

# Package ‘HCAExplorer’

December 14, 2020

**Type** Package

**Title** Browse the Human Cell Atlas data portal

**Version** 1.4.0

**Description** Search, browse, reference, and download resources from the Human Cell Atlas data portal. Development of this package is supported through funds from the Chan / Zuckerberg initiative.

**Depends** R (>= 3.6.0), dplyr

**Imports** BiocFileCache, HCAMatrixBrowser, S4Vectors, LoomExperiment, vctrs, curl, httr, jsonlite, methods, plyr, readr, rlang, tibble, tidygraph, utils, xml2

**Suggests** BiocStyle, knitr, rmarkdown, testthat (>= 2.1.0)

**License** Artistic-2.0

**Encoding** UTF-8

**LazyData** true

**URL** <https://github.com/Bioconductor/HCABrowser>

**RoxygenNote** 7.1.0

**biocViews** DataImport, Sequencing

**BugReports** <https://github.com/Bioconductor/HCABrowser/issues>

**VignetteBuilder** knitr

**git\_url** <https://git.bioconductor.org/packages/HCAExplorer>

**git\_branch** RELEASE\_3\_12

**git\_last\_commit** 7ea8f44

**git\_last\_commit\_date** 2020-10-27

**Date/Publication** 2020-12-13

**Author** Daniel Van Twisk [aut],  
Martin Morgan [aut],  
Bioconductor Package Maintainer [cre]

**Maintainer** Bioconductor Package Maintainer <maintainer@bioconductor.org>

## R topics documented:

activate.HCAExplorer . . . . .	2
download-expression-matrices . . . . .	3
fields,HCAExplorer-method . . . . .	4
filter.HCAExplorer . . . . .	5
getManifest . . . . .	6
getManifestFileFormats . . . . .	7
HCAExplorer . . . . .	8
HCAExplorer-class . . . . .	9
nextResults . . . . .	9
ProjectView-class . . . . .	10
resetQuery . . . . .	10
resetSelect . . . . .	11
results . . . . .	12
select.HCAExplorer . . . . .	13
show,HCAExplorer-method . . . . .	14
undoQuery . . . . .	14
values,HCAExplorer-method . . . . .	15
viewProjects . . . . .	16
[,HCAExplorer,ANY,ANY,ANY-method . . . . .	17
<b>Index</b>	<b>19</b>

---

activate.HCAExplorer    *Activate projects, samples, or files to display in the HCAExplorer Object*

---

### Description

The HCAExplorer can display its results in a variety of ways. Choose whether to display entries by project, samples, or files. The HCAExplorer class always defaults to projects.

### Usage

```
## S3 method for class 'HCAExplorer'
activate(.data, what = c("projects", "samples", "files"))
```

### Arguments

.data	An HCAExplorer object
what	character(1). Either 'projects', 'samples', or 'files'.

### Value

An HCAExplorer object with modified activation.

### See Also

[HCAExplorer](#) for the HCAExplorer class.

**Examples**

```
## Initiate an HCAExplorer object.
x <- HCAExplorer()
## Display the object. Notice "projects" are shown by default.
x

## Now activate "samples" and now notice that "samples" are displayed.
x <- activate(x, 'samples')
x

## Now activate "files" and now notice that "files" are displayed.
x <- activate(x, 'files')
x

## Now activate "projects" and now notice that the original "projects" are
## displayed.
x <- activate(x, 'projects')
x
```

---

download-expression-matrices

*Download Expression Matrices with HCAExplorer*

---

**Description**

Methods used to download expression matrices available through the HCA Data Portal. The method ‘checkExpressionMatricesAvailability’ specifies whether expression matrices can be downloaded for a project. The method ‘downloadExpressionMatrices’ downloads and returns expression matrices in a variety of formats.

**Usage**

```
checkExpressionMatricesAvailability(x, ...)
downloadExpressionMatrices(x, ...)

## S4 method for signature 'HCAExplorer'
checkExpressionMatricesAvailability(
  x,
  format = c("loom", "csv", "mtx"),
  organism
)

## S4 method for signature 'HCAExplorer'
downloadExpressionMatrices(
  x,
  format = c("loom", "csv", "mtx"),
  organism,
  useBiocFileCache = TRUE
)
```

**Arguments**

x	an HCAExplorer object. All project entries in the object will attempt to be downloaded. Use subsetting to reduce the projects that are to be downloaded.
format	either "loom", "csv", or "mtx". The default value is "loom". Which format the expression matrices should be downloaded as. If "loom", a 'LoomExperiment' object will be returned. If "csv", a tibble will be returned. If "mtx", a 'Single-CellExperiment' object will be returned.
organism	character(1). Which organism file to download. Usually either "Homo sapiens" or "Mus musculus". If not specified, all organism files will be included in the download.
useBiocFileCache	logical(1). Whether to save file with 'BiocFileCache'. Defaults to 'TRUE'. If 'TRUE', 'BiocFileCache' will be used to save a persistent copy of the file to the user's computer. This persistent copy will be used instead of redownloading the expression matrices in subsequent uses. If 'FALSE', the file will be saved to a temporary directory and will be removed when the session ends.
...	Additional arguments.

**Value**

For 'checkExpressionMatrixAvailability', returns a tibble. For 'downloadExpressionMatrices', returns a list of downloaded expression matrices in a format specified by the 'format' argument.

**Author(s)**

Daniel Van Twisk

**Examples**

```
hca <- HCAExplorer()
checkExpressionMatricesAvailability(hca)

hca <- hca[5]
hca <- downloadExpressionMatrices(hca, useBiocFileCache = FALSE)
```

---

fields,HCAExplorer-method

*List supported fields of an HCAExplorer object*

---

**Description**

This function is used to discover possible fields that can be queried upon in an HCAExplorer object.

**Usage**

```
## S4 method for signature 'HCAExplorer'
fields(x)
```

**Arguments**

x                    An HCAExplorer object.

**Value**

A tibble indicating fields that can be queried upon.

**See Also**

[HCAExplorer](#) for the HCAExplorer class, [values](#) for how to obtain possible values of a field.

**Examples**

```
## Intiate an HCAExplorer object.
x <- HCAExplorer()

## Find which fields can be queried upon.
fields(x)
```

---

filter.HCAExplorer      *Filter an HCAExplorer object using fields and values*

---

**Description**

Given some amount of fields and values associated with them, modify the search performed by the HCAExplorer object. This function adds terms to the query that is ultimately performed.

**Usage**

```
## S3 method for class 'HCAExplorer'
filter(.data, ..., .preserve)
```

**Arguments**

.data                An HCAExplorer object to query.  
 ...                 Any number of fields and values to queried against. The binary operators '==' and '&' is allowed. See examples.  
 .preserve           Unused argument

**Value**

An HCAExplorer object with the desired query performed.

**See Also**

[HCAExplorer()]  
[HCAExplorer](#) for the HCAExplorer class.

**Examples**

```
## Initiate an HCAExplorer Object
x <- HCAExplorer()

## First we want to perform a search for certain organs.
## Display possible fields looking for organs.
fields(x)
## organs can be queried with "organ".
## What values can the field "organ" have?
values(x, "organ")

## Construct a query looking for projects that involve blood or brain.
y <- x %>% filter(organ %in% c('blood', 'brain'))
y

## Now suppose one would also like to find projects that have a certain
## disease. What field corresponds to disease?
fields(y)
## The "disease" field looks right.
## What possible disease values can be queried upon?
values(y, "disease")

## Add a query for a 'normal' disease state to our search.
y <- y %>% filter(disease == 'normal')
y

## This entire query can also be performed at once in several ways.
x %>% filter(organ == c('blood', 'brain') & disease == 'normal')
x %>% filter(organ == c('blood', 'brain'), disease == 'normal')
```

---

**getManifest***Obtain metadata information from an HCAExplorer object*

---

**Description**

Obtain metadata information from an HCAExplorer object. This metadata can then be passed on to download files from other services.

**Usage**

```
## S4 method for signature 'HCAExplorer'
getManifest(x, fileFormat)
```

**Arguments**

<code>x</code>	An HCAExplorer object
<code>fileFormat</code>	character. A character vector of file formats of metadata to obtain. The possible arguments can be found using the <code>getManifestFileFormats</code> method.

**Value**

a tibble of metadata information.

**See Also**

[HCAExplorer](#) for the HCAExplorer class, [getManifestFileFormats](#) for how to obtain file formats that can be used as arguments in this function.

