



Hello and welcome to TC25!

This is your lab environment for Build, Automate, and Scale: A Tableau REST API Deep Dive

First we'll need to do a bit of setup.

To log into your VM, you will need to use the following credentials:

Username: **LabUser**

Password: **Pa\$\$w0rd**

The username and password applies only to the VM. In this HOT Session, we will be using Tableau Cloud. When signing into Tableau Cloud you will utilize the following credentials:


Username: @lab.CloudCredential(TC25HOT1942).Username

Password: @lab.CloudCredential(TC25HOT1942).Password

Let us know if you need assistance with developing your lab. The Room Assistants are here to help!


Exercise #1a Environment Setup & Authentication

1. Sign into the VM with the credentials provided if not already.
2. Today we will be using Postman, an API platform for building, testing and using APIs. You will need a free account setup with Postman. If you already have an account, skip to Step 6.
3. Go ahead and open up the Postman application by double clicking the Postman icon (insert logo) on the desktop.
4. Click the Create Account button in the upper right. This will open up a browser window where you can create a quick account as well as sign in with Google if preferred.



Why sign up?

- Organize all your API development within Postman Workspaces
- Sync your Postman data across devices
- Back up your data to the Postman cloud
- It's free!



Create Postman account

Work email


Username

Password Show

☐ Receive product updates, news, and other marketing communications
☒ Stay signed in

Create Free Account

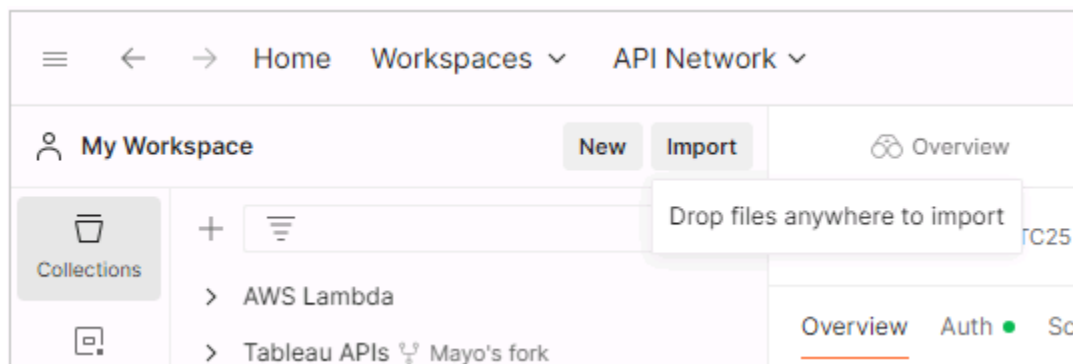
or

 Sign Up with Google

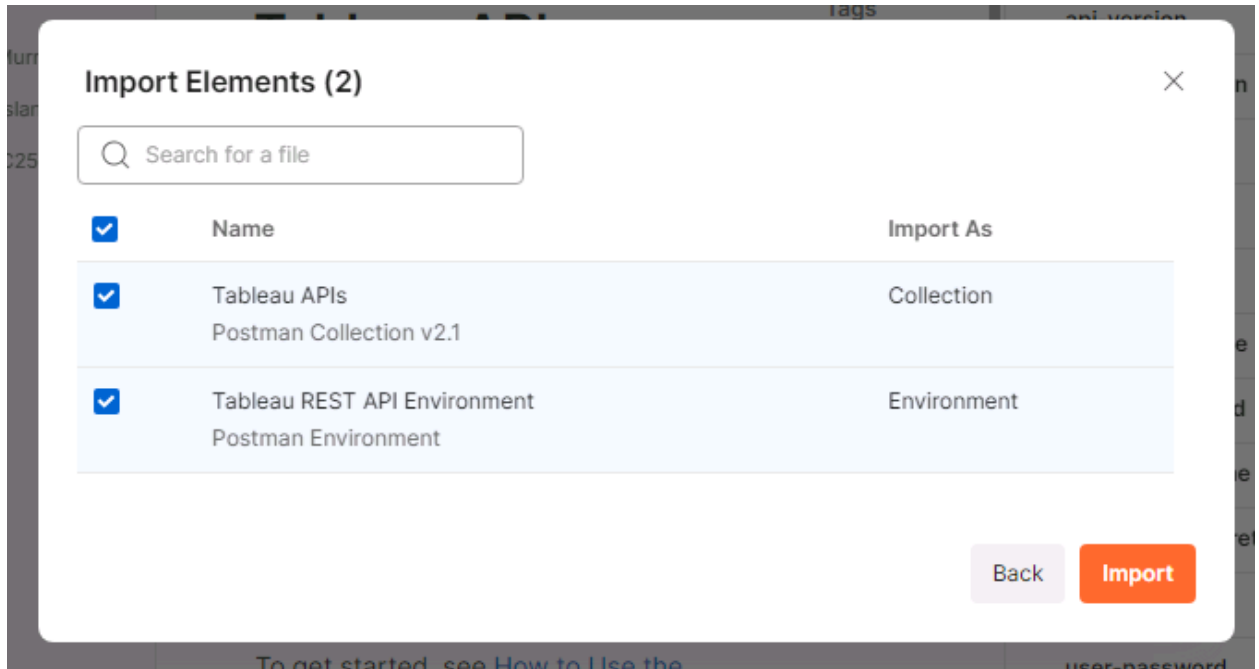
Sign In with SSO

By creating an account, you agree to our [terms](#) and [privacy policy](#).

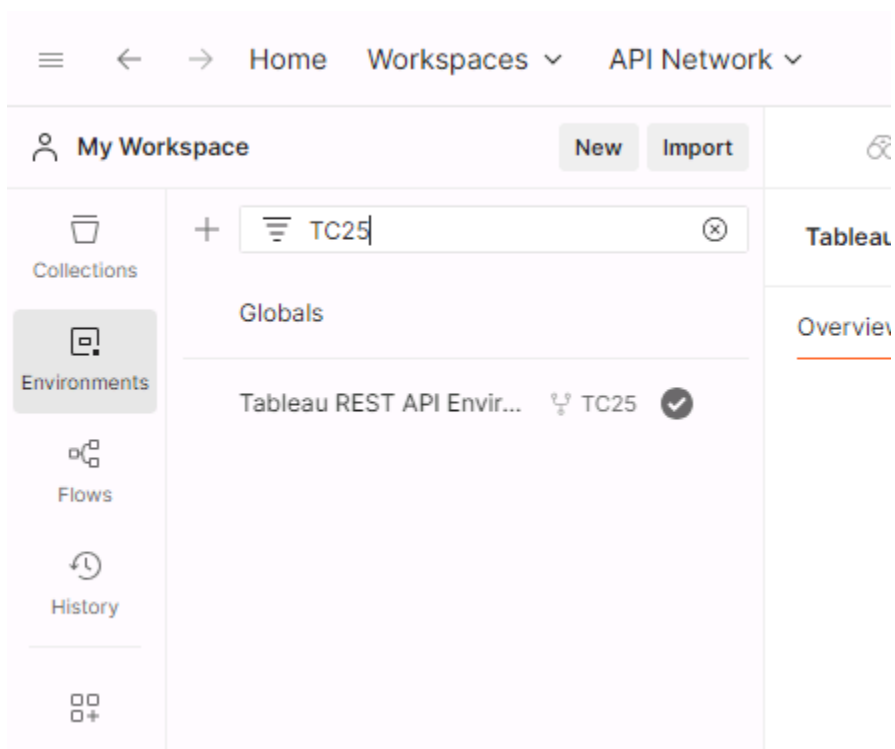
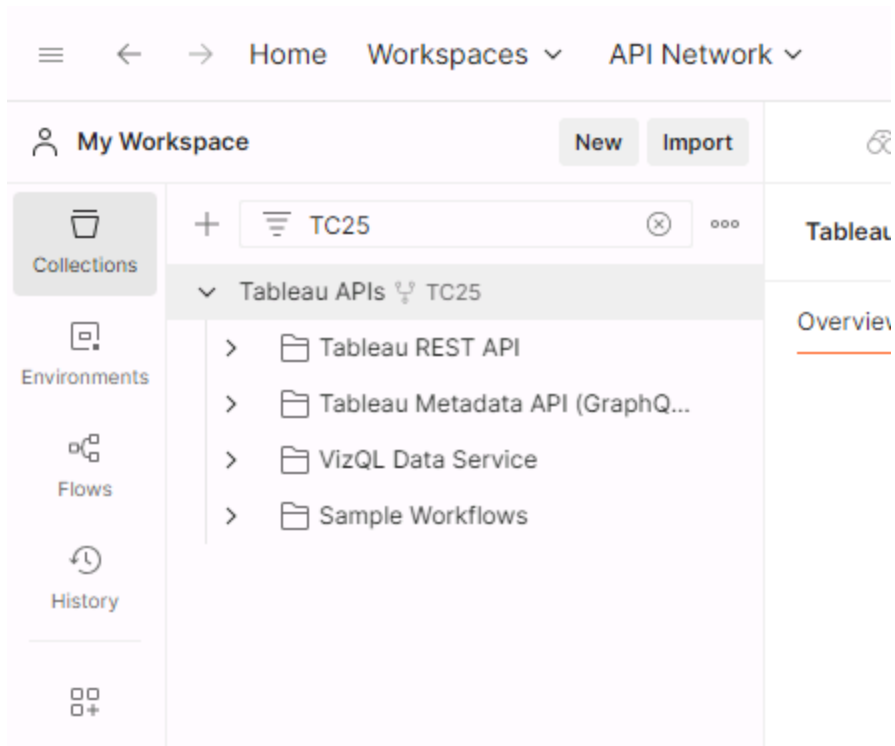
- Once you've created your account, go back to the Postman application on the desktop and sign-in using your credentials. Strangely, this re-opens a browser tab to sign-in, only to route you back to the application. 🤖
- Now that we're signed in, we can go get our Tableau REST API Collection. On the Desktop, you will see a folder called Lab Files. Go ahead and open it up. There should be 2 files.
 - TC25_Tableau APIs.postman_collection.json
 - TC25_Tableau REST API Environment.postman_environment.json
- We first need to import both the collection and the environment file to Postman. Click the **Import** button next to **My Workspace**.



