandex.Market docs: https://yandex.ru/support/partnermarket/export/books.html  
age  
author  
barcodes  
bid  
binding  
category_id  
condition  
create_dict (**kwargs) → dict  
    Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.  
create_xml (**kwargs) → xml.etree.ElementTree.Element  
    Must be inherited by each child class. Describes the logic for creating a xml with data from a model.  
credit_template_id  
currency  
description  
dimensions  
static from_xml (offer_el: xml.etree.ElementTree.Element, **mapping) → yan-  
                    dex_market_language.models.offers.BookOffer  
    Abstract method for parsing the xml element into a dictionary.  
isbn  
language  
name  
offer_id  
old_price  
page_extent  
pictures  
price  
publisher  
sales_notes  
series  
supplier  
table_of_contents  
url  
vendor
```

`vendor_code`

```
class yandex_market_language.models.AudioBookOffer (performed_by: str = None, performance_type: str = None, storage: str = None, audio_format: str = None, recording_length: str = None, **kwargs)
```

Bases: `yandex_market_language.models.offers.AbstractBookOffer`

Audio book offer.

Docs: <https://yandex.ru/support/partnermarket/export/audiobooks.html>

`age`

`audio_format`

`author`

`barcodes`

`bid`

`category_id`

`condition`

`create_dict` (*\*\*kwargs*) → dict

Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.

`create_xml` (*\*\*kwargs*) → `xml.etree.ElementTree.Element`

Must be inherited by each child class. Describes the logic for creating a xml with data from a model.

`credit_template_id`

`currency`

`description`

`dimensions`

`static from_xml` (*offer\_el: xml.etree.ElementTree.Element, \*\*mapping*) → `yandex_market_language.models.offers.AudioBookOffer`

Abstract method for parsing the xml element into a dictionary.

`isbn`

`language`

`name`

`offer_id`

`old_price`

`performance_type`

`performed_by`

`pictures`

`price`

`publisher`

`recording_length`

`sales_notes`

**series**

**storage**

**supplier**

**table\_of\_contents**

**url**

**vendor**

**vendor\_code**

```
class yandex_market_language.models.MusicVideoOffer(title: str, artist: str = None,
                                                    year=None, media: str = None,
                                                    starring: str = None, director:
                                                    str = None, original_name: str
                                                    = None, country: str = None,
                                                    **kwargs)
```

Bases: `yandex_market_language.models.fields.year.YearField`,  
`yandex_market_language.models.offers.AbstractOffer`

Music or video offer.

Docs: <https://yandex.ru/support/partnermarket/export/music-video.html>

**age**

**artist**

**barcodes**

**bid**

**category\_id**

**condition**

**country**

**create\_dict** (\*\*kwargs) → dict

Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.

**create\_xml** (\*\*kwargs) → xml.etree.ElementTree.Element

Must be inherited by each child class. Describes the logic for creating a xml with data from a model.

**credit\_template\_id**

**currency**

**description**

**dimensions**

**director**

**static from\_xml** (offer\_el: `xml.etree.ElementTree.Element`, \*\*mapping) → `yandex_market_language.models.offers.MusicVideoOffer`

Abstract method for parsing the xml element into a dictionary.

**media**

**offer\_id**

**old\_price**

**original\_name**

`pictures`  
`price`  
`sales_notes`  
`starring`  
`supplier`  
`title`  
`url`  
`vendor`  
`vendor_code`

**class** `yandex_market_language.models.MedicineOffer` (*name, delivery, pickup, \*\*kwargs*)  
Bases: `yandex_market_language.models.offers.AbstractOffer`

Medicine offer.

Docs: <https://yandex.ru/support/partnermarket/export/medicine.html>

`age`  
`barcodes`  
`bid`  
`category_id`  
`condition`

**create\_dict** (*\*\*kwargs*) → dict

Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.

**create\_xml** (*\*\*kwargs*) → `xml.etree.ElementTree.Element`

Must be inherited by each child class. Describes the logic for creating a xml with data from a model.

`credit_template_id`

`currency`

`description`

`dimensions`

**static from\_xml** (*offer\_el: xml.etree.ElementTree.Element, \*\*mapping*) → `yandex_market_language.models.offers.MedicineOffer`

Abstract method for parsing the xml element into a dictionary.

`name`  
`offer_id`  
`old_price`  
`pictures`  
`price`  
`sales_notes`  
`supplier`  
`url`  
`vendor`

**vendor\_code**

```
class yandex_market_language.models.EventTicketOffer(name: str, place: str, date,  
hall: str = None, hall_part:  
str = None, is_premiere=None,  
is_kids=None, **kwargs)
```

Bases: `yandex_market_language.models.offers.AbstractOffer`

EventTicket offer.

Docs: <https://yandex.ru/support/partnermarket/export/event-tickets.html>

**age**

**barcodes**

**bid**

**category\_id**

**condition**

**create\_dict** (*\*\*kwargs*) → dict

Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.

**create\_xml** (*\*\*kwargs*) → xml.etree.ElementTree.Element

Must be inherited by each child class. Describes the logic for creating a xml with data from a model.

**credit\_template\_id**

**currency**

**date**

**description**

**dimensions**

**static from\_xml** (*offer\_el: xml.etree.ElementTree.Element, \*\*mapping*) → yandex\_market\_language.models.offers.EventTicketOffer

Abstract method for parsing the xml element into a dictionary.

**hall**

**hall\_part**

**is\_kids**

**is\_premiere**

**name**

**offer\_id**

**old\_price**

**pictures**

**place**

**price**

**sales\_notes**

**supplier**

**url**

**vendor**

**vendor\_code**

```
class yandex_market_language.models.AlcoholOffer (name: str, vendor: str, bar-  
codes: List[str], parameters:  
List[yandex_market_language.models.parameter.Parameter],  
**kwargs)
```

Bases: `yandex_market_language.models.offers.AbstractOffer`

Alcohol offer.

Docs: <https://yandex.ru/support/partnermarket/export/alcohol.html>

**age**

**barcodes**

**bid**

**category\_id**

**condition**

**create\_dict** (*\*\*kwargs*) → dict

Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.

**create\_xml** (*\*\*kwargs*) → `xml.etree.ElementTree.Element`

Must be inherited by each child class. Describes the logic for creating a xml with data from a model.

**credit\_template\_id**

**currency**

**description**

**dimensions**

**static from\_xml** (*offer\_el: xml.etree.ElementTree.Element, \*\*mapping*) → `yan-`  
`dex_market_language.models.offers.AlcoholOffer`

Abstract method for parsing the xml element into a dictionary.

**name**

**offer\_id**

**old\_price**

**parameters**

**pictures**

**price**

**sales\_notes**

**supplier**

**url**

**vendor**

**vendor\_code**

```
class yandex_market_language.models.Parameter (name: str, value: str, unit: str = None)
```

Bases: `yandex_market_language.models.abstract.AbstractModel`

Model of offer characteristics and parameters.

Docs: <https://yandex.ru/support/partnermarket/elements/param.html>

**create\_dict** (*\*\*kwargs*) → dict  
 Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.

**create\_xml** (*\*\*kwargs*) → xml.etree.ElementTree.Element  
 Must be inherited by each child class. Describes the logic for creating a xml with data from a model.

**static from\_xml** (*el*: *xml.etree.ElementTree.Element*) → *yandex\_market\_language.models.parameter.Parameter*  
 Must be inherited by each child class. Describes the logic for creating a model from an xml element.

**name**  
**unit**  
**value**

**class** *yandex\_market\_language.models.Condition* (*condition\_type: str, reason: str*)  
 Bases: *yandex\_market\_language.models.abstract.AbstractModel*

Condition model. Used for used goods and goods discounted due to deficiencies.

Docs: <https://yandex.ru/support/partnermarket/elements/condition.html>

**condition\_type**

**create\_dict** (*\*\*kwargs*) → dict  
 Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.

**create\_xml** (*\*\*kwargs*) → xml.etree.ElementTree.Element  
 Must be inherited by each child class. Describes the logic for creating a xml with data from a model.

**static from\_xml** (*el*: *xml.etree.ElementTree.Element*) → *yandex\_market\_language.models.condition.Condition*  
 Must be inherited by each child class. Describes the logic for creating a model from an xml element.

**reason**

**class** *yandex\_market\_language.models.Dimensions* (*length, wight, height*)  
 Bases: *yandex\_market\_language.models.abstract.AbstractModel*

Offer dimensions model (length, width, height) in the package in centimeters.

**create\_dict** (*\*\*kwargs*) → dict  
 Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.

**create\_xml** (*\*\*kwargs*) → xml.etree.ElementTree.Element  
 Must be inherited by each child class. Describes the logic for creating a xml with data from a model.

**static from\_xml** (*el*: *xml.etree.ElementTree.Element*) → *yandex\_market\_language.models.dimensions.Dimensions*  
 Must be inherited by each child class. Describes the logic for creating a model from an xml element.

**height**  
**length**  
**width**

**class** *yandex\_market\_language.models.Age* (*unit: str, value*)  
 Bases: *yandex\_market\_language.models.abstract.AbstractModel*

Age category of the offer.

**create\_dict** (*\*\*kwargs*) → dict  
 Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.

**create\_xml** (*\*\*kwargs*) → `xml.etree.ElementTree.Element`  
 Must be inherited by each child class. Describes the logic for creating a xml with data from a model.

**static from\_xml** (*el*: `xml.etree.ElementTree.Element`) → `yandex_market_language.models.age.Age`  
 Must be inherited by each child class. Describes the logic for creating a model from an xml element.

**unit**

**value**

**class** `yandex_market_language.models.Gift` (*id*: `str`, *name*: `str`, *pictures*: `List[str] = None`)  
 Bases: `yandex_market_language.models.abstract.AbstractModel`

Gift model.

Docs: <https://yandex.ru/support/partnermarket/elements/promo-gift.html>

**create\_dict** (*\*\*kwargs*) → `dict`  
 Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.

**create\_xml** (*\*\*kwargs*) → `xml.etree.ElementTree.Element`  
 Must be inherited by each child class. Describes the logic for creating a xml with data from a model.

**static from\_xml** (*gift\_el*: `xml.etree.ElementTree.Element`) → `yandex_market_language.models.gift.Gift`  
 Must be inherited by each child class. Describes the logic for creating a model from an xml element.

**id**

**name**

**pictures**

**class** `yandex_market_language.models.Promo` (*promo\_id*: `str`, *promo\_type*: `str`, *purchase*: `yandex_market_language.models.promo.Purchase`, *promo\_gifts*: `List[PromoGift]`, *start\_date*=`None`, *end\_date*=`None`, *description*=`None`, *url*=`None`)

Bases: `yandex_market_language.models.abstract.AbstractModel`

Docs: <https://yandex.ru/support/partnermarket/elements/promo-gift.html>

**MAPPING** = {'description': 'description', 'end-date': 'end\_date', 'start-date': 'start\_date'}

**create\_dict** (*\*\*kwargs*) → `dict`  
 Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.

**create\_xml** (*\*\*kwargs*) → `xml.etree.ElementTree.Element`  
 Must be inherited by each child class. Describes the logic for creating a xml with data from a model.

**description**

**end\_date**

**classmethod from\_xml** (*promo\_el*: `xml.etree.ElementTree.Element`) → `yandex_market_language.models.promo.Promo`  
 Must be inherited by each child class. Describes the logic for creating a model from an xml element.

**promo\_gifts**

**promo\_id**

**promo\_type**

**purchase**

`start_date`

`url`

**class** `yandex_market_language.models.Purchase` (`products: List[Product]`, `required_quantity='1'`)

Bases: `yandex_market_language.models.abstract.AbstractModel`

Docs: <https://yandex.ru/support/partnermarket/elements/promo-gift.html>

**create\_dict** (`**kwargs`) → dict

Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.

**create\_xml** (`**kwargs`) → `xml.etree.ElementTree.Element`

Must be inherited by each child class. Describes the logic for creating a xml with data from a model.

**static from\_xml** (`purchase_el: xml.etree.ElementTree.Element`) → `yandex_market_language.models.promo.Purchase`

Must be inherited by each child class. Describes the logic for creating a model from an xml element.

`products`

`required_quantity`

**class** `yandex_market_language.models.Product` (`offer_id: str = None`, `category_id: str = None`)

Bases: `yandex_market_language.models.abstract.AbstractModel`

Docs: <https://yandex.ru/support/partnermarket/elements/promo-gift.html>

`category_id`

**create\_dict** (`**kwargs`) → dict

Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.

**create\_xml** (`**kwargs`) → `xml.etree.ElementTree.Element`

Must be inherited by each child class. Describes the logic for creating a xml with data from a model.

**static from\_xml** (`product_el: xml.etree.ElementTree.Element`) → `yandex_market_language.models.promo.Product`

Must be inherited by each child class. Describes the logic for creating a model from an xml element.

`offer_id`

**class** `yandex_market_language.models.PromoGift` (`offer_id: str = None`, `gift_id: str = None`)

Bases: `yandex_market_language.models.abstract.AbstractModel`

Docs: <https://yandex.ru/support/partnermarket/elements/promo-gift.html>

**create\_dict** (`**kwargs`) → dict

Must be inherited by each child class. Describes the logic for creating a dictionary with data from a model.

**create\_xml** (`**kwargs`) → `xml.etree.ElementTree.Element`

Must be inherited by each child class. Describes the logic for creating a xml with data from a model.

**static from\_xml** (`el: xml.etree.ElementTree.Element`) → `yandex_market_language.models.promo.PromoGift`

Must be inherited by each child class. Describes the logic for creating a model from an xml element.

`gift_id`

`offer_id`

## 4.1.2 Submodules

### 4.1.3 `yandex_market_language.exceptions` module

**exception** `yandex_market_language.exceptions.ParseError`  
Bases: `yandex_market_language.exceptions.YMLException`  
Base parse exception.

**exception** `yandex_market_language.exceptions.ValidationError`  
Bases: `yandex_market_language.exceptions.YMLException`  
Data validation exception.

**exception** `yandex_market_language.exceptions.YMLException`  
Bases: `Exception`  
Basic error for the whole module.

### 4.1.4 `yandex_market_language.yml` module

**class** `yandex_market_language.yml.YML` (*file\_or\_path*)  
Bases: `object`

Main class for feed parse and conversion.

**convert** (*feed: yandex\_market\_language.models.feed.Feed, pretty: bool = True*)  
Converts Feed model to XML file.

**parse** () → `yandex_market_language.models.feed.Feed`  
Parses an XML feed file to the Feed model.

**static prettify\_el** (*el: xml.etree.ElementTree.Element*) → `xml.etree.ElementTree.Element`  
Return a pretty-printed XML string for the Element.

`yandex_market_language.yml.convert` (*file\_or\_path, feed: yandex\_market\_language.models.feed.Feed, pretty: bool = True*)

`yandex_market_language.yml.parse` (*file\_or\_path*)

### 4.1.5 Module contents

Top-level package for Yandex Market Language (YML) for Python.

`yandex_market_language.parse` (*file\_or\_path*)

`yandex_market_language.convert` (*file\_or\_path, feed: yandex\_market\_language.models.feed.Feed, pretty: bool = True*)

Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given. You can contribute in many ways:

## 5.1 Types of Contributions

### 5.1.1 Report Bugs

Report bugs at [https://github.com/stefanitsky/yandex\\_market\\_language/issues](https://github.com/stefanitsky/yandex_market_language/issues).

If you are reporting a bug, please include:

- Your operating system name and version.
- Any details about your local setup that might be helpful in troubleshooting.
- Detailed steps to reproduce the bug.

### 5.1.2 Fix Bugs

Look through the GitHub issues for bugs. Anything tagged with “bug” and “help wanted” is open to whoever wants to implement it.

### 5.1.3 Implement Features

Look through the GitHub issues for features. Anything tagged with “enhancement” and “help wanted” is open to whoever wants to implement it.

## 5.1.4 Write Documentation

Yandex Market Language (YML) for Python could always use more documentation, whether as part of the official Yandex Market Language (YML) for Python docs, in docstrings, or even on the web in blog posts, articles, and such.

## 5.1.5 Submit Feedback

The best way to send feedback is to file an issue at [https://github.com/stefanitsky/yandex\\_market\\_language/issues](https://github.com/stefanitsky/yandex_market_language/issues).

If you are proposing a feature:

- Explain in detail how it would work.
- Keep the scope as narrow as possible, to make it easier to implement.
- Remember that this is a volunteer-driven project, and that contributions are welcome :)

## 5.2 Get Started!

Ready to contribute? Here's how to set up *yandex\_market\_language* for local development.

1. Fork the *yandex\_market\_language* repo on GitHub.
2. Clone your fork locally:

```
$ git clone git@github.com:your_name_here/yandex_market_language.git
```

3. Install your local copy into a virtualenv. Assuming you have *virtualenvwrapper* installed, this is how you set up your fork for local development:

```
$ mkvirtualenv yandex_market_language
$ cd yandex_market_language/
$ python setup.py develop
```

4. Create a branch for local development:

```
$ git checkout -b name-of-your-bugfix-or-feature
```

Now you can make your changes locally.

5. When you're done making changes, check that your changes pass *flake8* and the tests, including testing other Python versions with *tox*:

```
$ flake8 yandex_market_language tests
$ python setup.py test or pytest
$ tox
```

To get *flake8* and *tox*, just *pip* install them into your virtualenv.

6. Commit your changes and push your branch to GitHub:

```
$ git add .
$ git commit -m "Your detailed description of your changes."
$ git push origin name-of-your-bugfix-or-feature
```

7. Submit a pull request through the GitHub website.

## 5.3 Pull Request Guidelines

Before you submit a pull request, check that it meets these guidelines:

1. The pull request should include tests.
2. If the pull request adds functionality, the docs should be updated. Put your new functionality into a function with a docstring, and add the feature to the list in README.rst.
3. The pull request should work for Python 3.5, 3.6, 3.7 and 3.8, and for PyPy. Check [https://travis-ci.com/stefanitsky/yandex\\_market\\_language/pull\\_requests](https://travis-ci.com/stefanitsky/yandex_market_language/pull_requests) and make sure that the tests pass for all supported Python versions.

