

# Tutorial: Managing Current User Preferences in Documentum REST Services

Version 12

created by John Chen on Nov 9, 2016 8:57 PM, last modified by John Chen on Dec 8, 2016 11:31 PM

rest-version 7.3+

## Overview

Preferences are configurations for each login user to customize their display or operation defaults in client applications. Documentum REST Services 7.3 adds the current user preference(s) resources for REST client applications to store preference configurations into Documentum repositories. Each login user can have his own user preferences in a repository. We need to explain some terms used in this article firstly.

### Client code

Client code is one of the attribute of the preference instance. It is a name to identify the client application by which any single preference instance is used. A REST client must provide the client code when trying to create a preference in Documentum REST Services.

### Preference content

The content of preference could be saved in the preference instance. It should be parsed into the plain text and safely escaped for being kept in the preference instance. Documentum REST Services simply stores the content of the preference without doing any content processing.

## Get Current User Preferences

Assumed that we get a repository resource.

```
GET http://rest-demo/dctm-rest/repositories/REPO
Accept: application/vnd.emc.documentum+json
Authorization: Basic ZG1hZG1pbjpwYXNzd29yZA==
```

From the response, we'll find link relation "<http://identifiers.emc.com/linkrel/current-user-preferences>" which points to the current login user's preferences collection.

```
01. {
02.     .....
03.     "links": [
04.         .....
05.         {
06.             "rel": "http://identifiers.emc.com/linkrel/current-user-preferences",
07.             "href": "http://rest-demo/dctm-rest/repositories/REPO/currentuser-preferences"
08.         },
09.         .....
10.     ]
11. }
12.
```

We can get preferences by making a GET method on the href of this link relation.

```
GET http://rest-demo/dctm-rest/repositories/REPO/currentuser-preferences
Accept: application/vnd.emc.documentum+json
Authorization: Basic ZG1hZG1pbjpwYXNzd29yZA==
```

The preferences that belongs to the current login user is returned in the response.

```
01. {
02.     "id": "http://rest-demo/dctm-rest/repositories/REPO/currentuser-preferences",
03.     "title": "Preferences",
04.     "author": [
05.         {
06.             "name": "EMC Documentum"
07.         }
08.     ]
09. }
```

```

08. ],
09.   "updated": "....",
10.   "page": 1,
11.   "items-per-page": 100,
12.   "links": [
13.     {
14.       "rel": "self",
15.       "href": "http://rest-demo/dctm-rest/repositories/REPO/currentuser-preferences"
16.     }
17.   ],
18.   "entries": [
19.     ....preferences list here...
20.   ]
21. }

```

## Get a Preference

In the current user preferences resource, atom entries point to specific preference resources. On any single entry, there is a link pointing to the preference resource.

```

01. {
02.   ....
03.   "entries": [
04.     ....
05.     {
06.       ....
07.       "content": {
08.         "type": "application/vnd.emc.documentum+json"
09.         "src": "http://rest-demo/dctm-rest/repositories/REPO/currentuser-preferences/client-a"
10.       },
11.       ....
12.     },
13.     ....
14.   ]
15. }

```

Please get the preference instance by making a GET method on this link.

```

GET http://rest-demo/dctm-rest/repositories/REPO/currentuser-preferences/client-a
Accept: application/vnd.emc.documentum+json
Authorization: Basic ZG1hZG1pbjpwYXNzd29yZA==

```

The preference instance is returned in the response body.

```

01. {
02.   "name": "preference",
03.   "client": "client-a",
04.   "owner-name": "demo-user",
05.   "title": "....",
06.   "subject": ".....",
07.   "creation-date": ".....",
08.   "modify-date": ".....",
09.   "preference-content": "{\"pageCount\":10,\"lazyFetch\":true,\"theme\":\"blue\"}",
10.   "links": [
11.     {
12.       "rel": "self",
13.       "href": "http://rest-demo/dctm-rest/repositories/REPO/currentuser-preferences/client-a"
14.     },
15.     {
16.       "rel": "edit",
17.       "href": "http://rest-demo/dctm-rest/repositories/REPO/currentuser-preferences/client-a"
18.     },
19.     {
20.       "rel": "http://identifiers.emc.com/linkrel/delete",

```

```

21.     "href": "http://rest-demo/dctm-rest/repositories/REPO/currentuser-preferences/client-a"
22.   }
23. ]
24. }

```

## Create a Preference

We can create a new preference by making a POST method on current user preferences resource. The request body should be the new preference, which contains the properties to define the preference. Below is the a table of these available properties.

| Property           | Mandatory | Description                               |
|--------------------|-----------|---|
| client             | yes       | client code                               |
| preference-content | no        | the preference content. plain text.       |
| title              | no        | the title of this preference instance.    |
| subject            | no        | the subject of this preference instance.  |
| keywords           | no        | the keywords of this preference instance. |

Here is the snippet of the POST method to create a new preference.

```

POST http://rest-demo/dctm-rest/repositories/REPO/currentuser-preferences
Accept: application/vnd.emc.documentum+json
Content-Type: application/vnd.emc.documentum+json
Authorization: Basic ZG1hZG1pbjpwYXNzd29yZA==

{"properties":{"client":"client-a","preference-content":{"pagecount":10, "theme":"colorred"}}}

```

The response returns the link of the created user in the response header "Location". The response body contains the new preference representation.

```

HTTP 201 Created
Location:http://rest-demo/dctm-rest/repositories/REPO/currentuser-preferences/client-a

```

## Update a Preference

A preference representation uses the link relation *edit* to indicate whether the accessor has the permission to update the preference. If it appears, It means that the accessor has the permission to update the preference. We can update a preference by making POST method with the changing properties in the request body.

```

01. {
02.     ....
03.     "links":[
04.         .....
05.         {
06.             "rel":"edit",
07.             "href":"http://rest-demo/dctm-rest/repositories/REPO/currentuser-preferences/client-a"
08.         },
09.         .....
10.     ]
11. }

```

Below is the snippet to update a preference resource. Please note that properties "owner-name" and "client" are not allowed to update.

```

POST http://rest-demo/dctm-rest/repositories/REPO/currentuser-preferences/client-a
Accept: application/vnd.emc.documentum+json
Content-Type: application/vnd.emc.documentum+json
Authorization: Basic ZG1hZG1pbjpwYXNzd29yZA==

{"properties":{"preference-content":{"pagecount":20, "theme":"colorblue"}}}

```

The response returns 200 if the update is successful.

## Delete a Preference

A preference representation uses the link relation <http://identifiers.emc.com/linkrel/delete> to indicate whether the accessor has the permission to delete the preference. If it appears, It means that the accessor has the permission to delete the preference. We can delete a preference by making a DELETE method the resource of this preference.

```
01. {  
02.     ....  
03.     "links": [  
04.         .....  
05.         {  
06.             "rel": "http://identifiers.emc.com/linkrel/delete",  
07.             "href": "http://rest-demo/dctm-rest/repositories/REPO/currentuser-preferences/client-a"  
08.         },  
09.         .....  
10.     ]  
11. }
```

Please DELETE to the link to remove this preference instance. The response returns 204 if the deletion is successful.

```
DELETE http://rest-demo/dctm-rest/repositories/REPO/currentuser-preferences/client-a  
Accept: application/vnd.emc.documentum+json  
Content-Type: application/vnd.emc.documentum+json  
Authorization: Basic ZG1hZG1pbjpwYXNzd29yZA==
```

[Learn more about Documentum REST Services >>](#)