Critical manufacturing
an ASM PT company

OData Access to Data Sets

11.3

April 2026

DOCUMENT ACCESS

Public

OData Access to Data Sets

Estimated time to read: 5 minutes

OData (Open Data Protocol) is an open standard protocol that enables simple and standardized access to data. **Data Platform** allows accessing the data in all data sets using this protocol.

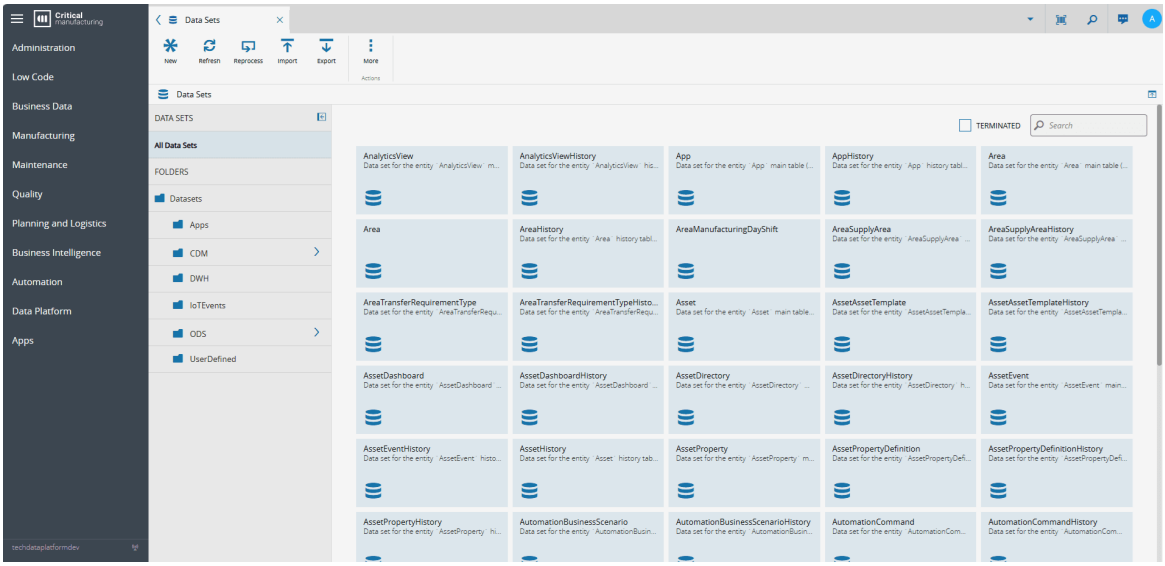
Overview

In this tutorial we describe how **OData** clients can easily access data in CM MES data sets stored in **Data Platform**.

Endpoints

Through **Data Platform**, all data sets can be accessed using the **OData** V4 protocol.

If you navigate to the **Data Sets** page in the **Data Platform** section, you can see that the data sets are organized in 6 root folders:



Each folder contains different types of data sets:

- **Apps** - folder for data sets created and used by MES Apps.
- **CDM** - folder for system data sets containing the **Canonical Data Model** (CDM) data.
- **DWH** - folder for data warehouse data sets containing aggregated data.
- **IotEvents** - folder for data sets associated with iot events.
- **ODS** - folder for data sets containing MES ODS data.
- **UserDefined** - folder for data sets manually created by CM MES users.

Each of these folders can be accessed independently via **OData**. For example, to list all available data sets in the **CDM** root folder, the OData client should use one of these endpoints:

- Service Document - `https://<hostname>/datamanager/odata/CDM`
- Metadata - `https://<hostname>/datamanager/odata/CDM/$metadata`

To access the data of data set `MaterialMovement` in the **CDM** root folder:

`https://<hostname>/datamanager/odata/CDM/MaterialMovement`

To access the data of data set ResourceLayout in the Resource folder (which is inside the CDM root folder):

`https://<hostname>/datamanager/odata/CDM/Resource.ResourceLayout`

Thus, to access data sets that are not in root folders, the path to the data set should be built using the dot separator.

Query Options

The standard OData query options are supported, plus a few extensions:

Query Option	Description	Example
$select	Pick the columns to get.	`https://<hostname>/datamanager/odata/CDM/MaterialMovement?$select=Enterprise_Name,Facility_Name,Area_Name,Step_Name,Material_Name,InProcessPrimaryQty`
$filter	Filter the data to get.	`https://<hostname>/datamanager/odata/CDM/MaterialMovement?$filter=contains(Material_Name,'Cookie') and Step_Name eq 'Mixing'`
$orderby	Columns to order the data.	`https://<hostname>/datamanager/odata/CDM/MaterialMovement?$orderby=Material_Name asc,Step_Name desc`
$top	Number of rows to get. If `$top` is not set, Data Manager will return up to the default maximum number of rows defined by the `DATAMANAGER_DEFAULT_MAX_ROWS` environment variable (set to 1000 by default).	`https://<hostname>/datamanager/odata/CDM/MaterialMovement?$top=50`
$skip	Number of rows to skip before returning the result.	`https://<hostname>/datamanager/odata/CDM/MaterialMovement?$skip=50`
$count	Add the total number of rows in the unfiltered data set to the reply.	`https://<hostname>/datamanager/odata/CDM/MaterialMovement?$count=true`

Query Option	Description	Example
$apply	Group and/or aggregate data.	`https://<hostname>/datamanager/odata/CDM/MaterialMovement?$select=Area_Name,Material_Name,TotalTrackIns,TotalTrackOuts&$apply=groupby((Area_Name,Material_Name),aggregate(TrackInCount with sum as TotalTrackIns,TrackOutCount with sum as TotalTrackOuts))`
$query	If the query needs to be sent via POST instead of a GET request (for example, the query is too long to be sent with a GET request), the `$query` option should be used, and the actual query sent as plain text in the body of the POST request.	`https://<hostname>/datamanager/odata/CDM/MaterialMovement?$query`, POST request body in plain text: `$select=Step_Name,Material_Name&$filter=Material_Name eq 'Cookie01'`
parameters	If the dataset is of type `Query` and the user-defined query string requires database parameters, they can be sent using the parameters option.	`https://<hostname>/datamanager/odata/UserDefined/MyDataset?parameters=var1=123,var2='example'`

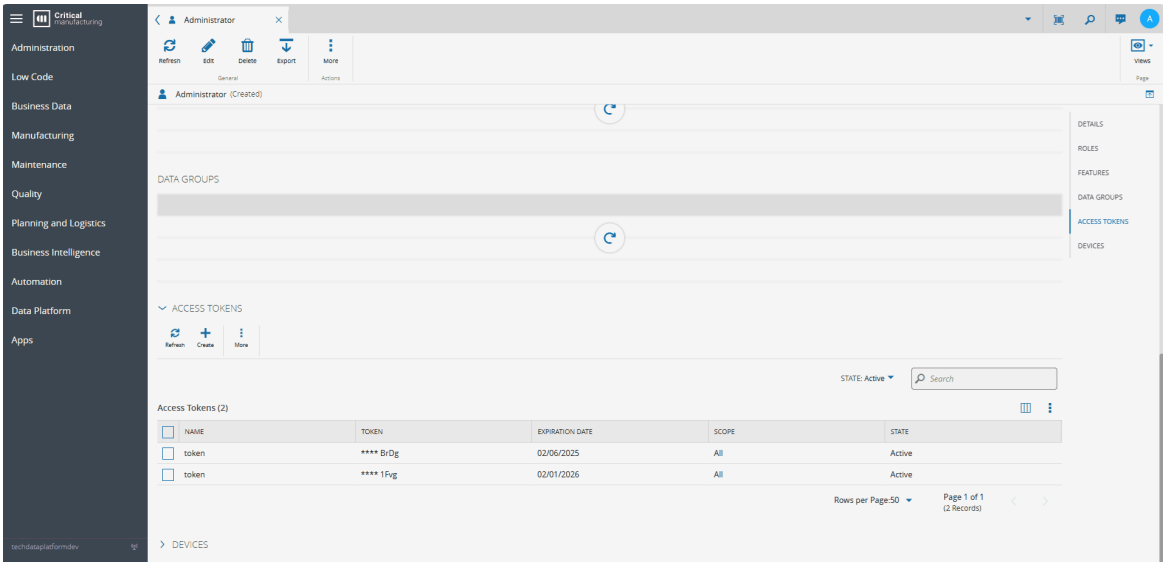
Note

All the options run server-side, so all OData queries should be as specific as possible to obtain the best performance.

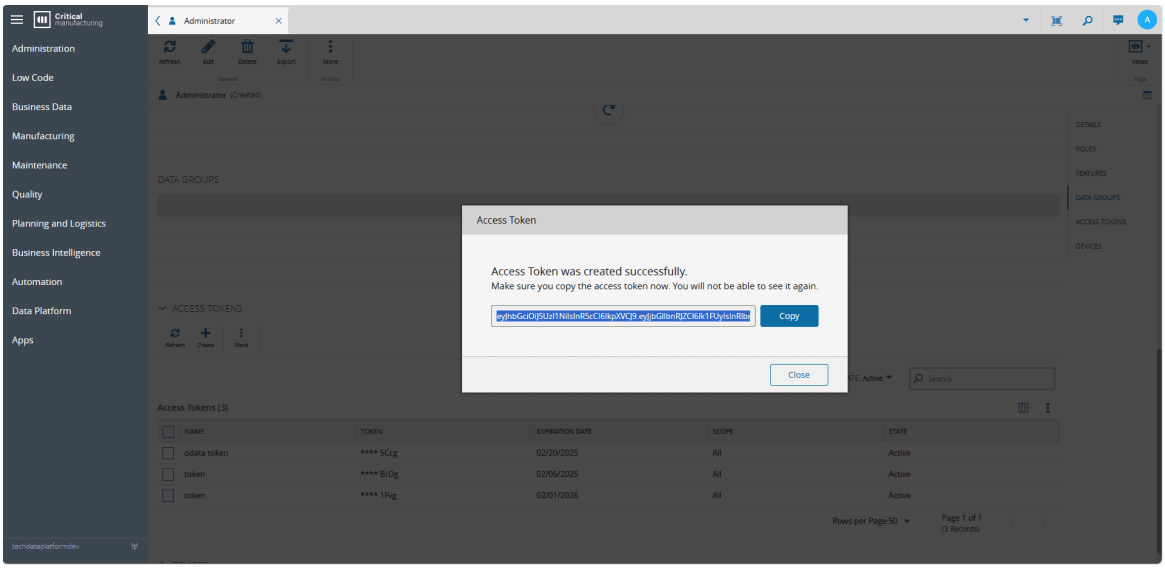
Accessing Data Sets With an OData Client

Any OData client can connect to **Data Platform**. However, the client will need an **Access Token** to retrieve the data.

To create an **Access Token** go to your user page, scroll down to the **Access Tokens** section and **Create** a new **Access Token**:

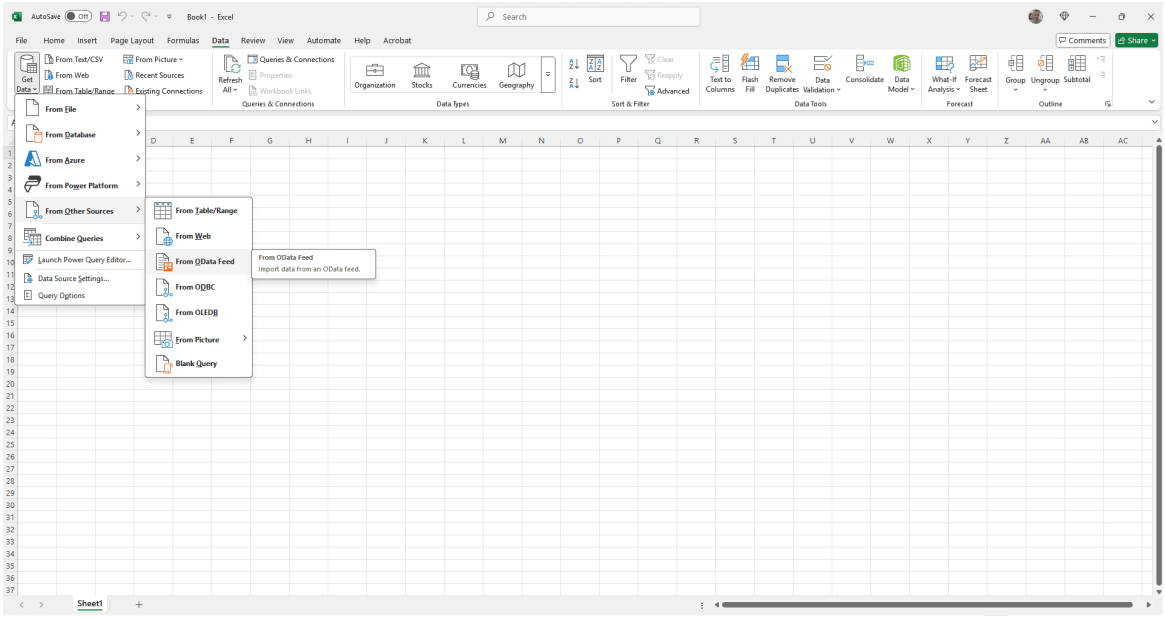


Save the token, so that it can be used in the OData Client to access the data:

