PeopleSpheres

Date
07/20

API DOCUMENTATION GUIDE

Table of Contents

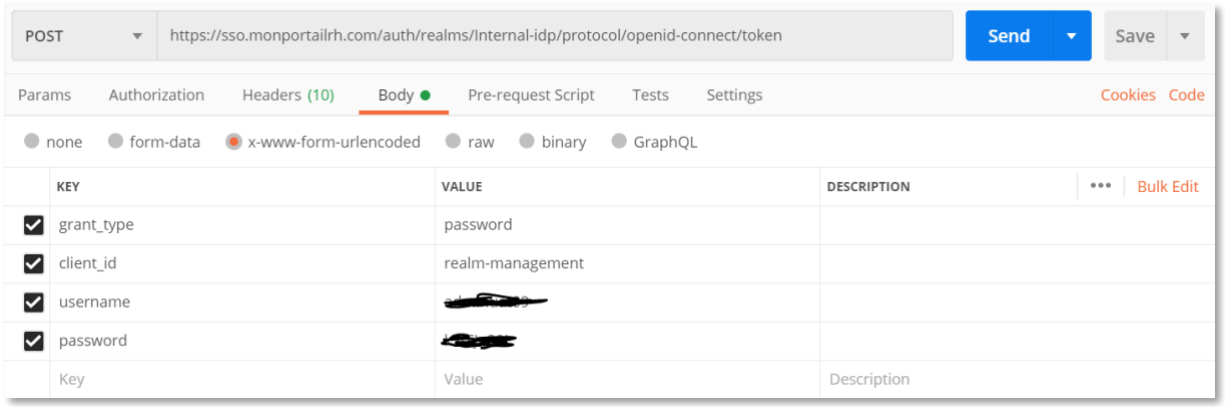
1 Introduction

Old documentation: **https://rest.monportailrh.com/swagger/**
New documentation: **https://rest.monportailrh.com/docs/**

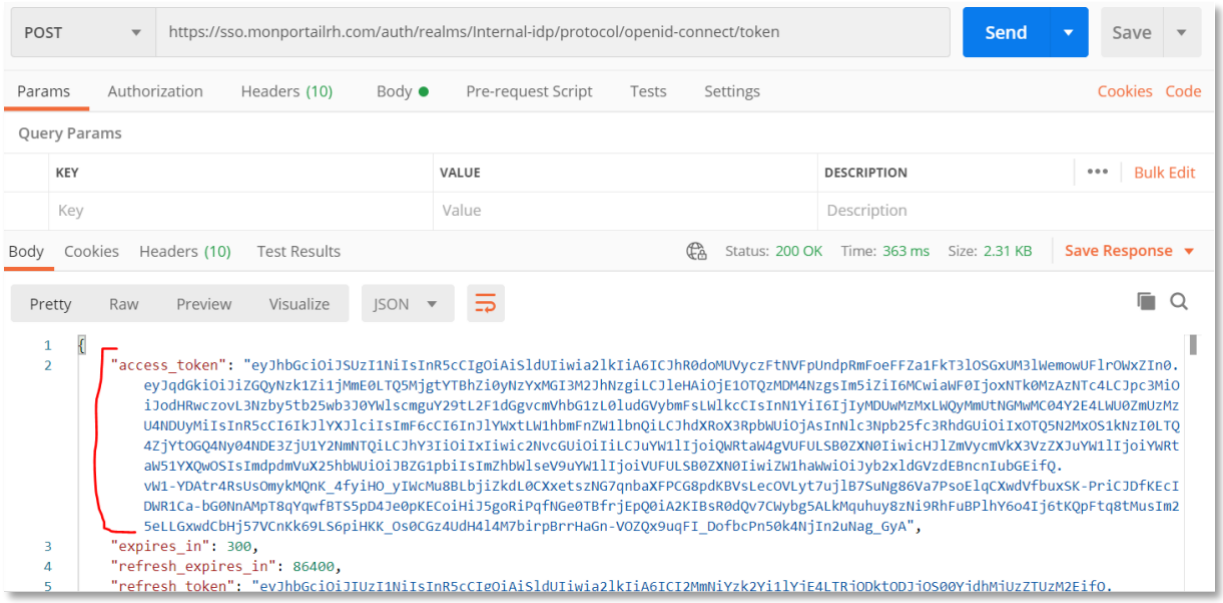
2 Authentication

To get a Bearer token, you should use the API request below:

