

Project Resources and Files API

Contents

- [Get A Listing Of Resource Folders Stored With A Project](#)
- [Get A Listing Of Resource Files Stored With A Project](#)
- [Get The Contents Of A Project Resource File](#)
- [Create A New Project Resource Folder](#)
- [Upload A New Project Resource File](#)
- [Delete A Project Resource Folder](#)
- [Delete A Project Resource File](#)

Related

- [How To Download Files via the XNAT REST API](#)

Get A Listing Of Resource Folders Stored With A Project

```
GET - /data/projects/{project-id}/resources
```

Parameters

{project-id}	Required path parameter
format	Optional querystring parameter. Specify the format of the returned response. <ul style="list-style-type: none">• json (default)• html• xml• csv
sortBy	Optional querystring parameter: Sort the returned results by one or more parameters in the Result array. Multiple parameters can be provided, separated by commas.

Response

```
{
  "ResultSet": {
    "Result": [
      {
        "file_count": "integer",
        "xnat_abstractresource_id": "integer {resource-id}",
        "cat_desc": "string",
        "cat_id": "",
        "format": "string",
        "label": "string {resource-label}",
        "category": "resources",
        "element_name": "xnat:resourceCatalog",
        "file_size": "integer (bytes)",
        "content": "",
        "tags": ""
      }
    ],
    "totalRecords": "integer",
    "title": "Resources"
  }
}
```

Get A Listing Of Resource Files Stored With A Project

```
GET - /data/projects/{project-id}/files
```

Parameters

{project-id}	Required path parameter
format	Optional querystring parameter. Specify the format of the returned response. <ul style="list-style-type: none">• json (default)• html• xml• csv
sortBy	Optional querystring parameter: Sort the returned results by one or more parameters in the Result array. Multiple parameters can be provided, separated by commas.

Response

```
{
  "ResultSet": {
    "Columns": [
      {
        "key": "URI",
        "serverRoot": ""
      }
    ],
    "Result": [
      {
        "file_content": "",
        "Size": "integer",
        "file_tags": "",
        "cat_ID": "integer {resource-id}",
        "digest": "hash",
        "collection": "string {resource-label}",
        "URI": "/data/projects/{project-id}/resources/{resource-id}/files/{filename}",
        "file_format": "",
        "Name": "string {filename}"
      },
      ...
    ],
    "title": "Files"
  }
}
```

Get The Contents Of A Project Resource File

```
GET - /data/projects/{project-id}/resources/{resource-id | resource-label}/files/{filename}
```

Parameters

{project-id}	Required path parameter
{resource-id} or {resource-label}	Required path parameter
{filename}	Required path parameter

Response: Success (200) returns the file contents in its native format.

Create A New Project Resource Folder

```
PUT - /data/projects/{project-id}/resources/{resource-label}
```

Parameters

{project-id}	Required path parameter
---------------------	-------------------------

<code>{resource-label}</code>	Required path parameter
-------------------------------	-------------------------

Response: none

Upload A New Project Resource File

```
PUT - /data/projects/{project-id}/resources/{resource-id | resource-label}/files/{filename}
```

Parameters

<code>{project-id}</code>	Required path parameter
<code>{resource-id}</code> or <code>{resource-label}</code>	Required path parameter
<code>{filename}</code>	Required path parameter
<code>inbody=true</code>	Required querystring parameter (if including the file in the body of the request)
<code>file</code>	Required body parameter.

Response: none

Delete A Project Resource Folder



This URI also permanently deletes all contents of a resource folder

```
DELETE - /data/projects/{project-id}/resources/{resource-id | resource-label}
```

Parameters

<code>{project-id}</code>	Required path parameter
<code>{resource-id}</code> or <code>{resource-label}</code>	Required path parameter

Response: none

Delete A Project Resource File

```
DELETE - /data/projects/{project-id}/resources/{resource-id | resource-label}/files/{filename}
```

Parameters

<code>{project-id}</code>	Required path parameter
<code>{resource-id}</code> or <code>{resource-label}</code>	Required path parameter
<code>{filename}</code>	Required path parameter

Response: none