SimonFraserUniversity-Corrie   👤▾   ?

immunoSEQ ANALYZER
(/)

Help

Search
[   ] Go

Browse
[choose a topic   ▾]

Need to talk?
Contact Us

The immunoSEQ REST API: documentation

The immunoSEQ REST API enables customers to directly access their immunosequencing data programmatically, using Basic or Token based authentication via HTTPS. POST and GET requests can be made to the endpoints to retrieve data for the specified user, workspace, and project. Both synchronous and asynchronous version are available.

Usage Requirements

Use HTTPS

Calls to the immunoSEQ REST API must be made using an HTTPS connection.

Specify Authorization Type (Basic or Token)

The Authorization header must be specified in one of two forms: Basic or Token. For **Basic auth**, the "Authorization" header must be `"Basic "` + a base 64 encoding of username followed by a colon, then the user's password (e.g. `"user@domain.com:password"`). For **Token auth**, the "Authorization" header must be `"Token "` + a token retrieved by the `Authenticate` endpoint (see details below).

Set CSRF-token=nocheck

immunoSEQ implements CSRF validation to prevent cross-site request forgery. By setting the `CSRF-token` header to `"nocheck"`, API endpoints are able to bypass this check.

Base URL

The base URL for all endpoints: `https://analyzer.adaptivebiotech.com/api/`

API Endpoints

`authenticate`
This endpoint uses Basic Auth to generate an authorization token that can be used for Token based auth on the other endpoints.

Request Method: `GET`

URL `authenticate?workspace=<workspace_name or workspace_id>&project=<project_name or project_id>`

Parameters:

`workspace` - the name or unique identifier of the workspace that the user has access to that will be accessed via the other endpoints

`project` - the name or unique identifier of the project in the workspace that contains the data that will be accessed via the other endpoints. The project name "Summary" can be used to query all samples in the workspace

Body: None

Result:

The body of the result is plain/text containing and encrypted authorization token that can be sent in the Authorization header of future requests to the other endpoints as an alternative to Basic authentication.

Notes:

The token will remain valid for up to 8 hours.

`queryProject`
This is a synchronous call to query sample and/or sequence data for a workspace and project.

(http://www.adaptivebiotech.com/)

Request Method: POST

FOR RESEARCH USE ONLY

URL: queryProject?workspace=<workspace_name or workspace_id>&project=<project_name or project_id>&separator=
<separator_char>&useExportFormat=<true | false>

Parameters:

`workspace` - the name or unique identifier of the workspace that the user has access to that contains the data to be queried (ignored if Token auth header provided)

`project` - the name or unique identifier of the project in the workspace that contains the data to be queried (ignored if Token auth header provided). The project name "Summary" can be used to query all samples in the workspace

`separator` - the character (URL-encoded) to use as the field separator of the result data that is returned. For example, `%2C` for a csv file.

`useExportFormat` - optional; defaults to "true". When set to "false" result values are returned with minimal transformation, specifically: doubles/fractions are returned with full precision, null columns are returned as "null" and sample_tags are returned with tag family prepended.

Body:

Set the `content-type` header to `text/plain`

Body contains a SQL-like query where "from" and "join" expressions reference the `samples` or `sequences` tables. See *Building Queries* below.

Result:

The body of the result is plain/text containing a header row containing the names of the columns separated by the separator character passed as a parameter. Each additional line represents the data for a row of results from the query with each column matching the corresponding header column name separated by the character specified in the "separator" parameter.

Notes:

This is a synchronous call and may timeout after 2 minutes. If you need to make calls that can run longer, use the `queryProjectAsync` API endpoint.

queryProjectAsync

This is an asynchronous call to query sample and/or sequence data for a workspace and project. The call returns a token that can be used to call the `getResult` API endpoint to poll for the query results.

Request Method: POST

URL: `queryProjectAsync?workspace=<workspace_name or workspace_id>&project=<project_name or project_id>&separator=` FOR RESEARCH USE ONLY

`<separator_char>&useExportFormat=<true | false>`

Parameters:

`workspace` - the name or unique identifier of the workspace that the user has access to that contains the data to be queried (ignored if Token auth header provided)

`project` - the name or unique identifier of the project in the workspace that contains the data to be queried (ignored if Token auth header provided). The project name "Summary" can be used to query all samples in the workspace

`separator` - the character (URL-encoded) to use as the field separator of the result data that is returned. For example, `%2C` for a csv file.

`useExportFormat` - optional; defaults to "true". When set to "false" result values are returned with minimal transformation, specifically: doubles/fractions are returned with full precision, null columns are returned as "null" and sample_tags are returned with tag family prepended.

Body:

Set the `content-type` header to `text/plain`

Body contains a SQL-like query where "from" and "join" expressions reference the `samples` or `sequences` tables. See *Building Queries* below.

Result:

A token that can be used to poll the `getResult` endpoint for the query results.

getResult

This is a synchronous call to poll for result data from a previous call to `queryProjectAsync`.

Request Method: `GET`

URL (Basic auth): `getResult/<result_token>?workspace=<workspace_name or workspace_id>&project=<project_name or project_id>`

URL (Token auth): `getResult/<result_token>`

Parameters:

`workspace` - the name or unique identifier of the workspace that the user has access to that contains the data to be queried (ignored if Token auth header provided)

`project` - the name or unique identifier of the project in the workspace that contains the data to be queried (ignored if Token auth header provided). The project name "Summary" can be used to query all samples in the workspace

`result_token` - the token that was returned in the body of a previous call to `queryProjectAsync`

Body: None

Result:

If the data isn't available yet, the response will contain HTTP status code `"202"` and the body of the response will be `"FALSE"`. If the data is available, the body of the response will contain data from the call made to initiate the data query. Data returned from the `getResult` call will contain a header row containing the names of the columns separated by the separator character passed as a parameter. Each additional line represents the data for a row of results from the query with each column matching the corresponding header column name separated by the character specified in the "separator" parameter of the initiating call.

Notes:

If any errors occurred during the process of the original endpoint call that initiated the data query `getResult` will return an appropriate HTTP status code indicating the error.

Building Queries

Query formats expected in the body of certain immunoSEQ API endpoints use the same schemas as the Analyzer's Advanced Query view, with one exception: endpoints do not inherit any analysis filters, so will query all samples in the project.

Queries may explicitly reference the `samples` or `sequences` table (matching Advanced Query syntax), or reference these (http://www.adaptivebiotech.com/)

tables by tokens; {samples_table} or {sequences_table}

To see exhaustive documentation on these fields, choose Help(/help?target=!immunoSEQ%20Overview) from the top-right
corner of any Analyzer page, and then choose "Column Info" from the dropdown menu of topics.

FOR RESEARCH USE ONLY

(http://www.adaptivebiotech.com/)