## 5.4 Tips

To run a subset of tests:

```
$ pytest tests.test_yandex_market_language
```

## 5.5 Deploying

A reminder for the maintainers on how to deploy. Make sure all your changes are committed (including an entry in HISTORY.rst). Then run:

```
$ bump2version patch # possible: major / minor / patch
$ git push
$ git push --tags
```

Travis will then deploy to PyPI if tests pass.



### 6.1 Development Lead

- Alexandr Stefanitsky-Mozdor <stefanitsky.mozdor@gmail.com>

### 6.2 Contributors

@Divalentor



### 7.1 0.6.2 (2021-30-03)

- Change license type to from GPLv2 to MIT.

### 7.2 0.6.1 (2020-07-28)

- Add `__slots__` for all models to reduce RAM usage.

### 7.3 0.6.0 (2020-04-06)

- Added all missed models: Gifts & Promos.
- Added `cbid` warning message on set (field is deprecated).
- Added creation of an XML file from the Feed model.

### 7.4 0.5.0 (2020-04-02)

- Added missed offers of types: audiobook, artist.title, medicine, event-ticket & alco.
- Some fixes for fields.

### 7.5 0.4.0 (2020-04-01)

- Added xml parsing for all models, except: Gifts, Promos and another types of offers like audiobooks, medicine etc.

- Fixed fields parsing for datetime fields & fields that can be None.
- Added new field for offer: supplier.

## 7.6 0.3.0 (2020-03-30)

- All missing fields and models were added for the BaseOffer.
- SimplifiedOffer is now fully supported for xml / dict.
- Custom exception classes removed and replaced with ValidationError.

## 7.7 0.2.0 (2020-03-29)

- Added models for xml to dict and backward support: Category, Currency, Feed, Option (delivery / pickup), Price.
- Added basic models implementation (WIP): Shop, Offers.
- Added basic validation support (WIP, will be improved after finishing of models).

## 7.8 0.1.0 (2020-03-28)

- First release on PyPI.

## CHAPTER 8

---

### Indices and tables

---

- `genindex`
- `modindex`
- `search`



## y

- [yandex\\_market\\_language](#), 36
- [yandex\\_market\\_language.exceptions](#), 36
- [yandex\\_market\\_language.models](#), 22
  - [yandex\\_market\\_language.models.category](#), 8
  - [yandex\\_market\\_language.models.currency](#), 8
  - [yandex\\_market\\_language.models.feed](#), 9
  - [yandex\\_market\\_language.models.fields](#), 7
    - [yandex\\_market\\_language.models.fields.enable\\_auto\\_discounts](#), 7
  - [yandex\\_market\\_language.models.offers](#), 9
  - [yandex\\_market\\_language.models.option](#), 20
  - [yandex\\_market\\_language.models.price](#), 21
  - [yandex\\_market\\_language.models.shop](#), 21
- [yandex\\_market\\_language.yml](#), 36



A

- AbstractBookOffer (class in yandex\_market\_language.models.offers), 9
- AbstractModel (class in yandex\_market\_language.models), 22
- AbstractOffer (class in yandex\_market\_language.models.offers), 10
- adult (yandex\_market\_language.models.offers.AbstractOffer attribute), 11
- Age (class in yandex\_market\_language.models), 33
- age (yandex\_market\_language.models.AlcoholOffer attribute), 32
- age (yandex\_market\_language.models.ArbitraryOffer attribute), 26
- age (yandex\_market\_language.models.AudioBookOffer attribute), 28
- age (yandex\_market\_language.models.BookOffer attribute), 27
- age (yandex\_market\_language.models.EventTicketOffer attribute), 31
- age (yandex\_market\_language.models.MedicineOffer attribute), 30
- age (yandex\_market\_language.models.MusicVideoOffer attribute), 29
- age (yandex\_market\_language.models.offers.AbstractBookOffer attribute), 9
- age (yandex\_market\_language.models.offers.AbstractOffer attribute), 11
- age (yandex\_market\_language.models.offers.AlcoholOffer attribute), 13
- age (yandex\_market\_language.models.offers.ArbitraryOffer attribute), 13
- age (yandex\_market\_language.models.offers.AudioBookOffer attribute), 14
- age (yandex\_market\_language.models.offers.BookOffer attribute), 15
- age (yandex\_market\_language.models.offers.EventTicketOffer attribute), 17
- age (yandex\_market\_language.models.offers.MedicineOffer attribute), 18
- age (yandex\_market\_language.models.offers.MusicVideoOffer attribute), 18
- age (yandex\_market\_language.models.offers.SimplifiedOffer attribute), 19
- age (yandex\_market\_language.models.SimplifiedOffer attribute), 25
- agency (yandex\_market\_language.models.Shop attribute), 23
- agency (yandex\_market\_language.models.shop.Shop attribute), 21
- AlcoholOffer (class in yandex\_market\_language.models), 32
- AlcoholOffer (class in yandex\_market\_language.models.offers), 12
- ArbitraryOffer (class in yandex\_market\_language.models), 26
- ArbitraryOffer (class in yandex\_market\_language.models.offers), 13
- artist (yandex\_market\_language.models.MusicVideoOffer attribute), 29
- artist (yandex\_market\_language.models.offers.MusicVideoOffer attribute), 18
- audio\_format (yandex\_market\_language.models.AudioBookOffer attribute), 28
- audio\_format (yandex\_market\_language.models.offers.AudioBookOffer attribute), 14
- AudioBookOffer (class in yandex\_market\_language.models), 28
- AudioBookOffer (class in yandex\_market\_language.models.offers), 14
- author (yandex\_market\_language.models.AudioBookOffer attribute), 28
- author (yandex\_market\_language.models.BookOffer attribute), 27
- author (yandex\_market\_language.models.offers.AbstractBookOffer attribute), 9
- author (yandex\_market\_language.models.offers.AudioBookOffer attribute), 18

*attribute*), 14  
 author (*yandex\_market\_language.models.offers.BookOffer attribute*), 15  
 available (*yandex\_market\_language.models.offers.AbstractOffer attribute*), 11  
**B**  
 barcodes (*yandex\_market\_language.models.AlcoholOffer attribute*), 32  
 barcodes (*yandex\_market\_language.models.ArbitraryOffer attribute*), 26  
 barcodes (*yandex\_market\_language.models.AudioBookOffer attribute*), 28  
 barcodes (*yandex\_market\_language.models.BookOffer attribute*), 27  
 barcodes (*yandex\_market\_language.models.EventTicketOffer attribute*), 31  
 barcodes (*yandex\_market\_language.models.MedicineOffer attribute*), 30  
 barcodes (*yandex\_market\_language.models.MusicVideoOffer attribute*), 29  
 barcodes (*yandex\_market\_language.models.offers.AbstractBookOffer attribute*), 9  
 barcodes (*yandex\_market\_language.models.offers.AbstractOffer attribute*), 11  
 barcodes (*yandex\_market\_language.models.offers.AlcoholOffer attribute*), 13  
 barcodes (*yandex\_market\_language.models.offers.ArbitraryOffer attribute*), 13  
 barcodes (*yandex\_market\_language.models.offers.AudioBookOffer attribute*), 14  
 barcodes (*yandex\_market\_language.models.offers.BookOffer attribute*), 16  
 barcodes (*yandex\_market\_language.models.offers.EventTicketOffer attribute*), 17  
 barcodes (*yandex\_market\_language.models.offers.MedicineOffer attribute*), 18  
 barcodes (*yandex\_market\_language.models.offers.MusicVideoOffer attribute*), 19  
 barcodes (*yandex\_market\_language.models.offers.SimplifiedOffer attribute*), 19  
 barcodes (*yandex\_market\_language.models.offers.AlcoholOffer attribute*), 25  
 barcodes (*yandex\_market\_language.models.offers.ArbitraryOffer attribute*), 27  
 barcodes (*yandex\_market\_language.models.offers.AudioBookOffer attribute*), 16  
 barcodes (*yandex\_market\_language.models.offers.BookOffer attribute*), 26  
 barcodes (*yandex\_market\_language.models.offers.EventTicketOffer attribute*), 15  
 barcodes (*yandex\_market\_language.models.offers.MedicineOffer attribute*), 18  
 barcodes (*yandex\_market\_language.models.offers.MusicVideoOffer attribute*), 23  
 barcodes (*yandex\_market\_language.models.offers.SimplifiedOffer attribute*), 21  
 barcodes (*yandex\_market\_language.models.SimplifiedOffer attribute*), 24  
 bid (*yandex\_market\_language.models.AlcoholOffer attribute*), 32  
 bid (*yandex\_market\_language.models.ArbitraryOffer attribute*), 26  
 bid (*yandex\_market\_language.models.AudioBookOffer attribute*), 28  
 bid (*yandex\_market\_language.models.BookOffer attribute*), 27  
 bid (*yandex\_market\_language.models.EventTicketOffer attribute*), 31  
 bid (*yandex\_market\_language.models.MedicineOffer attribute*), 30  
 bid (*yandex\_market\_language.models.MusicVideoOffer attribute*), 29  
 bid (*yandex\_market\_language.models.offers.AbstractBookOffer attribute*), 9  
 bid (*yandex\_market\_language.models.offers.AbstractOffer attribute*), 11  
 bid (*yandex\_market\_language.models.offers.AlcoholOffer attribute*), 13  
 bid (*yandex\_market\_language.models.offers.ArbitraryOffer attribute*), 13  
 bid (*yandex\_market\_language.models.offers.AudioBookOffer attribute*), 14  
 bid (*yandex\_market\_language.models.offers.BookOffer attribute*), 16  
 bid (*yandex\_market\_language.models.offers.EventTicketOffer attribute*), 17  
 bid (*yandex\_market\_language.models.offers.MedicineOffer attribute*), 18  
 bid (*yandex\_market\_language.models.offers.MusicVideoOffer attribute*), 19  
 bid (*yandex\_market\_language.models.offers.SimplifiedOffer attribute*), 19  
 binding (*yandex\_market\_language.models.BookOffer attribute*), 27  
 binding (*yandex\_market\_language.models.offers.BookOffer attribute*), 16  
 BookOffer (class in *yandex\_market\_language.models*), 26  
 BookOffer (class in *yandex\_market\_language.models.offers*), 15  
**C**  
 categories (*yandex\_market\_language.models.Shop attribute*), 23  
 categories (*yandex\_market\_language.models.shop.Shop attribute*), 21  
 Category (class in *yandex\_market\_language.models*), 24  
 Category (class in *yandex\_market\_language.models.category*), 8  
 category\_id (*yandex\_market\_language.models.AlcoholOffer attribute*), 32  
 category\_id (*yandex\_market\_language.models.ArbitraryOffer attribute*), 26  
 category\_id (*yandex\_market\_language.models.AudioBookOffer attribute*), 28  
 category\_id (*yandex\_market\_language.models.BookOffer attribute*), 27  
 category\_id (*yandex\_market\_language.models.EventTicketOffer attribute*), 31

category\_id (yandex\_market\_language.models.Category attribute), 24

category\_id (yandex\_market\_language.models.category attribute), 8

category\_id (yandex\_market\_language.models.EventTicketOffer attribute), 31

category\_id (yandex\_market\_language.models.MedicineOffer attribute), 30

category\_id (yandex\_market\_language.models.MusicVideoOffer attribute), 29

category\_id (yandex\_market\_language.models.offers.AbstractBookOffer attribute), 9

category\_id (yandex\_market\_language.models.offers.AbstractOffer attribute), 12

category\_id (yandex\_market\_language.models.offers.AlcoholOffer attribute), 13

category\_id (yandex\_market\_language.models.offers.ArbitraryOffer attribute), 14

category\_id (yandex\_market\_language.models.offers.AudioBookOffer attribute), 15

category\_id (yandex\_market\_language.models.offers.BookOffer attribute), 16

category\_id (yandex\_market\_language.models.offers.EventTicketOffer attribute), 17

category\_id (yandex\_market\_language.models.offers.MedicineOffer attribute), 18

category\_id (yandex\_market\_language.models.offers.MusicVideoOffer attribute), 19

category\_id (yandex\_market\_language.models.offers.SimplifiedOffer attribute), 20

category\_id (yandex\_market\_language.models.offers.MusicVideoOffer type (yandex\_market\_language.models.Condition attribute), 19

category\_id (yandex\_market\_language.models.offers.SimplifiedOffer attribute), 20

category\_id (yandex\_market\_language.models.Product attribute), 35

category\_id (yandex\_market\_language.models.SimplifiedOffer attribute), 25

cbid (yandex\_market\_language.models.offers.AbstractOffer attribute), 12

clean\_dict (yandex\_market\_language.models.AbstractModel attribute), 22

company (yandex\_market\_language.models.Shop attribute), 23

company (yandex\_market\_language.models.shop.Shop attribute), 21

Condition (class in yandex\_market\_language.models), 33

condition (yandex\_market\_language.models.AlcoholOffer attribute), 32

condition (yandex\_market\_language.models.ArbitraryOffer attribute), 26

condition (yandex\_market\_language.models.AudioBookOffer attribute), 28

condition (yandex\_market\_language.models.BookOffer attribute), 27

condition (yandex\_market\_language.models.EventTicketOffer attribute), 31

condition (yandex\_market\_language.models.MedicineOffer attribute), 30

condition (yandex\_market\_language.models.MusicVideoOffer attribute), 29

condition (yandex\_market\_language.models.offers.AbstractBookOffer attribute), 9

condition (yandex\_market\_language.models.offers.AbstractOffer attribute), 12

condition (yandex\_market\_language.models.offers.AlcoholOffer attribute), 13

condition (yandex\_market\_language.models.offers.ArbitraryOffer attribute), 14

condition (yandex\_market\_language.models.offers.AudioBookOffer attribute), 15

condition (yandex\_market\_language.models.offers.BookOffer attribute), 16

condition (yandex\_market\_language.models.offers.EventTicketOffer attribute), 17

condition (yandex\_market\_language.models.offers.MedicineOffer attribute), 18

condition (yandex\_market\_language.models.offers.MusicVideoOffer attribute), 19

condition (yandex\_market\_language.models.offers.SimplifiedOffer attribute), 20

convert () (in module yandex\_market\_language), 36

convert () (in module yandex\_market\_language.yml), 36

convert () (yandex\_market\_language.yml.YML method), 36

cost (yandex\_market\_language.models.Option attribute), 24

create\_dict () (yandex\_market\_language.models.option.Option attribute), 20

country (yandex\_market\_language.models.MusicVideoOffer attribute), 29

country (yandex\_market\_language.models.offers.MusicVideoOffer attribute), 19

create\_dict () (yandex\_market\_language.models.AbstractModel method), 22

create\_dict () (yandex\_market\_language.models.Age method), 33

create\_dict () (yandex\_market\_language.models.AlcoholOffer method), 32

create\_dict () (yandex\_market\_language.models.ArbitraryOffer method), 26

create\_dict () (yandex\_market\_language.models.AudioBookOffer method), 28

create\_dict () (yandex\_market\_language.models.BookOffer method), 27

create\_dict () (yandex\_market\_language.models.EventTicketOffer method), 26

<code>create_dict()</code> <i>dex_market_language.models.AudioBookOffer</i> method), 28	(yan-	<code>create_dict()</code> <i>dex_market_language.models.offers.AudioBookOffer</i> method), 15	(yan-
<code>create_dict()</code> <i>dex_market_language.models.BookOffer</i> method), 27	(yan-	<code>create_dict()</code> <i>dex_market_language.models.offers.BookOffer</i> method), 16	(yan-
<code>create_dict()</code> <i>dex_market_language.models.Category</i> method), 24	(yan-	<code>create_dict()</code> <i>dex_market_language.models.offers.EventTicketOffer</i> method), 17	(yan-
<code>create_dict()</code> <i>dex_market_language.models.category.Category</i> method), 8	(yan-	<code>create_dict()</code> <i>dex_market_language.models.offers.MedicineOffer</i> method), 18	(yan-
<code>create_dict()</code> <i>dex_market_language.models.Condition</i> method), 33	(yan-	<code>create_dict()</code> <i>dex_market_language.models.offers.MusicVideoOffer</i> method), 19	(yan-
<code>create_dict()</code> <i>dex_market_language.models.Currency</i> method), 24	(yan-	<code>create_dict()</code> <i>dex_market_language.models.offers.SimplifiedOffer</i> method), 20	(yan-
<code>create_dict()</code> <i>dex_market_language.models.currency.Currency</i> method), 8	(yan-	<code>create_dict()</code> <i>dex_market_language.models.Option</i> method), 24	(yan-
<code>create_dict()</code> <i>dex_market_language.models.Dimensions</i> method), 33	(yan-	<code>create_dict()</code> <i>dex_market_language.models.option.Option</i> method), 20	(yan-
<code>create_dict()</code> <i>dex_market_language.models.EventTicketOffer</i> method), 31	(yan-	<code>create_dict()</code> <i>dex_market_language.models.Parameter</i> method), 32	(yan-
<code>create_dict()</code> <i>dex_market_language.models.Feed</i> method), 22	(yan-	<code>create_dict()</code> <i>dex_market_language.models.Price</i> method), 25	(yan-
<code>create_dict()</code> <i>dex_market_language.models.feed.Feed</i> method), 9	(yan-	<code>create_dict()</code> <i>dex_market_language.models.price.Price</i> method), 21	(yan-
<code>create_dict()</code> <i>dex_market_language.models.Gift</i> method), 34	(yan-	<code>create_dict()</code> <i>dex_market_language.models.Product</i> method), 35	(yan-
<code>create_dict()</code> <i>dex_market_language.models.MedicineOffer</i> method), 30	(yan-	<code>create_dict()</code> <i>dex_market_language.models.Promo</i> method), 34	(yan-
<code>create_dict()</code> <i>dex_market_language.models.MusicVideoOffer</i> method), 29	(yan-	<code>create_dict()</code> <i>dex_market_language.models.PromoGift</i> method), 35	(yan-
<code>create_dict()</code> <i>dex_market_language.models.offers.AbstractBookOffer</i> method), 10	(yan-	<code>create_dict()</code> <i>dex_market_language.models.Purchase</i> method), 35	(yan-
<code>create_dict()</code> <i>dex_market_language.models.offers.AbstractOffer</i> method), 12	(yan-	<code>create_dict()</code> <i>dex_market_language.models.Shop</i> method), 23	(yan-
<code>create_dict()</code> <i>dex_market_language.models.offers.AlcoholOffer</i> method), 13	(yan-	<code>create_dict()</code> <i>dex_market_language.models.shop.Shop</i> method), 21	(yan-