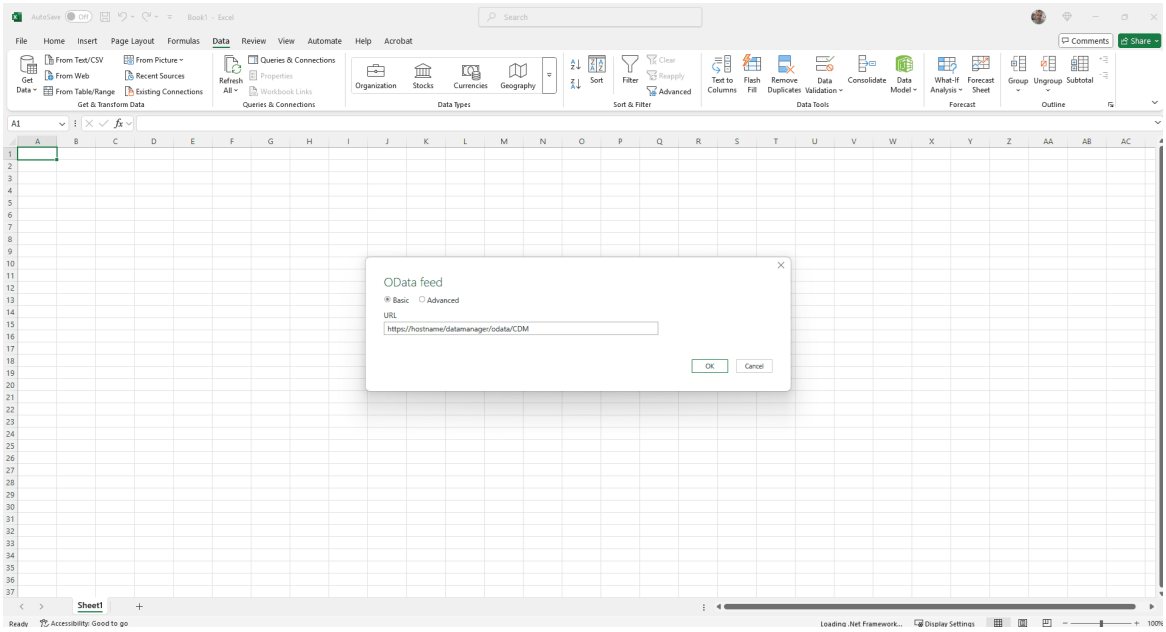


Data Sets in Excel

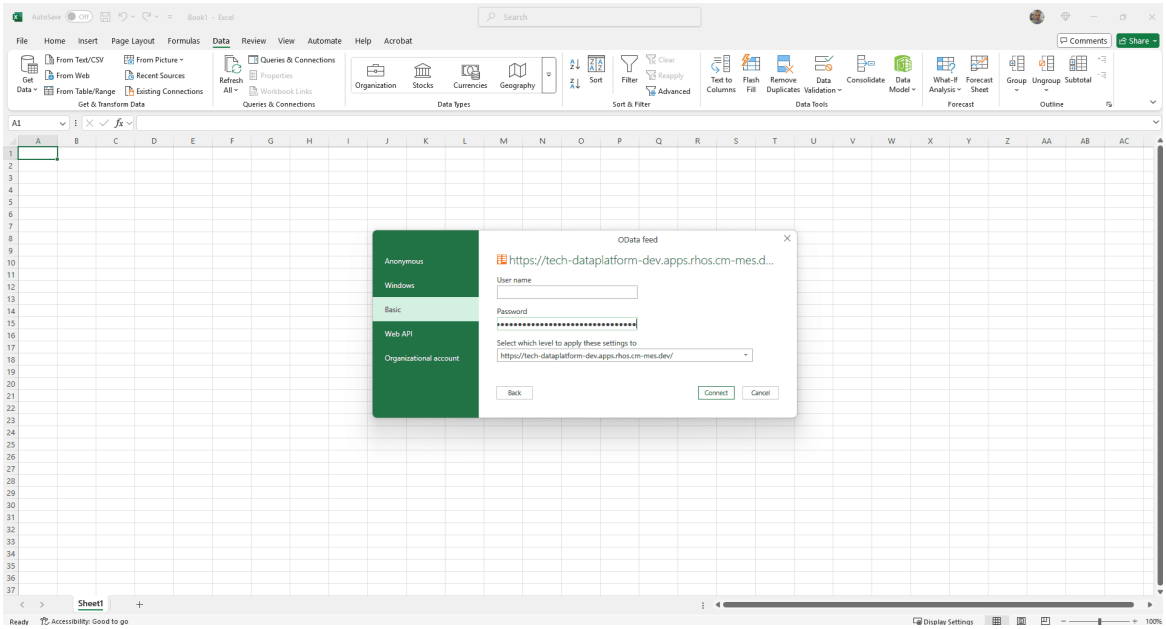
To retrieve data sets in **Excel** go to **Data > Get Data > From Other Sources > From OData Feed**:



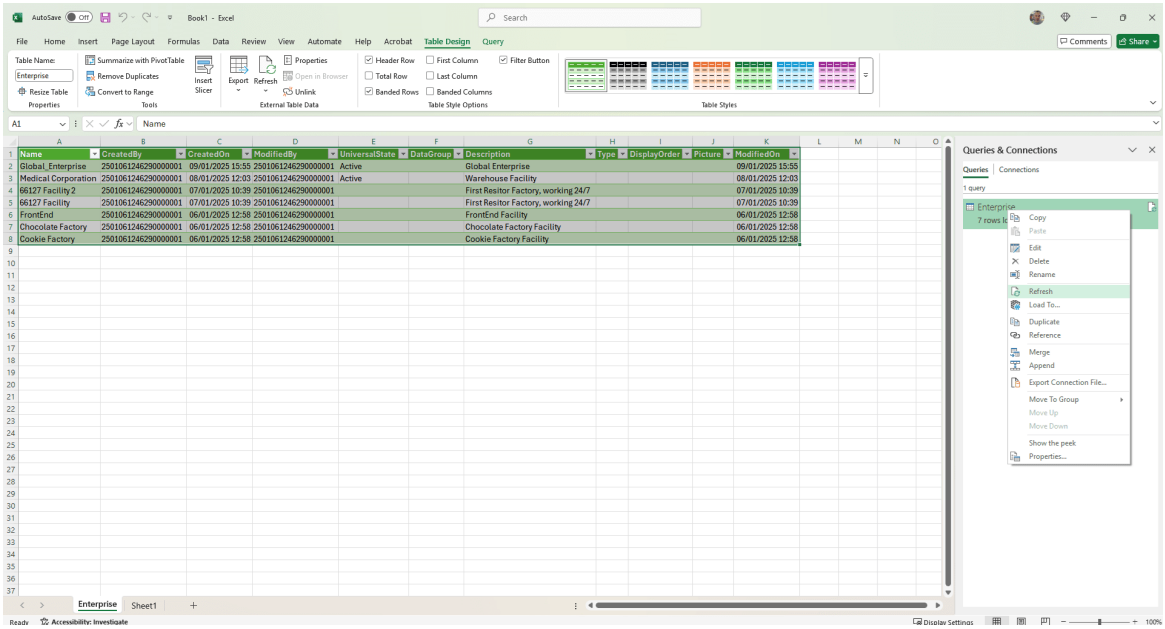
Type the URL for the root folder you want to access, or an OData query to access a specific data set with query options, and select **OK**:



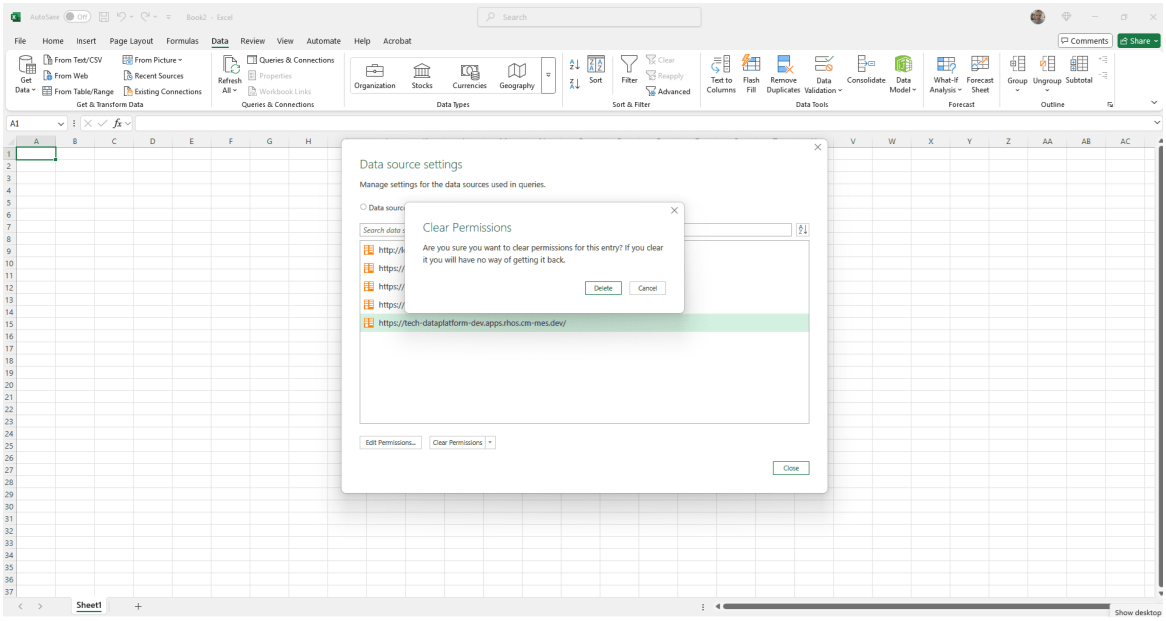
An authentication window will pop-up. Select **Basic**, paste your **Access Token** in the **Password** field and select **Connect** (you can leave the **User name** field empty):



You should now have access to the data in the data set, or to the list of data sets if you gave it an URL to a root folder (in which case you can pick the data set from the list). The data is now available to be processed and refreshed in **Excel**:



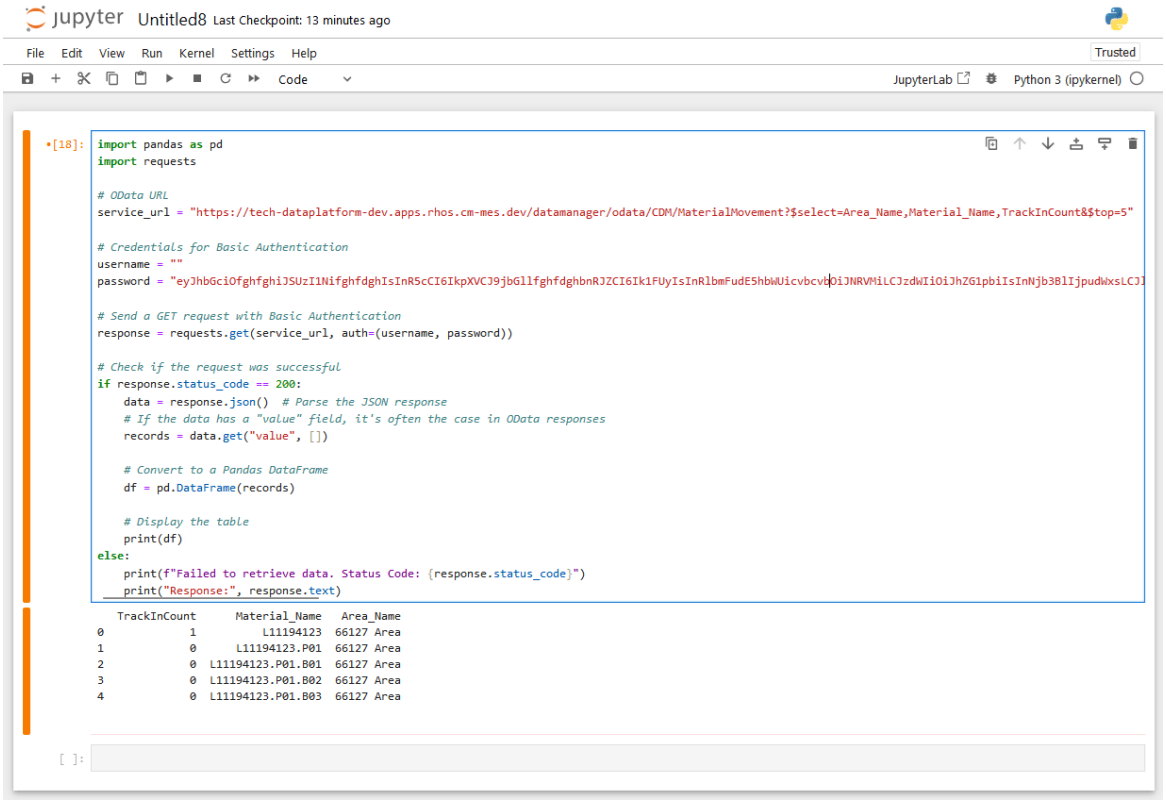
Excel caches the access tokens. Thus, if you are having trouble accessing the data (for example, your token expired) you should go to **Get Data > Data Source Settings** and select **Clear Permissions**, select the permissions you want to clear and **Delete** them:



Next time you try to connect to the data set, **Excel** will ask for the token again.

Data Sets in Jupyter Notebook

There are several ways to access **OData** data sets in **Jupyter Notebook**. For example, with a regular HTTP request with an OData query:



Just make sure to use **Basic Authentication** in the request, and setting your **Access Token** as the password.

Legal Information

Disclaimer

The information contained in this document represents the current view of Critical Manufacturing on the issues discussed as of the date of publication. Because Critical Manufacturing must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Critical Manufacturing, and Critical Manufacturing cannot guarantee the accuracy of any information presented after the date of publication. This document is for informational purposes only.

Critical Manufacturing makes no warranties, express, implied or statutory, as to the information herein contained.

Confidentiality Notice

All materials and information included herein are being provided by Critical Manufacturing to its Customer solely for Customer internal use for its business purposes. Critical Manufacturing retains all rights, titles, interests in and copyrights to the materials and information herein. The materials and information contained herein constitute confidential information of Critical Manufacturing and the Customer must not disclose or transfer by any means any of these materials or information, whether total or partial, to any third party without the prior explicit consent by Critical Manufacturing.

Copyright Information

All title and copyrights in and to the Software (including but not limited to any source code, binaries, designs, specifications, models, documents, layouts, images, photographs, animations, video, audio, music, text incorporated into the Software), the accompanying printed materials, and any copies of the Software, and any trademarks or service marks of Critical Manufacturing are owned by Critical Manufacturing unless explicitly stated otherwise. All title and intellectual property rights in and to the content that may be accessed through use of the Software is the property of the respective content owner and is protected by applicable copyright or other intellectual property laws and treaties.

Trademark Information

Critical Manufacturing is a registered trademark of Critical Manufacturing.

All other trademarks are property of their respective owners.