POST https://sso.monportailrh.com/auth/realms/Internal-idp/protocol/openid-connect/token
Header: Content-Type = application/x-www-form-urlencoded
Body:



API response:



Token expires within 5 minutes, during this time you can:

1. Login again
2. Or, Refresh the token by sending following request:

- POST https://sso.monportailrh.com/auth/realms/Internal-idp/protocol/openid-connect/token
- Header: "Content-Type" = application/x-www-form-urlencoded
- Body:
  - grant_type: refresh_token
  - client_id: realm-management
  - refresh_token: {the expired token}

3 http Header for the Rest API

All the API requests to https://rest.monportailrh.com.

URL should contain following headers:

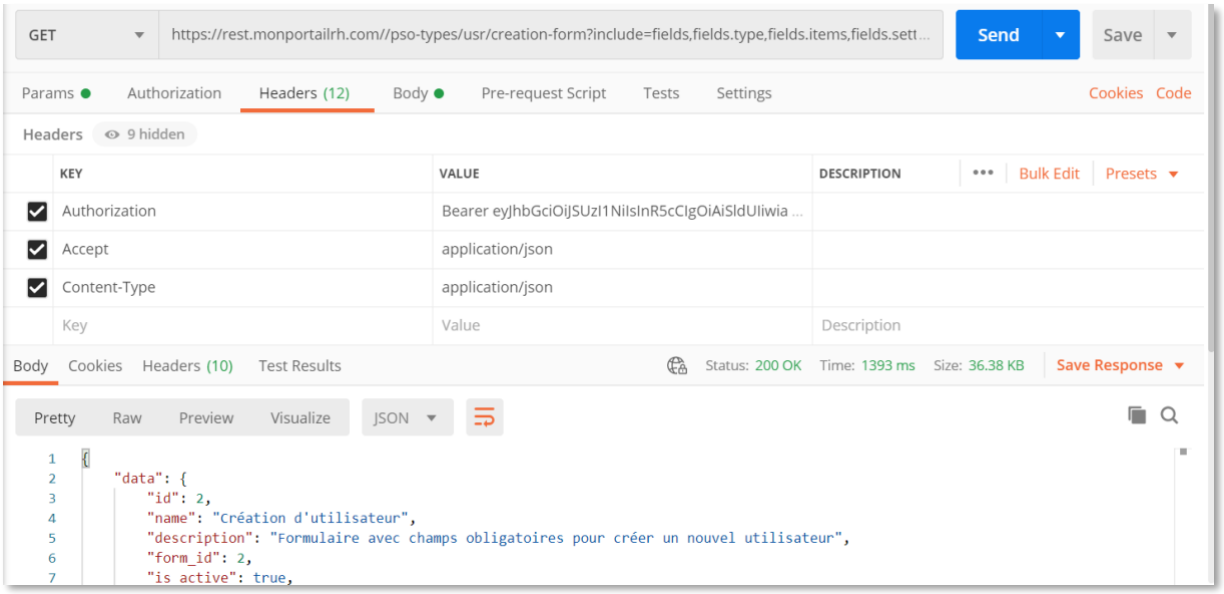
☑	Authorization	Bearer eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia ...
☑	Accept	application/json
☑	Content-Type	application/json

4 Create a user

The user creation has to be done on two steps:

4.1 Step 1: Get the User Creation Form

GET https://rest.monportailrh.com//pso-types/usr/creation-form?include=fields,fields.type,fields.items,fields.settings,fields.options,fields.assignment_settings,fields.items.type,fields.items.option



4.2 Step 2: Create a user

POST https://rest.monportailrh.com/pso-types/usr
Body:

Body in Json format:

{"data":[{"field_id":272,"value":["228"]},{"field_id":550,"value":["321"]},{"field_id":123,"value":["Riad"]},{"field_id":125,"value":["Lasfer"]},{"field_id":127,"value":["RiadLasferTestJuly9"]},{"field_id":28,"value":["RiadLasferTestJuly9@gmail.com"]},{"field_id":25,"value":["00000"]},{"field_id":27,"value":["1111"]},{"field_id":182,"value":["6"]},{"field_id":344,"value":["268"]},{"field_id":185,"value":["284"]},{"field_id":711,"value":["2020-07-10"]},{"field_id":186,"value":["285"]},{"field_id":199,"value":["74"]},{"field_id":540,"value":["2020-07-16"]},{"field_id":542,"value":["2020-07-16"]},{"field_id":544,"value":["111.1"]},{"field_id":546,"value":["2020-07-10"]},{"field_id":548,"value":["2020-07-10"]},{"field_id":716,"value":["2020-07-16"]},{"field_id":718,"value":["417"]},{"field_id":720,"value":["458"]},{"field_id":722,"value":["400"]},{"field_id":724,"value":["111"]},{"field_id":726,"value":["111"]},{"field_id":728,"value":["aaaa"]},{"field_id":730,"value":["aaaaa"]},{"field_id":732,"value":["2020-07-31"]},{"field_id":734,"value":["aaaa"]},{"field_id":708,"value":["383"]}]}

5 Get Users

Get users per pagination, the request below is to get the first 25 active users

GET https://rest.monportailrh.com/search?name=&pso-type=usr&page=1&active=1&per-page=25

- pso-type=usr
- page=1
- active=1
- per-page=25

This API helps you to retrieve the list of users with the main fields :

- ID
- Email
- First name
- Last name
- …

Once you retrieve the list of users, you can use the id to get the whole profile of a specific user, using the API below.

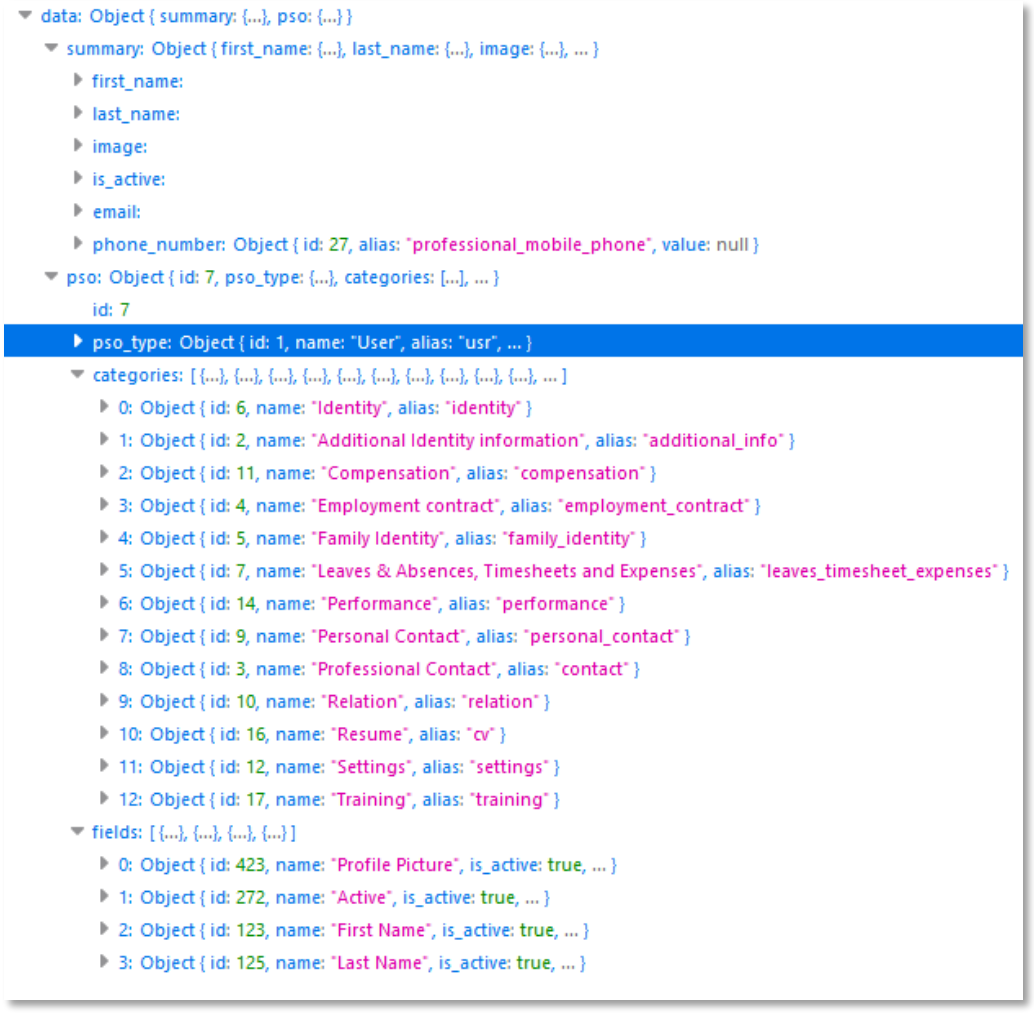

data: [{...}]
0: Object { meta: {...}, pso_type: {...}, data: [...] }
meta: Object { pagination: {...} }
pagination: Object { total: 22, count: 22, per_page: 25, ... }
total: 22
count: 22
per_page: 25
current_page: 1
total_pages: 1
links: []
pso_type: Object { id: 1, name: "User", alias: "usr", ... }
data: [{...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, ...]
0: Object { id: 6, status: "default", role: [...], ... }
1: Object { id: 7, status: "default", role: [...], ... }
2: Object { id: 8, status: "default", role: [...], ... }
3: Object { id: 9, status: "default", role: [...], ... }
4: Object { id: 10, status: "default", role: [...], ... }
5: Object { id: 11, status: "default", role: [...], ... }
6: Object { id: 12, status: "default", role: [...], ... }
7: Object { id: 13, status: "default", role: [...], ... }
8: Object { id: 20, status: "default", role: [...], ... }
9: Object { id: 23, status: "default", role: [...], ... }
10: Object { id: 25, status: "default", role: [...], ... }
11: Object { id: 27, status: "default", role: [...], ... }
12: Object { id: 38, status: "default", role: [...], ... }
13: Object { id: 39, status: "default", role: [...], ... }
14: Object { id: 41, status: "default", role: [...], ... }
15: Object { id: 47, status: "default", role: [...], ... }
16: Object { id: 51, status: "default", role: [...], ... }
17: Object { id: 56, status: "default", role: [...], ... }
18: Object { id: 57, status: "default", role: [...], ... }
19: Object { id: 58, status: "default", role: [...], ... }
20: Object { id: 59, status: "default", role: [...], ... }
21: Object { id: 60, status: "default", role: [...], ... }

6 Get a specific user

You have to use the ID, to get the profile of a specific user.

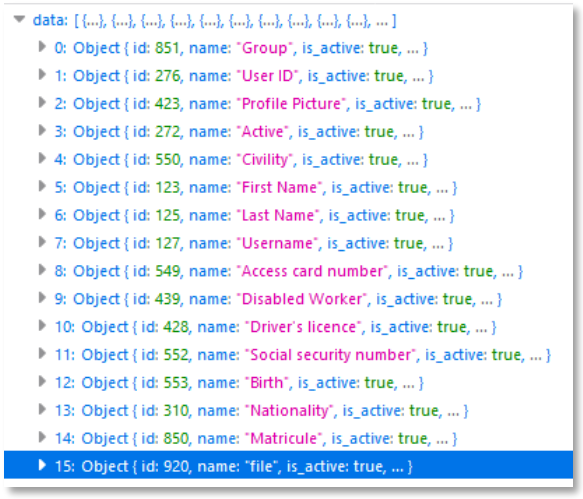
6.1 Get a profile summary and the fields categories

GET
https://rest.monportailrh.com/psos/**7**?include=pso.pso_type.profile_form_instance_id,pso.categories,pso.fields



6.2 Get the user profile per category

GET
https://rest.monportailrh.com/psos/**7**/fields?category=identity&active=true&include=type,items,options,settings,assignment_settings



6.3 Get the whole user profile

GET https://rest.monportailrh.com/psos/**7**/fields?active=true&include=type,items,options,settings,assignment_settings

7 Update a user

You have to use the id, to edit the profile of a specific user

7.1 Edit a Dropdown:

POST https://rest.monportailrh.com/psos/7/form-instances/1
Body in Json format: {"data":[{"field_id":550,"value":["322"]}]}

7.2 Edit a Composite field

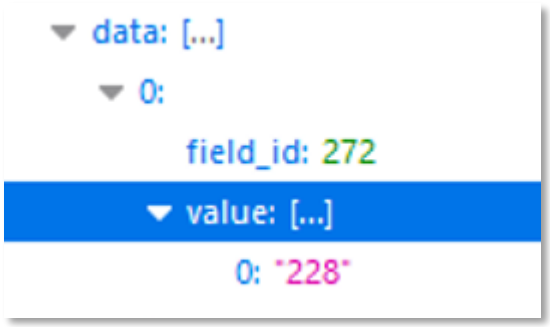
POST https://rest.monportailrh.com/psos/7/form-instances/1

Body in Json format:
{"data":[{"field_id":441,"value":["1"]},{"field_id":443,"value":["293"]},{"field_id":445,"value":["2020-07-10"]},{"field_id":447,"value":["2020-07-17"]},{"field_id":786,"value":["FLATCHR LOGO.png"]},{"field_id":789,"value":["https://rest.monportailrh.com/storage/demo_uat09/files/4a/59/Szg1dMXEaOTYJi9ejled6L1taVzCikdXVXhDDd5w5f073d79e4470.png"]}]}

8 Data Types

8.1 Dropdown

To use the first option « Yes », you have to use the id of this option in your API request « 228 »



8.2 Hierarchy

The Hierarchy type is similar to the dropdown type, and each parent option can have more options as children.

To use an option as a value, you have to use the id of this option in your API request

8.3 Date

Date format: yyyy-MM-dd

8.4 File

The File type has the same structure as a Composite field.

It contains 2 items:

1. File name

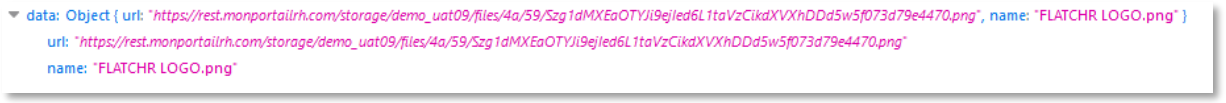
2. File Link

To attach a File to a user profile, you need to follow these 2 steps:

- Step 1: upload the File

POST https://rest.monportailrh.com/files/forms

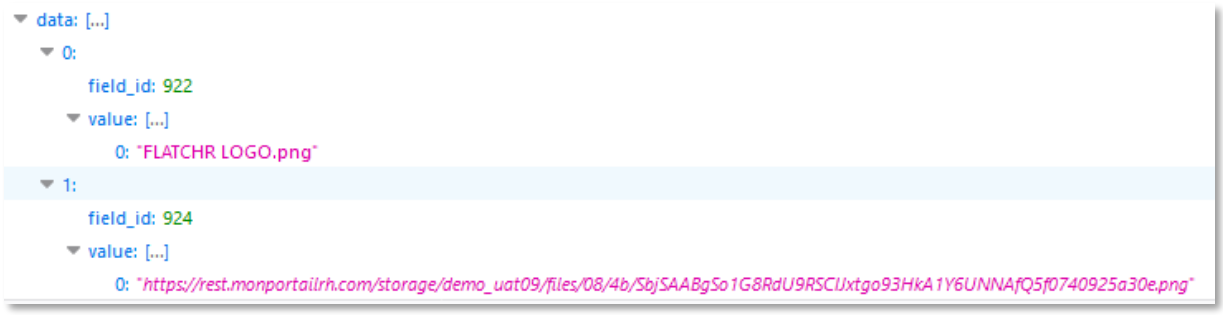
API response:



The API will return you the File name and the File Link that should be used as a field value.

- Step 2: Edit the File Field

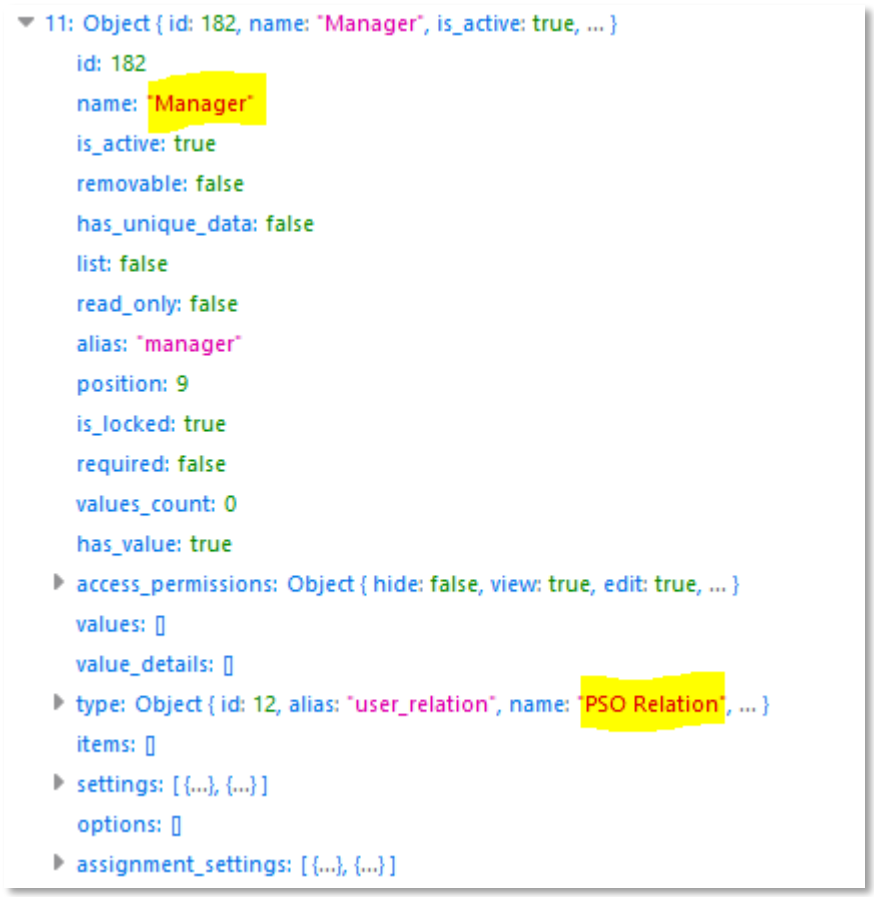
POST https://rest.monportailrh.com/psos/7/form-instances/1



Body in Json format: {"data":[{"field_id":922,"value":["FLATCHR LOGO.png"]},{"field_id":924,"value":["https://rest.monportailrh.com/storage/demo_uat09/files/08/4b/SbjSAABgSo1G8RdU9RSCIJxtgo93HkA1Y6UNNAfQ5f0740925a30e.png"]}]}

8.5 Relation

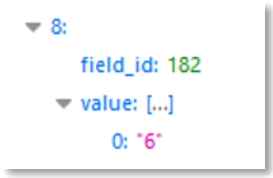
For the Relation type, like « Manager » can be used to link a user to his manager.



To link a user A to his manager (user B):

1. You can do it during the user creation of the user A, only if the user B already exists, else you need to create the user B first;