<code>create_dict()</code> <i>dex_market_language.models.offers.ArbitraryOffer</i> method), 14	(yan-	<code>create_dict()</code> <i>dex_market_language.models.SimplifiedOffer</i> method), 25	(yan-

`create_xml ()` (*yandex\_market\_language.models.AbstractModel* *method*), 10  
`create_xml ()` (*yandex\_market\_language.models.offers.AbstractOffer* *method*), 22  
`create_xml ()` (*yandex\_market\_language.models.Age* *method*), 33  
`create_xml ()` (*yandex\_market\_language.models.AlcoholOffer* *method*), 13  
`create_xml ()` (*yandex\_market\_language.models.offers.AlcoholOffer* *method*), 13  
`create_xml ()` (*yandex\_market\_language.models.ArbitraryOffer* *method*), 26  
`create_xml ()` (*yandex\_market\_language.models.offers.ArbitraryOffer* *method*), 14  
`create_xml ()` (*yandex\_market\_language.models.AudioBookOffer* *method*), 28  
`create_xml ()` (*yandex\_market\_language.models.offers.AudioBookOffer* *method*), 15  
`create_xml ()` (*yandex\_market\_language.models.BookOffer* *method*), 27  
`create_xml ()` (*yandex\_market\_language.models.offers.BookOffer* *method*), 16  
`create_xml ()` (*yandex\_market\_language.models.Category* *method*), 24  
`create_xml ()` (*yandex\_market\_language.models.offers.EventTicketOffer* *method*), 17  
`create_xml ()` (*yandex\_market\_language.models.category.Category* *method*), 8  
`create_xml ()` (*yandex\_market\_language.models.offers.MedicineOffer* *method*), 18  
`create_xml ()` (*yandex\_market\_language.models.Condition* *method*), 33  
`create_xml ()` (*yandex\_market\_language.models.offers.MusicVideoOffer* *method*), 19  
`create_xml ()` (*yandex\_market\_language.models.Currency* *method*), 24  
`create_xml ()` (*yandex\_market\_language.models.offers.SimplifiedOffer* *method*), 20  
`create_xml ()` (*yandex\_market\_language.models.currency.Currency* *method*), 8  
`create_xml ()` (*yandex\_market\_language.models.Option* *method*), 24  
`create_xml ()` (*yandex\_market\_language.models.option.Option* *method*), 20  
`create_xml ()` (*yandex\_market\_language.models.Dimensions* *method*), 33  
`create_xml ()` (*yandex\_market\_language.models.Parameter* *method*), 33  
`create_xml ()` (*yandex\_market\_language.models.EventTicketOffer* *method*), 31  
`create_xml ()` (*yandex\_market\_language.models.Price* *method*), 25  
`create_xml ()` (*yandex\_market\_language.models.Feed* *method*), 22  
`create_xml ()` (*yandex\_market\_language.models.price.Price* *method*), 21  
`create_xml ()` (*yandex\_market\_language.models.feed.Feed* *method*), 9  
`create_xml ()` (*yandex\_market\_language.models.Product* *method*), 35  
`create_xml ()` (*yandex\_market\_language.models.Gift* *method*), 34  
`create_xml ()` (*yandex\_market\_language.models.Promo* *method*), 34  
`create_xml ()` (*yandex\_market\_language.models.MedicineOffer* *method*), 30  
`create_xml ()` (*yandex\_market\_language.models.PromoGift* *method*), 35  
`create_xml ()` (*yandex\_market\_language.models.MusicVideoOffer* *method*), 29  
`create_xml ()` (*yandex\_market\_language.models.PromoGift* *method*), 35  
`create_xml ()` (*yandex\_market\_language.models.offers.AbstractBookOffer* *method*), 29  
`create_xml ()` (*yandex\_market\_language.models.Purchase* *method*), 35

`method`), 35

`create_xml()` (`yandex_market_language.models.Shop` `method`), 23

`create_xml()` (`yandex_market_language.models.shop.Shop` `method`), 21

`create_xml()` (`yandex_market_language.models.SimplifiedOffer` `method`), 25

`credit_template_id` (`yandex_market_language.models.AlcoholOffer` `attribute`), 32

`credit_template_id` (`yandex_market_language.models.ArbitraryOffer` `attribute`), 26

`credit_template_id` (`yandex_market_language.models.AudioBookOffer` `attribute`), 28

`credit_template_id` (`yandex_market_language.models.BookOffer` `attribute`), 27

`credit_template_id` (`yandex_market_language.models.EventTicketOffer` `attribute`), 31

`credit_template_id` (`yandex_market_language.models.MedicineOffer` `attribute`), 30

`credit_template_id` (`yandex_market_language.models.MusicVideoOffer` `attribute`), 29

`credit_template_id` (`yandex_market_language.models.offers.AbstractBookOffer` `attribute`), 10

`credit_template_id` (`yandex_market_language.models.offers.AbstractOffer` `attribute`), 12

`credit_template_id` (`yandex_market_language.models.offers.AlcoholOffer` `attribute`), 13

`credit_template_id` (`yandex_market_language.models.offers.ArbitraryOffer` `attribute`), 14

`credit_template_id` (`yandex_market_language.models.offers.AudioBookOffer` `attribute`), 15

`credit_template_id` (`yandex_market_language.models.offers.BookOffer` `attribute`), 16

`credit_template_id` (`yandex_market_language.models.offers.EventTicketOffer` `attribute`), 17

`credit_template_id` (`yandex_market_language.models.offers.MedicineOffer` `attribute`), 18

`credit_template_id` (`yandex_market_language.models.offers.MusicVideoOffer` `attribute`), 19

`credit_template_id` (`yandex_market_language.models.offers.SimplifiedOffer` `attribute`), 20

`credit_template_id` (`yandex_market_language.models.SimplifiedOffer` `attribute`), 25

`currencies` (`yandex_market_language.models.Shop` `attribute`), 23

`currencies` (`yandex_market_language.models.shop.Shop` `attribute`), 22

`Currency` (`class in yandex_market_language.models`), 23

`Currency` (`class in yandex_market_language.models.currency`), 8

`currency` (`yandex_market_language.models.AlcoholOffer` `attribute`), 32

`currency` (`yandex_market_language.models.ArbitraryOffer` `attribute`), 26

`currency` (`yandex_market_language.models.AudioBookOffer` `attribute`), 28

`currency` (`yandex_market_language.models.BookOffer` `attribute`), 27

`currency` (`yandex_market_language.models.Currency` `attribute`), 24

`currency` (`yandex_market_language.models.currency.Currency` `attribute`), 8

`currency` (`yandex_market_language.models.EventTicketOffer` `attribute`), 31

`currency` (`yandex_market_language.models.MedicineOffer` `attribute`), 30

`currency` (`yandex_market_language.models.MusicVideoOffer` `attribute`), 29

`currency` (`yandex_market_language.models.offers.AbstractBookOffer` `attribute`), 10

`currency` (`yandex_market_language.models.offers.AbstractOffer` `attribute`), 12

`currency` (`yandex_market_language.models.offers.AlcoholOffer` `attribute`), 13

`currency` (`yandex_market_language.models.offers.ArbitraryOffer` `attribute`), 14

`currency` (`yandex_market_language.models.offers.AudioBookOffer` `attribute`), 15

`currency` (`yandex_market_language.models.offers.BookOffer` `attribute`), 16

`currency` (`yandex_market_language.models.offers.EventTicketOffer` `attribute`), 17

`currency` (`yandex_market_language.models.offers.MedicineOffer` `attribute`), 18

`currency` (`yandex_market_language.models.offers.MusicVideoOffer` `attribute`), 19



*attribute*), 12

**E**

email (*yandex\_market\_language.models.Shop attribute*), 23

email (*yandex\_market\_language.models.shop.Shop attribute*), 22

enable\_auto\_discounts (*yandex\_market\_language.models.fields.enable\_auto\_discounts attribute*), 7

enable\_auto\_discounts (*yandex\_market\_language.models.fields.EnableAutoDiscountField attribute*), 7

EnableAutoDiscountField (*class in yandex\_market\_language.models.fields*), 7

EnableAutoDiscountField (*class in yandex\_market\_language.models.fields.enable\_auto\_discounts*), 7

end\_date (*yandex\_market\_language.models.Promo attribute*), 34

EventTicketOffer (*class in yandex\_market\_language.models*), 31

EventTicketOffer (*class in yandex\_market\_language.models.offers*), 16

expiry (*yandex\_market\_language.models.offers.AbstractOffer attribute*), 12

**F**

Feed (*class in yandex\_market\_language.models*), 22

Feed (*class in yandex\_market\_language.models.feed*), 9

from\_xml () (*yandex\_market\_language.models.AbstractModel static method*), 22

from\_xml () (*yandex\_market\_language.models.Age static method*), 34

from\_xml () (*yandex\_market\_language.models.AlcoholOffer static method*), 32

from\_xml () (*yandex\_market\_language.models.ArbitraryOffer static method*), 26

from\_xml () (*yandex\_market\_language.models.AudioBookOffer static method*), 28

from\_xml () (*yandex\_market\_language.models.BookOffer static method*), 27

from\_xml () (*yandex\_market\_language.models.Category static method*), 24

from\_xml () (*yandex\_market\_language.models.category.Category static method*), 8

from\_xml () (*yandex\_market\_language.models.Condition static method*), 33

from\_xml () (*yandex\_market\_language.models.Currency static method*), 24

from\_xml () (*yandex\_market\_language.models.currency.Currency static method*), 8

