

# Package ‘libraryCurator’

March 24, 2021

**Title** libraryCurator

**Version** 1.36.0-1

**API version** v0.1

**Description** This swagger page describes the libraryCurator APIs. Before carrying out any API calls you will need an API token. API tokens can be obtained under your profile within the Genestack software. To try out calls in this swagger page: 1. Click the 'Authorize' button below to enter your API token 2. Scroll to the 'Parameters' section for the method you wish to try out and click the 'Try it out' button 3. Enter parameter values that you wish to try 4. Scroll to the bottom of the Parameters section and click the 'Execute' bar that appears The server response will be in the section that follows.

**Depends** R (>= 3.3.2)

**License** Proprietary

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.1.1

## R topics documented:

LibrarySPoTApi_add_libraries . . . . .	1
LibrarySPoTApi_delete_library . . . . .	2
LibrarySPoTApi_delete_library_group . . . . .	2
LibrarySPoTApi_get_libraries_by_group . . . . .	3
LibrarySPoTApi_get_library . . . . .	3
LibrarySPoTApi_get_library_by_version . . . . .	4
LibrarySPoTApi_get_library_versions . . . . .	5
LibrarySPoTApi_search_libraries . . . . .	5
LibrarySPoTApi_update_library . . . . .	6

LibrarySPoTApi\_add\_libraries

*Add new library object*

---

**Description**

Add new library object

**Usage**

LibrarySPoTApi\_add\_libraries(source, ...)

**Arguments**

source

---

LibrarySPoTApi\_delete\_library

*Delete a library object*

---

**Description**

Delete a library object

**Usage**

LibrarySPoTApi\_delete\_library(id, ...)

**Arguments**

id                      Unique identifier (accession) of the object.

---

LibrarySPoTApi\_delete\_library\_group

*Delete a library group with all related library objects*

---

**Description**

Delete a library group with all related library objects

**Usage**

LibrarySPoTApi\_delete\_library\_group(id, ...)

**Arguments**

id                      Unique identifier (accession) of the object.

---

`LibrarySPoTApi_get_libraries_by_group`*Retrieve libraries related to the given group*

---

**Description**

Retrieve libraries related to the given group

**Usage**

```
LibrarySPoTApi_get_libraries_by_group(  
    id,  
    returned_metadata_fields,  
    page_offset,  
    page_limit,  
    ...  
)
```

**Arguments**

<code>id</code>	Unique identifier (accession) of the object.
<code>returned_metadata_fields</code>	The parameter defines amount of metadata attributes to return: 1. <code>template</code> - return metadata attributes according to applied template, if object doesn't have applied template default template will be used. This is the default for User endpoints. 2. <code>all</code> - return all metadata attributes with values and null attributes, if they are present in the applied template. This is the default for Curator endpoints.
<code>page_offset</code>	Show the page <code>pageOffset+1</code> results from the start of the results. E.g. 100 will show a page of results starting from the 101st result. The default value is 0.
<code>page_limit</code>	Maximum number of results to return per page (see Paging above). This value must be between 0 and 2000 (inclusive). The default is 2000.

---

`LibrarySPoTApi_get_library`*Retrieve a single library object by ID (accession)*

---

**Description**

Retrieve a single library object by ID (accession)

**Usage**

```
LibrarySPoTApi_get_library(id, returned_metadata_fields, ...)
```

**Arguments**

id	Unique identifier (accession) of the object.
returned_metadata_fields	The parameter defines amount of metadata attributes to return: 1. template - return metadata attributes according to applied template, if object doesn't have applied template default template will be used. This is the default for User endpoints. 2. all - return all metadata attributes with values and null attributes, if they are present in the applied template. This is the default for Curator endpoints.

---

LibrarySPoTApi\_get\_library\_by\_version

*Retrieve a single library object by ID (accession)*

---

**Description**

Retrieve a single library object by ID (accession)

**Usage**

```
LibrarySPoTApi_get_library_by_version(  
    id,  
    version,  
    returned_metadata_fields,  
    ...  
)
```

**Arguments**

id	Unique identifier (accession) of the object.
version	Unique version of the object.
returned_metadata_fields	The parameter defines amount of metadata attributes to return: 1. template - return metadata attributes according to applied template, if object doesn't have applied template default template will be used. This is the default for User endpoints. 2. all - return all metadata attributes with values and null attributes, if they are present in the applied template. This is the default for Curator endpoints.

---

`LibrarySPoTApi_get_library_versions`*Retrieve a list of object versions by ID*

---

**Description**

Retrieve a list of object versions by ID

**Usage**

```
LibrarySPoTApi_get_library_versions(id, ...)
```

**Arguments**

<code>id</code>	Unique identifier (accession) of the object.
-----------------	--

---

`LibrarySPoTApi_search_libraries`*List or search for library metadata objects*

---

**Description**

List or search for library metadata objects

**Usage**

```
LibrarySPoTApi_search_libraries(  
    filter,  
    query,  
    search_specific_terms,  
    returned_metadata_fields,  
    sort,  
    page_limit,  
    page_offset,  
    ...  
)
```

**Arguments**

<code>filter</code>	Filter by library metadata (key-value metadata pair(s)). E.g. "Library Type"=RNA-Seq-1
<code>query</code>	Search for library objects via a full-text query over all library metadata fields. E.g. "illumina HiSeq500"

**search\_specific\_terms**

If the full-text query term is present in an ODM dictionary, enabling this parameter will modify the query to include child terms of the full-text query. For example, the search query "Body fluid" can be expanded to include the term "Blood" (a child term of "Body fluid") so files containing either "Body fluid" or "Blood" in their metadata will be returned in the search results. The parent-child relationship is defined by the key "broaders" in the dictionary. If the full query term is not present in a dictionary then this parameter has no effect.

**returned\_metadata\_fields**

The parameter defines amount of metadata attributes to return: 1. `template` - return metadata attributes according to applied template, if object doesn't have applied template default template will be used. This is the default for User endpoints. 2. `all` - return all metadata attributes with values and null attributes, if they are present in the applied template. This is the default for Curator endpoints.

**sort**

Attribute to sort by, with optional ascending/descending marker, of the form "[+|-]<key\_name>". No marker or "+" indicates ascending sort, and "-" indicates descending sort. \*Default:\* sort by ID in ascending order.

**page\_limit**

Maximum number of results to return per page (see Paging above). This value must be between 0 and 2000 (inclusive). The default is 2000.

**page\_offset**

Show the page `pageOffset+1` results from the start of the results. E.g. 100 will show a page of results starting from the 101st result. The default value is 0.

---

**LibrarySPoTApi\_update\_library**

*Update a library object*

---

**Description**

Update a library object

**Usage**

```
LibrarySPoTApi_update_library(id, body, ...)
```

**Arguments**

<code>id</code>	Unique identifier (accession) of the object.
<code>body</code>	Metadata in the form of {key: value, key2: value2, ...}