**Examples**

```
## Initiate an HCAExplorer object.
x <- HCAExplorer()
## View the HCAExplorer object to decide which projects to subset.
x

## Decide to subset first project.
x <- x[1,]
x

## Get all possible manifest file formats for the project.
formats <- getManifestFileFormats(x)
formats

## Obtain the manifest for the file using only the first format
manifest <- getManifest(x, formats[1])
manifest
```

---

getManifestFileFormats

*Show all possible manifest file formats for current selection*

---

**Description**

Show all possible manifest file formats for the current selection of projects in the HCAExplorer object. To be used in conjunction with 'getManifest()'.

**Usage**

```
## S4 method for signature 'HCAExplorer'
getManifestFileFormats(x)
```

**Arguments**

x                    An HCAExplorer object

**Value**

A character vector of information about possible file formats.

**See Also**

[HCAExplorer](#) for the HCAExplorer class [getManifest](#) for how to obtain the manifest file.

**Examples**

```
## Initiate an HCAExplorer object.
x <- HCAExplorer()
## View the HCAExplorer object to decide which projects to subset.
x

## Decide to subset to just obtain the first two projects.
x[1:2,]
x

## Get all possible manifest file formats for projects.
formats <- getManifestFileFormats(x)
formats
```

---

HCAExplorer

*The Project Browser Class*

---

**Description**

A class that displays Human Cell Atlas(HCA) information obtained from a digested version of the Human Cell Atlas referred to as the Azul Backend. This class allows one to view, filter, and select entries in the to ultimately download metadata files and then other data such as matrices.

**Usage**

```
HCAExplorer(
  url = "https://service.explore.data.humancellatlas.org",
  per_page = 15
)
```

**Arguments**

`url` character(1) The url of the Human Cell Atlas Backend to access.  
`per_page` the number of results to display per request.

**Value**

a HCAExplorer object

**Author(s)**

Daniel Van Twisk

**Examples**

```
hca <- HCAExplorer(per_page = 15,
                  url = 'https://service.explore.data.humancellatlas.org')
hca
```



---

HCAExplorer-class      *The Project Browser Class*

---

**Description**

A class that displays Human Cell Atlas(HCA) information obtained from a digested version of the Human Cell Atlas referred to as the Azul Backend. This class allows one to view, filter, and select entries in the to ultimately download metadata files and then other data such as matrices.

**Arguments**

per\_page            the number of results to display per request.  
url                  character(1) The url of the Human Cell Atlas Backend to access.

**Author(s)**

Daniel Van Twisk

---

nextResults            *Next Results*

---

**Description**

Fetch the next set of entries from a HCAExplorer Object

**Usage**

```
## S4 method for signature 'HCAExplorer'  
nextResults(x)
```

**Arguments**

x                    An HCAExplorer object

**Value**

A HCAExplorer object that displays the next results

**Author(s)**

Daniel Van Twisk

**See Also**

[HCAExplorer](#) for the HCAExplorer class,

**Examples**

```
## Intitate an HCAExplorer object.
x <- HCAExplorer()
x

## View the next set of results in the HCAExplorer object.
x <- nextResults(x)
x
```

---

ProjectView-class      *The ProjectView class*

---

**Description**

A class that displays all metadata information from a list of projects in an HCAExplorer object. Extends lists.

**Arguments**

...                      Arguments to be passed to list constructor.

---

resetQuery              *Reset all queries performed on an object*

---

**Description**

Reset all queries performed on an HCAExplorer object then return the result.

**Usage**

```
## S4 method for signature 'HCAExplorer'
resetQuery(x)
```

**Arguments**

x                          An HCAExplorer object

**Value**

An HCAExplorer object with the changes applied to it.

**See Also**

[HCAExplorer](#) for the HCAExplorer class, [undoQuery](#) for how to undo some queries.

**Examples**

```

## Initiate an HCAExplorer Object
x <- HCAExplorer()

## First we want to perform a search for certain organs.
## Display possible fields looking for organs.
fields(x)
## organs can be queried with "organ".
## What values can the field "organ" have?
values(x, "organ")

## Construct a query looking for projects that involve brain.
x <- x %>% filter(organ == brain)
x

## Now select the first two projects by row.
x <- x[1:2,]
x

## Finally we can remove the previous two queries to get the original empty
## search.
x <- resetQuery(x)
x

```

---

resetSelect

*Reset a select perform on an HCAExplorer object.*


---

**Description**

This function will reset the selected slot of an HCAExplorer object to its default selection.

**Usage**

```

## S4 method for signature 'HCAExplorer'
resetSelect(x)

```

**Arguments**

x                    An HCAExplorer Object

**Value**

An HCAExplorer object with its selected slot returned to its default value.

**See Also**

[HCAExplorer](#) for the HCAExplorer class, [select](#) for how to make the initial selection.

**Examples**

```
## Initiate an HCAExplorer Object.
x <- HCAExplorer()
## View object to decide which columns to select.
x

## Decide to select columns 'projects.projectTitle' and 'samples.organ'.
x <- x %>% select('project.projectTitle', 'samples.organ')
x

## Revert select() with resetSelect()
x <- resetSelect(x)
x
```

---

results

*Obtain search unattenuated results tibble from an HCAExplorer Object*

---

**Description**

Returns a tibble either showing bundles or files based on whichever is activated. The tibble returned will display the full results of the search instead of the attenuated table shows when displaying the HCAExplorer object.

**Usage**

```
## S4 method for signature 'HCAExplorer'
results(x)
```

**Arguments**

x                    An HCAExplorer object.

**Value**

a tibble giving the unattenuated search results of the HCAExplorer object.

**See Also**

[HCAExplorer](#) for the HCAExplorer class.

**Examples**

```
## Initiate an HCAExplorer object.
x <- HCAExplorer()
## Display the object. Note that the tibble contains very few columns.
x

## Access the tibble and display at available columns using results()
results(x)
```

---

select.HCAExplorer	<i>Select columns to display upon showing the object</i>
--------------------	--

---

### Description

An HCAExplorer object is initiated with certain default columns being displayed upon showing the object. This method allows a user to display columns other than the default columns when displaying the object.

### Usage

```
## S3 method for class 'HCAExplorer'  
select(.data, ...)
```

### Arguments

.data	An HCAObject to filter.
...	Columns to be displayed.

### Value

An HCAExplorer object with the applied filter.

### See Also

[HCAExplorer](#) for the HCAExplorer class, [resetSelect](#) for how to reset the selection applied to an HCAExplorer object.

### Examples

```
## Intiate an HCAExplorer object  
x <- HCAExplorer()  
x  
  
## Use the results() method to display which columns are present.  
results(x)  
  
## Select the 'projects.projectTitle' and 'samples.organ' columns.  
x <- x %>% select('projects.projectTitle', 'samples.organ')  
x  
  
## Use resetSelect() to return to the original selection  
x <- resetSelect(x)  
x
```

show, HCAExplorer-method

*Show HCAExplorer*

---

### Description

Show HCAExplorer

### Usage

```
## S4 method for signature 'HCAExplorer'  
show(object)
```

### Arguments

object            a HCAExplorer object to show

### Value

outputs a text representation of the object

### See Also

[HCAExplorer](#) for the HCAExplorer class.

### Examples

```
## Initiate an HCAExplorer object.  
x <- HCAExplorer()  
## Invoke show() by simply displaying the object.  
x
```

---

undoQuery

*Undo one or multiple queries performed on an object*

---

### Description

Undo one or multiple queries performed on an HCAExplorer object then return the result.

### Usage

```
## S4 method for signature 'HCAExplorer'  
undoQuery(x, n = 1L)
```

### Arguments

x                    An HCAExplorer object  
n                    integer(1). The number of queries to step back from.

**Value**

An HCAExplorer object with the changes applied to it.

**See Also**

[HCAExplorer](#) for the HCAExplorer class, [resetQuery](#) for resetting the entire query.

**Examples**

```
## Initiate an HCAExplorer Object
x <- HCAExplorer()

## First we want to perform a search for certain organs.
## Display possible fields looking for organs.
fields(x)
## organs can be queried with "organ".
## What values can the field "organ" have?
values(x, "organ")

## Construct a query looking for projects that involve brain.
x <- x %>% filter(organ == brain)
x

## Now select the first two projects by row.
x <- x[1:2,]
x

## Finally we can remove the previous query using undoQuery().
## search.
x <- undoQuery(x, n = 1)
x
```

---

values,HCAExplorer-method

*List all values for certain fields in a HCAExplorer Object*

---

**Description**

To be used in conjunction with the fields function. This function allows one to find which values can be queried upon in a given field.

**Usage**

```
## S4 method for signature 'HCAExplorer'
values(x, fields)
```

**Arguments**

x	An HCAExplorer Object.
fields	A character vector of fields to display available values for.