from\_xml () (*yandex\_market\_language.models.Dimensions static method*), 33

from\_xml () (*yandex\_market\_language.models.EventTicketOffer static method*), 31

from\_xml () (*yandex\_market\_language.models.Feed static method*), 23

from\_xml () (*yandex\_market\_language.models.feed.Feed static method*), 9

from\_xml () (*yandex\_market\_language.models.Gift static method*), 34

from\_xml () (*yandex\_market\_language.models.MedicineOffer static method*), 30

from\_xml () (*yandex\_market\_language.models.MusicVideoOffer static method*), 29

from\_xml () (*yandex\_market\_language.models.offers.AbstractBookOffer static method*), 10

from\_xml () (*yandex\_market\_language.models.offers.AbstractOffer static method*), 12

from\_xml () (*yandex\_market\_language.models.offers.AlcoholOffer static method*), 13

from\_xml () (*yandex\_market\_language.models.offers.ArbitraryOffer static method*), 14

from\_xml () (*yandex\_market\_language.models.offers.AudioBookOffer static method*), 15

from\_xml () (*yandex\_market\_language.models.offers.BookOffer static method*), 16

from\_xml () (*yandex\_market\_language.models.offers.EventTicketOffer static method*), 17

from\_xml () (*yandex\_market\_language.models.offers.MedicineOffer static method*), 18

from\_xml () (*yandex\_market\_language.models.offers.MusicVideoOffer static method*), 19

from\_xml () (*yandex\_market\_language.models.offers.SimplifiedOffer static method*), 20

from\_xml () (*yandex\_market\_language.models.Option static method*), 24

from\_xml () (*yandex\_market\_language.models.option.Option static method*), 20

from\_xml () (*yandex\_market\_language.models.Parameter static method*), 33

from\_xml () (*yandex\_market\_language.models.Price static method*), 25

from\_xml () (*yandex\_market\_language.models.price.Price static method*), 21

from\_xml () (*yandex\_market\_language.models.Product static method*), 35

from\_xml () (*yandex\_market\_language.models.Promo class method*), 34

from\_xml () (*yandex\_market\_language.models.PromoGift static method*), 35

from\_xml () (*yandex\_market\_language.models.Purchase static method*), 35

from\_xml () (*yandex\_market\_language.models.Shop static method*), 23

from\_xml () (*yandex\_market\_language.models.shop.Shop static method*), 22

`from_xml()` (`yandex_market_language.models.SimplifiedOffer` language (`yandex_market_language.models.BookOffer` static method), 25 attribute), 27  
**G**  
`Gift` (class in `yandex_market_language.models`), 34  
`gift_id` (`yandex_market_language.models.PromoGift` language (`yandex_market_language.models.offers.AbstractBookOffer` attribute), 35 attribute), 10  
`gifts` (`yandex_market_language.models.Shop` attribute), 23  
`gifts` (`yandex_market_language.models.shop.Shop` language (`yandex_market_language.models.offers.AudioBookOffer` attribute), 22 attribute), 15  
`group_id` (`yandex_market_language.models.offers.AbstractOffer` language (`yandex_market_language.models.offers.BookOffer` attribute), 12 attribute), 16  
**H**  
`hall` (`yandex_market_language.models.EventTicketOffer` length (`yandex_market_language.models.Dimensions` attribute), 31 attribute), 33  
`hall` (`yandex_market_language.models.offers.EventTicketOffer` MAPPING (`yandex_market_language.models.Promo` attribute), 17 attribute), 34  
`hall_part` (`yandex_market_language.models.EventTicketOffer` Offer (`yandex_market_language.models.MusicVideoOffer` attribute), 31 attribute), 29  
`hall_part` (`yandex_market_language.models.offers.EventTicketOffer` Offer (`yandex_market_language.models.offers.MusicVideoOffer` attribute), 17 attribute), 19  
`height` (`yandex_market_language.models.Dimensions` Offer (class in `yandex_market_language.models`), 30 attribute), 33 Offer (class in `yandex_market_language.models.offers`), 17  
**I**  
`id` (`yandex_market_language.models.Gift` attribute), 34  
`is_kids` (`yandex_market_language.models.EventTicketOffer` model (`yandex_market_language.models.ArbitraryOffer` attribute), 31 attribute), 26  
`is_kids` (`yandex_market_language.models.offers.EventTicketOffer` Offer (`yandex_market_language.models.offers.ArbitraryOffer` attribute), 17 attribute), 14  
`is_premiere` (`yandex_market_language.models.EventTicketOffer` MusicVideoOffer (class in `yandex_market_language.models`), 31 attribute), 29  
`is_premiere` (`yandex_market_language.models.offers.EventTicketOffer` MusicVideoOffer (class in `yandex_market_language.models.offers`), 17 attribute), 18  
`is_starting` (`yandex_market_language.models.Price` attribute), 25  
**N**  
`is_starting` (`yandex_market_language.models.price.Price` name (`yandex_market_language.models.AlcoholOffer` attribute), 21 attribute), 32  
`isbn` (`yandex_market_language.models.AudioBookOffer` name (`yandex_market_language.models.AudioBookOffer` attribute), 28 attribute), 28  
`isbn` (`yandex_market_language.models.BookOffer` name (`yandex_market_language.models.BookOffer` attribute), 27 attribute), 27  
`isbn` (`yandex_market_language.models.offers.AbstractBookOffer` (`yandex_market_language.models.Category` attribute), 10 attribute), 24  
`isbn` (`yandex_market_language.models.offers.AudioBookOffer` name (`yandex_market_language.models.category.Category` attribute), 15 attribute), 8  
`isbn` (`yandex_market_language.models.offers.BookOffer` name (`yandex_market_language.models.EventTicketOffer` attribute), 16 attribute), 31  
**L**  
`language` (`yandex_market_language.models.AudioBookOffer` name (`yandex_market_language.models.Gift` attribute), attribute), 28 attribute), 34  
`language` (`yandex_market_language.models.MedicineOffer` attribute), 30

name (`yandex_market_language.models.offers.AbstractBookOffer` attribute), 10

name (`yandex_market_language.models.offers.AlcoholOffer` attribute), 13

name (`yandex_market_language.models.offers.AudioBookOffer` attribute), 15

name (`yandex_market_language.models.offers.BookOffer` attribute), 16

name (`yandex_market_language.models.offers.EventTicketOffer` attribute), 17

name (`yandex_market_language.models.offers.MedicineOffer` attribute), 18

name (`yandex_market_language.models.offers.SimplifiedOffer` attribute), 20

name (`yandex_market_language.models.Parameter` attribute), 33

name (`yandex_market_language.models.Shop` attribute), 23

name (`yandex_market_language.models.shop.Shop` attribute), 22

name (`yandex_market_language.models.SimplifiedOffer` attribute), 25

## O

offer\_id (`yandex_market_language.models.AlcoholOffer` attribute), 32

offer\_id (`yandex_market_language.models.ArbitraryOffer` attribute), 26

offer\_id (`yandex_market_language.models.AudioBookOffer` attribute), 28

offer\_id (`yandex_market_language.models.BookOffer` attribute), 27

offer\_id (`yandex_market_language.models.EventTicketOffer` attribute), 31

offer\_id (`yandex_market_language.models.MedicineOffer` attribute), 30

offer\_id (`yandex_market_language.models.MusicVideoOffer` attribute), 29

offer\_id (`yandex_market_language.models.offers.AbstractBookOffer` attribute), 10

offer\_id (`yandex_market_language.models.offers.AbstractOffer` attribute), 12

offer\_id (`yandex_market_language.models.offers.AlcoholOffer` attribute), 13

offer\_id (`yandex_market_language.models.offers.ArbitraryOffer` attribute), 14

offer\_id (`yandex_market_language.models.offers.AudioBookOffer` attribute), 15

offer\_id (`yandex_market_language.models.offers.BookOffer` attribute), 16

offer\_id (`yandex_market_language.models.offers.EventTicketOffer` attribute), 17

offer\_id (`yandex_market_language.models.offers.MedicineOffer` attribute), 18

offer\_id (`yandex_market_language.models.offers.MusicVideoOffer` attribute), 19

offer\_id (`yandex_market_language.models.offers.SimplifiedOffer` attribute), 20

offer\_id (`yandex_market_language.models.SimplifiedOffer` attribute), 25

offer\_id (`yandex_market_language.models.offers.EventTicketOffer` attribute), 17

offer\_id (`yandex_market_language.models.offers.MedicineOffer` attribute), 18

`dex_market_language.models.Option` attribute), 24

`order_before` (`YandexMarketLanguage.models.option.Option` attribute), 21

`original_name` (`YandexMarketLanguage.models.MusicVideoOffer` attribute), 29

`original_name` (`YandexMarketLanguage.models.offers.MusicVideoOffer` attribute), 19

**P**

`page_extent` (`YandexMarketLanguage.models.BookOffer` attribute), 27

`page_extent` (`YandexMarketLanguage.models.offers.BookOffer` attribute), 16

`Parameter` (class in `YandexMarketLanguage.models`), 32

`parameters` (`YandexMarketLanguage.models.AlcoholOffer` attribute), 32

`parameters` (`YandexMarketLanguage.models.offers.AbstractOffer` attribute), 12

