{JIRA Case: <http://jira.pentaho.com/browse/ESR-5610>, which is also related to http://jira.pentaho.com/browse/BISERVER-12489}

{MindTouch Article: <https://help.pentaho.com/Documentation/7.0/0R0/070/010/0A0/0O0>}

{GitHiub Code: <https://github.com/pentaho/pentaho-platform/blob/master/extensions/src/org/pentaho/platform/web/http/api/resources/FileResource.java#L212-L263>}

**File Resource**

# /repo/files/backup

## GET

Performs a complete system back up of the Pentaho system. This includes content, schedules, users, roles, datasources, and the metastore.

**Example Request:**  
GET pentaho/api/repo/files/backup

### Parameters

|  |  |  |
| --- | --- | --- |
| name | description | type |
| user-agent | A string representing the type of browser to use. Currently only applicable if contains 'FireFox' as FireFox requires a header with encoding information (UTF-8) and a quoted filename, otherwise encoding information is not supplied and the filename is not quoted. | header |

### Response Body

|  |  |
| --- | --- |
| element: | (custom) |
| media types: | \*/\* application/xml application/octet-stream |

A jax-rs Response object with the appropriate status code, header, and body.

**Example Response:**

|  |
| --- |
| Encrypted file stream |

### Status Codes

|  |  |
| --- | --- |
| code | description |
| 200 | Successfully exported the existing Pentaho System |
| 403 | User does not have administrative permissions |
| 500 | Failure to complete the export. |

# /repo/files/systemRestore

## POST

Performs a system restore of the Pentaho system. This includes content, schedules, users, roles, datasources, and the metastore.

**Example Request:**  
POST pentaho/api/repo/files/systemRestore

### Parameters

|  |  |  |
| --- | --- | --- |
| name | description | type |
| fileUpload | The zip file generated using the backup endpoint, used to do a full system restore. | formdata |
| overwrite | If kept at the default of true, overwrites any value found on the system with the matching value that is being imported. Values that exist on the system, but do not exist in the import will not be deleted.  When the overwrite flag is equal to false, any value that is found in the import process that already exists will not be imported. | boolean |

### Response Body

|  |  |
| --- | --- |
| element: | (custom) |
| media types: | \*/\* application/xml application/octet-stream |

A jax-rs Response object with the appropriate status code, header, and body.

### Status Codes

|  |  |
| --- | --- |
| code | description |
| 200 | Successfully imported the Pentaho System |
| 403 | User does not have administrative permissions |
| 500 | Failure to complete the import. |

# /repo/files/delete

## PUT

Moves a list of files to the user's trash folder.

**Example Request:**  
PUT pentaho/api/repo/files/delete

***Note****: This REST call operates in the same manner as when a file is moved to the trash folder through the Pentaho User Console user interface. It is not intended for concurrent execution by the same user or session.*

### Parameters

|  |  |  |
| --- | --- | --- |
| name | Description | type |
| params | A comma-separated list of the files to be moved to the trash folder. | string |

### Response Body

|  |  |
| --- | --- |
| element: | (custom) |
| media types: | \*/\* application/xml application/octet-stream |

A jax-rs Response object with the appropriate status code, header, and body.

### Status Codes

|  |  |
| --- | --- |
| code | Description |
| 200 | Successfully moved the files specified in the comma-separated list to the trash. |
| 500 | Failure to move the files specified in the comma-separated list to the trash. |

# /repo/files/deletepermanent

## PUT

Permanently deletes the selected list of files from the repository.

**Example Request:**  
PUT pentaho/api/repo/files/deletepermanent

***Note****: This REST call operates in the same manner as when a file is deleted through the Pentaho User Console user interface. It is not intended for concurrent execution by the same user or session.*

### Parameters

|  |  |  |
| --- | --- | --- |
| name | Description | type |
| params | A comma-separated list of the files to be deleted. | string |

### Response Body

|  |  |
| --- | --- |
| element: | (custom) |
| media types: | \*/\* application/xml application/octet-stream |

Server Response indicating the success of the operation.

### Status Codes

|  |  |
| --- | --- |
| code | description |
| 200 | Successfully deleted the files specified in the comma-separated list from the system. |
| 403 | Failure to delete the files specified in the comma-separated list from the system due to path not found. |

# /repo/files/{pathId }

## PUT

Creates a new file with the provided contents at a given path.

**Example Request:**  
PUT pentaho/api/repo/files/:jmeter-test:test\_file\_1.xml  
**PUT data:**

|  |
| --- |
| This PUT body does not contain data. |

### Parameters

|  |  |  |
| --- | --- | --- |
| name | description | type |
| pathId | The path from the root folder to the root node of the tree to return using colon characters in place of / or \ characters. To clarify /path/to/file, the encoded pathId would be :path:to:file. | path |

### Request Body

|  |  |
| --- | --- |
| element: | (custom) |
| media types: | \*/\* application/xml application/octet-stream |

An Input Stream with the contents of the file to be created.

### Response Body

|  |  |
| --- | --- |
| element: | (custom) |
| media types: | \*/\* application/xml application/octet-stream |

A jax-rs Response object with the appropriate status code, header, and body.

**Example Response:**

|  |  |
| --- | --- |
| 1 | This response does not contain data. |

### Status Codes

|  |  |
| --- | --- |
| code | description |
| 200 | Successfully created the file. |
| 403 | Failure to create the file due to permissions, file already exists, or invalid path id. |

## GET

Takes a pathId and returns a Response with the output stream based on the file located at the pathId.

**Example Request:**  
GET pentaho/api/repo/files/:jmeter-test:test\_file\_1.xml

### Parameters

|  |  |  |
| --- | --- | --- |
| name | description | type |
| pathId | Colon separated path for the repository file. | path |

### Response Body

|  |  |
| --- | --- |
| element: | (custom) |
| media types: | \*/\* application/xml application/octet-stream |

A jax-rs Response object with the appropriate status code, header, and body.

**Example Response:**

|  |  |
| --- | --- |
| 1 | This response does not contain data. |

### Status Codes

|  |  |
| --- | --- |
| code | description |
| 200 | Successfully get the file or directory. |
| 404 | Failed to find the file or resource. |
| 500 | Failed to open content. |

:

:

: