

Term Collection



If the API changes in an incompatible manner this is listed explicitly in the "important release notes" section of each [releases changelog](#).

Description

The database-side storage of terms is extended by the "TermCollection", which makes it possible, for example, to group terms that belong to a specific client. This way an end client can be assigned to one or more TermCollections.

Provides access to the Term collections available in the application.

URL:	/editor/termcollection/[ID]
Available Methods:	GET / POST / DELETE
Specialities:	<ul style="list-style-type: none">Deleting term collection will remove all terms, terms attributes and term entry attributes assigned to this term collection

Resource Layout

Name	Type	Info
id	int	GET only
name	string	The name of the term collection (unique)
timestamp	DateTime	Date and time when the term collection is created
customerId	integer	Foreign key to the customers table

POST request parameters example :

- create new empty term collection

```
{
  data: {
    "name": "The name of the term collection",
    "customerId": "the id of the customer to whom this collection is associated"
  }
}
```

GET request parameters example :

- return the term collection name, timestamp and customerId for the termcollection with id = 1

```
{
  data: {
    "id": 1
  }
}
```

Search term in term collection.

URL:	editor/termcollection/search
Available Methods:	GET
Specialities:	<ul style="list-style-type: none"> Find term/terms in the given term collection by searched string

Resource Layout

Name	Type	Info
term	string	Search term string. The search string can be combined with wildcards also.
language	array	Term language/s ids from lek_languages table
disableLimit	boolean	Disable the result limit configured in searchTermsCount config variable.

Search term entry and term attributes in term collection.

URL:	editor/termcollection/searchattribute
Available Methods:	GET
Specialities:	<ul style="list-style-type: none"> Find term entry and term attributes by groupId(termEntry id). The result array will return term attributes and term entry attribute values

Info: For search and searchattribute actions, will search in the collection assigned to the customers which customers are assigned to the currently logged user assigned customers.

Resource Layout

Name	Type	Info
groupId	string	Lek_terms table groupId field.

- Response array example

```

{
  "rows":{
    "termAttributes":[
      {
        "definition":"","
        "groupId":"001",
        "label":"gravure printing",
        "value":"11413",
        "desc":"gravure printing",
        "termId":"11413",
        "collectionId":"122",
        "languageId":"5",
        "labelId":"323",
        "attributeId":"73178",
        "parentId":null,
        "internalCount":null,
        "name":"transac",
        "attrType":"origination",
        "attrTarget":null,
        "attrId":null,
        "attrLang":"en",
        "attrValue":"origination",
        "attrCreated":"2018-05-07 17:32:07",
        "attrUpdated":"2018-05-07 17:32:07",
        "attrDataType":null,
        "language":"en"
      }
    ],
    "termEntryAttributes":[
      {
        "id":"8737",
        "labelId":"331",
        "collectionId":"122",
        "termEntryId":"4965",
        "parentId":null,
        "internalCount":"1",
        "language":null,
        "name":"note",
        "attrType":null,
        "attrDataType":null,
        "attrTarget":null,
        "attrId":null,
        "attrLang":null,
        "value":"Aleksandar",
        "created":"2018-05-07 17:32:07",
        "updated":"2018-05-07 17:32:07",
        "attributeId":"8737",
        "attrValue":"Aleksandar",
        "attrCreated":"2018-05-07 17:32:07",
        "attrUpdated":"2018-05-07 17:32:07"
      }
    ]
  }
}

```

Import terms to the term collection

URL:	editor/termcollection/import
Available Methods:	POST
Specialities:	Import the provided terms from the tbx file to the term collection

POST request parameters example(json):

```
{
  "data":{
    "collectionId": 1,
    "customerId":1000,
    "mergeTerms":true,
    "tbxUpload": ,
    "deleteTermsLastTouchedOlderThan": "2018-05-01",
    "deleteTermsOlderThanCurrentImport": true,
    "deleteProposalsLastTouchedOlderThan": "2018-05-01",
    "deleteProposalsOlderThanCurrentImport" : true
  }
}
```

Details for the resource parameters:

collectionId

The id of the term collection where those terms from the tbx file are going to

customerId

Customer to that this term collection is associated

mergeTerms

The merging of terms into an existing term collection is controlled by this boolean parameter.

[The merge process is complex and completely described here.](#)

tbxUpload

tbx file/files uploaded as multipart/form-data

deleteTermsLastTouchedOlderThan (date format Y-m-d)

Deletes all terms in all listed termCollections, that have a updated date older than the date configured for "deleteTermsLastTouchedOlderThan". Every term in a termCollection has a creation date, a modification date and an updated dated. The creation date and the modification date are taken from the dates as they appear in the TBX file.

The updated date is not shown in the GUI and is changed every time a user changes the term through touching it in the GUI or through an import.

deleteTermsOlderThanCurrentImport (boolean)

Deletes all terms in all listed termCollections, that have an "updated date" older than the start of the current import. To learn, what the "updated date" is, please read above possible values: true or false.

deleteProposalsLastTouchedOlderThan (date format Y-m-d)

Delete all proposals in a listed termCollections, that have a creation date older than this configs value.

deleteProposalsOlderThanCurrentImport (boolean)

Delete all proposals in a listed termCollections, that have a creation date older than the start of the current import.

Important: with the import call, one or multiple tbx files must be provided.

There can be additional possibility to import terms to the term collection.

1. Import tbx files via language resources panel with term collection as resource
2. Using the term import plugin (see more at [Plug-In TermlImport: TermlImport](#))