`parameters` (`YandexMarketLanguage.models.offers.AlcoholOffer` attribute), 13

`parameters` (`YandexMarketLanguage.models.offers.AlcoholOffer` attribute), 13

`parent_id` (`YandexMarketLanguage.models.Category` attribute), 24

`parent_id` (`YandexMarketLanguage.models.category.Category` attribute), 8

`parse()` (in module `YandexMarketLanguage`), 36

`parse()` (in module `YandexMarketLanguage.yml`), 36

`parse()` (`YandexMarketLanguage.yml.YML` method), 36

`ParseError`, 36

`part` (`YandexMarketLanguage.models.offers.AbstractBookOffer` attribute), 10

`performance_type` (`YandexMarketLanguage.models.AudioBookOffer` attribute), 28

`performance_type` (`YandexMarketLanguage.models.offers.AudioBookOffer` attribute), 15

`performed_by` (`YandexMarketLanguage.models.AudioBookOffer` attribute), 28

`performed_by` (`YandexMarketLanguage.models.offers.AudioBookOffer` attribute), 15

`pickup` (`YandexMarketLanguage.models.offers.AbstractOffer` attribute), 12

`pickup_options` (`YandexMarketLanguage.models.fields.PickupOptionsField` attribute), 8

`PickupOptionsField` (class in `YandexMarketLanguage.models.fields`), 7

`pictures` (`YandexMarketLanguage.models.AlcoholOffer` attribute), 32

`pictures` (`YandexMarketLanguage.models.ArbitraryOffer` attribute), 26

`pictures` (`YandexMarketLanguage.models.AudioBookOffer` attribute), 28

`pictures` (`YandexMarketLanguage.models.BookOffer` attribute), 27

`pictures` (`YandexMarketLanguage.models.EventTicketOffer` attribute), 31

`pictures` (`YandexMarketLanguage.models.Gift` attribute), 34

`pictures` (`YandexMarketLanguage.models.MedicineOffer` attribute), 30

`pictures` (`YandexMarketLanguage.models.MusicVideoOffer` attribute), 29

`pictures` (`YandexMarketLanguage.models.offers.AbstractBookOffer` attribute), 10

`pictures` (`YandexMarketLanguage.models.offers.AbstractOffer` attribute), 12

`pictures` (`YandexMarketLanguage.models.offers.AlcoholOffer` attribute), 13

`pictures` (`YandexMarketLanguage.models.offers.ArbitraryOffer` attribute), 14

`pictures` (`YandexMarketLanguage.models.offers.AudioBookOffer` attribute), 15

`pictures` (`YandexMarketLanguage.models.offers.BookOffer` attribute), 16

`pictures` (`YandexMarketLanguage.models.offers.EventTicketOffer` attribute), 17

`pictures` (`YandexMarketLanguage.models.offers.MedicineOffer` attribute), 18

`pictures` (`YandexMarketLanguage.models.offers.MusicVideoOffer` attribute), 19

`pictures` (`YandexMarketLanguage.models.offers.SimplifiedOffer` attribute), 20

`pictures` (`YandexMarketLanguage.models.SimplifiedOffer` attribute), 25

`place` (`YandexMarketLanguage.models.EventTicketOffer` attribute), 31

`place` (`YandexMarketLanguage.models.offers.EventTicketOffer` attribute), 17

`platform` (`YandexMarketLanguage.models.Shop` attribute), 23

`platform` (`YandexMarketLanguage.models.shop.Shop` attribute), 22

`plus` (`YandexMarketLanguage.models.currency.Currency` attribute), 24

`plus` (`YandexMarketLanguage.models.currency.Currency` attribute), 8

`prettify_el()` (`YandexMarketLanguage.yml.YML` static method), 36

Price (class in `yandex_market_language.models`), 24  
 Price (class in `yandex_market_language.models.price`), 21  
 price (`yandex_market_language.models.AlcoholOffer` attribute), 32  
 price (`yandex_market_language.models.ArbitraryOffer` attribute), 26  
 price (`yandex_market_language.models.AudioBookOffer` attribute), 28  
 price (`yandex_market_language.models.BookOffer` attribute), 27  
 price (`yandex_market_language.models.EventTicketOffer` attribute), 31  
 price (`yandex_market_language.models.MedicineOffer` attribute), 30  
 price (`yandex_market_language.models.MusicVideoOffer` attribute), 30  
 price (`yandex_market_language.models.offers.AbstractBookOffer` attribute), 10  
 price (`yandex_market_language.models.offers.AbstractOffer` attribute), 12  
 price (`yandex_market_language.models.offers.AlcoholOffer` attribute), 13  
 price (`yandex_market_language.models.offers.ArbitraryOffer` attribute), 14  
 price (`yandex_market_language.models.offers.AudioBookOffer` attribute), 15  
 price (`yandex_market_language.models.offers.BookOffer` attribute), 16  
 price (`yandex_market_language.models.offers.EventTicketOffer` attribute), 17  
 price (`yandex_market_language.models.offers.MedicineOffer` attribute), 18  
 price (`yandex_market_language.models.offers.MusicVideoOffer` attribute), 19  
 price (`yandex_market_language.models.offers.SimplifiedOffer` attribute), 20  
 price (`yandex_market_language.models.SimplifiedOffer` attribute), 25  
 Product (class in `yandex_market_language.models`), 35  
 products (`yandex_market_language.models.Purchase` attribute), 35  
 Promo (class in `yandex_market_language.models`), 34  
 promo\_gifts (`yandex_market_language.models.Promo` attribute), 34  
 promo\_id (`yandex_market_language.models.Promo` attribute), 34  
 promo\_type (`yandex_market_language.models.Promo` attribute), 34  
 PromoGift (class in `yandex_market_language.models`), 35  
 promos (`yandex_market_language.models.Shop` attribute), 23  
 promos (`yandex_market_language.models.shop.Shop` attribute), 22  
 publisher (`yandex_market_language.models.AudioBookOffer` attribute), 28  
 publisher (`yandex_market_language.models.BookOffer` attribute), 27  
 publisher (`yandex_market_language.models.offers.AbstractBookOffer` attribute), 10  
 publisher (`yandex_market_language.models.offers.AudioBookOffer` attribute), 15  
 publisher (`yandex_market_language.models.offers.BookOffer` attribute), 16  
 Purchase (class in `yandex_market_language.models`), 35  
 purchase (`yandex_market_language.models.Promo` attribute), 34  
**R**  
 rate (`yandex_market_language.models.Currency` attribute), 24  
 rate (`yandex_market_language.models.currency.Currency` attribute), 9  
 reason (`yandex_market_language.models.Condition` attribute), 33  
 recording\_length (`yandex_market_language.models.AudioBookOffer` attribute), 28  
 recording\_length (`yandex_market_language.models.offers.AudioBookOffer` attribute), 15  
 required\_quantity (`yandex_market_language.models.Purchase` attribute), 35  
**S**  
 sales\_notes (`yandex_market_language.models.AlcoholOffer` attribute), 32  
 sales\_notes (`yandex_market_language.models.ArbitraryOffer` attribute), 26  
 sales\_notes (`yandex_market_language.models.AudioBookOffer` attribute), 28  
 sales\_notes (`yandex_market_language.models.BookOffer` attribute), 27  
 sales\_notes (`yandex_market_language.models.EventTicketOffer` attribute), 31  
 sales\_notes (`yandex_market_language.models.MedicineOffer` attribute), 30  
 sales\_notes (`yandex_market_language.models.MusicVideoOffer` attribute), 30  
 sales\_notes (`yandex_market_language.models.offers.AbstractBookOffer` attribute), 10  
 sales\_notes (`yandex_market_language.models.offers.AbstractOffer` attribute), 12

<code>sales_notes (yandex_market_language.models.offers.AlcoholOffer attribute), 13</code>	<code>supplier (yandex_market_language.models.AudioBookOffer attribute), 29</code>
<code>sales_notes (yandex_market_language.models.offers.ArbitraryOffer attribute), 14</code>	<code>supplier (yandex_market_language.models.BookOffer attribute), 27</code>
<code>sales_notes (yandex_market_language.models.offers.AudioBookOffer attribute), 15</code>	<code>supplier (yandex_market_language.models.EventTicketOffer attribute), 31</code>
<code>sales_notes (yandex_market_language.models.offers.BookOffer attribute), 16</code>	<code>supplier (yandex_market_language.models.MedicineOffer attribute), 30</code>
<code>sales_notes (yandex_market_language.models.offers.EventTicketOffer attribute), 17</code>	<code>supplier (yandex_market_language.models.MusicVideoOffer attribute), 30</code>
<code>sales_notes (yandex_market_language.models.offers.MedicineOffer attribute), 18</code>	<code>supplier (yandex_market_language.models.offers.AbstractBookOffer attribute), 10</code>
<code>sales_notes (yandex_market_language.models.offers.MusicVideoOffer attribute), 19</code>	<code>supplier (yandex_market_language.models.offers.AbstractOffer attribute), 12</code>
<code>sales_notes (yandex_market_language.models.offers.SimplifiedOffer attribute), 20</code>	<code>supplier (yandex_market_language.models.offers.AlcoholOffer attribute), 13</code>
<code>sales_notes (yandex_market_language.models.SimplifiedOffer attribute), 25</code>	<code>supplier (yandex_market_language.models.offers.ArbitraryOffer attribute), 14</code>
<code>series (yandex_market_language.models.AudioBookOffer attribute), 28</code>	<code>supplier (yandex_market_language.models.offers.AudioBookOffer attribute), 15</code>
<code>series (yandex_market_language.models.BookOffer attribute), 27</code>	<code>supplier (yandex_market_language.models.offers.BookOffer attribute), 16</code>
<code>series (yandex_market_language.models.offers.AbstractBookOffer attribute), 10</code>	<code>supplier (yandex_market_language.models.offers.EventTicketOffer attribute), 17</code>
<code>series (yandex_market_language.models.offers.AudioBookOffer attribute), 15</code>	<code>supplier (yandex_market_language.models.offers.MedicineOffer attribute), 18</code>
<code>series (yandex_market_language.models.offers.BookOffer attribute), 16</code>	<code>supplier (yandex_market_language.models.offers.MusicVideoOffer attribute), 19</code>
<code>Shop (class in yandex_market_language.models), 23</code>	<code>supplier (yandex_market_language.models.offers.SimplifiedOffer attribute), 20</code>
<code>Shop (class in yandex_market_language.models.shop), 21</code>	<code>supplier (yandex_market_language.models.SimplifiedOffer attribute), 25</code>
<code>shop (yandex_market_language.models.Feed attribute), 23</code>	
<code>shop (yandex_market_language.models.feed.Feed attribute), 9</code>	
<code>SimplifiedOffer (class in yandex_market_language.models), 25</code>	
<code>SimplifiedOffer (class in yandex_market_language.models.offers), 19</code>	
<code>starring (yandex_market_language.models.MusicVideoOffer attribute), 30</code>	
<code>starring (yandex_market_language.models.offers.MusicVideoOffer attribute), 19</code>	
<code>start_date (yandex_market_language.models.Promo attribute), 34</code>	
<code>storage (yandex_market_language.models.AudioBookOffer attribute), 29</code>	
<code>storage (yandex_market_language.models.offers.AudioBookOffer attribute), 15</code>	
<code>storage (yandex_market_language.models.offers.AudioBookOffer attribute), 15</code>	
<code>store (yandex_market_language.models.offers.AbstractOffer attribute), 12</code>	
<code>supplier (yandex_market_language.models.AlcoholOffer attribute), 32</code>	
<code>supplier (yandex_market_language.models.ArbitraryOffer attribute), 19</code>	

**T**

<code>table_of_contents (yandex_market_language.models.AudioBookOffer attribute), 29</code>
<code>table_of_contents (yandex_market_language.models.BookOffer attribute), 27</code>
<code>table_of_contents (yandex_market_language.models.offers.AbstractBookOffer attribute), 10</code>
<code>table_of_contents (yandex_market_language.models.offers.AudioBookOffer attribute), 15</code>
<code>table_of_contents (yandex_market_language.models.offers.BookOffer attribute), 16</code>
<code>title (yandex_market_language.models.MusicVideoOffer attribute), 30</code>
<code>title (yandex_market_language.models.offers.MusicVideoOffer attribute), 19</code>

`to_dict()` (`yandex_market_language.models.AbstractModel` method), 22  
`to_xml()` (`yandex_market_language.models.AbstractModel` method), 22  
`type_prefix` (`yandex_market_language.models.ArbitraryOffer` attribute), 26  
`type_prefix` (`yandex_market_language.models.offers.ArbitraryOffer` attribute), 14  
**U**  
`unit` (`yandex_market_language.models.Age` attribute), 34  
`unit` (`yandex_market_language.models.Parameter` attribute), 33  
`url` (`yandex_market_language.models.AlcoholOffer` attribute), 32  
`url` (`yandex_market_language.models.ArbitraryOffer` attribute), 26  
`url` (`yandex_market_language.models.AudioBookOffer` attribute), 29  
`url` (`yandex_market_language.models.BookOffer` attribute), 27  
`url` (`yandex_market_language.models.EventTicketOffer` attribute), 31  
`url` (`yandex_market_language.models.MedicineOffer` attribute), 30  
`url` (`yandex_market_language.models.MusicVideoOffer` attribute), 30  
`url` (`yandex_market_language.models.offers.AbstractBookOffer` attribute), 10  
`url` (`yandex_market_language.models.offers.AbstractOffer` attribute), 12  
`url` (`yandex_market_language.models.offers.AlcoholOffer` attribute), 13  
`url` (`yandex_market_language.models.offers.ArbitraryOffer` attribute), 14  
`url` (`yandex_market_language.models.offers.AudioBookOffer` attribute), 15  
`url` (`yandex_market_language.models.offers.BookOffer` attribute), 16  
`url` (`yandex_market_language.models.offers.EventTicketOffer` attribute), 17  
`url` (`yandex_market_language.models.offers.MedicineOffer` attribute), 18  
`url` (`yandex_market_language.models.offers.MusicVideoOffer` attribute), 19  
`url` (`yandex_market_language.models.offers.SimplifiedOffer` attribute), 20  
`url` (`yandex_market_language.models.Promo` attribute), 35  
`url` (`yandex_market_language.models.Shop` attribute), 23  
`url` (`yandex_market_language.models.shop.Shop` attribute), 22  
`value` (`yandex_market_language.models.SimplifiedOffer` attribute), 25  
`value` (`yandex_market_language.models.Age` attribute), 36  
`value` (`yandex_market_language.models.Parameter` attribute), 33  
`value` (`yandex_market_language.models.Price` attribute), 25  
`value` (`yandex_market_language.models.price.Price` attribute), 21  
`vendor` (`yandex_market_language.models.AlcoholOffer` attribute), 32  
`vendor` (`yandex_market_language.models.ArbitraryOffer` attribute), 26  
`vendor` (`yandex_market_language.models.AudioBookOffer` attribute), 29  
`vendor` (`yandex_market_language.models.BookOffer` attribute), 27  
`vendor` (`yandex_market_language.models.EventTicketOffer` attribute), 31  
`vendor` (`yandex_market_language.models.MedicineOffer` attribute), 30  
`vendor` (`yandex_market_language.models.MusicVideoOffer` attribute), 30  
`vendor` (`yandex_market_language.models.offers.AbstractBookOffer` attribute), 10  
`vendor` (`yandex_market_language.models.offers.AbstractOffer` attribute), 12  
`vendor` (`yandex_market_language.models.offers.AlcoholOffer` attribute), 13  
`vendor` (`yandex_market_language.models.offers.ArbitraryOffer` attribute), 14  
`vendor` (`yandex_market_language.models.offers.AudioBookOffer` attribute), 15  
`vendor` (`yandex_market_language.models.offers.BookOffer` attribute), 16  
`vendor` (`yandex_market_language.models.offers.EventTicketOffer` attribute), 17  
`vendor` (`yandex_market_language.models.offers.MedicineOffer` attribute), 18  
`vendor` (`yandex_market_language.models.offers.MusicVideoOffer` attribute), 19  
`vendor` (`yandex_market_language.models.offers.SimplifiedOffer` attribute), 20  
`vendor` (`yandex_market_language.models.SimplifiedOffer` attribute), 26  
`vendor_code` (`yandex_market_language.models.AlcoholOffer` attribute), 32  
`vendor_code` (`yandex_market_language.models.ArbitraryOffer` attribute), 26

[vendor\\_code \(yandex\\_market\\_language.models.AudioBookOffer attribute\), 29](#)  
[vendor\\_code \(yandex\\_market\\_language.models.BookOffer attribute\), 27](#)  
[vendor\\_code \(yandex\\_market\\_language.models.EventTicketOffer attribute\), 31](#)  
[vendor\\_code \(yandex\\_market\\_language.models.MedicineOffer attribute\), 30](#)  
[vendor\\_code \(yandex\\_market\\_language.models.MusicVideoOffer attribute\), 30](#)  
[vendor\\_code \(yandex\\_market\\_language.models.offers.AbstractBookOffer attribute\), 10](#)  
[vendor\\_code \(yandex\\_market\\_language.models.offers.AbstractOffer attribute\), 12](#)  
[vendor\\_code \(yandex\\_market\\_language.models.offers.AlcoholOffer attribute\), 13](#)  
[vendor\\_code \(yandex\\_market\\_language.models.offers.ArbitraryOffer attribute\), 14](#)  
[vendor\\_code \(yandex\\_market\\_language.models.offers.AudioBookOffer attribute\), 15](#)  
[vendor\\_code \(yandex\\_market\\_language.models.offers.BookOffer attribute\), 16](#)  
[vendor\\_code \(yandex\\_market\\_language.models.offers.EventTicketOffer attribute\), 17](#)  
[vendor\\_code \(yandex\\_market\\_language.models.offers.MedicineOffer attribute\), 18](#)  
[vendor\\_code \(yandex\\_market\\_language.models.offers.MusicVideoOffer attribute\), 19](#)  
[vendor\\_code \(yandex\\_market\\_language.models.offers.SimplifiedOffer attribute\), 20](#)  
[vendor\\_code \(yandex\\_market\\_language.models.SimplifiedOffer attribute\), 26](#)  
[version \(yandex\\_market\\_language.models.Shop attribute\), 23](#)  
[version \(yandex\\_market\\_language.models.shop.Shop attribute\), 22](#)  
[volume \(yandex\\_market\\_language.models.offers.AbstractBookOffer attribute\), 10](#)

## W

[weight \(yandex\\_market\\_language.models.offers.AbstractOffer attribute\), 12](#)  
[width \(yandex\\_market\\_language.models.Dimensions attribute\), 33](#)

## Y

[yandex\\_market\\_language \(module\), 36](#)  
[yandex\\_market\\_language.exceptions \(module\), 36](#)  
[yandex\\_market\\_language.models \(module\), 22](#)  
[yandex\\_market\\_language.models.category \(module\), 8](#)  
[yandex\\_market\\_language.models.currency \(module\), 8](#)