2. If the user B already exists, you can use his user_ID as a value of this field as seen below (id=6):

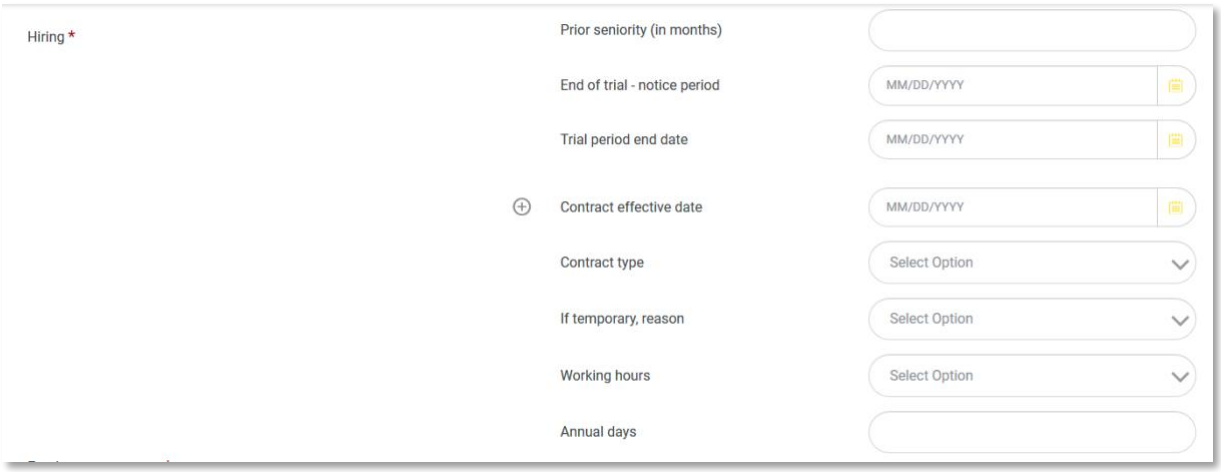


8.6 Numeric

It can be an integer, or a Double with 14 digits for the precision.

8.7 Composite

A Composite field is a set of subfields, called items
Below is how it looks on the web application:



And here is the API response:

8.8 ListOf

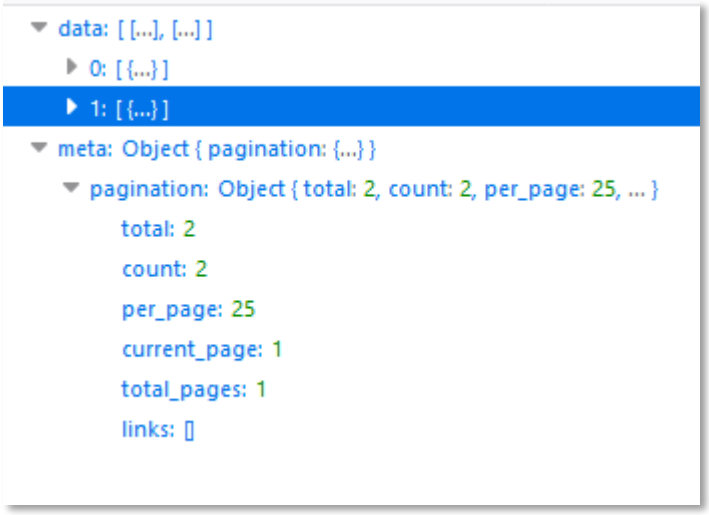
A ListOf field can be any field type (text, numeric, dropdown, etc.), we can append new values to the list, we can also edit/delete the existing values.

8.8.1 Append a new value

POST https://rest.monportailrh.com/psos/7/form-instances/1/fields/934/values
Body in Json format: {"data":[{"field_id":934,"value":["2841"]}]}

8.8.2 Get the list of values

GET https://rest.monportailrh.com/psos/7/form-instances/1/fields/934/values?page=1



8.8.3 Edit an existing value from the list

PATCH https://rest.monportailrh.com/psos/7/form-instances/1/fields/934/values
Body in Json format: {"data":[{"value_id":144154,"value":["2843"]}]}

9 Field Settings

- id: the ID is used as an identifier to edit the field through API
- is_active: it should be « true » to be able to edit it
- has_unique_data: if « true », the data should be unique across all users for this specific field
- list: if « true », this field can be used as a list of values, we can append a new value or edit/delete the existing values
- required: if « true », the field value is required in the API request
- values_count: counter of values for the listOf field

10 Recommendations and notes

- The API requests Body and API responses are dynamic, they depend on the customer configurations (Fields, Forms and Roles configurations);
- It's highly recommended to use an admin account, to have access to all Data, Forms and Fields;
- The options of dropdown/hierarchy fields are translatable, for the field « Active »
    - If the user language of the user who is used by the API is EN: the API will return Yes/No
    - If the user language is FR: the API will return Oui/Non

  So, you need to always keep the same user language for the user used by the API.
- If a Composite field is marked as required, all items should be included in request;
- « Username » and « Professional email address » should be unique;
- To retrieve all users and all listOf fields values, it's highly recommended to use the default export named « Rapport_Utilisateurs_customerName.zip » which is available on your sFTP following this path /customerName /incoming/reports.