8. Drag both files from the **Lab Files** folder to the import window in Postman. **Click Import.**

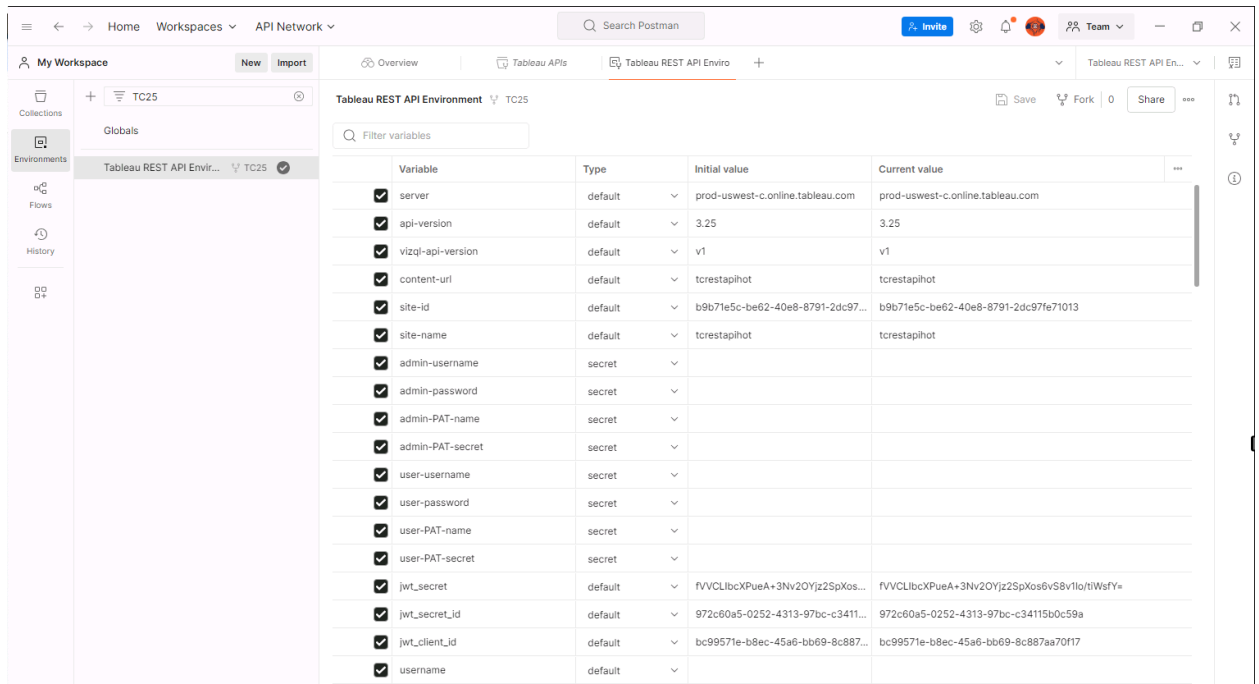


9. You should now have both the Collection and Environment in your Postman application.

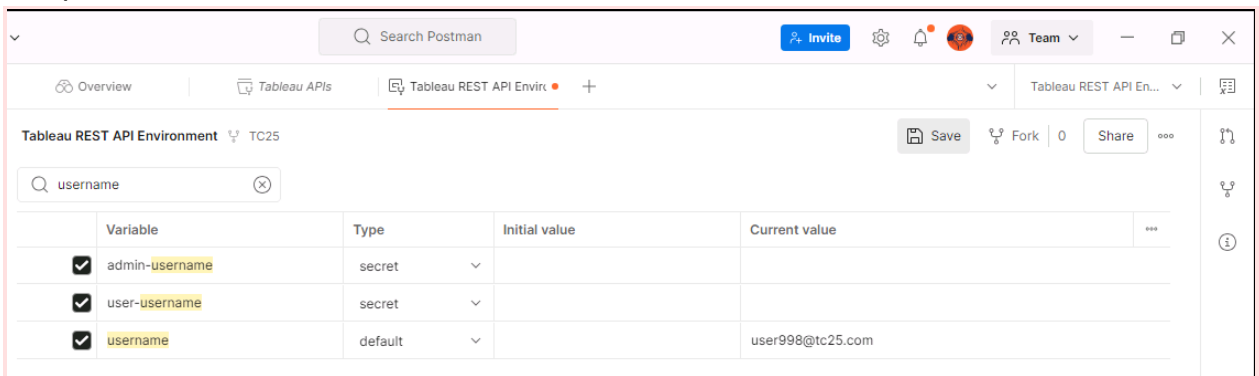


10. Go ahead and click on Environments and select the Tableau REST API Environment collection (if not already selected). There should be a number of variables already loaded into your Environment file. We only need to add one

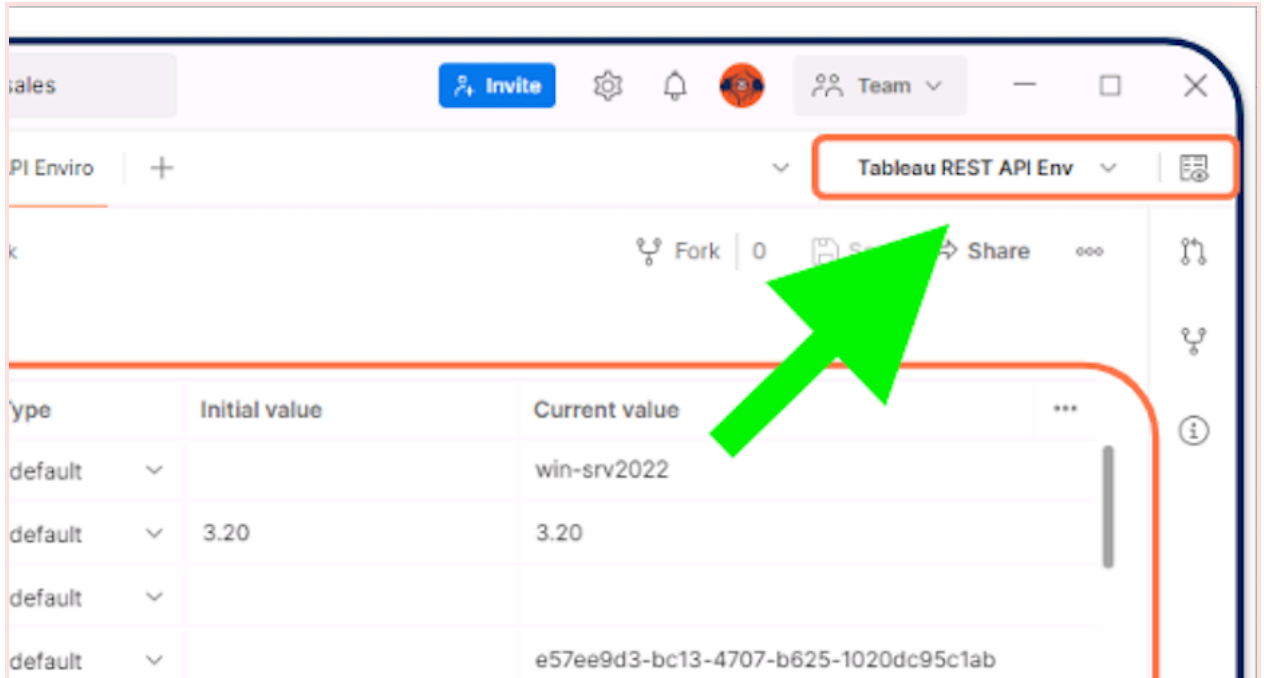
more... YOUR USERNAME.



- Place your cursor in the cell for Current Value for the **username** variable and then from the top of the instruction click the Username: [@lab.CloudCredential\(TC25HOT1942\).Username](#) to paste your assigned user email address into the environment. Click the Save button to save your new variable entry.

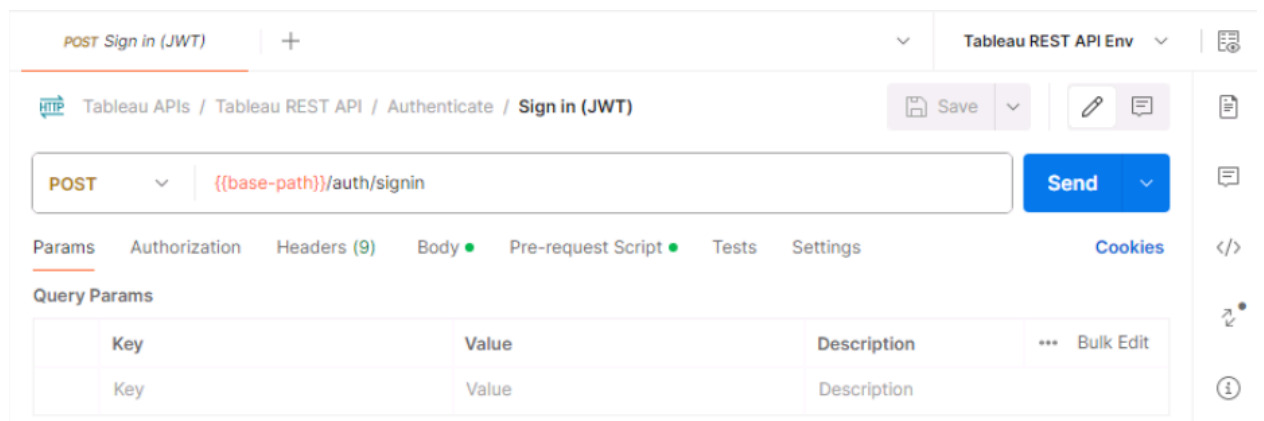


⚠ IMPORTANT!! Don't forget to select your environment in the upper right corner. Failing to do so will cause unnecessary mental anguish.



***The Collection also has some variables to leverage when working with the Tableau REST API. We don't need to modify any of the variables from the Collection before we can get started. But if you're curious about what `{{base-path}}` is or your `{{api-key}}` is set you can click on **Collections->Tableau APIs** then select the **Variables** tab to view those.

- Now let's test some **Authentication** requests. Find the **Authentication** folder under the Tableau REST API section of the API collection. Click on the **Sign in (JWT)** API request. You'll notice the **Body** and **Pre-request Script** sections under the URI have a green dot . This indicates additional information that needs our attention.

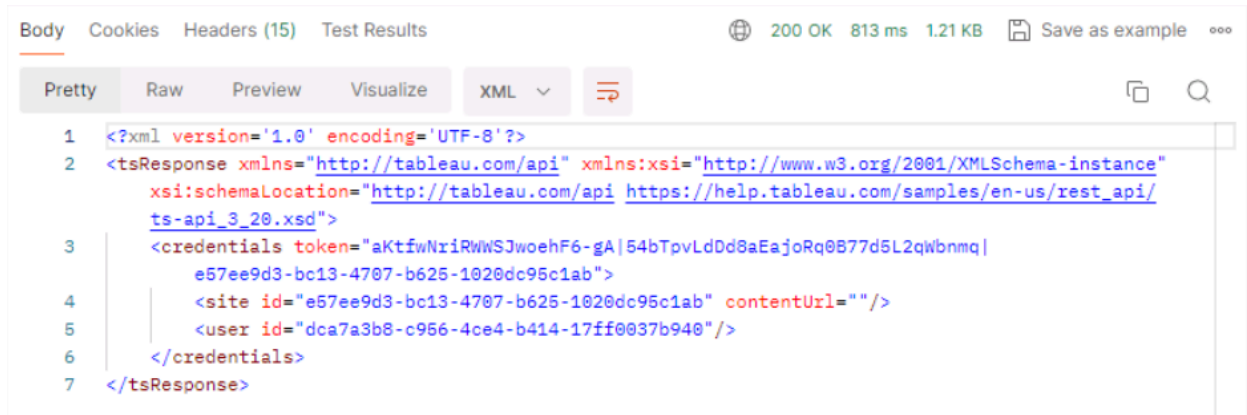


13. Most of the API calls we make will have information in the **Body** to send with the request. Here we're sending a JSON Web Token `{{jwt}}` being generated in the **Pre-request Script** as our credentials. Additionally, we're sending the variable value for `{{content-url}}`, which is the equivalent of Site name.

```
1 <tsRequest>
2   ...<credentials jwt="{{jwt}}">
3   ...<site contentUrl="{{content-url}}" />
4   ...</credentials>
5 </tsRequest>
```

Send

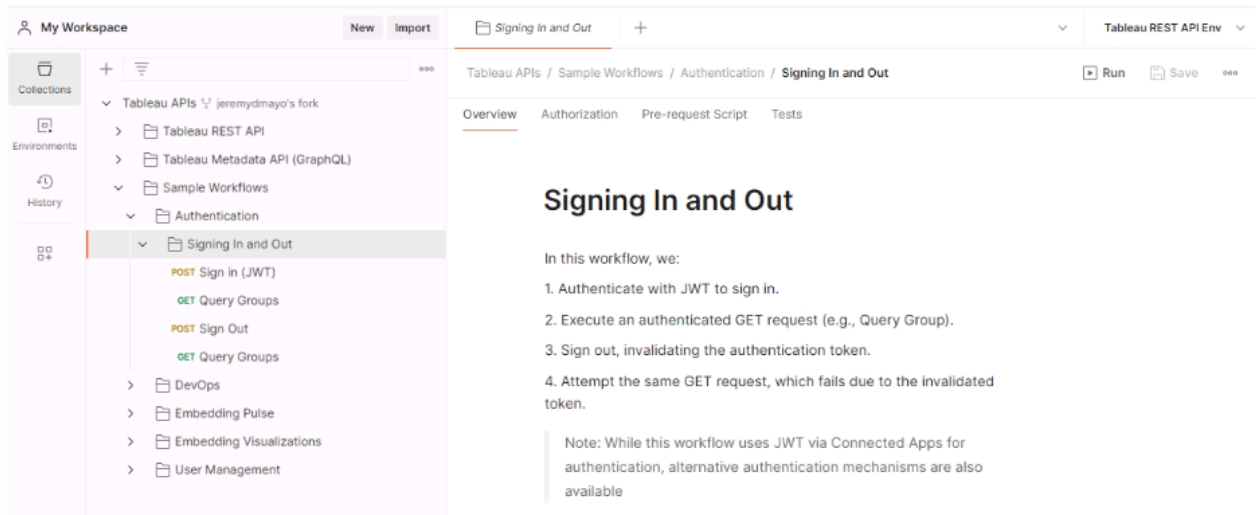
14. Click the button. You should receive a response that looks similar to the image below. Notice the credentials token. This token is your `{{api-key}}` to each subsequent call, until a **Sign-out** request is made or the token expires.

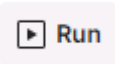


Exercise #1b - Authentication Workflow

1. Postman has a really great feature that allows you to run a complete test of APIs in a sequence. We call them **Workflows** and have included a few samples in our **Collection**. Find the **Sample Workflows** folder at the bottom of the **Tableau APIs** Collection. Under **Authentication**, there's a **Signing In and Out**

workflow. Select that and you will see all of the calls that make up the workflow.



2. Click the  button to configure the workflow. Select the checkbox ☒ to **Persist responses for a session** and leave the rest as default. Click

Run Tableau APIs

and watch the magic happen. 🪄🎩

Signing In and Out

Runner

+

Tableau REST API Env

Run order

Deselect All

Select All

Reset

✓

POST

Sign in (JWT)

✓

GET

Query Groups

✓

POST

Sign Out

✓

GET

Query Groups

Functional

Choose how to run your collection

●

Run manually

Run this collection in the Collection Runner.

○

Schedule runs

Periodically run collection at a specified time on the Postman Cloud.

○

Automate runs via CLI

Configure CLI command to run on your build pipeline.

Run configuration

Iterations

1

Delay

0ms

Data

Select File

✓

Persist responses for a session ⓘ

> Advanced settings

Run Tableau APIs

Tableau APIs - Run results

Run Again

Automate Run

+ New Run

Export Results

Ran today at 14:06:16 · View all runs

Source	Environment	Iterations	Duration	All tests	Avg. Resp. Time
Runner	Tableau REST API Environment	1	4s 347ms	5	157 ms

All Tests

Passed (4)

Failed (1)

Skipped (0)

View Summary

Iteration 1

POST

Sign in (JWT)

http://win-srv2022/api/3.20/auth/signin

200 OK 371 ms 1.238 KB

PASS

Status code is 200 - Successful sign in

GET

Query Groups

http://win-srv2022/api/3.20/sites/e57ee9d3-bc13-4707-b625-1020dc95c1ab/groups/

200 OK 203 ms 1.172 KB

PASS

Status code is 200 - Success! Data was retrieved thanks to the authentication token.

POST

Sign Out

http://win-srv2022/api/3.20/auth/signout

204 No Content 29 ms 629 B

PASS

Status code is 200 - Successful sign out

GET

Query Groups

http://win-srv2022/api/3.20/sites/e57ee9d3-bc13-4707-b625-1020dc95c1ab/groups/

401 Unauthorized 26 ms 1.213 KB

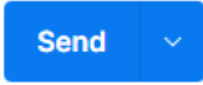

FAIL

Tableau Error: 401002 Unauthorized Access - Cause: Invalid authentication credentials were provided. (see https://help.tableau.com/current/api/rest_api/en-us/REST/rest_api...