**Value**

a character vector of possible values for a filter.

**See Also**

[HCAExplorer](#) for the HCAExplorer class, [fields](#) for how to search for fields in an HCAExplorer object.

**Examples**

```
## Intiate an HCAExplorer object.
x <- HCAExplorer()

## Find which firelds can be queried upon.
fields(x)

## Which values can "organ" be searched with?
values(x, 'organ')
```

---

viewProjects

*View all metadata about a selection of projects*

---

**Description**

Returns a list of all the metadata from the current selection of entries in the HCAExplorer object as a projectView object. This method is is meant to return a clear and useful representation of the metadata of a selection of projects.

**Usage**

```
## S4 method for signature 'HCAExplorer'
viewProjects(x)
```

**Arguments**

x                    An HCAExplorer object

**Value**

A list of all metadata in the selected entries. This list will contain mulitple lists representing the metadata.

**See Also**

[HCAExplorer](#) for the HCAExplorer class.

[HCAExplorer()][projectView()]



**Examples**

```
## Initiate an HCAExplorer object.
x <- HCAExplorer()
x

## Use viewProjects to show all metadata information relating to a project.
view <- viewProjects(x)
view

## Subset the data to obtain the first two rows.
x <- x[1:2,]
x

## Fewer projects have there metadata shown due to our previous subset.
view <- viewProjects(x)
view
```

---

[,HCAExplorer,ANY,ANY,ANY-method

*Subset an HCAExplorer Object by row number or project name*

---

**Description**

Allows subsetting an HCAExplorer object by row number or project name. This method internally acts as a filter and actually performs a query using "projectIds".

**Usage**

```
## S4 method for signature 'HCAExplorer,ANY,ANY,ANY'
x[i, j, ..., drop = TRUE]
```

**Arguments**

x	An HCAExplorer object.
i	Either a numeric vector indicating which rows to choose or a character vector of project titles indicating which projects to choose.
j	Unused argument.
...	Unused argument.
drop	Unused argument.

**Value**

An HCAExplorer object with the applied subset.

**See Also**

[HCAExplorer](#) for the HCAExplorer class, [filter](#) for how to perform a query on an HCAExplorer object.

**Examples**

```
## Initiate an HCAExplorer Object.  
x <- HCAExplorer()  
## View object to decide which projects to choose.  
x  
## Decide to select projects 1, 2, 3, and 7  
x <- x[c(1:3, 7)]  
x
```

# Index

.HCAExplorer (HCAExplorer-class), 9  
.ProjectView (ProjectView-class), 10  
[, HCAExplorer, ANY, ANY, ANY-method, 17  
'[' , HCAExplorer-method  
    ([], HCAExplorer, ANY, ANY, ANY-method),  
    17  
activate.HCAExplorer, 2  
checkExpressionMatricesAvailability  
    (download-expression-matrices),  
    3  
checkExpressionMatricesAvailability, HCAExplorer-method,  
    (download-expression-matrices),  
    10, 14  
    3  
download-expression-matrices, 3  
downloadExpressionMatrices  
    (download-expression-matrices),  
    3  
downloadExpressionMatrices, HCAExplorer-method  
    (download-expression-matrices),  
    3  
fields, 16  
fields, HCAExplorer-method, 4  
filter, 17  
filter.HCAExplorer, 5  
getManifest, 6, 7  
getManifest, HCAExplorer-method  
    (getManifest), 6  
getManifestFileFormats, 7, 7  
getManifestFileFormats, HCAExplorer-method  
    (getManifestFileFormats), 7  
HCAExplorer, 2, 5, 7, 8, 9–17  
HCAExplorer-class, 9  
nextResults, 9  
nextResults, HCAExplorer-method  
    (nextResults), 9  
ProjectView-class, 10  
resetQuery, 10, 15  
resetQuery, HCAExplorer-method  
    (resetQuery), 10  
resetSelect, 11, 13  
resetSelect, HCAExplorer-method  
    (resetSelect), 11  
results, 12  
results, HCAExplorer-method (results), 12  
select, 11  
select.HCAExplorer, 13  
show, HCAExplorer-method, 14  
undoQuery, HCAExplorer-method  
    (undoQuery), 14  
values, 5  
values, HCAExplorer-method, 15  
viewProjects, 16  
viewProjects, HCAExplorer-method  
    (viewProjects), 16