PASS

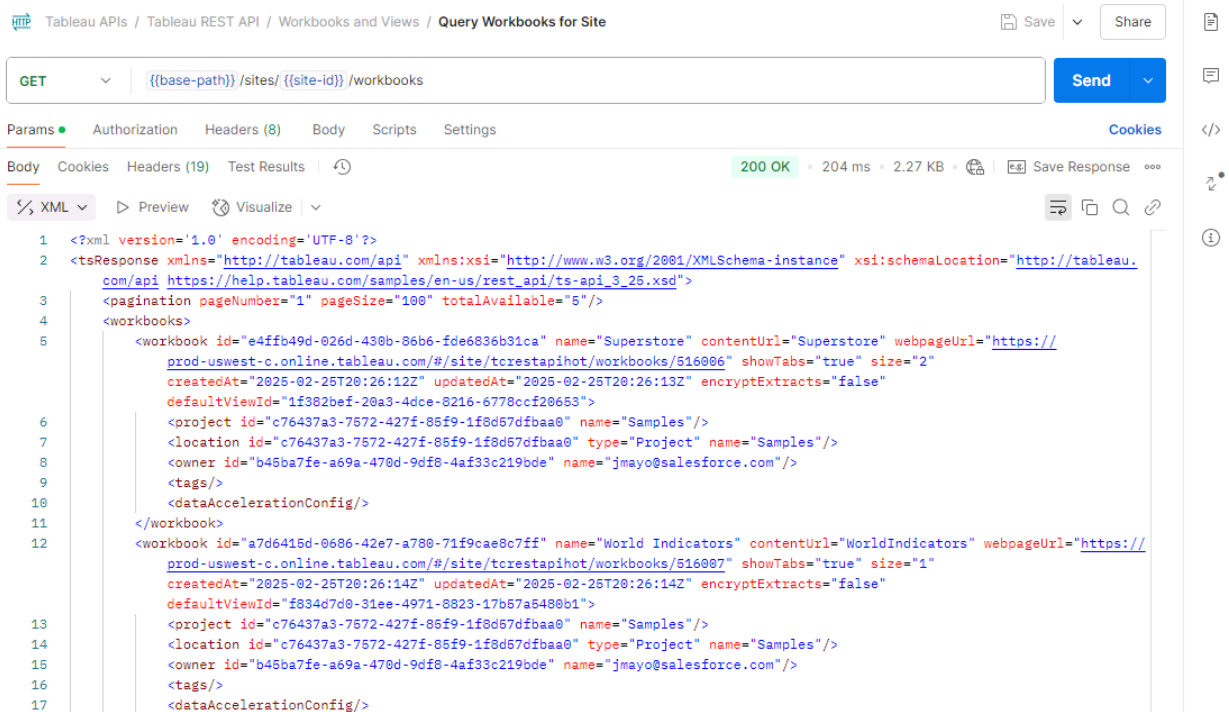
Status code is 401 - Authentication failed! No data is retrieved which is expected since the user signed out.

Exercise #2a - Working with Variables

1. Let's **find a workbook** and see what variables are available. First we need to get a new **authentication** token.
2. Find and open the request to Sign Out and click . This clears the `{{api-key}}` token.
3. Now let's re-authenticate by hitting the request to **Sign In (JWT)** and get a new token .
4. Under the Tableau REST APIs folder, scroll to **Workbooks and Views**. Open the

Query Workbooks for Site API request. Go ahead and click .

5. With a successful response  you should also get all of the workbooks on the site.



The screenshot shows a REST client interface with the following details:

- URL:** `{{base-path}}/sites/{{site-id}}/workbooks`
- Method:** GET
- Status:** 200 OK
- Response (XML):**

```
<?xml version='1.0' encoding='UTF-8'?>
<tsResponse xmlns="http://tableau.com/api" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://tableau.com/api https://help.tableau.com/samples/en-us/rest_api/ts-api_3_25.xsd">
  <pagination pageNumber="1" pageSize="100" totalAvailable="5"/>
  <workbooks>
    <workbook id="e4ffb49d-026d-430b-86b6-fde6836b31ca" name="Superstore" contentUrl="Superstore" webpageUrl="https://prod-uswest-c.online.tableau.com/#/site/tcrestapihot/workbooks/516006" showTabs="true" size="2"
      createdAt="2025-02-25T20:26:12Z" updatedAt="2025-02-25T20:26:13Z" encryptExtracts="false"
      defaultViewId="1f382bef-20a3-4dce-8216-6778ccf20653">
      <project id="c76437a3-7572-427f-85f9-1f8d57dfbaa0" name="Samples"/>
      <location id="c76437a3-7572-427f-85f9-1f8d57dfbaa0" type="Project" name="Samples"/>
      <owner id="b45ba7fe-a69a-470d-9df8-4af33c219bde" name="jmayo@salesforce.com"/>
      <tags/>
      <dataAccelerationConfig/>
    </workbook>
    <workbook id="a7d6415d-0686-42e7-a780-71f9cae8c7ff" name="World Indicators" contentUrl="WorldIndicators" webpageUrl="https://prod-uswest-c.online.tableau.com/#/site/tcrestapihot/workbooks/516007" showTabs="true" size="1"
      createdAt="2025-02-25T20:26:14Z" updatedAt="2025-02-25T20:26:14Z" encryptExtracts="false"
      defaultViewId="f834d7d0-31ee-4971-8823-17b57a5480b1">
      <project id="c76437a3-7572-427f-85f9-1f8d57dfbaa0" name="Samples"/>
      <location id="c76437a3-7572-427f-85f9-1f8d57dfbaa0" type="Project" name="Samples"/>
      <owner id="b45ba7fe-a69a-470d-9df8-4af33c219bde" name="jmayo@salesforce.com"/>
      <tags/>
      <dataAccelerationConfig/>
    </workbook>
  </workbooks>
</tsResponse>
```


6. This time let's do the same but narrow the search using a filter expression. Click on the **Params** tab and check the box next to `filter`. Under the Value (where it says filter-expression) let's add an expression to filter the search results to a workbook named **Superstore**, and owned by jmayo@salesforce.com. You can separate the expressions by a single comma.

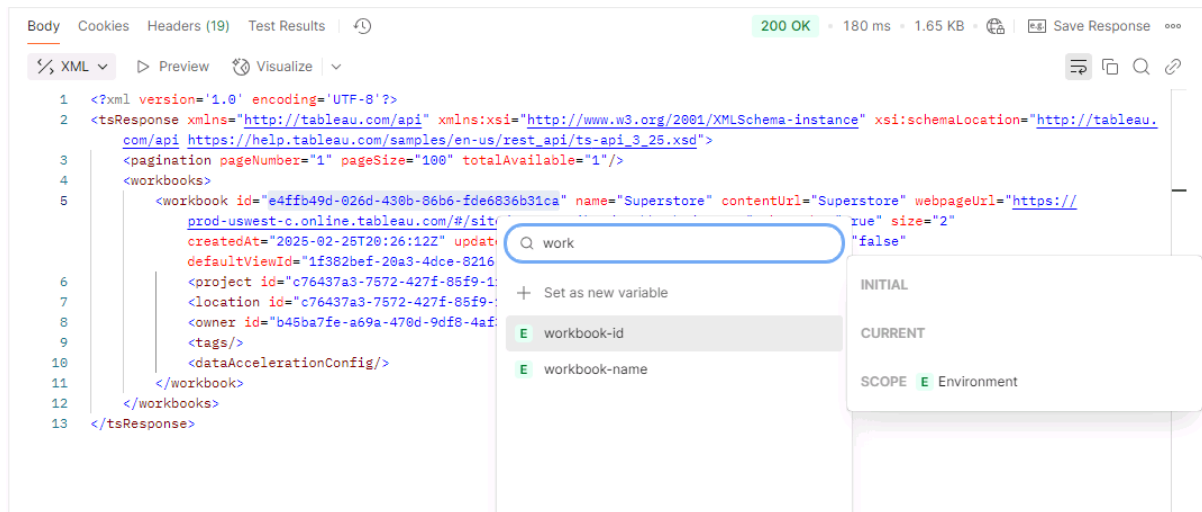
`ownerEmail:eq:jmayo@salesforce.com,name:eq:Superstore`

7. This time your response should be limited to just the one workbook named **Superstore** owned by jmayo@salesforce.com.



```
1 <?xml version='1.0' encoding='UTF-8'?>
2 <tsResponse xmlns="http://tableau.com/api" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://tableau.com/api https://help.tableau.com/samples/en-us/rest_api/ts-api_3_25.xsd">
3   <pagination pageNumber="1" pageSize="100" totalAvailable="1"/>
4   <workbooks>
5     <workbook id="e4ffb49d-026d-430b-86b6-fde6836b31ca" name="Superstore" contentUrl="Superstore" webpageUrl="https://prod-uswest-c.online.tableau.com/#/site/tcrestapihot/workbooks/516006" showTabs="true" size="2"
6       createdAt="2025-02-25T20:26:12Z" updatedAt="2025-02-25T20:26:13Z" encryptExtracts="false"
7       defaultViewId="1f382bef-20a3-4dce-8216-6778ccf20653">
8       <project id="c76437a3-7572-427f-85f9-1f8d57dfbaa0" name="Samples"/>
9       <location id="c76437a3-7572-427f-85f9-1f8d57dfbaa0" type="Project" name="Samples"/>
10      <owner id="b45ba7fe-a69a-470d-9df8-4af33c219bde" name="jmayo@salesforce.com"/>
11      <tags/>
12      <dataAccelerationConfig/>
13    </workbook>
  </workbooks>
</tsResponse>
```

8. Since we're going to need the workbook id in future calls, let's go ahead and grab that id (between the quotes) and right click . Select the option to **Set as Variable** and type workbook to locate `{{workbook-id}}`.



```
1 <?xml version='1.0' encoding='UTF-8'?>
2 <tsResponse xmlns="http://tableau.com/api" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://tableau.com/api https://help.tableau.com/samples/en-us/rest_api/ts-api_3_25.xsd">
3   <pagination pageNumber="1" pageSize="100" totalAvailable="1"/>
4   <workbooks>
5     <workbook id="e4ffb49d-026d-430b-86b6-fde6836b31ca" name="Superstore" contentUrl="Superstore" webpageUrl="https://prod-uswest-c.online.tableau.com/#/site/tcrestapihot/workbooks/516006" showTabs="true" size="2"
6       createdAt="2025-02-25T20:26:12Z" updatedAt="2025-02-25T20:26:13Z" encryptExtracts="false"
7       defaultViewId="1f382bef-20a3-4dce-8216-6778ccf20653">
8       <project id="c76437a3-7572-427f-85f9-1f8d57dfbaa0" name="Samples"/>
9       <location id="c76437a3-7572-427f-85f9-1f8d57dfbaa0" type="Project" name="Samples"/>
10      <owner id="b45ba7fe-a69a-470d-9df8-4af33c219bde" name="jmayo@salesforce.com"/>
11      <tags/>
12      <dataAccelerationConfig/>
13    </workbook>
  </workbooks>
</tsResponse>
```

9. Now that we have our `{{workbook-id}}` we can go get some more info about our workbook in a later exercise.

Exercise #3 - Requesting Images, PDFs & Data

1. Sticking with **Workbooks and Views**, open the **Query Views for Workbook** API request. On the **Params** tab, notice under **Path Variables** we're going to

use the `{{workbook-id}}` we just grabbed, only this time we're getting a detailed list of the views that make up this workbook. Go ahead and click

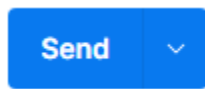


Tableau APIs / Tableau REST API / Workbooks and Views / Query Views for Workbook

GET `{{base-path}}/sites/{{site-id}}/workbooks/workbook-id/views` Send

Params Authorization Headers (8) Body Scripts Settings Cookies

Query Params

Key	Value	Description	Bulk Edit
<input type="checkbox"/> includeUsageStatistics	get-usage-information	(Optional) true to return usage statistics. The defa...	
Key	Value	Description	

Path Variables

Key	Value	Description	Bulk Edit
workbook-id	{{workbook-id}}	Description	

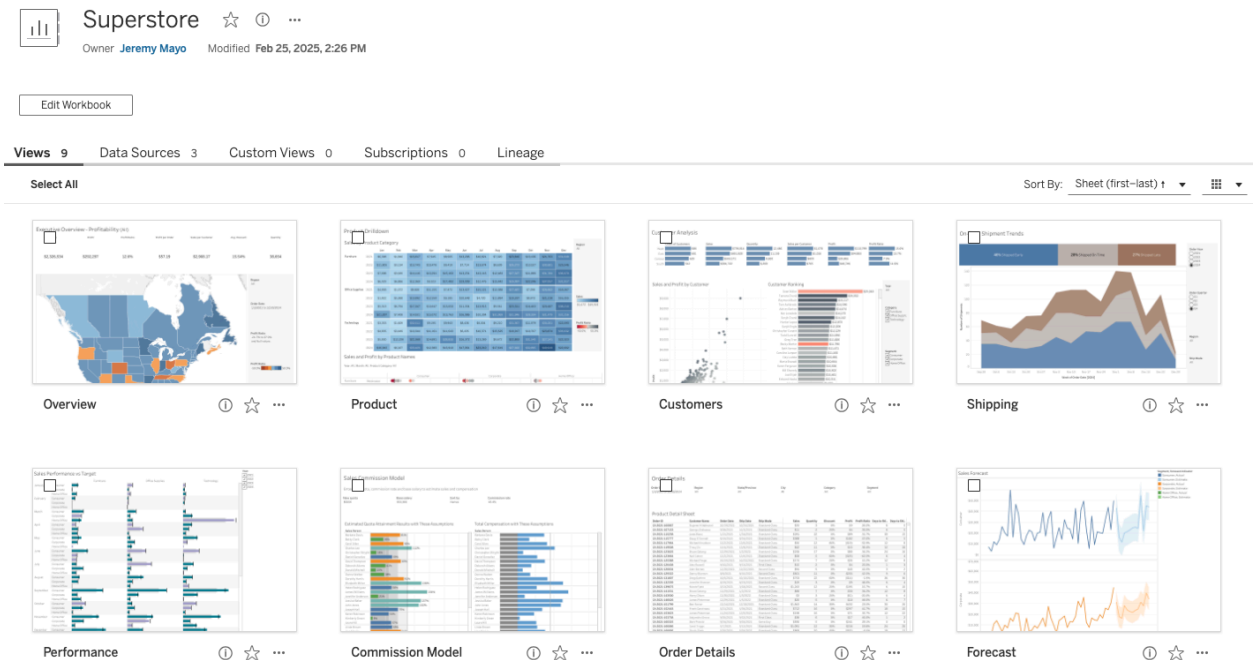
2. There we go! Now we have all the view ids and some other info for the views that are in this workbook.

Body Cookies Headers (20) Test Results Visualize

200 OK • 3.67 s • 1.81 KB Save Response

```
<?xml version='1.0' encoding='UTF-8'?>
<tsResponse xmlns="http://tableau.com/api" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://tableau.com/api https://help.tableau.com/samples/en-us/rest_api/ts-api_3_25.xsd">
  <views>
    <view id="1f382bef-20a3-4dce-8216-6778ccf20653" name="Overview" contentUrl="Superstore/sheets/Overview"
      createdAt="2025-02-25T20:26:12Z" updatedAt="2025-02-25T20:26:12Z" viewUrlName="Overview">
      <tags/>
    </view>
    <view id="346a5a5f-bfdb-4d18-8736-1c746601bc44" name="Product" contentUrl="Superstore/sheets/Product"
      createdAt="2025-02-25T20:26:12Z" updatedAt="2025-02-25T20:26:12Z" viewUrlName="Product">
      <tags/>
    </view>
    <view id="37ac5d39-cc4d-4320-b6d6-586b5b11b0c" name="Customers" contentUrl="Superstore/sheets/Customers"
      createdAt="2025-02-25T20:26:12Z" updatedAt="2025-02-25T20:26:12Z" viewUrlName="Customers">
      <tags/>
    </view>
    <view id="95fce7b5-3880-428f-b513-997e3a1aad3f" name="Shipping" contentUrl="Superstore/sheets/Shipping"
      createdAt="2025-02-25T20:26:12Z" updatedAt="2025-02-25T20:26:12Z" viewUrlName="Shipping">
      <tags/>
    </view>
    <view id="0d109ae9-901e-4306-8cd8-314c2860e76d" name="Performance" contentUrl="Superstore/sheets/Performance"
      createdAt="2025-02-25T20:26:12Z" updatedAt="2025-02-25T20:26:12Z" viewUrlName="Performance">
      <tags/>
    </view>
    <view id="7e5736a5-642c-43c2-9838-ba53197daf5c" name="Commission Model" contentUrl="Superstore/sheets/CommissionModel"
      createdAt="2025-02-25T20:26:12Z" updatedAt="2025-02-25T20:26:12Z" viewUrlName="CommissionModel">
      <tags/>
    </view>
    <view id="541ea30b-911d-47d4-8d1d-9cabd4b0b94b" name="Order Details" contentUrl="Superstore/sheets/OrderDetails"
      createdAt="2025-02-25T20:26:12Z" updatedAt="2025-02-25T20:26:12Z" viewUrlName="OrderDetails">
      <tags/>
    </view>
    <view id="955e9771-3b56-422a-8115-4a333ba2fc67" name="Forecast" contentUrl="Superstore/sheets/Forecast"
      createdAt="2025-02-25T20:26:12Z" updatedAt="2025-02-25T20:26:12Z" viewUrlName="Forecast">
      <tags/>
    </view>
    <view id="a9b941ec-ad58-488b-af63-6d8847977242" name="What If Forecast" contentUrl="Superstore/sheets/WhatIfForecast"
      createdAt="2025-02-25T20:26:12Z" updatedAt="2025-02-25T20:26:12Z" viewUrlName="WhatIfForecast">
      <tags/>
  </views>
</tsResponse>
```

3. Pick any of the views from the response and save it as your `{{view-id}}` variable. We're going to use that `{{view-id}}` to grab an image so for context below is a screenshot of what each view looks like.



As a reminder, to save the `{{view-id}}`, select the **view id** of the view you want to query, then right click → **Set as variable** → Search for **view-id** → Click **view-id** to set your variable.

```
<view id="1f382bef-20a3-4dce-8216-6778ccf20653" name="Overview" contentUrl="Superstore/sheets/Overview"
  createdAt="2025-02-25T20:26:12Z" updatedAt="2025-02-25T20:26:12Z">
  <tags/>
</view>
<view id="346a5a5f-bfdb-4d18-8736-1c746601bc" name="Product" contentUrl="Superstore/sheets/Product"
  createdAt="2025-02-25T20:26:12Z" updatedAt="2025-02-25T20:26:12Z">
  <tags/>
</view>
<view id="37ac5d39-cc4d-4320-b6d6-586b5b111b" name="Customers" contentUrl="Superstore/sheets/Customers"
  createdAt="2025-02-25T20:26:12Z" updatedAt="2025-02-25T20:26:12Z">
  <tags/>
</view>
<view id="95fce7b5-3880-428f-b513-997e3a1aac" name="Shipping" contentUrl="Superstore/sheets/Shipping"
  createdAt="2025-02-25T20:26:12Z" updatedAt="2025-02-25T20:26:12Z">
  <tags/>
</view>
<view id="0d109ae9-901e-4306-8cd8-314c2860e7" name="Performance" contentUrl="Superstore/sheets/Performance"
  createdAt="2025-02-25T20:26:12Z" updatedAt="2025-02-25T20:26:12Z">
  <tags/>
</view>
```

A search bar with the text 'view' is shown. Below it, a list of suggestions is displayed:

- + Set as new variable
- E view-id
- E custom-view-luid
- E view-name
- E send-if-view-empty

On the right side, a sidebar shows the 'INITIAL' tab selected, with 'CURRENT' and 'SCOPE' tabs below it. The 'SCOPE' tab is highlighted, showing 'Environment'.

4. Under the **Workbook and Views** folder, locate the **Query View Image** API request and select it to open it up. Notice that the **Params** tab has some preset

Path Variables using the `{{view-id}}` variable you just set.

Tableau APIs / Tableau REST API / Workbooks and Views / Query View Image

Save Share

GET

{{base-path}}/sites/{{site-id}}/views/view-id/image

Send

Params

Authorization

Headers (8)

Body

Scripts

Settings

Cookies

Query Params

<input type="checkbox"/>	Key	Value	Description	Bulk Edit
<input type="checkbox"/>	resolution	image-resolution	(Optional) The resolution of the image. Image widt...	
<input type="checkbox"/>	maxAge	max-age-minutes	(Optional) The maximum number of minutes a view...	
<input type="checkbox"/>	vf_<fieldname>	filter-value	The value of the field that you want to use to filter ...	
	Key	Value	Description	

Path Variables

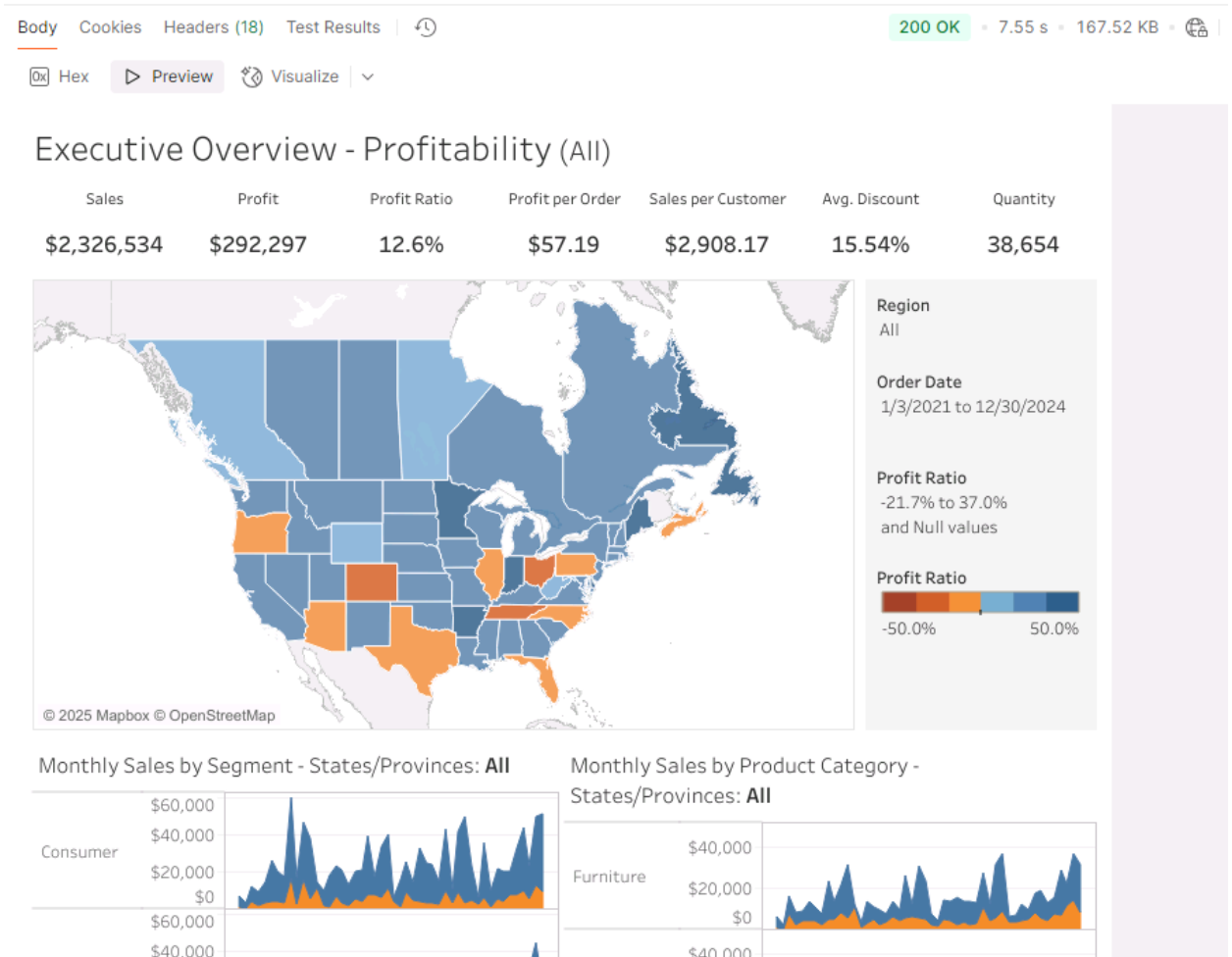
	Key	Value	Description	Bulk Edit
	view-id	{{view-id}}	Description	

5. Let's give this one a go. Hit

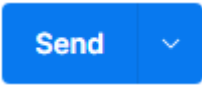
Send

. You should get an image in the response body below, similar to the output here. Obviously, if you selected a

different `{{view-id}}` your image will be different.

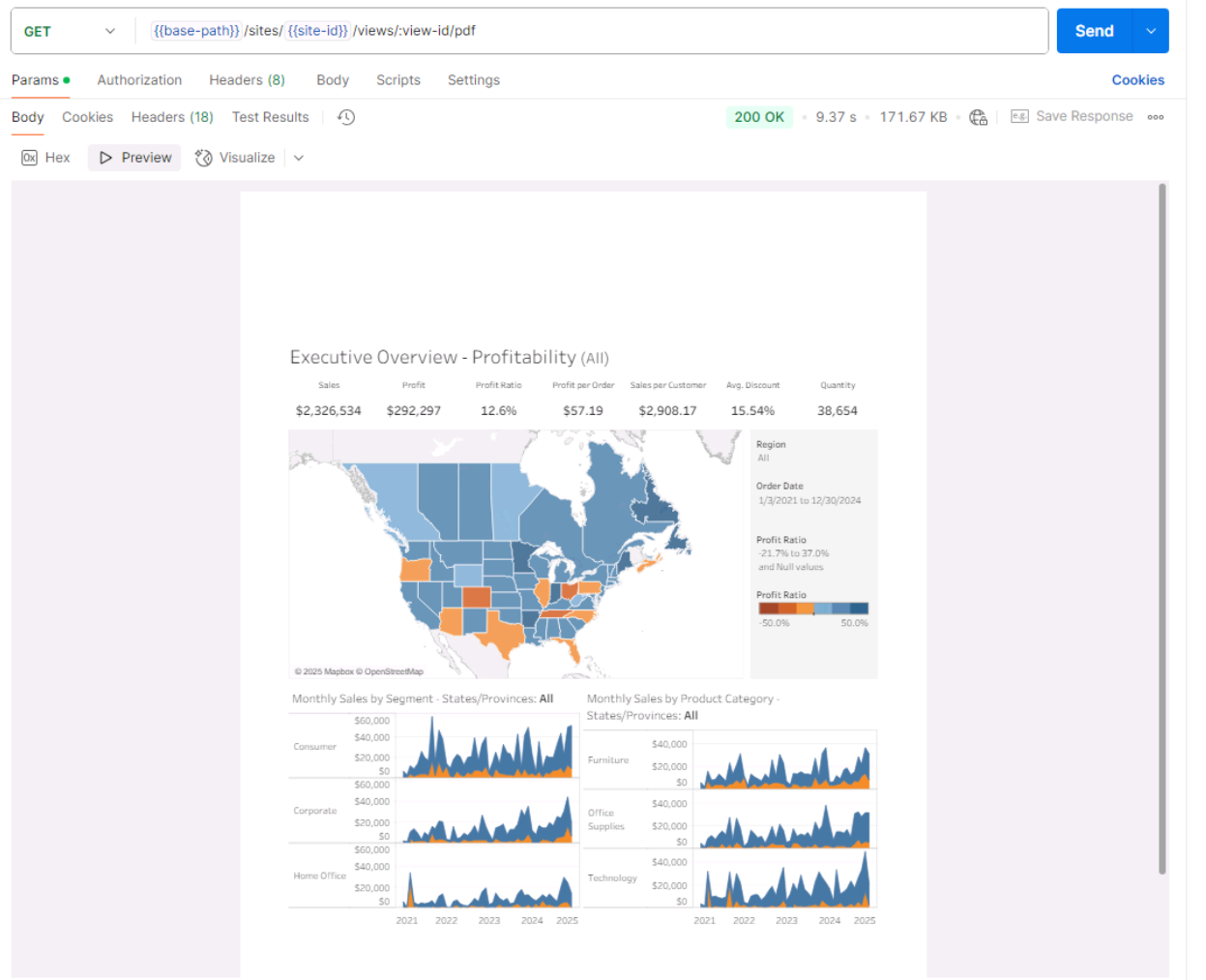


6. Now let's do the same but get a PDF instead. Back under the **Workbook and Views** folder, locate the **Query View PDF** API request and select it to open it up. Notice that, similar to **Query View Image**, the **Params** tab has a preset

Path Variable using the `{{view-id}}` variable. Click .

7. This time the output should be a formatted one page Portrait image of the same view. You can play around with some of the other **Query Params** to modify the

output.



8. Finally, let's grab data from the view. One more time under the **Workbook and Views** folder, locate the **Query View Data** API request and select it to open it up. Similar to the first 2 calls, the **Params** tab has a preset **Path Variable** using

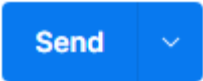



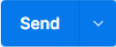


the `{{view-id}}` variable. Hit .

Tableau APIs / Tableau REST API / Workbooks and Views / Query View Data  Save  Share

GET  `{{base-path}}/sites/{{site-id}}/views/view-id/data` 

Params  Authorization Headers (8) Body Scripts Settings 

Query Params

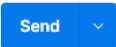
<input type="checkbox"/> Key	Value	Description	*** Bulk Edit
<input type="checkbox"/> maxAge	max-age-minutes	(Optional) The maximum number of minutes view ...	
<input type="checkbox"/> vf_<fieldname>	filter-value	The value of the field that you want to use to filter...	
Key	Value	Description	


Path Variables


Key	Value	Description	*** Bulk Edit
view-id	<code>{{view-id}}</code>	Description	


9. Depending on the view you chose, your results may vary.




Tableau APIs / Tableau REST API / Workbooks and Views / Query View Data  Save  Share

GET  `{{base-path}}/sites/{{site-id}}/views/view-id/data` 

Params  Authorization Headers (8) Body Scripts Settings 

Body Cookies Headers (18) Test Results  200 OK • 5.73 s • 3.54 KB •  Save Response ***

 Raw  Preview  Visualize 

```
1 Country/Region,State/Province,Profit Ratio,Latitude (generated),Longitude (generated)
2 Canada,Alberta,19.6%,53.41,-114.42
3 Canada,British Columbia,4.2%,54.9464,-125.1024
4 Canada,Manitoba,8.2%,55.0085,-97.1771
5 Canada,Newfoundland and Labrador,36.2%,53.63,-61.37
6 Canada,Nova Scotia,-13.8%,45,-63.58
7 Canada,Ontario,26.8%,50.94,-84.75
8 Canada,Prince Edward Island,30.3%,46.32,-63.19
9 Canada,Quebec,28.5%,53.02,-72.58
10 Canada,Saskatchewan,21.6%,54.49,-105.68
11 United States,Alabama,29.7%,32.75,-86.75
12 United States,Arizona,-9.7%,34.6,-111.5
13 United States,Arkansas,34.3%,34.75,-92.5
14 United States,California,16.7%,37.25,-119.75
```

Exercise #4 - Data & Filter Expressions

1. From the **Workbooks and Views** folder, open the **Query View Data** request. This requires a `{{view-id}}` variable and we still have ours from the last

exercise (**Overview**). Go ahead and click .

Tableau APIs / Tableau REST API / Workbooks and Views / **Query View Data**

GET `{{(base-path)}}/sites/{{(site-id)}}/views/:view-id/data` **Send**

Params • Authorization Headers (8) Body Pre-request Script Tests Settings **Cookies**

Query Params

Key	Value	Description
<input type="checkbox"/> maxAge	max-age-minutes	(Optional) The maximum number of minutes view dat...
<input type="checkbox"/> vf_<fieldname>	filter-value	The value of the field that you want to use to filter the...
Key	Value	Description

Path Variables

Key	Value	Description
view-id	<code>{{(view-id)}}</code>	Description

Body Cookies Headers (13) Test Results **Status: 200 OK Time: 4.41 s Size: 3.19 KB Save as example**

Pretty Raw Preview Visualize Text

```

1 Country/Region,State/Province,Profit Ratio,Latitude (generated),Longitude (generated)
2 Canada,Alberta,19.5%,53.41,-114.42
3 Canada,British Columbia,4.2%,54.9464,-125.1024
4 Canada,Manitoba,8.2%,55.0085,-97.1771
5 Canada,Newfoundland and Labrador,36.2%,63.63,-61.37
6 Canada,Nova Scotia,-13.8%,45,-63.58
7 Canada,Ontario,26.8%,50.94,-84.75
8 Canada,Prince Edward Island,30.3%,46.32,-63.19
9 Canada,Quebec,28.5%,53.02,-72.58
10 Canada,Saskatchewan,21.6%,54.49,-105.68
11 United States,Alabama,29.7%,32.75,-86.75
12 United States,Arizona,-9.7%,34.5,-111.6
13 United States,Arkansas,34.3%,34.75,-92.6

```

Activate Windows
Go to Settings to activate Windows.

- This isn't quite what we expected. Why didn't it return the full data table for the **Overview** view? That's because when using **Query View Data** from a dashboard, only data from the first sheet¹ in the dashboard view gets returned.
- Let's fix that. Open and run **Query Views for Workbook**. This takes a `{{(workbook-id)}}` as a path variable and you should already have that set from the previous exercise. From the response, scroll down until you see the view named **Order Details**. Grab the view id for **Order Details** and set this to your `{{(view-id)}}` variable.

GET {{base-path}}/sites/{{site-id}}/workbooks/workbook-id/views

Send

Params Authorization Headers (8) Body Scripts Settings

Cookies

Body Cookies Headers (20) Test Results

200 OK 3.67 s 1.81 KB Save Response

XML Preview Visualize

```
3 <views>
4   <view id="1f382bef-28a3-4dce-8216-6778ccf28653" name="Overview" contentUrl="Superstore/sheets/Overview"
5     createdAt="2025-02-25T20:26:12Z" updatedAt="2025-02-25T20:26:12Z" viewUrlName="Overview">
6     <tags/>
7   </view>
8   <view id="346a5a5f-bfdb-4d18-8736-1c746601bc44" name="Product" contentUrl="Superstore/sheets/Product"
9     createdAt="2025-02-25T20:26:12Z" updatedAt="2025-02-25T20:26:12Z" viewUrlName="Product">
10    <tags/>
11  </view>
12  <view id="37ac5d39-cc4d-4320-8225-02-25T20:26:12Z" name="Customers" contentUrl="Superstore/sheets/Customers"
13    createdAt="2025-02-25T20:26:12Z" updatedAt="2025-02-25T20:26:12Z" viewUrlName="Customers">
14    <tags/>
15  </view>
16  <view id="95fce7b5-3880-428f-8225-02-25T20:26:12Z" name="Performance" contentUrl="Superstore/sheets/Performance"
17    createdAt="2025-02-25T20:26:12Z" updatedAt="2025-02-25T20:26:12Z" viewUrlName="Performance">
18    <tags/>
19  </view>
20  <view id="0d109ae9-901e-4306-8225-02-25T20:26:12Z" name="CommissionModel" contentUrl="Superstore/sheets/CommissionModel"
21    createdAt="2025-02-25T20:26:12Z" updatedAt="2025-02-25T20:26:12Z" viewUrlName="CommissionModel">
22    <tags/>
23  </view>
24  <view id="641ea38b-911d-47d4-8d1d-9cabd4b8b94b" name="Order Details" contentUrl="Superstore/sheets/OrderDetails"
25    createdAt="2025-02-25T20:26:12Z" updatedAt="2025-02-25T20:26:12Z" viewUrlName="OrderDetails">
26    <tags/>
27  </view>
28  <view id="965e9771-3b56-422a-8115-4a333ba2fc67" name="Forecast" contentUrl="Superstore/sheets/Forecast"
29    createdAt="2025-02-25T20:26:12Z" updatedAt="2025-02-25T20:26:12Z" viewUrlName="Forecast">
30    <tags/>
31  </view>
```

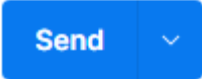

5. Go back to the **Query View Data** request. Click . That's more like it. This gives me more order specific detail from Superstore.

Tableau APIs / Tableau REST API / Workbooks and Views / Query View Data

GET `{{(base-path)}/sites/{{(site-id)}}/views/view-id/data` 



Params • Authorization Headers (8) Body Pre-request Script Tests Settings Cookies


Query Params

<input type="checkbox"/> Key	Value	Description	*** Bulk Edit
<input type="checkbox"/> maxAge	max-age-minutes	(Optional) The maximum number of minutes view dat...	
<input type="checkbox"/> vf_<fieldname>	filter-value	The value of the field that you want to use to filter the...	
Key	Value	Description	

Path Variables

Key	Value	Description	*** Bulk Edit
view-id	{{(view-id)}}	Description	

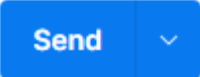
Body Cookies Headers (13) Test Results  Status: 200 OK Time: 17.38 s Size: 2.78 MB  Save as example

Pretty Raw Preview Visualize Text 


```
1 Customer Name,Measure Names,Order Date,Order ID,Ship Date,Ship Mode,Measure Values
2 Aaron Bergman,Sales,2/18/2020,US-2020-152905,2/24/2020,Standard Class,12.624
3 Aaron Bergman,Quantity,2/18/2020,US-2020-152905,2/24/2020,Standard Class,2
4 Aaron Bergman,Discount,2/18/2020,US-2020-152905,2/24/2020,Standard Class,0.2
5 Aaron Bergman,Profit,2/18/2020,US-2020-152905,2/24/2020,Standard Class,-2.5248
6 Aaron Bergman,Profit Ratio,2/18/2020,US-2020-152905,2/24/2020,Standard Class,-0.2
7 Aaron Bergman,Days to Ship Scheduled,2/18/2020,US-2020-152905,2/24/2020,Standard Class,6
8 Aaron Bergman,Days to Ship Actual,2/18/2020,US-2020-152905,2/24/2020,Standard Class,6
9 Aaron Bergman,Sales,3/7/2020,US-2020-156587,3/8/2020,First Class,309.592
10 Aaron Bergman,Quantity,3/7/2020,US-2020-156587,3/8/2020,First Class,7
11 Aaron Bergman,Discount,3/7/2020,US-2020-156587,3/8/2020,First Class,0.2
12 Aaron Bergman,Profit,3/7/2020,US-2020-156587,3/8/2020,First Class,15.0033
13 Aaron Bergman,Profit Ratio,3/7/2020,US-2020-156587,3/8/2020,First Class,0.048461524
```

Activate Windows
Go to Settings to activate Windows


6. Let's filter this output so that we only get orders with a **Ship Mode of First Class**. Under the **Params** tab, check the box for vf_<fieldname>. Modify the vf_<fieldname> to read **vf_Ship Mode** and set the filter-value to **First Class**.





Click . You should now have a filtered data set result.

HTTP Tableau APIs / Tableau REST API / Workbooks and Views / Query View Data

GET  `{{base-path}}/sites/{{site-id}}/views/:view-id/data?vf_Ship Mode=First Class`


Params • Authorization Headers (8) Body Scripts Settings


Body Cookies Headers (18) Test Results  200 OK • 3.57 s

 Raw  Preview  Visualize 

```
1 Customer Name,Measure Names,Order Date,Order ID,Ship Date,Ship Mode,Measure Values
2 Aaron Bergman,Sales,3/7/2021,US-2021-156587,3/8/2021,First Class,309.592
3 Aaron Bergman,Quantity,3/7/2021,US-2021-156587,3/8/2021,First Class,7
4 Aaron Bergman,Discount,3/7/2021,US-2021-156587,3/8/2021,First Class,0.2
5 Aaron Bergman,Profit,3/7/2021,US-2021-156587,3/8/2021,First Class,15.0033
6 Aaron Bergman,Profit Ratio,3/7/2021,US-2021-156587,3/8/2021,First Class,0.048461524
7 Aaron Bergman,Days to Ship Scheduled,3/7/2021,US-2021-156587,3/8/2021,First Class,3
8 Aaron Bergman,Days to Ship Actual,3/7/2021,US-2021-156587,3/8/2021,First Class,3
9 Aaron Bergman,Sales,11/10/2023,US-2023-140935,11/12/2023,First Class,563.94
10 Aaron Bergman,Quantity,11/10/2023,US-2023-140935,11/12/2023,First Class,4
11 Aaron Bergman,Discount,11/10/2023,US-2023-140935,11/12/2023,First Class,0
12 Aaron Bergman,Profit,11/10/2023,US-2023-140935,11/12/2023,First Class,116.868
13 Aaron Bergman,Profit Ratio,11/10/2023,US-2023-140935,11/12/2023,First Class,0.207234812
14 Aaron Bergman,Days to Ship Scheduled,11/10/2023,US-2023-140935,11/12/2023,First Class,2
15 Aaron Bergman,Days to Ship Actual,11/10/2023,US-2023-140935,11/12/2023,First Class,4
16 Aaron Hawkins,Sales,10/25/2021,US-2021-158400,10/28/2021,First Class,49.408
17 Aaron Hawkins,Quantity,10/25/2021,US-2021-158400,10/28/2021,First Class,4
18 Aaron Hawkins,Discount,10/25/2021,US-2021-158400,10/28/2021,First Class,0.2
19 Aaron Hawkins,Profit,10/25/2021,US-2021-158400,10/28/2021,First Class,18.528
```

Exercise #5 - Headless BI Applications

- Up to this point, we've focused exclusively on API calls made against workbooks or views that are published to Tableau Server or Cloud. But what if all you want is the data. Enter the VizQL Data Service API, also known as Headless BI API. Through the use of this newly introduced API, and its quick add to the Postman collection, we are able to query directly against a  Tableau Published Data Source.
- In the interest of time, we're gonna speed up this exercise a bit. Postman has a really great feature that allows you to run a complete test of APIs in a sequence. We call them **Workflows** and have included a few samples in our **Collection**. Find the **Sample Workflows** folder at the bottom of the **Tableau APIs** Collection. Select the **VizQL Data Service Queries**, and read thru the Overview to get an idea of what is happening in this workflow.

3. Click the  **Run** button to configure the workflow. Select the checkbox ☒ to **Persist responses for a session** and leave the rest as default. Click

Run Tableau APIs

and watch the magic happen. 🪄🎩

Run Sequence

Deselect All | Select All | Reset

1

☒

POST

Sign in (JWT)

2

☒

GET

Query Data Sources

3

☒

POST

Read Metadata

4

☒

POST

Query Datasource

5

☒

POST

Sign Out

Functional

Performance

Choose how to run your collection

☒ Run manually
Run this collection in the Collection Runner.

☐ Schedule runs
Periodically run collection at a specified time on the Postman Cloud.

☐ Automate runs via CLI
Configure CLI command to run on your build pipeline.

Run configuration

Iterations ⓘ
1

Delay ⓘ
0 ms

Test data file ⓘ
Only JSON and CSV files are accepted.
Select File

☐ Persist responses for a session ⓘ
☐ Turn off logs during run ⓘ
Advanced settings

Run Tableau APIs

4. Inspect the results set for the **READ METADATA** call

Tableau APIs - Run results

Run today at 02:08:13 · View all runs

Source

Environment

Iterations

Duration

All tests

Avg. Resp. Time

Runner

Tableau REST API Environment

1

2s 396ms

1

198 ms

All Tests Passed (1) Failed (0) Skipped (0)

View Summary

Iteration 1

POST Sign in (JWT)
https://prod-uswest-c.online.tableau.com/api/3.25/auth/signin
200 OK
No tests found

GET Query Data Sources
https://prod-uswest-c.online.tableau.com/api/3.25/sites/b9b71e5c-be62-40e8-8791-2dc97fe710...
200 OK
No tests found

POST Read Metadata
https://prod-uswest-c.online.tableau.com/api/v1/vizq-data-service/read-metadata
200 OK
No tests found

POST Query Datasource
https://prod-uswest-c.online.tableau.com/api/v1/vizq-data-service/query-datasource
200 OK
No tests found

POST Sign Out
https://prod-uswest-c.online.tableau.com/api/3.25/auth/signout
204 No Content
Running request...

1

POST

Tableau APIs / Sample Workflows / VizQL Data Service Queries / Read Metadata

Response

Headers

Request

Pretty

200 155 ms 3.886 KB

1

{

2

"data": [

3

{

4

"fieldName": "Calculation_1368249927221915648",

5

"fieldCaption": "Profit Ratio",

6

"dataType": "REAL",

7

"logicalTableId": ""

8

},

9

{

10

"fieldName": "Category",

11

"fieldCaption": "Category",

12

"dataType": "STRING",

13

"logicalTableId": "Orders_ECFCA1FB690A41FE8038C071773BA862"

14

},

15

{

16

"fieldName": "City",

17

"fieldCaption": "City",

18

"dataType": "STRING",

19

"logicalTableId": "Orders_ECFCA1FB690A41FE8038C071773BA862"

20

},

21

{

22

"fieldName": "Country/Region",

23

"fieldCaption": "Country/Region",

24

"dataType": "STRING",

25

"logicalTableId": "Orders_ECFCA1FB690A41FE8038C071773BA862"

26

},

27

{

28

"fieldName": "Customer Name",

29

"fieldCaption": "Customer Name",

30

"dataType": "STRING",

31

"logicalTableId": "Orders_ECFCA1FB690A41FE8038C071773BA862"

32

},

33

{

34

"fieldName": "Discount",

35

"fieldCaption": "Discount",

5. Inspect the results set for the Query Datasource call

Tableau APIs - Run results Run Again Automate Run ▾ + New Run Export Results

● Ran today at 02:08:13 · [View all runs](#)

Source	Environment	Iterations	Duration	All tests	Avg. Resp. Time
Runner	Tableau REST API Environment	1	2s 396ms	1	198 ms

All Tests Passed (1) Failed (0) Skipped (0) [View Summary](#)

Iteration 1

- POST Sign in (JWT)**
https://prod-uswest-c.online.tableau.com/api/3.25/auth/signin 200 OK
No tests found
- GET Query Data Sources**
https://prod-uswest-c.online.tableau.com/api/3.25/sites/b9b71e5c-be62-40e8-8791-2dc97fe710... 200 OK
No tests found
- POST Read Metadata**
https://prod-uswest-c.online.tableau.com/api/v1/vizql-data-service/read-metadata 200 OK
No tests found
- POST Query Datasource**
https://prod-uswest-c.online.tableau.com/api/v1/vizql-data-service/query-datasource 200 OK
No tests found
- POST Sign Out**
https://prod-uswest-c.online.tableau.com/api/3.25/auth/signout 204 No Content
Running request...

1 POST Tableau APIs / Sample Workflows / VizQL Data Service Queries / Query Datasource

Response Headers Request

200 259 ms 1.147 KB

Pretty ▾

```
1 {
2   "data": [
3     {
4       "Category": "Technology",
5       "SUM(Sales)": 839893.2789999967
6     },
7     {
8       "Category": "Furniture",
9       "SUM(Sales)": 754747.7612999984
10    },
11    {
12      "Category": "Office Supplies",
13      "SUM(Sales)": 731893.3140000021
14    }
15  ]
16 }
```



Thanks so much for attending **Build, Automate, and Scale: A Tableau REST API Deep Dive Hands-on Training Session!!**

We hope this was valuable material to help you on your journey with Tableau. Please don't forget to leave feedback on the session. See ya next year!

